

The **hypcap** package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2008/04/14 v1.8

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

1	Usage	1
1.1	Package options	2
1.2	User commands	2
1.3	Limitations	2
2	Implementation	2
3	Installation	5
3.1	Download	5
3.2	Bundle installation	5
3.3	Package installation	5
3.4	Refresh file name databases	5
3.5	Some details for the interested	6
4	History	6
	[1999/02/13 v1.0]	6
	[2000/08/14 v1.1]	6
	[2000/09/07 v1.2]	6
	[2001/08/27 v1.3]	6
	[2001/09/06 v1.4]	6
	[2006/02/20 v1.5]	7
	[2007/02/19 v1.6]	7
	[2007/04/09 v1.7]	7
	[2008/04/14 v1.8]	7
5	Index	7

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 `\hyccapredef`.

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

- Packages that redefine `\caption` or `\@caption`.

2 Implementation

```
1 (*package)
```

Package identification.

```
2 \NeedsTeXFormat{LaTeX2e}
```

```
3 \ProvidesPackage{hyccap}%
```

```
4 [2008/04/14 v1.8 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{hypcap}{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   \global\let\hc@currentHref\@currentHref
17   \vspace*{-\hypcapspace}%
18   \begingroup
19     \let\leavevmode\relax
20     \hyper@@anchor\@currentHref\relax
21   \endgroup
22   \vspace*{\hypcapspace}%
23   \hc@hyperref{\let\caption\hc@caption}%
24   \global\@capstarttrue
25   \global\advance\csname c@\@captype\endcsname\m@ne
26 }

27 \@ifpackagelater{hyperref}{2007/04/09}{%
28   \let\hc@hyperref\@gobble
29 }{%
30   \let\hc@hyperref\@firstofone
31 }

\hc@caption The new \caption command without the first part is defined in the macro
\hc@caption.
32 \def\hc@caption{%
33   \global\advance\csname c@\@captype\endcsname\@ne
34   \@dblarg{\hc@caption\@captype}%
35 }

\hc@@caption This is a copy of package hyperref's \@caption macro without making the anchor,
because this is already done in \capstart.
36 \long\def\hc@@caption#1[#2]#3{%
37   \let\caption\hc@org@caption
38   \global\@capstartfalse
39   \ifHy@hypertextnames
40     \hyper@makecurrent\@captype
41   \else
42     \global\let\@currentHref\hc@currentHref
43   \fi
44   \par\addcontentsline{%
45     \csname ext@#1\endcsname}{#1}{%
46     \protect\numberline{%
47       \csname the#1\endcsname

```

```

48   }\ignorespaces #2}%
49 }%
50 \begingroup
51   \@parboxrestore
52   \normalsize
53   \@makecaption{\csname fnum@#1\endcsname}{%
54     \ignorespaces#3%
55   }%
56   \par
57 \endgroup
58 }

```

`\hycapredef` The macro `\hycapredef` prepares the call of `\hc@redef` that will redefine the environment that is given in the argument.

```

59 \def\hycapredef#1{%
60   \expandafter\hc@redef\csname hc@org#1\expandafter\endcsname
61   \csname hc@orgend#1\expandafter\endcsname
62   \expandafter{#1}%
63 }

```

`\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).

```

64 \def\hc@redef#1#2#3{%
65   \newcommand#1{%
66     \expandafter\let\expandafter#1\csname#3\endcsname
67     \expandafter\let\expandafter#2\csname end#3\endcsname
68     \renewenvironment*{#3}[1][1]{%
69       \ifx\##1\%
70         #1\relax
71       \else
72         #1[\##1]% hash-ok (compatibility for float)
73       \fi
74       \capstart
75     }{%
76       \if@capstart
77         \PackageError{hycap}{You have forgotten to use \string\caption}%
78         \global\@capstartfalse
79       \else
80         \fi
81       #2%
82     }%
83 }

```

At last the options are defined and processed.

```

84 \DeclareOption{figure}{\hycapredef{\CurrentOption}}
85 \DeclareOption{figure*}{\hycapredef{\CurrentOption}}
86 \DeclareOption{table}{\hycapredef{\CurrentOption}}
87 \DeclareOption{table*}{\hycapredef{\CurrentOption}}
88 \DeclareOption{all}{%
89   \hycapredef{figure}%
90   \hycapredef{figure*}%
91   \hycapredef{table}%
92   \hycapredef{table*}%
93 }
94 \ProcessOptions\relax
95 \</package>

```

3 Installation

3.1 Download

Package. This package is available on CTAN¹:

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

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

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for \TeX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

3.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

3.3 Package installation

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`.

3.4 Refresh file name databases

If your \TeX distribution (te \TeX , mi \TeX , ...) relies on file name databases, you must refresh these. For example, te \TeX users run `texhash` or `mktextlsr`.

¹<http://ftp.ctan.org/tex-archive/>

3.5 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 \LaTeX . The `.dtx` chooses its action depending on the format:

plain- \TeX : Run `docstrip` and extract the files.

\LaTeX : Generate the documentation.

If you insist on using \LaTeX for `docstrip` (really, `docstrip` does not need \LaTeX), then inform the autodetect routine about your intention:

```
latex \let\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 \LaTeX` :

```
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.

[2007/02/19 v1.6]

- Fix for `hypertextnames=false`.

[2007/04/09 v1.7]

- Stuff in `\caption` moved to `hyperref`. This avoids redefinitions of `\caption` and `\@caption` (idea of Axel Sommerfeldt).
- Fix for subfigure (Marco Kuhlmann, Amilcar do Carmo Lucas).

[2008/04/14 v1.8]

- `\hc@redef` fixed to get package float work (Axel Sommerfeldt).

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		\endinput 7	
\@capstartfalse	38, 78	H	
\@capstarttrue	24		
\@captype	14, 15, 25, 33, 34, 40	\H@refstepcounter	14
\@currentHref	16, 20, 42	\hc@caption	34, 36
\@dblarg	34	\hc@caption	23, 32
\@ehc	6	\hc@currentHref	16, 42
\@firstofone	30	\hc@hyperref	23, 28, 30
\@gobble	28	\hc@org@caption	9, 37
\@ifpackagelater	27	\hc@redef	60, 64
\@ifundefined	5	\hycapredef	2,
\@makecaption	53	59, 84, 85, 86, 87, 89, 90, 91, 92	
\@ne	33	\hycapspace	2, 12, 17, 22
\@parboxrestore	51	\hyper@anchor	20
\\	69	\hyper@makecurrent	15, 40
A		I	
\addcontentsline	44	\if@capstart	11, 11, 76
\advance	25, 33	\ifHy@hypertextnames	39
B		\ifx	69
\baselineskip	12	\ignorespaces	48, 54
C		L	
\capstart	2, 13, 74	\leavevmode	19
\caption	10, 23, 37, 77	M	
\csname 25, 33, 45, 47, 53, 60, 61, 66, 67		\m@ne	25
\CurrentOption	84, 85, 86, 87	N	
D		\NeedsTeXFormat	2
\DeclareOption	84, 85, 86, 87, 88	\newcommand	9, 12, 13, 65
E		\newif	11
\endcsname		\normalsize	52
25, 33, 45, 47, 53, 60, 61, 66, 67		\numberline	46

P		R	
<code>\PackageError</code>	6, 77	<code>\renewenvironment</code>	68
<code>\par</code>	44, 56		
<code>\ProcessOptions</code>	94		
<code>\protect</code>	46	V	
<code>\ProvidesPackage</code>	3	<code>\vspace</code>	17, 22