

The subcaption package*

Axel Sommerfeldt

<https://gitlab.com/axelsommerfeldt/caption>

2018/05/13

Abstract

This package supports typesetting of sub-captions (by using the the sub-caption feature of the caption package).

- ▲ At the end of each section, text marked with the mountain symbol will contain background knowledge on how the particular command or environment is actually implemented. If you just want to use this package as it is, you don't have to read or understand them.
- ▲ This package demonstrates the usage of `\DeclareCaptionSubType`, `\captionsetup{sub-type}`, and the internal hook `\caption@subtypehook` (offered by the caption package).



Please note: This package is incompatible with the subfigure and subfig packages.

*This package has version number v1.2.

Contents

1	Loading the package	3
2	The <code>\subcaptionbox</code> command	4
3	The <code>subfigure</code> & <code>subtable</code> environments	6
4	The <code>\subcaption</code> command	7
5	The <code>\DeclareCaptionSubType</code> command	8
6	References	9
6.1	The <code>\subref</code> command	10
6.2	The <code>subrefformat=</code> option	10
6.3	Referencing sub-figures without sub-captions	10
6.4	Where to place the <code>\label</code> command?	11
6.5	Where do hyperlinks jump?	12
7	Beyond this package	12
8	Thanks	12

1 Loading the package

Load this package using

```
\usepackage[\langle options \rangle]{subcaption} .
```

The options for the subcaption package are the same ones as for the caption package, but specify settings which are used for sub-captions *additionally*. In fact

```
\usepackage[\langle options \rangle]{subcaption}
```

is identical to

```
\usepackage{subcaption}  
\captionsetup[sub]{\langle options \rangle} .
```

The default settings for subcaptions are:

```
margin=0pt, size=smaller, labelformat=parens, labelsep=space,  
skip=6pt, list=false, hypcap=false1
```

Options specified with `\usepackage[...]{subcaption}` and `\captionsetup[sub]{...}` will override the ones specified by `\captionsetup{...}` and `\captionsetup[figure]{...}`, but are again overwritten by `\captionsetup[subfigure]{...}` (same for ‘table’). So finally we have the following order how settings for sub-captions are applied:

1. Global settings (`\usepackage[...]{caption}` and `\captionsetup{...}`)
2. Environmental settings (`\captionsetup[figure -or- table]{...}`)
3. Local settings (`\captionsetup{...}` inside figure or table environment)
4. Default ‘sub’ settings (`margin=0pt, size=smaller, ...`, see above)
5. Custom ‘sub’ settings (`\usepackage[...]{subcaption}` and `\captionsetup[sub]{...}`)
6. Environmental ‘sub’ settings (`\captionsetup[subfigure -or- subtable]{...}`)
7. Local ‘sub’ settings (`\captionsetup{...}` inside subfigure or subtable)

An example:

```
\usepackage[labelsep=quad, indentation=10pt]{caption}  
\usepackage[labelfont=bf, list=true]{subcaption}  
\captionsetup[table]{textfont=it, position=top}  
\captionsetup[subtable]{textfont=sf}
```

causes the captions inside subtable environments to be typeset with the settings

```
indentation=10pt, position=top, margin=0pt, font=small,  
labelformat=parens, labelsep=space, skip=6pt, hypcap=false,  
labelfont=bf, list=true, textfont=sf .
```

¹This means that sub-captions are not listed in the List of Figures or Tables by default, but you can enable that by specifying the option `list=true`.

2 The `\subcaptionbox` command

`\subcaptionbox` The easiest way to apply a sub-caption is by using the `\subcaptionbox` command, which automatically aligns the sub-figures resp. sub-tables by their very first caption line. Its syntax is:

```
\subcaptionbox[⟨list entry⟩]{⟨heading⟩}[⟨width⟩][⟨inner-pos⟩]{⟨contents⟩}
\subcaptionbox*{⟨heading⟩}[⟨width⟩][⟨inner-pos⟩]{⟨contents⟩}
```

The arguments `⟨list entry⟩` & `⟨heading⟩` will be used for typesetting the `\caption`.

`⟨width⟩` is the width of the resulting `\parbox`; the default value is the width of the contents.

