The subfigure Package*

Steven Douglas Cochran

Digital Mapping Laboratory, School of Computer Science
Carnegie-Mellon University, 5000 Forbes Avenue
Pittsburgh, PA 15213–3890, USA

`sdc+@cs.cmu.edu`

2002/07/02

Abstract

This article documents the LaTeX package 'subfigure', which provides support for the inclusion of small, 'sub', figures and tables. It simplifies the positioning, captioning and labeling of such objects within a single `figure` or `table` environment. In addition, this package allows such subcaptions to be written to a List-of-Figures or List-of-Tables if desired. The 'subfigure' package also cooperates with the 'caption' and 'caption2' packages by H.A. Sommerfeldt [1, 2], the 'ccaption' and 'tocloft' packages [3, 4] by Peter Wilson, the 'hyperref' package by Sebastian Rahtz [5], the 'captcont' package [6], and should be compatible with all other packages that modify or extend the `float` environment or the `\caption` or `\label` commands.

---

*This paper documents the subfigure package v2.1.4, last revised 2002/07/30.

Contents

List of Tables

List of Figures

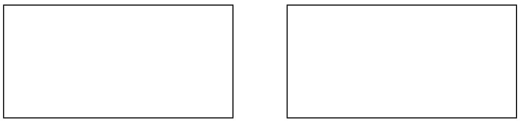


Figure 1: Here are two figures side-by-side.

1 Introduction

This package provides support for the manipulation and reference of small or ‘sub’ figures and tables within a single `figure` or `table` environment.[1] It is convenient to use this package when your subfigures are to be separately captioned, referenced, or when such subcaptions are to be included in the List-of-Figures.

Before using the `subfigure` package, consider the following to see if you really need it. If you simply want to center your figure, then you can use `\centerline`, `\centering` or the `center` environment to do so. If your figure has a short width or if you wrap your figure in a `\parbox` or a `minipage` of a short width, then you can place multiple figures or tables side-by-side. For example, the following will put two images side-by-side in a single figure as shown in figure 1:[2]

```
\begin{figure}%
  \centering
  \parbox{1.2in}{...figure code...}%
  \qquad
  \begin{minipage}{1.2in}%
    ...figure code...
  \end{minipage}%
  \caption{Here are two figures side-by-side.}%
  \label{fig:1figs}%
\end{figure}
```

Further, if you place the caption inside the `\parbox` or `minipage`, then the width of the caption will be limited to the width of the parbox or minipage as shown in figures 2 and 3:

```
\begin{figure}%
  \centering
  \parbox{1.2in}{%
    ...figure code...
    \caption{First.}%
    \label{fig:2figsA}}%
  \qquad
  \begin{minipage}{1.2in}%
    ...figure code...
    \caption{Second.}%
    \label{fig:2figsB}%
  \end{minipage}%
\end{figure}%
```

For more information on typesetting figures and tables, see the document “Using Imported Graphics in LaTeX 2ε” by Keith Reckdahl [7].

---

[1] Section 4.6 describes how to add support for additional `float` environments.

[2] You might have to use the optional position arguments ‘[b]’ or ‘[t]’ if the figures are of different heights.

Figure 2: First. Figure 3: Second.

2 The User Interface

To use this package place

```
\usepackage[⟨options⟩]{⟨subfigure⟩}
```

in the preamble of your document. The supported options are shown in table 1.

`\subfigure`
`\subtable`

Within a `figure` or `table` environment, you can use the following commands to create a subfigure or subtable "box" with an optional subcaption underneath.

```
\subfigure[⟨list_entry⟩][⟨subcaption⟩]{⟨figure⟩}
\subtable[⟨list_entry⟩][⟨subcaption⟩]{⟨figure⟩}
```

If a subcaption argument is given (including the null subcaption '`[ ]`') then the subfigure is labeled with a counter formatted by the command '`\thesubfigure`' which returns, by default, '(a)', '(b)', etc. The counter used for labeling the subfigures is *subfigure* and is incremented for each subfigure regardless of whether a subcaption was printed. The internals of the `\subtable` command are symmetric to those of the `\subfigure` command. Further, if a List-of-Figures (or List-of-Tables) is generated, then the ⟨*list_entry*⟩ argument controls how the caption text is used there. Table 2 shows the possibilities.

If you wish to reference a specific subfigure or subtable, you can include a `\label` inside the body of either the ⟨*subcaption*⟩ or ⟨*figure*⟩ argument to the command (but not the ⟨*list_entry*⟩ argument). If supplied by itself, the ⟨*subcaption*⟩ is a "moving argument"[3] and, therefore, any "fragile" commands contained within it must be `\protect`'ed. If the ⟨*list_entry*⟩ argument is supplied, then the ⟨*subcaption*⟩ is not a "moving argument"; however, the ⟨*list_entry*⟩ is.

Note: since the `\subfigure` and `\subtable` commands have optional arguments, delimited with square brackets, before their required argument, you cannot use the ']' character at the top level of either the ⟨*subcaption*⟩ or ⟨*list_entry*⟩ argument. To overcome this problem, you must wrap the portion of the text containing the ']' character (or the entire argument), in a pair of curly brackets (see [8, § C.1.1] for more detail). For example:

```
\subfigure[This does not $sqrt[3]{8}$ work.]{... figure text ...}
\subfigure[This works {$sqrt[3]{8}$} fine.]{... figure text ...}
\subfigure[{This also works $sqrt[3]{8}$ fine.}]{... figure text ...}
```

One final note, the `\subfigure` and `\subtable` commands are actually identical and it is the surrounding environment that defines actually identical and it is the surrounding environment that defines whether a subtable or subfigure will be generated and not which command is used. At the user level, the choice of names is purely cosmetic (and historical). Therefore you can use `\subfigure` for any float (*e.g.*, `figure`, `table`, or other) environment.

---

[3] See [8, § 4.7 and § C.1.3] for a more detailed description of "moving arguments" and "fragile" commands.

Table 1: subfigure package options.

Option	**Description**
normal	Provides ‘normal’ subcaptions, this is the default.
hang, isu	Causes the label to be a hanging indentation to the subcaption paragraph. (isu is a synonym for hang.)
center	Causes each line of the paragraph to be separately centered. Overrides centerlast.
centerlast, anne	Causes the last line only to be centered. Overrides nooneline. (anne is a synonym for centerlast.)
nooneline	If a subcaption fits on one line it will, by default, be centered. This option treats a single line like a mid-line of a multi-line caption.
raggedright	Causes the subcaption text to be raggedright. Overrides center and centerlast.
scriptsize, footnotesize, small, normalsize, large, Large	Sets the font size of the subcaptions (both the label and the text), footnotesize is default.
rm, sf, tt, md, bf, up, it, sl, sc, RM, SF, TT, MD, BF,UP, IT, SL, SC	The lowercase commands set the font attributes of the subcaption label. The capitalized version sets the font attributes of the text. Family, shape and style attributes may be mixed. The default is to set the document defaults for the family, series and shape.
figbotcap, tabbotcap, FIGBOTCAP, TABBOTCAP	Sets the figure or table numbering based on the assumption that the figure or table caption comes after the subfigures or subtables. The capitalized version also places the subcaption after the figure (“FIGBOTCAP” and “TABBOTCAP” are the default settings).
figtopcap, tabtopcap, FIGTOPCAP, TABTOPCAP	Sets the figure or table numbering based on the assumption that the figure or table caption precedes the subfigures or subtables. The capitalized version also places the subcaption before the figure (“TABTOPCAP” is the preferred table setting, see section 2.3 for details).
loose, tight	The (default) loose option sets the historically normal whitespace around the subfloat. The tight option sets less space around the subfigure (this is the preferred setting).

Table 2: `\subfigure` calling arguments.

Subfigure Command	LoF/LoT	Subfigure Caption
`\subfigure`{. . . fig. . . }		
`\subfigure`[ ]{. . . fig. . . }	(b)	(b)
`\subfigure`[Subcaption.]{. . . fig. . . }	(c) Subcaption. . . .	(c) Subcaption.
`\subfigure`[ ][Subcaption.]{. . . fig. . . }		(d) Subcaption.
`\subfigure`[ ][ ]{. . . fig. . . }		(e)
`\subfigure`[List_entry.][Subcaption.]{. . . fig. . . }	(f) List_entry. . . .	(f) Subcaption.
`\subfigure`[List_entry.][ ]{. . . fig. . . }	(g) List_entry. . . .	(g)

2.1 Format Options

There are six options for formatting the layout of the caption label and text. The first is normal, which produces the style shown in figure 4. The other options may be used in various combinations to produce the layouts shown in figures 5 thru 17. Note that some combinations, like center and centerlast do not make sense since center overrides centerlast. Also, nooneline, when combined with either center or centerlast has no effect (unless the hang option is also set); and, raggedright overrides both center and centerlast.

2.2 Font Size and Style Options

There are twenty-four options for setting the font of the subcaption. The first six set the size of both the subcaption label and text. They are: scriptsize, footnotesize (default), small, normalsize, large, and Large. Their effect is shown in figure 18.

The next nine, rm: sf, tt, md, bf, up, it, sl, and sc, set the family, series or shape of the subcaption label, as shown in figures 19(a)– 19(i). The last nine: RM, SF, TT, MD, BF, UP, IT, SL, and SC, do the same for the text of the subcaption, as shown in figure 19(j)–19(r). These size and style options may be combined in 3456 ways to set the label and text of the subcaption (as long as the selected font combination exists!) The font family for the text and label may be set as roman (rm/RM), sans serif (sf/SF), and typewriter (tt/TT). These may be combined with those for the font series, medium (md/MD) and bold (bf/BF); and the font shape, upright (up/UP), italic (it/IT), slanted (sl/SL), and small caps (sc/SC).

2.3 Caption Position Options

There are eight options that control the *position* of the subcaption and how the subcaption *numbering* is related to the "containing" figure or table's caption. The following shows only the subfigure-related options, but the subtable options are symmetric.

The first option, figbotcap tells the subfigure command that the "containing" figure's `\caption` occurs **after** the subfigures. This information is needed to decide if the current figure counter shows the number for the last figure (figbotcap) or for the current one (figtopcap, see below).

The second option, figtopcap tells the subfigure command that the "containing" figure's `\caption` occurs **before** the subfigures. The subcaption is automatically placed below the figure for each of these options.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 4: Format option [normal].

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 5: Format option [nooneline].

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 6: Format option [centerlast]; centerlast overrides nooneline.[4]

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 7: Format option [center]; center overrides nooneline and centerlast.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 8: Format option [hang].

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 9: Format options [hang,nooneline].

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 10: Subfigure format options [hang,centerlast].

---

[4] So this is the same as [centerlast,nooneline]. Only the shortest number of options to achieve an effect is shown. Adding any combination of overridden options has no effect.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 11: Format options [hang,centerlast,nooneline].

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 12: Format options [hang,center]; center overrides centerlast.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 13: Format options [hang,center,nooneline]; center overrides centerlast.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 14: Format option [raggedright]; raggedright overrides center and centerlast.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 15: Format options [raggedright,noonelirne]; raggedright overrides center and centerlast.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 16: Format options [hang,raggedright]; raggedright overrides center and centerlast.

(a) Short.

(b) This is a long caption of gibberish: Aafas d taed gaihgghn adf irin sadf aets etsd na.

Figure 17: Subfigure format options [hang,raggedright, nooneline]; raggedright overrides center and centerlast.

(a) Font option [scriptsize].

(b) Font option [footnotesize].

(c) Font option [small].

(d) Font option [normalsize].

(e) Font option [large].

(f) Font option [Large].

Figure 18: Subfigure font size options. (Default: footnotesize).

