The hypcap package

Heiko Oberdiek
<heiko.oberdiek at googlemail.com>

2008/09/08 v1.10

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

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.

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

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

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

```
\hypcapredef{myfloat}
```

Only environments with one optional parameter are supported.

`\capstartfalse` `\capstarttrue` **`\capstartfalse`, `\capstarttrue`:** Since 2008/09/08 v1.10. They disable and enable `\capstart`. They can be used to cancel the effect of a redefined float environment. Example:

```
\documentclass{article}
\usepackage{hyperref}
\usepackage[figure]{hypcap}[2008/09/08]

\begin{document}
  \section{Hello World}
  \begin{figure}
    \caption{Figure with caption A}
  \end{figure}
  \capstartfalse
  \begin{figure}
    Figure without caption
  \end{figure}
  \capstarttrue
  \begin{figure}
    \caption{Figure with caption B}
  \end{figure}
\end{document}
```

1.3 Limitations

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

2 Implementation

```
1 ⟨*package⟩
```

Package identification.

```
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{hypcap}%
4   [2008/09/08 v1.10 Adjusting anchors of captions (HO)]
```

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}
```

`\ifcapstart`

```
13 \newif\ifcapstart
14 \capstarttrue
```

`\capstart` The macro `\capstart` contains the first part of the `\caption` command: Incrementing the counter and setting the anchor.

```
15 \newcommand*\capstart{%
16   \ifcapstart
17     \H@refstepcounter\@captype % first part of caption
18     \hyper@makecurrent\@captype
```

```
    \global\let\hc@currentHref\@currentHref
    \vspace*{-\hypcapspace}%
    \begingroup
      \let\leavevmode\relax
      \hyper@@anchor\@currentHref\relax
    \endgroup
    \vspace*{\hypcapspace}%
    \hc@hyperref{\let\caption\hc@caption}%
    \global\@capstarttrue
    \global\advance\csname c@\@captype\endcsname\m@ne
  \fi
}

\@ifpackagelater{hyperref}{2007/04/09}{%
  \let\hc@hyperref\@gobble
}{%
  \let\hc@hyperref\@firstofone
}
```

`\hc@caption` The new `\caption` command without the first part is defined in the macro `\hc@caption`.

```
\def\hc@caption{%
  \global\advance\csname c@\@captype\endcsname\@ne
  \@dblarg{\hc@@caption\@captype}%
}
```

`\hc@@caption` This is a copy of package `hyperref`'s `\@caption` macro without making the anchor, because this is already done in `\capstart`.

```
\long\def\hc@@caption#1[#2]#3{%
  \let\caption\hc@org@caption
  \global\@capstartfalse
  \ifHy@hypertexnames
    \hyper@makecurrent\@captype
  \else
    \global\let\@currentHref\hc@currentHref
  \fi
  \par\addcontentsline{%
    \csname ext@#1\endcsname}{#1}{%
    \protect\numberline{%
      \csname the#1\endcsname
    }{\ignorespaces #2}%
  }%
  \begingroup
    \@parboxrestore
    \normalsize
    \@makecaption{\csname fnum@#1\endcsname}{%
      \ignorespaces#3%
    }%
    \par
  \endgroup
}
```

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

```
\def\hypcapredef#1{%
  \expandafter\hc@redef\csname hc@org#1\expandafter\endcsname
                       \csname hc@orgend#1\expandafter\endcsname
                       \expandafter{#1}%
}
```

`\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).

```
\def\hc@redef#1#2#3{%
  \newcommand#1{}%
  \expandafter\let\expandafter#1\csname#3\endcsname
  \expandafter\let\expandafter#2\csname end#3\endcsname
  \renewenvironment*{#3}[1][]{%
    \ifx\\##1\\%
      #1\relax
    \else
      #1[##1]% hash-ok (compatibility for float)
    \fi
    \capstart
  }{%
    \if@capstart
      \PackageError{hypcap}{You have forgotten to use \string\caption}%
      \global\@capstartfalse
    \else
    \fi
    #2%
  }%
}
```

At last the options are defined and processed.

```
\DeclareOption{figure}{\hypcapredef{\CurrentOption}}
\DeclareOption{figure*}{\hypcapredef{\CurrentOption}}
\DeclareOption{table}{\hypcapredef{\CurrentOption}}
\DeclareOption{table*}{\hypcapredef{\CurrentOption}}
\DeclareOption{all}{%
  \hypcapredef{figure}%
  \hypcapredef{figure*}%
  \hypcapredef{table}%
  \hypcapredef{table*}%
}
\ProcessOptions\relax
```

⟨/package⟩

3 Installation

3.1 Download

Package. This package is available on CTAN[1]:

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
```

[1] ftp://ftp.ctan.org/tex-archive/

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 (teTEX, mikTEX, ...) relies on file name databases, you must refresh these. For example, teTEX users run `texhash` or `mktexlsr`.

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 pdfLATEX:

```
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, published in newsgroup comp.text.tex: "Re: hyperref and figures"[2]

[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 hypertexnames=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).

[2008/08/11 v1.9]

- Code is not changed.
- URLs updated.

[2008/09/08 v1.10]

- `\capstartfalse` and `\capstarttrue` added.

[2] Url: http://groups.google.com/group/comp.text.tex/msg/5c9b47b001a9379c

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; plain numbers refer to the code lines where the entry is used.