`⟨inner-pos⟩` specifies how the contents will be justified inside the resulting `\parbox`; it can be either ‘c’ (for `\centering`), ‘l’ (for `\raggedright`), ‘r’ (for `\raggedleft`), or ‘s’ (for no special justification). The default is ‘c’. (But you can use any justification defined with `\DeclareCaptionJustification` as well, e.g. ‘centerlast’.)

Using `\subcaptionbox`, the baseline of the resulting box will be placed right between contents and heading, so usually you don’t have to care about the vertical alignment of the sub-figures for yourself. Also the hyperlink anchor is placed properly with respect to the `hypcap=` setting.

One example:

```
\begin{figure}
\centering
\subcaptionbox{A cat\label{cat}}
{\includegraphics{cat}}
\subcaptionbox{An elephant\label{elephant}}
{\includegraphics{elephant}}
\caption{Two animals}\label{animals}
\end{figure}
```

gives the result:

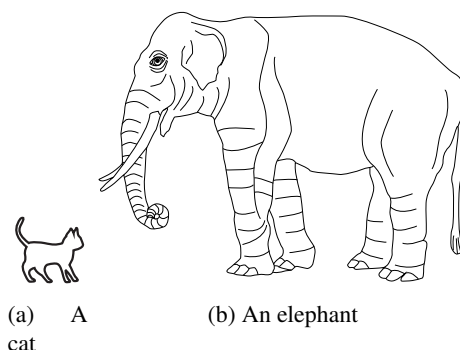


Figure 1: Two animals²

²The pictures were taken with permission from the L^AT_EX Companion[1] examples.

As you see the result is not satisfying; the caption below the cat looks ugly because of the small width of the graphic. This can be solved by using the optional arguments of `\subcaptionbox`, increasing the width of the resulting box:

```
...
\subcaptionbox{A cat\label{cat}}
[2.5cm]{\includegraphics{cat}}
...
```

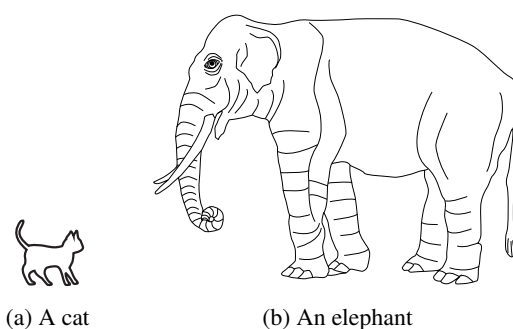


Figure 2: Two animals

Furthermore the main caption, which is centered with respect to the `\textwidth`, looks mis-aligned with respect to the sub-captions. This can (again) be solved by using the optional arguments of `\subcaptionbox`, giving both boxes the same width, for example:

```
...
\subcaptionbox{A cat\label{cat}}
[.4\linewidth]{\includegraphics{cat}}%
\subcaptionbox{An elephant\label{elephant}}
[.4\linewidth]{\includegraphics{elephant}}
...
```

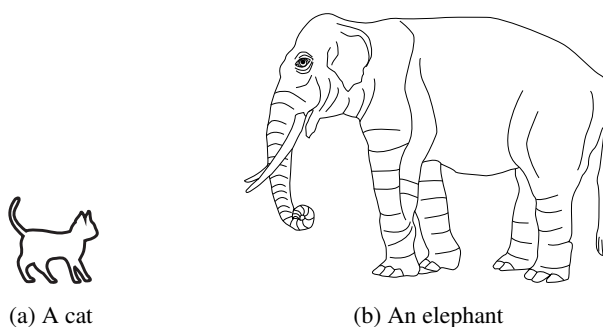


Figure 3: Two animals

▲ The `\subcaptionbox` is a `\parbox` with `\setcaptionsubtype` as first contents line.