(a) Subcaption label font option [rm].

(b) Subcaption label font option [sf].

(c) Subcaption label font option [tt].

(d) Subcaption label font option [md].

(e) Subcaption label font option [bf].

(f) Subcaption label font option [up].

(g) Subcaption label font option [it].

(h) Subcaption label font option [sl].

(i) Subcaption label font option [sc].

(j) Subcaption font option [RM].

(k) Subcaption font option [SF].

(l) Subcaption font option [TT].

(m) Subcaption font option [MD].

(n) **Subcaption font option [BF].**

(o) Subcaption font option [UP].

(p) *Subcaption font option [IT].*

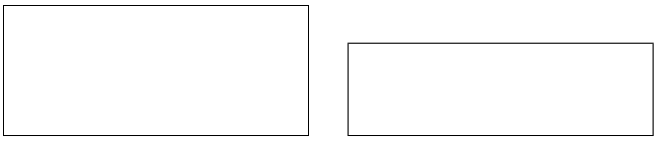
(q) *Subcaption font option [SL].*

(r) Subcaption font option [SC].

Figure 19: Subfigure font style options. (Default: rm,md,up,RM,MD,UP). Note: The above single options are loaded after the default settings and multiple options are allowed, see the text, section 2.2.

The third option, FIGBOTCAP, is similar to figbotcap, except that it also forces the subcaption to be placed under the figure. This is the default setting for figures (and TABBOTCAP for tables) and is shown in figure 20.

The fourth option, FIGTOPCAP, is similar to figtopcap, except that it forces the subcaption to be placed above the figure box. (While not the default, it is the preferred format for tables, which uses the option TABTOPCAP.) An example of this option is shown in figure 21. Note that the baseline of the subfigure is along the top of the two subfigures. See section 4.5 for another way of positioning the caption when captions are placed above the figure or table.

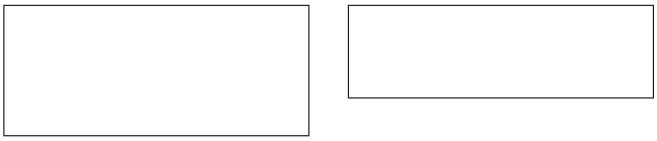


(a) First caption.

(b) Second long, long, long, long, long, long, caption.

Figure 20: Subcaption position option [FIGBOTCAP].

Figure 21: Subcaption position option: [FIGTOPCAP].



(a) First caption.

(b) Second long, long, long, long, long, long caption.

2.4 Recent Changes and Backward Compatibility

There are some significant changes in this version of the subfigure package. One of them was to pack the subfigure tighter together by removing the space at the top of the subfigure at the beginning of a page, or minipage, and to reduce the spacing around the subfigure, see table 3. If you have been using an older version of the subfigure package (*i.e.*, version 2.0 or eariler) than the default (loose) setting will not cause any change in your existing documents.[5]

If you want to use the new and preferred, tight option, your subfigures will take up less space and should provide a more balanced visual appearance for your paper.

The second significant change is the ability to make the text on the List-of-Figures page different than that in the subcaption. The use of a second optional argument to the `\subfigure` command is shown in table 2. This should not cause any compatibility problems.

The third significant change is the it is now possible to have the captions and subcaptions come before or after the corresponding figure/table portion. While the default settings support the old view of the subcaptions following the figure/table and, in turn, followed by the main caption. The preferred format is for figures to retain that layout and for tables to have both their subcaption and main captions come before the table portion. This preferred setting may be specified by adding the option TABTOPCAP when loading the subfigure package.

The fourth update is that the font style options have been generalized so that an option from each of the family, series and shape, may be combined, as long as that combination exists; and you can set the font of the label and text separately. In addition, the the `\space` that separated the label from the text in the subcaption has been replaced with a horizontal skip of `\subfiglabelskip` which has the default value of `0.33em plus 0.07em minus 0.03em`. This extension should not cause any compatibility problems.

The last major change is that there is now a `\subref` command that allows a reference to the subfigure without the figure number. An example of the use of this command is shown later in section 3.3. Associated with this change is

---

[5] If you have been using a beta release version of subfigure 2.1, than you will need to use the tight option in order to maintain the "look-and-feel" that you are used to.

Table 3: Subfigure spacing changes.

subfigure Constant	Old (v2.0) Value	**loose** Option	**tight** Option
`\subfigtopskip`	10 pt	10 pt	5 pt
`\subfigcapskip`	10 pt	10 pt	0 pt
`\subfigcaptopadj`	—	0 pt	3 pt
`\subfigbottomskip`	10 pt	10 pt	5 pt
`\subfigcapmargin`	10 pt	10 pt	0 pt
`\subfiglabelskip`	—	0.33 em	0.33 em plus 0.07 em minus 0.03 em

that the `\label` command will accept an optional argument, for use with the hyperref package, when used within the scope of the `\subfigure` or `\subtable`, see section 4.7.3 for details.

The subfigure package checks for and loads a configuration file called `subfigure.cfg` which is placed anywhere that LaTeX will look for classes or packages (see section 4). By default, the `subfigure.sty` file tries to look unchanged from older versions. In order to have it automatically use the preferred settings, you can add a configuration file containing the options tight to reduce the extra whitespace around the subfigures and TABTOPCAP to show that table captions will come before the table and the the subcaptions for tables should be set above the subtable. The following line is all you need in your configuration file:

```
\ExecuteOptions{tight,TABTOPCAP}
```

You could also load the subfigure package with the options with the following in your LaTeX preamble:

```
\usepackage[tight,TABTOPCAP]{subfigure}
```

2.5 Frequently Asked Questions

The four most frequently asked questions about the subfigure package are:

1. "My subfigures are not aligned along their bottoms. Why?"

   Remember! The subfigure package aligns subfigure along their baselines with the subcaption (if any) sticking out above or below. The above problem is usually due to using a `minipage`, `tabular` or `array` environment that, by default, places the baseline at the center of the box that it generates. If the two subfigures are different sizes, or if one subfigure is generated in some other way with its baseline not at the expected place (perhaps an `\includegraphics`), then the subfigures will be misaligned. One solution is to use the environment options '[t]' or '[b]' to move the baseline to the top or bottom rather than the center.

2. "How can I get my figures/subfigures to line up the way I want?"

   A similar question, but this one is caused by extra whitespace in the source text generating spaces next to the figures, and `\par`'s generated by blanklines. The main thing is *be aware* that extra whitespace can move figures and subfigures around, sometimes a lot and sometimes just a little so that they look “wrong”. Placing too many ‘%’s at the end of the lines is better than too few in the figure and table environments. (See the discussion of “white space” in section 3.)

3. “I have too many subfigures for one page, How can I spread them over two or more pages and continue the numbering?”

   Option 1: Adjust the *figure* and *subfigure* counters (or the *table* and *subtable* counters) as needed before and after each figure (or table) See, for example *Using Imported Graphics in LaTeX 2ε* [7, § 30].

   Option 2: Use the ccaption package by Peter Wilson [3].

   Option 3: Use the simpler captcont package by Steven Cochran [6].

   All of these options work well. Of the packages, the ccaption package is bigger and offers more control over what is done (and things to do) at the expense of being a little harder to use. The captcont package is easier to use, but only provides for continued floats.

4. “Why do I get a garbled caption or an error when I use square brackets?”

```
\subfigure[SHIFT: ‘‘register[3] $<<=$ 3;’’]{... figure text ...}
```

Since the `\subfigure` and `\subtable` commands have optional arguments, delimited with square brackets, before their required argument, you cannot use the ‘]’ character at the top level of either the ⟨*subcaption*⟩ or ⟨*list_entry*⟩ argument. To overcome this problem, you must wrap all or the portion of the text containing the ‘]’ character, in a pair of curly brackets (see [8, § C.1.1] for more detail). For example:

```
\subfigure[SHIFT: ‘‘register{[3]} $<<=$ 3;’’]{... figure text ...}
```

or

```
\subfigure[{SHIFT: ‘‘register[3] $<<=$ 3;’’}]{... figure text ...}.
```

3 Three Examples

The easiest way to show the use of this package is to give a few examples. The two most important things to remember when working with the subfigure package are that (1) the subfigures are aligned along their baselines (see figure 25 and section 4.1) and (2) that whitespace in the `figure` environment are significant and affect the layout.

The baseline of the subfigure is usually at the bottom of the subfigure or (when the subcaption appears at the top) at the bottom of the subcaption *and* the `\subfigcapskip` space—which is usually the same as the top of the subfigure. However sometimes, especially when using the `tabular`, `array`, or `minipage` environments to build the figure, the baseline appears elsewhere. The above environments are all aligned at their center by default but that may be changed with the optional ‘`[t]`’ or ‘`[b]`’ arguments. As a last resort you can wrap all of your figures in a `\vtop` box with a `\vbox to 0pt{\null}` at the top followed by the figure.

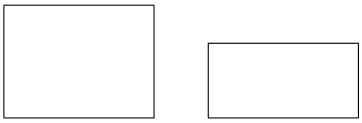
If your figure is not quite centered or where you want it to be, the problem is often a space character being placed to one side or the other of the figure. Some general rules of thumb are:[6]

- Two end-of-lines following each other (ignoring any whitespace) are turned into a `\par` or paragraph break.
- Multiple whitespace (including the end-of-line) are compressed into a single space.
- The spaces after a macro command name (*e.g.*, `\foo`) are ignored.
- A ‘%’ character at the end of the line suppresses the end-of-line and all of the spaces (if any) at the beginning of the next line.

To suppress significant extra whitespace, you can add some ‘%’ characters at the end of each line that doesn’t end with a command name. This is more than is required, but extra ‘%’ usually don’t cause a problem.

---

[6] See chapters 7 and 8 of “The TeXbook” [9] for details.



(a) First.

(b) Second figure.

(c) Third.

Figure 22: Three subfigures.

Figure 22 contains two top ‘subfigures’ and figure 22(c).[7]

The other case where things are not correctly centered is when the subfigure uses only the label for the subcaption. This is often the case when the description for each subfigure is given in the figure caption rather than in each subcaption. In this case, the default label has the form ‘(a) ’ where the trailing space is defined by `\subfiglabelskip`. In this case the style should redefine this space as ‘0pt’ so that the label is perfectly centered (see section 3.3, below for an example).

3.1 A Simple Example

`\subfigure` The first example, shown in figure 22, specifies `\centering` to horizontally center the set of subfigures, and uses `\\` and some horizontal space (using `\qquad`) to control the placement of the subfigures. Note that the alignment of the top two subfigures is along the bottom of the figure portion of each.

```
\begin{figure}%
  \centering
  \subfigure[First.]{...}\qquad
  \subfigure[Second figure.]{...}\\
  \subfigure[Third.]{\label{3figs-c}...}%
  \caption{Three subfigures.}
  \label{3figs}
\end{figure}
...
Figure~\ref{3figs} contains two top ‘subfigures’ and
figure~\ref{3figs-c}.
```

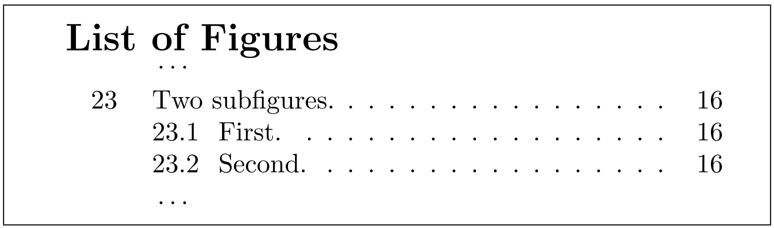
3.2 A More Advanced Example

A second example, shown in figure 23, demonstrates how to change the subfigure labels and have the subfigure captions printed in the List-of-Figures.

