

# The **hypcap** package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2006/02/20 v1.5

## Abstract

This package tries a solution of the problem with `hyperref`, that links to floats points below the caption and not at the beginning of the float. Therefore this package divides the task into two part, the link setting with `\capstart` or automatically at the beginning of a float and the rest in the `\caption` command.

## Contents

<b>1</b>	<b>Usage</b>	<b>1</b>
1.1	Package options	1
1.2	User commands	2
1.3	Limitations	2
<b>2</b>	<b>Implementation</b>	<b>2</b>
<b>3</b>	<b>Installation</b>	<b>4</b>
3.1	Some details for the interested	5
<b>4</b>	<b>History</b>	<b>5</b>
[1999/02/13 v1.0]		5
[2000/08/14 v1.1]		5
[2000/09/07 v1.2]		5
[2001/08/27 v1.3]		5
[2001/09/06 v1.4]		6
[2006/02/20 v1.5]		6
<b>5</b>	<b>Index</b>	<b>6</b>

## 1 Usage

The package `hypcap` requires that `hyperref` is loaded first:

```
\usepackage[...]{hyperref}
\usepackage[...]{hypcap}
```

### 1.1 Package options

The names of the four float environments `figure`, `figure*`, `table`, or `table*` can be used as option. Then the package redefines the environment in order to insert `\capstart` (see below) in the beginning of the environment automatically.

Option `all` enables the redefinitions of all four float environments. For other environments see the user command `\hypcapredef`.

## 1.2 User commands

`\capstart` **\capstart:** First this command increments the counter (`\@captype`). Then it makes an anchor for package `hyperref`. At last `\caption` is redefined to remove the anchor setting part from `hyperref`'s `\caption`.

The package expects the following structure of a float environment:

```
\begin{float}...
\capstart
...
\caption{...}
...
\end{float}
```

There can be several `\caption` commands. For these you need `\capstart` again:

```
\capstart ... \caption... \capstart ... \caption...
```

And the `\caption` command itself can be put in a group.

With the options, described above, the extra writing of `\capstart` can be avoided. Consequently, there must be a `\caption` in every environment of this type, specified by the option. If you want to use more than one `\caption` in this environment, you have to state `\capstart` again.

`\hyccapspace` **\hyccapspace:** Because it looks poor, if the link points exactly at top of the figure, there is additional space: `\hyccapspace`, the default is `0.5\baselineskip`, examples:

```
\renewcommand{\hyccapspace}{0pt} removes the space
\renewcommand{\hyccapspace}{1pt} sets a fix value
```

`\hyccapredef` **\hyccapredef:** If there are other float environments, that should automatically execute `\capstart`, then a redefinition with `\hyccapredef` can be tried:

```
\hyccapredef{myfloat}
```

Only environments with one optional parameter are supported.

## 1.3 Limitations

- Package `subfigure` does not work.
- Packages that redefine `\caption` or `\@caption`.

## 2 Implementation

```
1 <*package>
```

Package identification.

```
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{hyccap}%
4 [2006/02/20 v1.5 Adjusting anchors of captions (H0)]
```

For unique command names this package uses `hc@` as prefix for internal command names.

First we check, if package `hyperref` is loaded:

```
5 \@ifundefined{hyper@anchor}{%
6   \PackageError{hyccap}{You have to load 'hyperref' first}\@ehc
7   \endinput
8 }{}
```