3 The subfigure & subtable environments

subfigure After loading the subcaption package the new environments subfigure and sub-
subtable table are available, which have the same (optional & mandatory) arguments as the
minipage environment:

```
\begin{subfigure} [⟨outer-pos⟩] [⟨height⟩] [⟨inner-pos⟩] {⟨width⟩}  
...  
\end{subfigure}
```

and

```
\begin{subtable} [⟨outer-pos⟩] [⟨height⟩] [⟨inner-pos⟩] {⟨width⟩}  
...  
\end{subtable}
```

Inside these environments you use the ordinary `\caption` command for typesetting captions. So this example is the same as the last one, but uses the subfigure environment:

```
\begin{figure}  
  \begin{subfigure} [b] {.5\linewidth}  
    \centering\large A  
    \caption{A subfigure}\label{fig:1a}  
  \end{subfigure}%  
  \begin{subfigure} [b] {.5\linewidth}  
    \centering\large B  
    \caption{Another subfigure}\label{fig:1b}  
  \end{subfigure}  
  \caption{A figure}\label{fig:1}  
\end{figure}
```

Some additional notes:

- You can override the settings for a specific subcaption with a `\captionsetup` inside the subfigure or subtable environment, e.g.:

```
\begin{subfigure} [b] {.5\linewidth}  
  \centering\large A  
  \captionsetup{skip=3pt}  
  \caption{A subfigure}\label{fig:1a}  
\end{subfigure}
```

- Hyperlinks targeted to this subfigure will jump to the beginning of the subfigure, and not to the caption of the subfigure (if `hypcap=true` is set for sub-captions). (See [section 6.5: Where do hyperlinks jump?](#))

Beside the `⟨inner-pos⟩` values of ‘c’, ‘t’, and ‘b’, the subcaption package also offers the values ‘T’ and ‘B’ additionally which align the subfigure at the very top resp. bottom. (In contrast ‘t’ and ‘b’ align the subfigure at the top resp. bottom *baseline*.)

- ▲ The `subfigure` & `subtable` environments are just simple `minipage` environments with `\setcaptionsubtype` as first contents line. These environments are defined with the help of `\caption@For{subtypelist}`, which executes code for every sub-type declared with `\DeclareCaptionSubType`.

4 The `\subcaption` command

`\subcaption` The most flexible way to apply a sub-caption is by using the `\subcaption` command. Its syntax is analogous to the one of the `\caption` command and shares its features:

```
\subcaption[{list entry}]{{heading}}
\subcaption*{{heading}}
```

Please note that the `\subcaption` command *must* be applied inside its own box or environment.

An example:

```
\begin{figure}
  \begin{minipage}[b]{.5\linewidth}
    \centering\large A
    \subcaption{A subfigure}\label{fig:1a}
  \end{minipage}%
  \begin{minipage}[b]{.5\linewidth}
    \centering\large B
    \subcaption{Another subfigure}\label{fig:1b}
  \end{minipage}
  \caption{A figure}\label{fig:1}
\end{figure}
```

gives the result:

A (a) A subfigure	B (b) Another subfigure
-----------------------------	-----------------------------------

Figure 4: A figure

- ▲ Prepared with `\DeclareCaptionSubType` (offered by the `caption` package), the `caption` package command `\setcaptionsubtype` becomes available. Analogous to the `\setcaptiontype` command of the `caption` package, the `\setcaptionsubtype` command sets the sub-type of the box or environment (so `\caption` will typeset a sub-caption instead of an ordinary one), places a proper hyperlink anchor (non-starred variant only), executes options associated with the sub-type etc.

The `\subcaption` command is just a simple combination of `\setcaptionsubtype*` and `\caption`.

5 The `\DeclareCaptionSubType` command

`\DeclareCaptionSubType` For using the sub-caption feature of the caption package some commands and counters must be prepared. This is done with

```
\DeclareCaptionSubType[<numbering scheme>]{<type>}
\DeclareCaptionSubType*[<numbering scheme>]{<type>}
```

For the environments `figure` & `table`, and all the ones defined with `\DeclareFloatingEnvironment` offered by the `newfloat` package, this will be done automatically, but for other environments (e.g. the ones defined with `\newfloat` offered by the `float` package or `\DeclareNewFloatType` offered by the `floatrow` package) this has to be done manually.

The starred variant provides the sub-caption numbering format `<type> . <subtype>` (e.g. ‘1.2’) while the non-starred variant simply uses `<subtype>` (e.g. ‘a’).

Own numbering formats can be created by redefining `\thesub<type>`, e.g.

```
\DeclareCaptionSubType*{figure}
\renewcommand\thesubfigure{\thefigure\alph{subfigure}}
```

would give you sub-caption numbers like ‘1b’.

The default numbering scheme is `alph`, but you can use any \LaTeX (or self-defined) command name here which converts a counter to a text value, e.g. `arabic`, `roman`, `Roman`, `alph`, `Alph`, `fnsymbol`, ...