The first `\renewcommand` changes the reference to `\thesubfigure` to return both the figure number and the subfigure number separated with a period. The

---

[7] In this and later boxed figures, the boxes are intended to represent a portion of the page in which the figure occurs. This is usually to show the figure along with some text or to show the effect of some option on multiple pages.



List of Figures

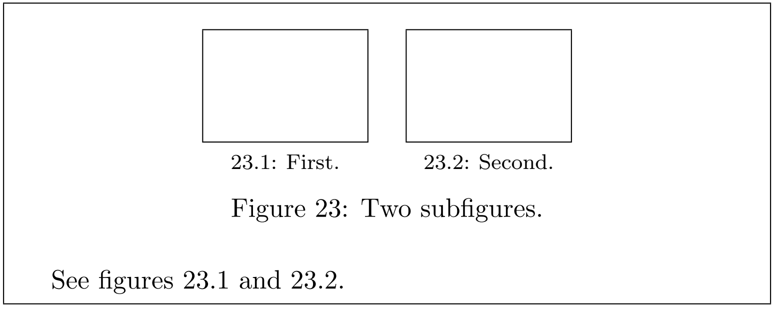
. . .

23 Two subfigures. 16

23.1 First. 16

23.2 Second. 16

. . .



23.1: First. 23.2: Second.

Figure 23: Two subfigures.

See figures 23.1 and 23.2.

next two `\renewcommand`'s turn off the `\p@subfigure` (since it is now included in `\thesubfigure` and adds the colon and space to the subfigure label. Later in the file, the *lofdepth* is set to "2" so allow the subfigure captions to show and the `\listoffigures` is loaded. Finally, the `figure` is defined and a little following text is given that refers to it.

```
\renewcommand{\thesubfigure}{\thefigure.\arabic{subfigure}}
\makeatletter
  \renewcommand{\p@subfigure}{}
  \renewcommand{\@thesubfigure}{\thesubfigure:\hskip\subfiglabelskip}
\makeatother
...
\setcounter{lofdepth}{2}
\listoffigures
...
\begin{figure}%
  \centering
  \subfigure[First.]{%
    \label{fig:first}%
    ...}%
  \qquad
  \subfigure[Second.]{%
    \label{fig:second}%
    ...}%
  \caption{Two subfigures.}
\end{figure}
...
See figures~\ref{fig:first} and \ref{fig:second}.
```

3.3 An Example Without Subcaption Text

`\subref` `\Subref` The last example, shown in figure 24, demonstrates a commonly required format where the subfigure are just labeled and the description occurs in the main caption. This is easy to do by using the "empty" optional caption arguments "[ ][ ]". This

List of Figures

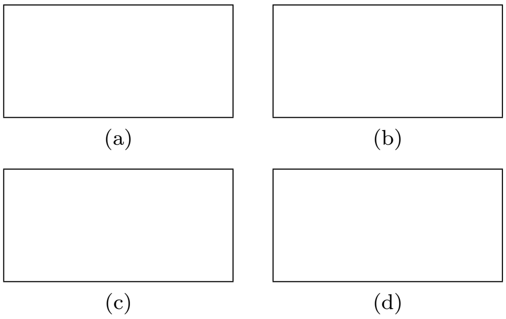

(a) (b)
(c) (d)

Figure 24: A set of four subfigures: (a) describes the first subfigure; (b) describes the second subfigure; (c) describes the third subfigure; and, (d) describes the last subfigure.

The text references the main figure as figure 24 or part of it as figures 24(a)–(c).

creates a label for the subfigure in the text, but it does not show on the List-of-Figures page. However, by default the caption may not be perfectly centered, so `\subfiglabelskip` is reduced to zero points to ensure that there is not any extra space hidden in the subcaption. To refer to the subfigure label within the text or the main caption, you can use the `\subref` command, which is similar to the `\ref` command, but does not carry the figure number. (The `\Subref` command is the same but sets it with `\subcaplabelfont`).

```
\subfiglabelskip=0pt
...
\listoffigures
...
\begin{figure}%
  \centering
  \subfigure[][]{%
    \label{fig:ex3-a}%
    ...figure code...}%
  \hspace{8pt}%
  \subfigure[][]{%
    \label{fig:ex3-b}%
    ...figure code...}\\
  \subfigure[][]{%
    \label{fig:ex3-c}%
    ...figure code...}%
  \hspace{8pt}%
  \subfigure[][]{%
    \label{fig:ex3-d}%
    ...figure code...}%
  \caption[A set of four subfigures.]{A set of four subfigures:
           \subref{fig:ex3-a} describes the first subfigure;
```

```
          \subref{fig:ex3-b} describes the second subfigure;
          \subref{fig:ex3-c} describes the third subfigure; and,
          \subref{fig:ex3-d} describes the last subfigure.}%
  \label{fig:ex3}%
\end{figure}
...
The text references the main figure as figure~\ref{fig:ex3} or part
of it as figures~\ref{fig:ex3-a}--\subref{fig:ex3-c}.
```

4 Customization

The following sections describe the internal parameters used by the subfigure package to define the layout of the subfigures or tables, as well as the labels and captions the accompany them. In addition, adjustments to the entries on a "List-of" page and the addition of new `float` environments are described.

Adjusting these values allows extensive customization of the subfigure package. If you want to customize the package, an alternative to actually changing the code is to create a a file called `subfigure.cfg` and place it anywhere that LaTeX will look for classes or packages. Any changes placed in the file will affect the predefined parameters and you can override the default settings. Any user options will be processed after this file is loaded.

In order to change the major commands in the subfigure package with this configure file, you will need to use the `\AtEndOfPackage` command to defer that portion of your changes until the end of the package.

4.1 Changing the Layout

The layout of the `\subfigure` or `\subtable` is defined by several internal values which may be changed to customize appearance of the object. The following illustration shows the relationship of these values. Figure 25(a) shows the standard layout with the caption following the figure. The figure is vertically centered with `\subfigtopskip` of space added above, then `\subfigcapskip` of space is added below the figure followed by the subcaption and, finally, `\subfigbottomskip` of space added at the bottom. The baseline is located at the bottom of the figure. It is along this baseline that adjacent subfigure boxes are aligned. Figure 25(c) shows the case where the caption precedes the figure (*ie.*, `\subfiguretopcaptrue` or `\subtabletopcaptrue`). In this case the various boxes and glue are reversed,[8] except that the `\subfigcapskip` is increased by `\subfigcaptopadj`. The other two cases, figures 25(b) and (d), show the cases where there is no caption. Note that the `\subfigcapskip` is left out when there is no caption. Note also, for all of these cases, that the space at the top of the subfigure is automatically removed for items that are the first box in a vertical list or other than the first box in a horizontal list. This allows tighter packing of the subfigures and the full use of the page or `minipage`.

---

[8] The `\subfigtopskip` and `\subfigbottomskip` actually follow the **figuretopcap** and **tabletopcap** flags, so that the actual top spacing used is `\subfigtopskip` when the flags are **false** and `\subfigbottomskip` when they are **true**.

Each of these values `\subfigtopskip`, `\subfigcapskip`, and `\subfigbottomskip`; as well as `\subfigcapmargin` and `\subfiglabelskip` (the latter not shown in figure 25) may be changed from their defaults (see table 4) to adjust the subfigure for the current layout style. In addition, they may all assume negative values, which in some cases may solve problems with the layout. Even though these constants are “skips”, only the last two (`\subfigcapmargin` and `\subfiglabelskip`) will shrink or expand since the others assume their natural size in the subfigure box and are fixed at that size.

4.2 Adjusting the Subcaption

`\subref` `\Subref`

The subfigure label has three forms. The first is the one that appears in the text when you use the `\ref` command; the second is the one that appears on the List-of-Figures page and may be used to reference individual subfigures within the figure and subfigure captions, using the `\subref` or `\Subref` commands; and the third is the fully formatted version used under the subfigure as the label part of the caption.

The `\ref` command yields the string, saved by the `\label` command, composed by concatenating the value of `\p@subfigure` to `\thesubfigure`. By default these are defined by: “`\thefigure`” and “`(\alph{subfigure})`”, respectively, which produces a reference of the figure number followed by the subfigure letter in parentheses.

The label used on the List-of-Figures page may be retrieved with the `\subref` command (this value is saved by the `\label` command when the `\label` command is used within the scope of the subfigure. This is the string defined by `\@@thesubfigure`, which, by default, is the value “`\thesubfigure`” (or “`(\alph{subfigure})`”).

The label used with the subcaption text is defined by the internal value `\@thesubfigure`, which, by default, has the value

“`\thesubfigure\hskip\subfiglabelskip`”.

It is prefixed by `\subcapsize\subcaplabelfont` and followed by the subcaption text which is set with `\subcapfont`.

Note that by default `\subcaplabelfont` has the default value “`{\familydefault \seriesdefault\shapedefault}`”. The package options described in table 1 allow you to set these values for your paper. If you update the `\@subfigure` command, you should include any separator character or spacing between the label and the start of the subcaption text. The default is `\hskip\subfiglabelskip` placed after the label.

Finally, the text of the subcaption is prefixed by `\subcapfont` which may be changed using the set of nine lower-case font options described in table 1.[9] One other way of changing the layout of the lapel and caption is by replacing the `\@makesubfigurecaption` or `\@makesubtablecaption` command (which by default are identical).

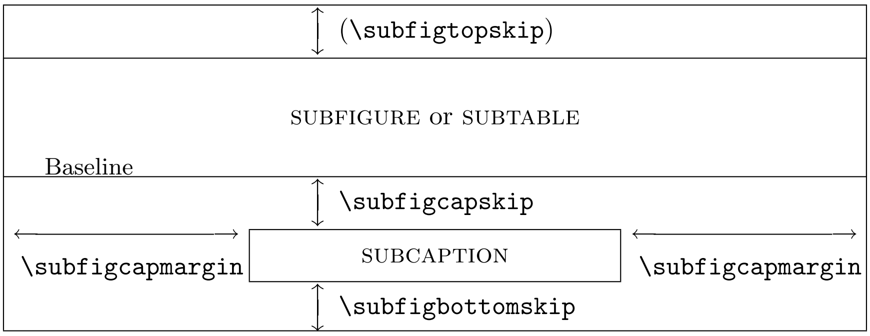
4.3 Adjusting the Subfigure and Subtable Counters

`\c@figure` `\c@table`

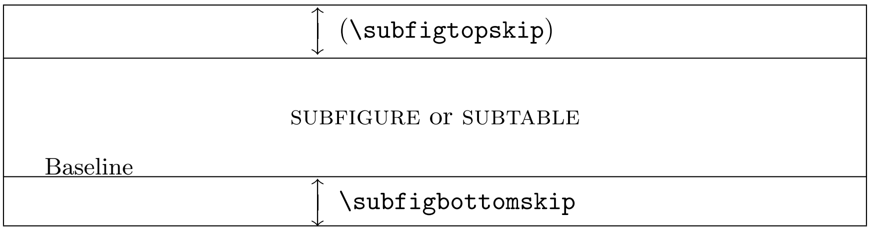
To create some special effects, such as continuing the subfigure numbering across

---

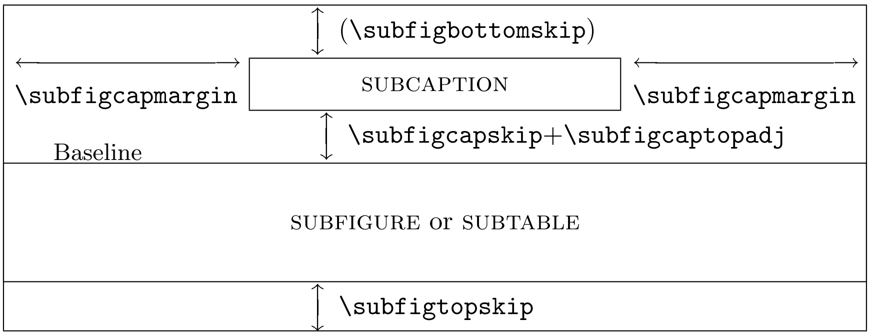
[9] It is also prefixed by `\subcapsize` as part of the overall label and caption.