`\hc@org@caption` Save the original meaning of `\caption`:

```

9 \newcommand*{\hc@org@caption{}}
10 \let\hc@org@caption\caption

\if@capstart The switch \if@capstart helps to detect \capstart commands with missing
\caption macros. Because \caption can occur inside a group, assignments to
the switch have to be made global.
11 \newif\if@capstart

\hypcapspace The anchor is raised by \hypcapspace.
12 \newcommand*\hypcapspace{.5\baselineskip}

\capstart The macro \capstart contains the first part of the \caption command: Incre-
menting the counter and setting the anchor.
13 \newcommand*\capstart{%
14   \H@refstepcounter\@captype % first part of caption
15   \hyper@makecurrent\@captype
16   \vspace*{~\hypcapspace}%
17   \begingroup
18     \let\leavevmode\relax
19     \hyper@@anchor\@currentHref\relax
20   \endgroup
21   \vspace*{\hypcapspace}%
22   \let\caption\hc@caption
23   \global\@capstarttrue
24 }

\hc@caption The new \caption command without the first part is defined in the macro
\hc@caption.
25 \def\hc@caption{%
26   \@dblarg{\hc@caption\@captype}%
27 }

\hc@@caption This is a copy of package hyperref's \@caption macro without making the anchor,
because this is already done in \capstart.
28 \long\def\hc@@caption#1[#2]#3{%
29   \let\caption\hc@org@caption
30   \global\@capstartfalse
31   \hyper@makecurrent\@captype
32   \par\addcontentsline{%
33     \csname ext@#1\endcsname}{#1}{%
34       \protect\numberline{%
35         \csname the#1\endcsname
36       }}{\ignorespaces #2}%
37   }%
38   \begingroup
39     \@parboxrestore
40     \normalsize
41     \@makecaption{\csname fnum@#1\endcsname}{%
42       \ignorespaces#3%
43     }%
44   \par
45   \endgroup
46 }

\hypcapredef The macro \hypcapredef prepares the call of \hc@redef that will redefine the
environment that is given in the argument.
47 \def\hypcapredef#1{%
48   \expandafter\hc@redef\csname hc@org#1\expandafter\endcsname
49     \csname hc@orgend#1\expandafter\endcsname
50     \expandafter{#1}%
51 }

```

```

\hc@redef The old meaning of the environment is saved. Then \capstart is appended in
the begin part. The end part contains a check that produces an error message in
case of \capstart without \capstart (\capstart has incremented the counter).
52 \def\hc@redef#1#2#3{%
53   \newcommand#1{%
54     \expandafter\let\expandafter#1\csname#3\endcsname
55     \expandafter\let\expandafter#2\csname end#3\endcsname
56     \renewenvironment*{#3}[1][]{%
57       \ifx\##1\%
58         #1\relax
59       \else
60         #1[\##1]%
61       \fi
62       \capstart
63     }{%
64       \if@capstart
65         \PackageError{hypcap}{You have forgotten to use \string\caption}%
66         \global\@capstartfalse
67       \else
68         \fi
69       #2%
70     }%
71 }

```

At last the options are defined and processed.

```

72 \DeclareOption{figure}{\hypcapredef{\CurrentOption}}
73 \DeclareOption{figure*}{\hypcapredef{\CurrentOption}}
74 \DeclareOption{table}{\hypcapredef{\CurrentOption}}
75 \DeclareOption{table*}{\hypcapredef{\CurrentOption}}
76 \DeclareOption{all}{%
77   \hypcapredef{figure}%
78   \hypcapredef{figure*}%
79   \hypcapredef{table}%
80   \hypcapredef{table*}%
81 }
82 \ProcessOptions\relax
83 \</package>

```

### 3 Installation

**CTAN.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/hypcap.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/hypcap.pdf](#) Documentation.

**Unpacking.** The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain-TeX:

```
tex hypcap.dtx
```

**TDS.** Now the different files must be moved into the different directories in your installation TDS tree (also known as texmf tree):

```

hypcap.sty → tex/latex/oberdiek/hypcap.sty
hypcap.pdf → doc/latex/oberdiek/hypcap.pdf
hypcap.dtx → source/latex/oberdiek/hypcap.dtx

```

If you have a docstrip.cfg that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

<sup>1</sup><http://ftp.ctan.org/tex-archive/>

**Refresh file databases.** If your  $\text{T}_{\text{E}}\text{X}$  distribution ( $\text{t}_{\text{E}}\text{X}$ ,  $\text{m}_{\text{ik}}\text{T}_{\text{E}}\text{X}$ , ...) rely on file databases, you must refresh these. For example,  $\text{t}_{\text{E}}\text{X}$  users run `texhash` or `mktextlsr`.