But `\DeclareCaptionSubType` is not only for defining new sub-caption types, you can use this command for re-definitions as well, e.g.

```
\DeclareCaptionSubType*[arabic]{table}
\captionsetup[subtable]{labelformat=simple,labelsep=colon}
```

will give you sub-captions in tables like these ones:

Table 1: Two tables			
1.1: Table one		1.2: Table two	
A	B	E	F
C	D	G	H

▲ `\DeclareCaptionSubType` is an integral part of the caption package kernel.

6 References

The macro `\the<counter>` is not only responsible for the look of the `<counter>`, but for the look of the references typeset with `\ref`, too. References will be prefixed by L^AT_EX with the internal macro `\p@<counter>`.

`\DeclareCaptionSubType` will define both of them for sub-captions (e.g. sub-figure and subtable), and as you have seen in the last section `\DeclareCaptionSubType` will give you some options to control the internal (re-)definition of `\the<counter>` and `\p@<counter>`.

`\thesubfigure` For example `\thesubfigure` and `\p@subfigure` are (as default) internally defined as
`\p@subfigure`

```
\newcommand\thesubfigure{\alph{subfigure}}
\newcommand\p@subfigure{\thefigure}
```

so the label of sub-captions will look like ‘a’ (decorated by the selected label format), while references will look like ‘1a’ since they are prefixed by `\p@subfigure = \thefigure`.

After `\DeclareCaptionSubType*[arabic]{figure}`, `\thesubfigure` and `\p@subfigure` will look like

```
\renewcommand\thesubfigure{\thefigure.\arabic{subfigure}}
\renewcommand\p@subfigure{}
```

But if you want detailed control on how the references will look like, the options of `\DeclareCaptionSubType` are potentially not sufficient. In this case one need to redefine these two macros on his/her own. Some examples:

If you want parentheses around the sub-figure part of the reference, so they will look like ‘1 (a)’, you may get them this way:

```
\usepackage[labelformat=simple]{subcaption}
\renewcommand\thesubfigure{(\alph{subfigure})}
```

(Note: Since `parens` is the default label format you will get double parentheses in sub-captions when not specifying a different label format, e.g. `simple`.)

But if you want only a closing parenthesis, so references will look like ‘1a)’, but the sub-captions itself should still look like ‘(a)’, this would be a possible solution:

```
\usepackage{subcaption}
\renewcommand\thesubfigure{\alph{subfigure}}
\DeclareCaptionLabelFormat{opening}{(#2}
\captionsetup[subfigure]{labelformat=opening}
```

(Please note that you need to surround redefinitions of `\p@<counter>` with `\makeatletter` and `\makeatother`. See <http://tex.stackexchange.com/questions/8351/> for details.)

6.1 The `\subref` command

While `\ref{⟨key⟩}` (and `\ref*{⟨key⟩}`, if the `hyperref` package is used) usually gives a combined result representing the main caption counter and the sub-caption one, it is sometimes useful to have a reference to the sub-caption only. For this purpose you can use

```
\subref{⟨key⟩}
\subref*{⟨key⟩} 3 .
```

So for example `\ref{cat}` gives the result ‘1a’ but `\subref{cat}` gives ‘a’.

Note: If the sub-caption was (re-)defined with the starred variant `\DeclareCaptionSubType*`, both `\ref` and `\subref` usually gives the same result.

▲ The `\subref` command demonstrates the usage of `\caption@subtyphook` which will be called during `\captionsetup{subtype}`.

6.2 The `subrefformat=` option

`subrefformat=` By applying `\DeclareCaptionSubType`, or by redefining `\the⟨counter⟩` and `\p@⟨counter⟩`, you will change the look of references typeset with `\ref` and `\subref`. But maybe you only want to change the output of `\subref` without affecting the references typeset with `\ref`? This is possible, too, by using the option `subrefformat`:

```
\captionsetup{subrefformat=⟨label format⟩}
```

This one will choose a label format (either a pre-defined one, or a one defined with `\DeclareCaptionLabelFormat`) as decorative element to sub-references. The default one is `simple` which has no decorative elements but simply typeset the reference as it is.