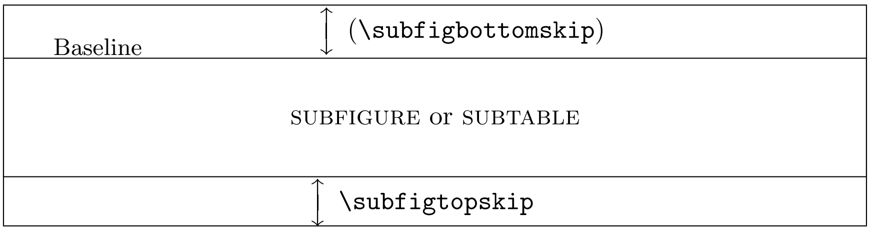
(a) Standard layout [FIGBOTCAP] or [TABBOTCAP].



(b) Standard layout [FIGBOTCAP] or [TABBOTCAP] with no caption present.



(c) Reversed layout [FIGTOPCAP] or [TABTOPCAP].



(d) Reversed layout [FIGTOPCAP] or [TABTOPCAP] with no caption present.

Figure 25: Subfigure and subtable layout.

several pages as part of one long continued figure, you can back off the number change from a caption with the command:

`\addtocounter{`*figure*`}{-1}`

within the `figure` environment. In addition, you can adjust for previous subfigures or subtables with one of the following (here we assume that two subfigures or subtables appeared in the previous pages:

`\addtocounter{`*subfigure*`}{2}`

or

`\addtocounter{`*subtable*`}{2}`.

Two other things that may be necessary, if you switch between figures and tables in the same `figure` environment (*e.g.*, by changing `\@captype`, see section 4.7.2), is to add the command `\listsubcaptions` following the last subfigure when the subfigure is using the TOPCAP or topcap option. This is necessary to flush the list of subcaptions before the next `subfigure` or `subtable`. This also may be necessary if you switch between `\figuretopcaptrue` and `\figuretopcapfalse`. The other thing that may be required in some cases, is to reset the subfigure counter by entering:

`\setcounter{`*subfigure*`}{0}`

This should only be necessary if you are dynamically switching between different subfigure options, or changing the `\@captype`, within a float environment.

4.4 Modifying the List-of-Figures and List-of-Tables

`\l@subfigure`
`\@dottedxxxline`

To generate a List-of-Figures, or List-of-Tables, page you need to add a `\listoffigures` or `\listoftables` command where you want the list to appear. These commands also cause the appropriate captions and subcaptions to be written to a file with the extensions `lof` (`lot`). If you want the subcaption text to appear in the List-of-Figures or List-of-Tables page, you need to change the value of the counter *lofdepth* (*lotdepth*) counter from its default of '1'. For example, to have the subfigure subcaptions to appear on the List-of-Figures, add the following to the preamble of your paper:

`\setcounter{`*lofdepth*`}{2}`

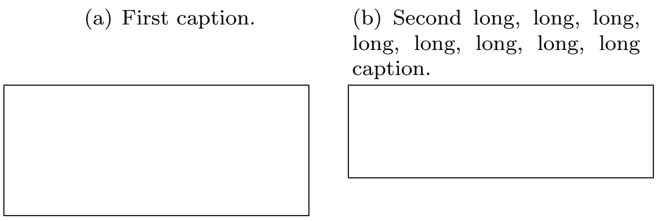
If you want to change how the subcaption appears on the "List-of" pages you can change its format by redefining the `\l@subfigure` or `\l@subtable` command. Usually you will want to use the `\@dottedxxxline` command (section 5.9, page 41) to help with the formatting. For instance the default value of `\l@subfigure` is:

```
\newcommand{\l@subfigure}{%
  \@dottedxxxline{\ext@subfigure}{2}{3.8em}{2.5em}}
```

The arguments of the `\@dottedxxxline` command are:

1. Type. The usual values are : `lof` or `lot`. The internal values `\ext@subfigure` and `\ext@subtable` stand for these extensions.

Figure 26: Caption position option: [figtopcap] with changing settings of `\subfiguretopcap`.



(a) First caption.

(b) Second long, long, long, long, long, long, long, long caption.

2. Level. By default this is ‘2’ for the *subfigure* and *subtable*. If the level is greater than ⟨*Type*⟩*depth* (where ⟨*Type*⟩ is the first argument, above), then no line is produced.
3. Indent. Total indentation from the left margin.
4. Numwidth. Width of box for the label number if the Title has a `\numberline` command. This is also the amount of extra indentation added to second and later lines of a multiple line entry.
5. Title. Contents of entry (*e.g.* the ⟨*list_entry*⟩ or ⟨*subcaption*⟩).
6. Page. The page number of the figure or table.

The final two arguments, title and page, are automatically appended to the value of `\l@subfigure` (and symmetrically for `\l@subtable`).

For example, to change the amount of space reserved for the label (if, for instance, you have a lot of figures and the and you need extra space for the figure number) you could widen the `2.5em` space for the label to `4.0em`:

```
\makeatletter
  \renewcommand{\l@subfigure}{%
    \@dottedxxxline{\ext@subfigure}{2}{3.8em}{4.0em}}
\makeatother
```

4.5 Aligning Captions Above the Figure

`\subfiguretopcaptrue`
`\subfiguretopcapfalse`
`\figuretopcaptrue`

For unbalanced sets of captions placed, above the figures or tables, the caption portion looks unbalanced, such as the ones in figure 21. If you prefer to have the caption portion aligned along the top rather than the bottom, as shown in figure 26, you can use the figtopcap option as in figure 21, but use two “empty” subfigures to position the captions followed by two more containing the figures, but without captions. The code to produce this example is:

```
\begin{figure}%
\centering
  \caption{Caption position option: [\Lopt{figtopcap}] with changing
          settings of \subfiguretopcap.}%
  \label{fig:position3}%
  \subfiguretopcapfalse
  \subfigure[First caption.]{\hbox to 1.5in{\hfil\null}}%
  \hspace{0.2in}%
  \subfigure[Second long, long, long, long,
            long, long, long, long caption.]{%
```

```
    \hbox to 1.5in{\hfil\null}}\\[5pt]
  \subfiguretopcaptrue
  \subfigure{\fbox{\hbox to 1.5in{\vbox to 15mm{\vfil\null}\hfil}}}%
  \hspace{0.2in}%
  \subfigure{\fbox{\hbox to 1.5in{\vbox to 10mm{\vfil\null}\hfil}}}%
\end{figure}
```

This example makes use of one of the four flags that control how the caption labels are numbered and where the subcaption appears. Two are for `\subfigure` and two for `\subtable`. The first of each set tell the `\subfigure` or `\subtable` command that the related main caption appears before or after the set of subfloats. `\figuretopcaptrue` and `\tabletopcaptrue` indicate that the caption appears before and `\figureotopcapfalse` and `\tabletopcapfalse` indicate that the it appears after. The other two flags force the subcaption to be placed before (`\subfiguretopcaptrue` and `\subtabletopcaptrue`) or after the actual subfigure or subtable (`\subfiguretopcapfalse` and `\subtabletopcapfalse`).

There are two difficulties with this approach, first, you need to keep changing the setting of `\subfiguretopcap`, and second, if you have more than one row of subfigures or subtables, then you will need to fiddle with the counter (see section 4.3 to keep the numbers straight. The reason that this format is not supported by the subfigure package is that you need information about all of the subfigure or subtables on a row to box the figures correctly and this information is not available locally.

4.6 Adding Subfloats to New Environments

It is easy to add a `subfloat` command to a new environment. For instance, let us assume we have a new `float` environment[10] called "`map`" in which various maps are displayed and for which a List-of-Maps is to be generated in the contents section. If we wanted to have `submaps`, then we could define the following:

```
\makeatletter
  \newcounter{submap}[map]
  \newif\ifmaptopcap
  \newif\ifsubmaptopcap
  \newcommand{\p@submap}{\themap}
  \newcommand{\thesubmap}{(\alph{submap})}
  \newcommand{\@thesubmap}{\themap\hskip\subfiglabelskip}
  \newcommand{\@@thesubmap}{\themap}
  \newcommand{\ext@submap}{\ext@map}
  \newcommand{\l@submap}{\@dottedxxxline{\ext@submap}{2}{3.8em}{2.5em}}
  \newcounter{lomdepth}
  \setcounter{lomdepth}{1}
  \newcommand{\submap}{\subfigure}
  \newcommand{\@makesubmapcaption}{\@makesubfigurecaption}
  \ifhyperrefloaded
    \newcommand\theHsubmap{\themap.\arabic{submap}}
    \newcommand{\toclevel@submap}{1}
  \fi
\makeatother
```

---

[10] For information on creating new float environments, see any of the following: [10], [3] or [11].

The first and last lines make the character '@' act like a letter between them and therefore it may be part of a command name used there. First a new counter for the `submap` is created along with two conditionals that define where the position (i.e., above or below) of the main caption and subcaption is with respect to the submap. Then the four commands that define the `submap` label are created. The first two, `\p@submap` and `\thesubmap` define the standard label returned by `\ref`. The next, `\@thesubmap` gives the label as shown under or over the submap with the subcaption and the last, `\@@thesubmap` shows how the label is displayed on the List-of-Maps and/or referenced with the `\subref` command.

The next four lines show where and how to print to the List-of-Maps page: `\ext@submap` gives the List-of-Maps file extension; `\l@submap` shows how to print the submap line on the List-of-Maps page; and the last two lines show how to create and set the counter *lomdepth*, which controls how many caption levels are shown on the page when it is printed.

The next two lines create the `\submap` and `\@makesubmapcaption` commands by making them the same as the `\subfigure` and `\@makesubfigurecaption` commands.

The last four lines conditionally create the `\theHsubmap` and `\toclevel@submap` commands which are used by the hyperref Package to name the item and to control the presence of the item bookmark.

Of course, this is where the ccaption package [3] comes in handy since it will do all of the above with one command:

```
\newsubfloat{map}
```

4.7 Interaction with Other Parts of LaTeX

In the following sections, the interaction of the subfigure package with other parts of LaTeX is documented. These "other parts" may be either part of the the LaTeX base or contributed packages or classes.

4.7.1 TeX's "Mouth"

The most important thing to remember when laying out your figures within a `float` environment is that spaces take room. If you have an extra space between two figures, then they will be separated by a little bit.

TeX's state varies as it reads a line of text from a file. It ignores some spaces and carriage-returns and converts others to `\space`'s or `\par`'s. You can use a '`%`' to insure that you only have real spaces where you want them. To understand which spaces are significant, you should read chapters 7 and 8 of the TeXbook [9]. However, the main source of unexpected extra spacing is carriage-returns which are turned in to `\space`'s. As a general rule: if in doubt, then add a '`%`' immediately after the last significant character of the line.