### 3.1 Some details for the interested

**Attached source.** The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk hypcap.pdf unpack_files output .
```

**Unpacking with  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .** The `.dtx` chooses its action depending on the format:

**plain- $\text{T}_{\text{E}}\text{X}$ :** Run `docstrip` and extract the files.

**$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ :** Generate the documentation.

If you insist on using  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  for `docstrip` (really, `docstrip` does not need  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ), then inform the autodetect routine about your intension:

```
latex \install=y\input{hypcap.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

**Generating the documentation.** You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with `pdf $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$` :

```
pdflatex hypcap.dtx
makeindex -s gind.ist hypcap.idx
pdflatex hypcap.dtx
makeindex -s gind.ist hypcap.idx
pdflatex hypcap.dtx
```

## 4 History

[1999/02/13 v1.0]

- A beginning version.

[2000/08/14 v1.1]

- Global assignments of `\if@capstart` in order to allow `\caption` in groups.
- Option `all` added.

[2000/09/07 v1.2]

- Package in `dtx` format.

[2001/08/27 v1.3]

- Bug fix with `hyperref`'s `pdfmark` driver  
(`\leavevmode` in `\hyper@@anchor/\pdf@rect`).

[2001/09/06 v1.4]

- Small fixes in the dtx file.

[2006/02/20 v1.5]

- Code is not changed.
- New DTX framework.

## 5 Index

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

Symbols	
<code>\@capstartfalse</code> .....	30, 66
<code>\@capstarttrue</code> .....	23
<code>\@captype</code> .....	14, 15, 26, 31
<code>\@currentHref</code> .....	19
<code>\@dblarg</code> .....	26
<code>\@ehc</code> .....	6
<code>\@ifundefined</code> .....	5
<code>\@makecaption</code> .....	41
<code>\@parboxrestore</code> .....	39
<code>\@</code> .....	57
A	
<code>\addcontentsline</code> .....	32
B	
<code>\baselineskip</code> .....	12
C	
<code>\capstart</code> .....	2, 13, 62
<code>\caption</code> .....	10, 22, 29, 65
<code>\csname</code> .....	33, 35, 41, 48, 49, 54, 55
<code>\CurrentOption</code> .....	72, 73, 74, 75
D	
<code>\DeclareOption</code> .....	72, 73, 74, 75, 76
E	
<code>\endcsname</code> ..	33, 35, 41, 48, 49, 54, 55
<code>\endinput</code> .....	7
H	
<code>\H@refstepcounter</code> .....	14
<code>\hc@caption</code> .....	26, <u>28</u>
<code>\hc@caption</code> .....	22, <u>25</u>
<code>\hc@org@caption</code> .....	<u>9</u> , 29
<code>\hc@redef</code> .....	48, <u>52</u>
<code>\hypcapredef</code> .....	2, <u>47</u> , 72, 73, 74, 75, 77, 78, 79, 80
<code>\hypcapspace</code> .....	2, <u>12</u> , 16, 21
<code>\hyper@@anchor</code> .....	19
<code>\hyper@makecurrent</code> .....	15, 31
I	
<code>\if@capstart</code> .....	<u>11</u> , 11, 64
<code>\ifx</code> .....	57
<code>\ignorespaces</code> .....	36, 42
L	
<code>\leavevmode</code> .....	18
N	
<code>\NeedsTeXFormat</code> .....	2
<code>\newcommand</code> .....	9, 12, 13, 53
<code>\newif</code> .....	11
<code>\normalsize</code> .....	40
<code>\numberline</code> .....	34
P	
<code>\PackageError</code> .....	6, 65
<code>\par</code> .....	32, 44
<code>\ProcessOptions</code> .....	82
<code>\protect</code> .....	34
<code>\ProvidesPackage</code> .....	3
R	
<code>\renewenvironment</code> .....	56
V	
<code>\vspace</code> .....	16, 21