For example

```
\captionsetup{subrefformat=parens}
```

will result in references (typeset with `\ref`) like ‘1a’ but sub-references (typeset with `\subref`) like ‘(a)’.

6.3 Referencing sub-figures without sub-captions

`\phantomsubcaption` If you don’t want to give a sub-figure a caption, because the picture itself already contains the caption, or for some other reason, you can use the command

```
\phantomsubcaption
```

instead of `\subcaption`, or – when inside a `subfigure` or `subtable` environment – `\phantomcaption` instead of `\caption`. `\phantomsubcaption` and `\phantomcaption` do not have any arguments, and they do not generate any output, but give you an anchor for a `\label` command which can be placed afterwards. Furthermore it increases the sub-figure resp. sub-table counter.

Please note that – just like `\subcaption` – the `\phantomsubcaption` command *must* be applied inside its own group, box, or environment.

³Like `\ref*`, `\subref*` is only available if the `hyperref` package^[3] is used.

An example:

```
\begin{figure}
\centering
{\includegraphics{cat_with_a}
\phantomsubcaption\label{cat}}
{\includegraphics{elephant_with_b}
\phantomsubcaption\label{elephant}}
\caption{Two animals: \subref{cat} a huge cat,
and \subref{elephant} an elephant}
\end{figure}
```

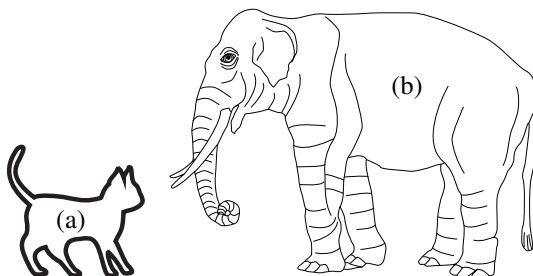


Figure 5: Two animals: (a) a huge cat, and (b) an elephant

6.4 Where to place the `\label` command?

When using `\subcaption` or `\phantomsubcaption`, or `\caption` or `\phantomcaption` inside a subfigure or subtable environment, the `\label` can be either placed inside the caption text or right after the `\subcaption` or `\caption` command, e.g:

```
\subcaption{Some text here\label{label1}}
...
\subcaption{Some other text}\label{label2}}
...
\subcaption{Something different}
\label{label3}
```

When using the `\subcaptionbox` command, the `\label` should be placed inside the caption text, e.g.:

```
\subcaptionbox{A description here\label{label4}}
{Some content here}
...
\subcaptionbox[List-of-Figures entry]
{A description here\label{label5}}
{Some content here}
```

6.5 Where do hyperlinks jump?

For the `subfigure` & `subtable` environments and `\subcaptionbox` boxes (and own constructs which use `\setcaptionsubtype`) the hyperlink anchors will be placed in respect to the `hypcap=` setting. While usage of this option is straight-forward for ordinary captions, the usage for sub-captions depends on the setting regarding the main captions. This table gives you an overview where the hyperlinks will jump:

caption subcaption	hypcap=false	hypcap=true
	hypcap=false	hypcap=true
hypcap=false	sub-caption	figure or table (default setting)
hypcap=true	sub-figure or sub-table	sub-figure or sub-table

But if `\subcaption` is used and `hypcap=true` is set for sub-captions, the `subcaption` package does not know where the sub-figure or sub-table actually begins, so it will jump to the sub-caption instead.

Remember: If you use the `hypcap` package[4], it controls the placement of the hyperlink anchors, making the rules above invalid.

(See also the documentation of the `caption` package, sections about `hyperref` & `hypcap`.)

7 Beyond this package

For a more advanced usage of the sub-caption feature of the `caption` package, please take a look at the `floatrow` package[2] which provides the powerful `subfloatrow` environment for typesetting sub-figures.

8 Thanks

I would like to thank Stephen Dalton who helped to make this package a better one.

References

- [1] Frank Mittelbach and Michel Goossens:
The L^AT_EX Companion (2nd. Ed.), Addison-Wesley, 2004.
- [2] Olga Lapko:
The floatrow package documentation, 2007/12/24
- [3] Sebastian Rahtz & Heiko Oberdiek:
Hypertext marks in L^AT_EX, November 12, 2007
- [4] Heiko Oberdiek:
The hypcap package – Adjusting anchors of captions, 2007/04/09