4.7.2 The Float Environment

Although the subfigure package was designed to work within a `float` environment (*e.g.*, `figure` or `table`), it can be used outside with the following two caveats:

1. You need to define `\@captype`. This is usually either `figure` or `table`. For example add the following to the preamble of your document:

   ```
   \makeatletter
     \newcommand{\change_cap_type}[1]{%
       \renewcommand{\@captype}{#1}}
   \makeatother
   ```

   Then use the new command to switch in the middle of a given `float` environment, say from “figure” to “table” with the command `\change_cap_type{table}`.

2. If you want to define references using `\label`, then you also need to redefine the LaTeX internal `\@currentlabel`. For example:

   ```
   \makeatletter
     \edef\@currentlabel{\p@subfigure\thesubfigure}
   \makeatother
   ```

   before using the `\label` command. NOTE: Many other commands change `\@currentlabel`, including all of the “section” commands, `\caption`, `equation`’s, and `theorem`’s.

4.7.3 Interaction with Other Packages

The only packages that directly interact with the subfigure package are the caption/caption2 packages by H.A. Sommerfeldt [1, 2], the ccaption package and tocloft packages by Peter Wilson [3, 4], and the captcont package by S.D. Cochran [6].

caption If you load the subfigure package **before** the caption package, then the caption package will detect that fact and will change the `\subcapsize` when the options scriptsize, ..., Large are specified (overriding such options used when loading the subfigure package). In addition, it redefines `\@thesubfigure` and `\@thesubtable` to use `\captionlabelfont`. It also uses an older layout of `\@thesubsubfigure` and `\@thesubtable`.

The best plan is to load the caption package **before** the subfigure package. In addition, you should try to coordinate the “look and feel” of the two packages. This limits you a little since, although the two packages have similar options, the options in the caption package do not combine the same way. You can pick one from each column:

normal, hang, center, centerlast	nooneline	scriptsize, footnotesize, small, normalsize, large, Large	up, it, sl, sc, md, bf, rm, sf, tt	(Other options—see package.)

caption2 This package acts similarly to the caption package. If you specify the subfigure it will try to support the subfigure package, if you specify nosubfigure than it will not. If neither option is specified, than load order matters. If loaded **before** the subfigure package, than it will not try to support the package and if loaded **after** it will.

Again, the best plan is to load the caption2 package **before** the subfigure package, and specify the nosubfigure option. In addition, you should try to coordinate the "look and feel" of the two packages. This limits you a little since, although the two packages have similar options, they are not exactly the same; however, most of the good looking combinations are easily available. You can pick one from each column:

nosubfigure, subfigure	normal, hang, center, centerlast flushleft, indent	oneline, nooneline	scriptsize, footnotesize, small, normalsize, large, Large	up, it, sl, sc, md, bf, rm, sf, tt	(Other options—see package.)

ccaption The ccaption package provides for all sorts of extensions and style options for `float` `\captions`. It also provides for the use of the `\caption` command outside of a `float` environment and a mechanism for creating new types of `float` environments.

In order to use it with the subfigure package, you need to pass the subfigure option when loading it:

```
\usepackage[subfigure]{ccaption}
```

tocloft The tocloft package gives the user the ability to easily configure the "List-of" pages. It takes a subfigure option so it doesn't matter which package is loaded first.

```
\usepackage[subfigure]{tocloft}
```

hyperref The hyperref package extends the functionality of all of the LaTeX cross-referencing commands to produce hypertext links. In addition, it provides new commands to allow the user to insert hypertext links. When used with the subfigure package, they may be loaded in any order; however, it might be better if the subfigure package is loaded **first**.

`\subfloat@label`

To more fully support the hyperref package, the `\label` command, when used within the scope of the `\subfigure` or `\subtable` commands takes an optional argument (note the parentheses rather than square brackets):

`\label(`⟨*bookmark*⟩`){`⟨*key*⟩`}`

We would like to use the subcaption as the bookmark text, but the `\label` command is often processed before the subcaption. Therefore, this optional argument may be used to supply this information if desired. By default a bookmark field of the form "Subfigure␣1(a)" will be generated. [11]

`\caption`
`\caption*`
`\captcont`
`\captcont*`

captcont This package may be used with or without the subfigure package to extend figure or table numbering across multiple pages. This package knows about

---

[11] If the document class is report or other class that defines `\thechapter`, than the default bookmark field will be of the form "Subfigure␣1 ␣1(a)".

```
   This text  should be
verbatim.     And   not
 messed with in any way  !
```

(a) First subcaption.

```
   This text (also)should be
verbatim.     And   not
 messed with in any way  !
```

(b) Second subcaption.

Figure 27: Subfigures (a) and (b) show examples of using verbatim text in a subfigure.

how the subfigure package interacts with the List-of-Figures and List-of-Tables and does the right thing when used with `subfigure`'s and `subtable`'s.

The captcont package may be loaded either **before** or **after** the subfigure and it has four options: figbotcap or figtopcap and tabbotcap or tabtopcap. These are the same as the subfigure options. When the captcont package is used with the subfigure package, only the subfigure options matter. Any given with the captcont package are ignored.

The thing to remember about the captcont package is that if you normally place the `\caption` **before** your subfigures or subtables (*i.e.*, figtopcap or tabtopcap respectively), then you start a series of continued `figure`'s with the `\caption[*]` and use `\contcapt[*]` in each of the rest of the figures. If you **follow** your subfigures or subtables with a caption (*i.e.*, figbotcap or tabbotcap respectively), then you start the series with the `\contcapt[*]` in the first `figure` environment and all but the last where you use the `\caption[*]`.

4.7.4 Creating a `subfigure` Environment

`subfloat` Some people have wanted to use the `verbatim` environment within the `\subfigure` command and run into the restriction that the `verbatim` environment cannot be nested. To include verbatim text in a subfigure, you can define a new environment, in which verbatim text may be enclosed, and which calls the `\subfigure` command.

```
\newbox\subfigbox             % Create a box to hold the subfigure.
\makeatletter
  \newenvironment{subfloat}%  % Create the new environment.
    {\def\caption##1{\gdef\subcapsave{\relax##1}}%
     \let\subcapsave=\@empty  % Save the subcaption text.
     \let\sf@oldlabel=\label
     \def\label##1{\xdef\sublabsave{\noexpand\label{##1}}}%
     \let\sublabsave\relax    % Save the label key.
     \setbox\subfigbox\hbox
       \bgroup}%              % Open the box...
      {\egroup                % ... close the box and call \subfigure.
     \let\label=\sf@oldlabel
     \subfigure[\subcapsave]{\box\subfigbox}}%
\makeatother
```

The following is an example of this `subfloat` environment begin used to produce figure 27. Note that you need to supply the width of the `verbatim`; here we use a section using a `minipage`).

```
\begin{figure}
  \centering  \begin{subfloat}%
```

```
    \begin{minipage}{2.1in}
      \begin{verbatim}
   This text  should be
verbatim.     And   not
 messed with in any way  !
      \end{verbatim}
    \end{minipage}%
    \caption{First subcaption.}%
    \label{fig:verbone}
  \end{subfloat}%
  \qquad
  \begin{subfloat}%
    \begin{minipage}{2.1in}
      \begin{verbatim}
  This text (also)should be
verbatim.     And   not
 messed with in any way  !
      \end{verbatim}
    \end{minipage}%
    \caption{Second subcaption.}%
    \label{fig:verbtwo}
  \end{subfloat}
  \caption{Subfigures~\subref{fig:verbone} and \subref{fig:verbtwo}
           show examples of using verbatim text in a subfigure.}
  \label{fig:verbatim}
\end{figure}
```

5 The Code

5.1 Identification

Announce the subfigure package.

```
\NeedsTeXFormat{LaTeX2e}[1994/12/01]
\ProvidesPackage{subfigure}[2002/07/30 v2.1.4 subfigure package]
```

5.2 Check for the **hyperref** Package

`\toclevel@subfigure`
`\toclevel@subtable`

After every package is loaded, check to see if the hyperref package was among them, if so, then make sure that the `\ifhyperrefloaded` switch is set so that the `\subfloat@label` command will write the correct form of the `\newlabel` to the `aux` file. Also define the "TOC level" of the subfigure and subtable. We fix these at one since the default figure and table levels are zero. Finally, we add definitions for `\theHsubfigure` and `\theHsubtable` to avoid duplicate names in the PDF file when using the hyperref Package.

```
\newif\ifhyperrefloaded
\AtBeginDocument{%
  \@ifpackageloaded{hyperref}{%
    \hyperrefloadedtrue
    \providecommand\theHsubfigure{\thefigure.\arabic{subfigure}}%
    \providecommand\theHsubtable{\thetable.\arabic{subtable}}%
    \providecommand{\toclevel@subfigure}{1}%
    \providecommand{\toclevel@subtable}{1}}{}}
```

5.3 Initialization and Shared Constants

`\ifsubcaphang`
`\ifsubcapcenter`
`\ifsubcapcenterlast`
`\ifsubcapnooneline`
`\ifsubcapraggedright`

These five flags control how the style in which the subfloat label and caption are printed. The **subcaphang** flag is first checked and if true, causes the subcaption label to be typeset separately and placed to the upper left of the space available for the subcaption. The **subcapcenter** flag centers each line of the subcaption. The **subcapcenterlast** centers the last line of the subcaption (this is a NOP if the **subcapcenter** flag is true. If the **subcapnooneline** is false, then the label plus the text of the subcaption are centered. If it is true, than the other flags may cause something different to happen. The purpose of this flag, generally, is to cause a single line to be left justified when there is a very short caption. The last flag is the **subcapraggedright** which typsets its text without lining up the right side. This is useful for the subcaptions since they are usually short and prone to generating hyphenated words unless allowed to be ragged.

```
\newif\ifsubcaphang
\newif\ifsubcapcenter
\newif\ifsubcapcenterlast
\newif\ifsubcapnooneline
\newif\ifsubcapraggedright
```

Table 4 gives the initial (default) values of the internals that are used to control the placement and printing of the subfloats.

`\subfigtopskip`
`\subfigcapskip`
`\subfigcaptopadj`
`\subfigbottomskip`

See figure 25 for details of where these take effect. Generally the `\subfigtopskip` appears between the figure or table and the edge of the box. `\subfigbottomskip` appears between the subcaption and the edge of the box. If the subcaption follows the figure or table, then `\subfigcapskip` is placed before it along with (*i.e.*, in addition to) a `\baselineskip`. If the subcaption comes before the figure or table then `\subfigcapskip` is placed after it along with `\subfigcaptopadj`. Although several of the above are `skip`'s they are typset at their base size and will not shrink or expand.

```
\newskip\subfigtopskip     \subfigtopskip    =  5\p@
\newskip\subfigcapskip     \subfigcapskip    =  0\p@
\newdimen\subfigcaptopadj  \subfigcaptopadj  =  3\p@
\newskip\subfigbottomskip  \subfigbottomskip =  5\p@
```

`\subfigcapmargin`
`\subfiglabelskip`

These two values are used to typeset the subcaption The width of the subcaption is the same as that of its associated figure or table width. `\subfigcapmargin` is placed on either side of the caption and `\subfiglabelskip` is placed between the subcaption label and the subcaption text. Depending on the manner of typesetting the subcaption, this may shrink or expand. By default, the `\subfigcapmargin` is zero to allow as much room of the subcaption as possible.

```
\newdimen\subfigcapmargin  \subfigcapmargin  =  \z@
\newskip\subfiglabelskip   \subfiglabelskip  =  0.33em plus 0.07em minus 0.03em
```

`\subcapsize`

`\subcapsize` is used to set the size of both the subcaption label and the subcaption text. The options allow it to be set to any of the following: `\scriptsize`,

Table 4: Default values of the Subfigure constants. These values are set during the options processing (see section 5.7).

Command	**loose** Option	**tight** Option	Description
`\subfigtopskip`	10 pt	5 pt	Length from the top of the subfloat box to the beginning of the figure.
`\subfigcapskip`	10 pt	0 pt	Length between the baseline of the subcaption and the figure.
`\subfigcaptopadj`	0 pt	3 pt	Length added to `\subfigcapskip` when the caption is above the figure.
`\subfigbottomskip`	10 pt	5 pt	Length from the bottom of the subcaption to the bottom of the subfloat.
`\subfigcapmargin`	10 pt	0 pt	Indentation of the subcaption from the sides of the subfloat box. (This should always be positive or zero.)
`\subfiglabelskip`	0.33 em	0.33 em plus 0.07 em minus 0.03 em	Space between the label and the text of the subcaption.
`\subcapsize`	footnotesize		Size for the text portion of the subcaption font.
`\subcaplabelfont`	(Default family, series and shape)		Font for the label portion of the subcaption.
`\subcapfont`	(Default family, series and shape)		Font for the text portion of the subcaption.

`\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`. It may also be set to `\tiny`, `\LARGE`, `\huge` or `\HUGE` by hand if need be for special instances.

```
\newcommand*{\subcapsize}{}
```

`\subcaplabelfont`
`\subcaplabelfont@f`
`\subcaplabelfont@c`
`\subcaplabelfont@s`

The `\subcaplabelfont` is composed of three parts, the font family, such as roman, san serif or typewriter; the font series, such as medium or bold; and the font shape, such as italic, slanted, small caps or upright. These are combined along with the `\subcapsize` to select the font for the subcaption label.

```
\newcommand*{\subcaplabelfont}{%
  \subcaplabelfont@f\subcaplabelfont@c\subcaplabelfont@s}
\newcommand*{\subcaplabelfont@f}{\fontfamily{\familydefault}\selectfont}
\newcommand*{\subcaplabelfont@c}{\fontseries{\seriesdefault}\selectfont}
\newcommand*{\subcaplabelfont@s}{\fontshape{\shapedefault}\selectfont}
```

`\subcapfont`
`\subcapfont@f`
`\subcapfont@c`
`\subcapfont@s`

The `\subcapfont` is the same as the `\subcaplabelfont` except that it is applied to the subcaption text rather than the label.

```
\newcommand*{\subcapfont}{%
  \subcapfont@f\subcapfont@c\subcapfont@s}
\newcommand*{\subcapfont@f}{\fontfamily{\familydefault}\selectfont}
```

```
\newcommand*{\subcapfont@c}{\fontseries{\seriesdefault}\selectfont}
\newcommand*{\subcapfont@s}{\fontshape{\shapedefault}\selectfont}
```

`\ifsf@tight`

Create an ‘if’ to control whether the check for the top-of-page is performed in the `\@subfloat` command. This is necessary to preserve the look-and-feel of the older versions of this package. The loose option turns this flag off (no check) and the tight option turns it on (do the check).

```
\newif\ifsf@tight           \sf@tighttrue
```

5.4 Subfigure Constants

`\c@subfigure`

Subfigure counter.

```
\newcounter{subfigure}[figure]
```

`\iffiguretopcap`
`\ifsubfiguretopcap`

These control how the subfigure *caption* numbering is obtained and where the figure caption and subcaption should appear relative to the body of the `subfigure`. The boolean `\iffiguretopcap` indicates that the *caption* counter is current and there is no need to increment it. The boolean `\ifsubfiguretopcap` indicates that the subcaption will be printed above the body portion of the subfigure.

```
\@ifundefined{figuretopcaptrue}{\newif\iffiguretopcap}{}
\newif\ifsubfiguretopcap
```

`\p@subfigure`
`\thesubfigure`
`\@thesubfigure`
`\@@thesubfigure`

The `\thesubfigure` command defines the label for text references (prefixed by `\p@subfigure`). This is the value saved by the `\label` and retrieved by the `\ref` commands. In the case of a conflict between this package and a prior one over the definition of `\thesubfigure`, this package will win. This is insured by first specifying the `\providecommand` for the `\thesubfigure` and then `\renewcommand`. This is necessary because some packages incorrectly insert this command.

The `\@thesubfigure` value defines the the caption label complete offset from the beginning of the caption text. It is used in the subfigure caption and normally takes the label portion as defined by `\thesubfigure`.

Finally, the value defined by `\@@thesubfigure` is also saved by the `\label` command and may be retrieved with the `\subref` command. This is often useful in the subcaption or caption text when referring to the individual subfigures. This value is also the one that is used in the List-of-Figures.

These multiple “views” of the *subfigure* counter allow a style to define the way the label looks in the figure, for example “**(a)**␣”. Then references to it with `\ref` have the form “2.1a”, and with `\subref` “(a)” (This latter form is also used to label references in the List-of-Figures section).

```
\let\p@subfigure=\thefigure
\providecommand*{\thesubfigure}{(\alph{subfigure})}
\renewcommand*{\thesubfigure}{(\alph{subfigure})}
\newcommand*{\@thesubfigure}{\thesubfigure\hskip\subfiglabelskip}
\newcommand*{\@@thesubfigure}{\thesubfigure}
```

`\ext@subfigure`
`\l@subfigure`
`\c@lofdepth`

These values define how and if the subfigure caption will appear in a List-of-Figures file. `\ext@subfigure` defines the default subfigure file extension (which is the same as `\ext@figure` — the List-of-Figures file, *lof*). `\l@subfigure` shows how to print an `lof` subfigure line and defines that line at level two. `\c@lofdepth` is an extension of the Table-of-Contents depth value and controls the depth to which captions in the file are printed to the actual page. By default, the subcaptions are not.

```
\let\ext@subfigure=\ext@figure
```

```
\newcommand*{\l@subfigure}{%
  \@dottedxxxline{\ext@subfigure}{2}{3.8em}{2.5em}}
```

```
\newcounter{lofdepth}
\setcounter{lofdepth}{1}
```

5.5 Subtable Constants

This section is symmetric to section 5.4.

`\c@subtable`

Subtable counter.

```
\newcounter{subtable}[table]
```

`\iftabletopcap`
`\ifsubtabletopcap`

These define the form that the subcaption prefix is generated. The boolean `\iftabletopcap` works with the numbering of the subcaption label and uses the current *table* counter value if true and the next value if false. The boolean `\ifsubtabletopcap` sets the subcaption before the main body of the subfigure, if true; and, after it, if false.

```
\@ifundefined{tabletopcaptrue}{\newif\iftabletopcap}{}
\newif\ifsubtabletopcap
```

`\p@subtable`
`\thesubtable`
`\@thesubtable`
`\@@thesubtable`

The `\thesubtable` command defines the label for text references (prefixed by `\p@subtable`), while the `\@thesubtable` command defines what appears in the subcaption under or over the subtable. The `\@@thesubtable` command defines an alternative reference to the label for use in the subcaption and caption of the table (see the discussion above for the equivalent figure values). The latter form is also used for the List-of-Tables label. As above, the `\thesubtable` command is twice specified

```
\let\p@subtable=\thetable
\providecommand*{\thesubtable}{(\alph{subtable})}
\renewcommand*{\thesubtable}{(\alph{subtable})}
\newcommand*{\@thesubtable}{\thesubtable\hskip\subfiglabelskip}
\newcommand*{\@@thesubtable}{\thesubtable}
```

`\ext@subtable`
`\l@subtable`
`\c@lotdepth`

These define how and if the subtable caption will appear in a List-of-Tables file. `\ext@subtable` defines the default subtable file extension (which is the same as `\ext@table` — the List-of-Tables file, *lot*). `\l@subtable` shows how to print an `lot` subtable line and defines that line at level two. `\c@lotdepth` is an extension of the table-of-contents depth value and controls the depth to which captions in the file are printed to the actual page. By default, the subcaptions are not printed.

```
\let\ext@subtable=\ext@table
```

```
\newcommand*{\l@subtable}{%
  \@dottedxxxline{\ext@subtable}{2}{3.8em}{2.5em}}
```

```
\newcounter{lotdepth}
\setcounter{lotdepth}{1}
```

5.6 Declaration of Options

The following options allow general compatibility with the caption and caption2 packages by H.A. Sommerfeldt [1]. There are six different subcaption layout options supported: normal, hang (or isu), center, centerlast (or anne), nooneline and raggedright. The hang subcaption may be combined with the center or centerlast options. The nooneline may be combined with any of the other options (but it's effect is negated or looks bad with either of center or centerlast unless the hang option is also used). raggedright overrides the center or centerlast options.

```
\DeclareOption{normal}{%
  \subcaphangfalse
  \subcapcenterfalse
  \subcapcenterlastfalse
  \subcapnoonelinefalse
  \subcapraggedrightfalse}

\DeclareOption{hang}{\subcaphangtrue}
\DeclareOption{center}{\subcapcentertrue}
\DeclareOption{centerlast}{\subcapcenterlasttrue}
\DeclareOption{nooneline}{\subcapnoonelinetrue}
\DeclareOption{raggedright}{\subcapraggedrighttrue}

\DeclareOption{isu}{\ExecuteOption{hang}}
\DeclareOption{anne}{\ExecuteOption{centerlast}}
```

There are options for six different font sizes available.

```
\DeclareOption{scriptsize}{\renewcommand*{\subcapsize}{\scriptsize}}
\DeclareOption{footnotesize}{\renewcommand*{\subcapsize}{\footnotesize}}
\DeclareOption{small}{\renewcommand*{\subcapsize}{\small}}
\DeclareOption{normalsize}{\renewcommand*{\subcapsize}{\normalsize}}
\DeclareOption{large}{\renewcommand*{\subcapsize}{\large}}
\DeclareOption{Large}{\renewcommand*{\subcapsize}{\Large}}
```

There are eighteen options available to set the font attributes of the subcaptions. The first nine affect only the subcaption label The last nine affect only the subcaption text.

```
\DeclareOption{rm}{\renewcommand*{\subcaplabelfont@f}{\rmfamily}}
\DeclareOption{sf}{\renewcommand*{\subcaplabelfont@f}{\sffamily}}
\DeclareOption{tt}{\renewcommand*{\subcaplabelfont@f}{\ttfamily}}
\DeclareOption{md}{\renewcommand*{\subcaplabelfont@c}{\mdseries}}
\DeclareOption{bf}{\renewcommand*{\subcaplabelfont@c}{\bfseries}}
\DeclareOption{up}{\renewcommand*{\subcaplabelfont@s}{\upshape}}
\DeclareOption{it}{\renewcommand*{\subcaplabelfont@s}{\itshape}}
\DeclareOption{sl}{\renewcommand*{\subcaplabelfont@s}{\slshape}}
\DeclareOption{sc}{\renewcommand*{\subcaplabelfont@s}{\scshape}}

\DeclareOption{RM}{\renewcommand*{\subcapfont@f}{\rmfamily}}
\DeclareOption{SF}{\renewcommand*{\subcapfont@f}{\sffamily}}
\DeclareOption{TT}{\renewcommand*{\subcapfont@f}{\ttfamily}}
\DeclareOption{MD}{\renewcommand*{\subcapfont@c}{\mdseries}}
\DeclareOption{BF}{\renewcommand*{\subcapfont@c}{\bfseries}}
\DeclareOption{IT}{\renewcommand*{\subcapfont@s}{\itshape}}
\DeclareOption{SL}{\renewcommand*{\subcapfont@s}{\slshape}}
\DeclareOption{SC}{\renewcommand*{\subcapfont@s}{\scshape}}
\DeclareOption{UP}{\renewcommand*{\subcapfont@s}{\upshape}}
```

There are eight options available to control the caption placement and the proper numbering in association with the figure or table caption placement. The first four affect only the caption numbering by informing the internals that the associated figure or table caption appears before or after the subfloat. The second four do this and, in addition, shift the subfloat caption to the bottom or top of the subfloat. The `\subfigure` and `\subtable` commands each have a set of flags since it is often the case that a document style requires that figure captions follow the figure and table captions precede the table.

```
\DeclareOption{figbotcap}{\figuretopcapfalse}
\DeclareOption{figtopcap}{\figuretopcaptrue}
\DeclareOption{tabbotcap}{\tabletopcapfalse}
\DeclareOption{tabtopcap}{\tabletopcaptrue}

\DeclareOption{FIGBOTCAP}{\ExecuteOptions{figbotcap}\subfiguretopcapfalse}
\DeclareOption{FIGTOPCAP}{\ExecuteOptions{figtopcap}\subfiguretopcaptrue}
\DeclareOption{TABBOTCAP}{\ExecuteOptions{tabbotcap}\subtabletopcapfalse}
\DeclareOption{TABTOPCAP}{\ExecuteOptions{tabtopcap}\subtabletopcaptrue}
```

`\subfigtopskip` `\subfigcapskip` `\subfigcaptopadj` `\subfigbottomskip` `\subfigcapmargin` `\subfiglabelskip`

The last two options control the overall “look-and-feel” of the subfloat. The loose option is the default and makes the subfloat look like it always has with lots of extra room around the subfigure and subcaption.

```
\DeclareOption{loose}{%
  \subfigtopskip    = 10\p@
  \subfigcapskip    = 10\p@
  \subfigcaptopadj  =  0\p@
  \subfigbottomskip = 10\p@
  \subfigcapmargin  = 10\p@
  \subfiglabelskip  =  0.33em
```

`\@thesubfigure` `\@thesubtable`

Next, it replaces the glue at the end of the subcaption label with a `\space` like the older version of the subfigure package.

```
  \renewcommand*{\@thesubfigure}{\thesubfigure\space}
  \renewcommand*{\@thesubtable}{\thesubtable\space}
```

`\ifsf@tight`

Finally, set the **sf@tight** flag to make the `\@subfloat` command skip its check for the top of a page or minipage and to always add its topmost vertical spacing. (For more details about the `\@subfloat` command, see section 5.8.)

```
  \sf@tightfalse}
```

`\subfigtopskip` `\subfigcapskip` `\subfigcaptopadj` `\subfigbottomskip` `\subfigcapmargin` `\subfiglabelskip`

The tight option is the is the preferred version and has less white space around the subfloat. It also will omit the space above the subfloat at the top of the page or minipage.

```
\DeclareOption{tight}{%
  \subfigtopskip    =  5\p@
  \subfigcapskip    =  0\p@
  \subfigcaptopadj  =  3\p@
  \subfigbottomskip =  5\p@
  \subfigcapmargin  =  \z@
  \subfiglabelskip  =  0.33em plus 0.07em minus 0.03em
```

`\@thesubfigure` `\@thesubtable`

Next, it keeps the glue at the end of the subcaption label to allow better subcaption fitting.

```
  \renewcommand*{\@thesubfigure}{\thesubfigure\hskip\subfiglabelskip}
  \renewcommand*{\@thesubtable}{\thesubtable\hskip\subfiglabelskip}
```

`\ifsf@tight`

Finally, set the **sf@tight** flag to make the `\@subfloat` command check for the top of a page or minipage and to skip adding any vertical space there. (For more details about the `\@subfloat` command, see section 5.8.)

```
  \sf@tighttrue}
```

5.7 Execution of Options

The normal type of subcaption is preselected, the standard subcaption size is set to footnotesize, and the font for both the subcaption label and text is set above to the global defaults for family, series, and shape. Also, the subcaptions for the subfigure and subtable are placed after the figure box and it is assumed that the figure or table caption follows all of the associated subfloats. Finally, the loose form is selected in order to cause minimal change to existing papers using the subfigure package.

The preferred form would be to have the TABTOPCAP and tight be the defaults, but this would adversely affect the existing papers that have used the official releases of this package.

```
\ExecuteOptions{normal,footnotesize,FIGBOTCAP,TABBOTCAP,loose}
\InputIfFileExists{subfigure.cfg}{%
  \typeout{****************************************^^J%
           * Local config file subfigure.cfg used *^^J%
           ****************************************}}{}
\ProcessOptions
```

5.8 The Subfigure and Subtable Commands

`\subfigure`

The `\subfigure` command acts as cover function for the `\@subfloat` command. It locally changes the `\label` command to our special version that supports the `\subref`'s (see section 5.9). It insures that the proper counter is used and has the correct value. Since the caption is usually generated later, we must locally anticipate the future value of its counter by adding one to it within a local group. Upon leaving `\subfigure`, the old value is restored.

```
\newcommand*{\subfigure}{%
  \bgroup
    \let\subfig@oldlabel=\label
    \let\label=\subfloat@label
    \@nameuse{if\@captype topcap}\else
      \advance\@nameuse{c@\@captype}\@ne
    \fi
    \refstepcounter{sub\@captype}%
    \@ifnextchar [%
      {\@subfigure}%
      {\@subfigure[\@empty]}}
```

`\subtable`

The `\subtable` command is identical to `\subfigure`. The of names at the user level is purely cosmetic (and historical).

```
\let\subtable=\subfigure
```

`\@subfigure`

Here we are still setting up for the main `\@subfloat` command. We check for a second optional argument. If one is not found, than any optional argument from the last `\subfigure` or `\subtable` becomes the main caption and we give `\@empty` as the default list-entry caption. If we see another optional argument, then we make that one the main caption and use any prior optional argument as the list-entry caption. See Table 2 for how this looks to the user.

```
\def\@subfigure[#1]{%
    \@ifnextchar [%
      {\@subfloat{sub\@captype}[{#1}]}%
      {\@subfloat{sub\@captype}[\@empty{#1}][{#1}]}}
```

`\@subfloat`

This is the common code for setting up the subfloat box and drawing the subcaption under it. The two skips are used only here to keep track of what vertical space is to be placed before and after the figure.

The first argument is the type of object being generated: that is, a `subfigure` or a `subtable`. The second and third are the `subcaption` and `subfigure` arguments from the calling `\subfigure` or `\subtable` command.

```
\newskip\subfig@top
\newskip\subfig@bottom
```

If **ifsf@tight** is true, then the `\@subfloat` command checks to see if it is at the top of a page or a minipage and will suppress the top vertical space in that case; otherwise, it always adds the space.

```
\long\def\@subfloat#1[#2][#3]#4{%
    \@tempcnta=1
    \ifsf@tight
      \if@minipage
        \@tempcnta=\z@
      \else\ifdim \lastskip=\z@ \else
        \@tempcnta=2
      \fi\fi
    \fi
```

Based on the `\iffiguretopcap` or `\iftabletopcap` flags we select which vertical space is to be placed above and below the figure or table and save it in `\subfig@top` and `\subfig@bottom`.

```
    \@nameuse{if\@captype topcap}%
      \subfig@top=\subfigbottomskip
      \subfig@bottom=\subfigtopskip
    \else
      \subfig@top=\subfigtopskip
      \subfig@bottom=\subfigbottomskip
    \fi
```

The `\leavevmode` is here to inhibit any LaTeX errors that the surreounding environment might generate if we stay in vertical mode. Then it determines the width of the figure or table by placing it in a box and testing the box.

```
    \leavevmode
    \setbox\@tempboxa \hbox{#4}%
    \@tempdima=\wd\@tempboxa
```

Finally we put the figure together in a vertical box. At the very top goes any vertical space, but only if we are not at the top of the page or minipage as determined above.

```
    \vtop\bgroup
      \vbox\bgroup
        \ifcase\@tempcnta
          \@minipagefalse
        \or
          \vspace{\subfig@top}%
        \or
          \ifdim \lastskip=\z@ \else
            \@tempskipb\subfig@top\relax\@xaddvskip
          \fi
        \fi
```

Next, based on the 'topcap' flags, we check if the subcaption or the figure goes next. If it is the subcaption, then we add some extra `\subfigcaptopadj` space between the subcaption and the figure and table in addtion to the regular `\subfigcapskip` space. This finishes off the top box and establishes our baseline.

After that we add in either the figure or subcaption (whichever we have not typeset yet and follow it with the bottom vertical space. (see figure 25(c) for a diagram of this layout).

Finally, we globally (**!**) reset the *figure* or *table* counter, if we incremented it at the beginning of the `\subfigure` or `\subtable` command so that any functions used inside the command body which globally sets the counters (*e.g.*, the tabularx package) will not cause problems.

```
    \@nameuse{if#1topcap}%
     \ifx \@empty#3\relax \else
      \@subcaption{#1}{#2}{#3}%
      \vskip\subfigcapskip
      \vskip\subfigcaptopadj
     \fi\egroup
     \box\@tempboxa
    \else
     \box\@tempboxa\egroup
     \ifx \@empty#3\relax \else
      \vskip\subfigcapskip
      \@subcaption{#1}{#2}{#3}%
     \fi
    \fi
   \vspace{\subfig@bottom}%
  \egroup
  \@nameuse{if\@captype topcap}\else
   \global\advance\@nameuse{c@\@captype}\m@ne
  \fi
 \egroup}
```

`\@subfigcaptionlist` `\@subcaption` `\listsubcaptions` `\@listsubcaptions` The following series of commands control exactly how the subcaption is typeset. The `\@subcaption` command adds the subcaption to the current list of subcaptions to be added to the "List-of" page as soon as the major caption is declared (see `\@caption` below). (NOTE: only one list is kept because that seems right; if there is a mix of tables and figures, they will be grouped under the next `\caption`.) Next `\@subcaption` calls the appropriate float-type specific command to decide how to size and shape the subcaption text.

```
\newcommand*{\@subfigcaptionlist}{}
```

```
\newcommand{\@subcaption}[3]{%
 \ifx \relax#2\relax \else
  \bgroup
   \let\label=\@gobble
   \let\protect=\string
   \def\@subcaplabel{\@nameuse{@@the#1}}%
   \xdef\@subfigcaptionlist{%
    \@subfigcaptionlist,%
    {\protect\numberline{\@subcaplabel}\noexpand{\ignorespaces #2}}}%
  \egroup
 \fi
 \@nameuse{@make#1caption}{\@nameuse{@the#1}}{#3}}
```

```
\newcommand*{\listsubcaptions}{%
  \@ifstar
    {\gdef\@subfigcaptionlist{}}%
    {\@listsubcaptions{\@captype}}}
```

```
\newcommand*{\@listsubcaptions}[1]{%
  \@ifundefined{@captype}{}{%
    \@ifundefined{ext@sub#1}{}{%
      \@for \sf@temp:=\@subfigcaptionlist \do {%
        \ifx \@empty\sf@temp\relax \else
          \addcontentsline
            {\@nameuse{ext@sub#1}}%
            {sub#1}%
            {\sf@temp}%
        \fi}}}%
  \gdef\@subfigcaptionlist{}}
```

`\@makesubfigurecaption`
`\@makesubtablecaption`

By default, the `\@subfigurecaption` and `\@subtablecaption` commands are identical. Unlike the standard `\@makecaption` command, we assume that the first argument (the label number produced by the `\@thesubfigure` or the `\@thesubtable`) contains any trailing separator characters or spacing (which makes it easier to customize).

The `\@makesubfigurecaption` command first checks the size of the caption typeset as a single line. It knocks off twice the `\subfigcapmargin` (at it's regular size) to determine the with of the caption and label.

```
\newcommand{\@makesubfigurecaption}[2]{%
  \setbox\@tempboxa\hbox{%
    \subcapsize
    {\subcaplabelfont #1}%
    {\subcapfont\ignorespaces #2}}%
  \@tempdimb=-\subfigcapmargin
  \multiply\@tempdimb\tw@
  \advance\@tempdimb\@tempdima
```

Next it creates a horizontal box of that width and if the label plus the text was too wide or if the **subcapnooneline** flag is true, then it sends off the label and subcaption to `\subfig@caption` to typset as a paragraph. NOTE: `\subfig@caption` assumes that `\@tempbdimb` has the calculated width for the paragraph.

If the label plus the text will fit and the **subcapnooneline** flag is false, then we just return them (from box `\@tempboxa`).

```
  \hbox to\@tempdima{%
    \hss
    \ifdim \wd\@tempboxa >\@tempdimb
      \subfig@caption{#1}{#2}%
    \else\ifsubcapnooneline
      \subfig@caption{#1}{#2}%
    \else
      \box\@tempboxa
    \fi\fi
    \hss}}
```

```
\let\@makesubtablecaption=\@makesubfigurecaption
```

`\subfig@caption`
`\subfig@captionpar`

These commands are called to typeset a multiple-line subcaption (or a single line when **subcapnooneline** is true). Depending on the **subcapcenter** and **subcapcenterlast** flags, the text will be justified (both false), centered (**subcapcenter** true), or justified with the last line centered (only the flag **subcapcenterlast** set true).

```
\newcommand{\subfig@caption}[2]{%
  \ifsubcaphang
    \sbox{\@tempboxa}{\subcapsize\subcaplabelfont #1}%
    \addtolength{\@tempdimb}{-\wd\@tempboxa}%
    \usebox{\@tempboxa}%
    \subfig@captionpar{\@tempdimb}{%
      {\subcapfont\ignorespaces #2}}%
  \else
    \subfig@captionpar{\@tempdimb}{%
      {\subcaplabelfont #1}%
      {\subcapfont\ignorespaces #2}}%
  \fi}
```

```
\newcommand{\subfig@captionpar}[2]{%
  \parbox[t]{#1}{%
    \subcapsize
    \ifsubcapraggedright
      \setlength{\leftskip}{\z@}%
```

```
     \setlength{\@rightskip}{\@flushglue}%
     \setlength{\rightskip}{\@rightskip}%
     \setlength{\parindent}{\z@}%
   \else\ifsubcapcenter
     \setlength{\leftskip}{\@flushglue}%
     \setlength{\rightskip}{\@flushglue}%
     \setlength{\parfillskip}{\z@skip}%
   \else\ifsubcapcenterlast
     \addtolength{\leftskip}{\z@ plus 1fil}%
     \addtolength{\rightskip}{\z@ plus -1fil}%
     \setlength{\parfillskip}{\z@ plus 2fil}%
   \fi\fi\fi
   #2}}
```

5.9 Patches to the Standard Environment

The following adjust the standard environment for the subfigure package. They are designed as wrappers to the current definition of the standard commands to minimize any chance of conflict with other packages or to extend LaTeX.

`\@dottedxxxline`

This is a generalized wrapper for the `\@dottedtocline` command. It checks for the level based on the output file (first argument) and not using only `\@tocdepth`. (See section 4.4 for a description of the arguments.)

```
\newcommand*{\@dottedxxxline}[6]{%
 \ifnum #2>\@nameuse{c@#1depth}\else
   \@dottedtocline{0}{#3}{#4}{#5}{#6}
 \fi}
```

`\subfig@end@float`
`\subfig@end@dblfloat`
`\end@float`
`\end@dblfloat`

These commands patch the end of the `float` environment so that it will dump out the subcaptions if any remain at this point. This can occur when using the TOPCAP options.

```
\let\subfig@end@float=\end@float
```

```
\renewcommand*{\end@float}{%
 \@listsubcaptions{\@captype}%
 \subfig@end@float}
```

```
\let\subfig@end@dblfloat=\end@dblfloat
```

```
\renewcommand*{\end@dblfloat}{%
 \@listsubcaptions{\@captype}%
 \subfig@end@dblfloat}
```

`\subfig@oldcaption`
`\@caption`

Next, we redefine the current `\@caption` command to dump any subcaptions saved. First the ‘old’ caption command is called to add the line to the “List-of” file and then the list of subcaptions, `\@subfigcaptionlist` is written to the same file. Lastly, the `\@subfigcaptionlist` is reinitialized.

```
\let\subfig@oldcaption=\@caption
```

```
\long\def\@caption#1[#2]#3{%
 \@ifundefined{if#1topcap}%
   {\subfig@oldcaption{#1}[{#2}]{#3}}%
   {\@nameuse{if#1topcap}%
      \@listsubcaptions{#1}%
      \subfig@oldcaption{#1}[{#2}]{#3}%
    \else
      \subfig@oldcaption{#1}[{#2}]{#3}%
      \@listsubcaptions{#1}%
    \fi}}
```

`\subfig@oldlabel`

To support the redefinition of the `\label` command within the body of the subfloats, we will use `\subfig@oldlabel` to save the current definition of `\label` and create the `\subfloat@label` command to take its place during the processing of the `\subfigure` command. Since the definition of `\label` may change as packages are loaded, we save the definition each time that `\label` is replaced with `\sub@label` (see 5.8 above).

```
\let\subfig@oldlabel=\relax
```

`\subfloat@label`
`\sub@label`

One difference from the regular `\label` command is that there is an optional argument (note with parentheses rather than square brackets) that is only used with the hyperref package to define the *bookmark* argument to the label. Typically, this would be a copy or paraphrase of the subcaption text. If this is not given and the hyperref package is being used, then the *bookmark* argument is of the form “Subfigure␣1(a)”.

```
\newcommand*{\subfloat@label}{%
 \@ifnextchar(
   {\sf@sub@label}
   {\sf@sub@label(Sub\@captype\space
                  \@ifundefined{thechapter}{}{%
                    \@nameuse{thechapter}\space}%
                  \@nameuse{p@sub\@captype}%
                  \@nameuse{thesub\@captype}.)}}
```

```
\let\sub@label\subfloat@label
```

`\sf@sub@label` The `\sf@sub@label` parses the optional argument and (if the hyperref Package is loaded) saves the *bookmark* text as `\@currentlabelname`. It then calls the `\sf@@sub@label` command to the real processing of the label.

```
\def\sf@sub@label(#1)#2{%
  \ifhyperrefloaded
    \protected@edef\@currentlabelname{%
      \expandafter\strip@period #1\relax.\relax\@@@}%
  \fi
  \sf@@sub@label{#2}}
```

`\sf@@sub@label` In order to support the hyperref package we check if it was loaded and use the proper form of the `\newlabel` command. `\sf@@sub@label` operates by first calling the old `\label` definition (which adds a `\newlabel` command to the `*.aux` file) and then adds another `\newlabel` command to the `*.aux` file with a similar reference name (with 'sub@' prepended) and the value of `\@@thesubfigure` or `\@@thesubtable`.

If the `\ifhyperrefloaded` flag is set, then the `\newlabel` command has three extra fields, the first is the value of `\@currentlabelname`, which is either of the form "Subfigure␣1(a)" or was defined by the optional argument to `\label` (actually `\sub@label`). The second extra field is the hypertext anchor name and the third is unused. Otherwise, the we us the standard `\newlabel` form to write the subreference.

```
\newcommand*{\sf@@sub@label}[1]{%
  \@bsphack
  \subfig@oldlabel{#1}%
  \ifhyperrefloaded
    \protected@write\@auxout{}{%
        \string\newlabel{sub@#1}%
            {{\@nameuse{@@thesub\@captype}}%
            {\thepage}%
            {\expandafter\strip@period\@currentlabelname\relax.\relax\@@@}%
            {\@currentHref}%
            {}}}%
  \else
    \protected@write\@auxout{}{%
        \string\newlabel{sub@#1}%
            {{\@nameuse{@@thesub\@captype}}%
            {\thepage}}}%
  \fi
  \@esphack}
```

`\subref` The `\subref` command is the same as the `\ref` command except that `\@@thesubtable` instead of `\p@subfigure\thesubfigure` or `\p@subtable\thesubtable`. This is often of use for local references within the figure where the figure number may be assumed; or, for ease in constructing a range of references within a figure with many subfigures.

```
\newcommand\subref[1]{%
  \ref{sub@#1}}
```

`\Subref` The `\Subref` command is the same as `\subref`, except that it adds `\subcaplabelfont` before the reference so that it uses the same font (except that the current font size is maintained).

```
\newcommand\Subref[1]{%
  {\subcaplabelfont
   \ref{sub@#1}}}
```

6 Acknowledgements

This package was originally written to automatically line up some figure boxes and place labels under them for a paper that I was writing. I thought it useful and uploaded it to the internet community and later to CTAN. Many people have asked questions or given comments which collectively have changed and improved the usefulness of this package.

A few people have contributed more than most and I want to thank them publicly, but in no particular order:

- **Harald Axel Sommerfeldt** for the work that he did to adjust his caption and caption2 packages as necessary to support the subfigure package when they are loaded together.
- **Peter Wilson** for the work that he did to adjust his ccaption package (and other packages) as necessary to support the subfigure package when they are loaded together.
- **William 'bil' L. Kleb** for his extensive list of errors and suggestions to this documentation.
- **Axel Reichert** for his request for a 'hang' caption style since the subcaptions tend to have a short width. And, for his request for some way of referencing the individual subfigures in the main caption without the figure number.
- **Harald Harders** for his suggestion of the `\subref` command and modifying `\label` within the subfigure package to save local references to the subfigures that are often needed.
- **Heiko Oberdiek** and **James A. Bednar** for their help with coexisting with the hyperref and html packages. Also, **Ingele Roelens** for pointing out some further compatibility problems when using the hyperref package with PDFLaTeX.
- **Frederic Darboux** for searching out and finding several incompatibilities with other packages.

References

[1] Harald Axel Sommerfeldt, *The caption Package*, Version 1.4b, 1995/04/05. (Available from CTAN as file `caption.dtx`.)

[2] Harald Axel Sommerfeldt, *The caption Package*, Version 2.0(beta), 1995/10/09. (Available from CTAN as file `caption2.dtx`.)

[3] Peter Wilson, *The ccaption Package*, Version 1.0a, 2001/08/15. (Available from CTAN as file `ccaption.dtx`.)

[4] Peter Wilson, *The tocloft Package*, Version 2.2, 2001/14/17. (Available from CTAN as file `ccaption.dtx`.)

[5] Sebastian Rahtz, *Hypertext marks in LaTeX*, 2002/04/05/. (Available from CTAN as file `hyperref.dtx`.)

[6] Steven Douglas Cochran, *The captcont Package*, 2002/02/14/. (Available from CTAN as file `captcont.dtx`.)

[7] Keith Reckdahl, *Using Imported Graphics in LaTeX 2ε*, 1997/12/15. (Available from CTAN as file `epslatex.pdf`)

[8] Leslie Lamport, *LaTeX User's Guide and Reference Manual*, 2nd edition, Addison-Wesley, Reading, Massachusetts, 1994.

[9] Donald Ervin Knuth, *The TeXbook*, Addison-Wesley, Reading, Massachusetts, 1986.

[10] Leslie Lamport, Frank Mittelbach, and Johannes Braams, *Standard Document Classes for LaTeX version 2e* Version 1.4e, 2001/04/01. (Available from CTAN as file `classes.dtx`.)

[11] Anselm Lingnau, *An Improved Environment for Floats.* Version 1.3d, 2001/11/08. (Available from CTAN as file `float.dtx`.)