The **hypbmsec** package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2006/02/20 v2.2

Abstract

This package expands the syntax of the sectioning commands. If the argument of the sectioning commands isn't usable as outline entry, a replacement for the bookmarks can be given.

Contents

1 Usage

1.1 Bookmarks restrictions

Outline entries (bookmarks) are written to a file and have to obey the pdf specification. Therefore they have several restrictions:

- Bookmarks have to be encoded in PDFDocEncoding[1].

---

[1] `hyperref` doesn't support Unicode.

- They should only expand to letters and spaces.

- The result of expansion have to be a valid pdf string.

- Stomach commands like `\relax`, box commands, math, assignments, or definitions aren't allowed.

- Short entries are recommended, which allow a clear view.

1.2 \texorpdfstring

The generic way in package `hyperref` is the use of `\texorpdfstring`[2]:

```
\section{Pythagoras:
  \texorpdfstring{$a^2+b^2=c^2}{%
    a\texttwosuperior\ + b\texttwosuperior\ =
    c\texttwosuperior}%
}
```

1.3 Sectioning commands

The package `hyperref` automatically generates bookmarks from the sectioning commands, unless it is suppressed by an option. Commands that structure the text are here called "sectioning commands":

```
\part, \chapter,
\section, \subsection, \subsubsection,
\paragraph, \subparagraph
```

1.4 Places for sectioning strings

The argument(s) of these commands are used on several places:

text The current text without restrictions.

toc The headlines and the table of contents with the restrictions of "moving arguments".

out The outlines with many restrictions: The outline have to expand to a valid pdf string with PDFDocEncoding (see 1.1).

1.5 Solution with optional arguments

If the user wants to use a footnote within a sectioning command, the LaTeX solution is an optional argument:

```
\section[Title]{Title\footnote{Footnote text}}
```

Now `Title` without the footnote is used in the headlines and the table of contents. Also `hyperref` uses it for the bookmarks.

This package `hypbmsec.drv` offers two possibilities to specify a separate outline entry:

- An additional second optional argument in square brackets.

- An additional optional argument in parentheses (in assoziation with a pdf string that is internally surrounded by parentheses, too).

Because `hypbmsec.drv` stores the original meaning of the sectioning commands and uses them again, there should be no problems with packages that redefine the sectioning commands, if these packages doesn't change the syntax.

---

[2]In versions of `hyperref` below 6.54 see `\ifbookmark`.

1.6 Syntax

The following examples show the syntax of the sectioning commands. For the places the strings appear the abbreviations are used, that are introduced in 1.4.

1.6.1 Star form

The behaviour of the star form isn't changed. The string appears only in the current text:

```
\section*{text}
```

1.6.2 Without optional arguments

The normal case, the string in the mandatory argument is used for all places:

```
\section{text, toc, out}
```

1.6.3 One optional argument

Also the form with one optional parameter in square brackets isn't new; for the bookmarks the optional parameter is used:

```
\section[toc, out]{text}
```

1.6.4 Two optional arguments

The second optional parameter in square brackets is introduced by this package to specify an outline entry:

```
\section[toc][out]{text}
```

1.6.5 Optional argument in parentheses

Often the toc and the text string would be the same. With the method of the two optional arguments in square brackets (see 1.6.4) this string must be given twice, if the user only wants to specify a different outline entry. Therefore this package offers another possibility: In association with the internal representation in the pdf file an outline entry can be given in parentheses. So the package can easily distinguish between the two forms of optional arguments and the order does not matter:

```
\section(out){toc, text}
\section[toc](out){text}
\section(out)[toc]{text}
```

1.7 Without hyperref

Package hypbmsec.drv uses hyperref for support of the bookmarks, but this package is not required. If hyperref isn't loaded, or is called with a driver that doesn't support bookmarks, package hypbmsec.drv shouldn't be removed, because this would lead to a wrong syntax of the sectioning commands. In any cases package hypbmsec.drv supports its syntax and ignores the outline entries, if there are no code for bookmarks. So it is possible to write texts, that are processed with several drivers to get different output formats.

1.8 Protecting parentheses

If the string itself contains parentheses, they have to be hidden from TEX's argument parsing mechanism. The argument should be surrounded by curly braces:

```
\section({outlines(bookmarks)}){text, toc}
```

With version 6.54 of `hyperref` the other standard method works, too: The closing parenthesis is protected:

```
\section(outlines(bookmarks{)}){text, toc}
```

2 Implementation

```
1 ⟨*package⟩
```

Package identification.

```
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{hypbmsec}%
4   [2006/02/20 v2.2 Bookmarks in sectioning commands (HO)]
```

Because of redifining the sectioning commands, it is dangerous to reload the package several times.

```
5 \@ifundefined{hbs@do}{}{%
6   \PackageInfo{hypbmsec}{Package 'hypbmsec' is already loaded}
7   \endinput
8 }
```

`\hbs@do`   The redefined sectioning commands calls `\hbs@do`. It does

- handle the star case.

- resets the macros that store the entries for the outlines (`\hbs@bmstring`) and table of contents (`\hbs@tocstring`).

- store the sectioning command #1 in `\hbs@seccmd` for later reuse.

- at last call `\hbs@checkarg` that scans and interprets the parameters of the redefined sectioning command.

```
 9 \def\hbs@do#1{%
10   \@ifstar{#1*}{%
11     \let\hbs@tocstring\relax
12     \let\hbs@bmstring\relax
13     \let\hbs@seccmd#1%
14     \hbs@checkarg
15   }%
16 }
```

`\hbs@checkarg`   `\hbs@checkarg` determines the type of the next argument:

- An optional argument in square brackets can be an entry for the table of contents or the bookmarks. It will be read by `\hbs@getsquare`

- An optional argument in parentheses is an outline entry. This is worked off by `\hbs@getbookmark`.

- If there are no more optional arguments, `\hbs@process` reads the mandatory argument and calls the original sectioning commands.

```
17 \def\hbs@checkarg{%
18   \@ifnextchar[\hbs@getsquare{%
19     \@ifnextchar(\hbs@getbookmark\hbs@process
20   }%
21 }
```

\hbs@getsquare \hbs@getsquare reads an optional argument in square brackets and determines, if this is an entry for the table of contents or the bookmarks.

```
22 \long\def\hbs@getsquare[#1]{%
23   \ifx\hbs@tocstring\relax
24     \def\hbs@tocstring{#1}%
25   \else
26     \hbs@bmdef{#1}%
27   \fi
28   \hbs@checkarg
29 }
```

\hbs@getbookmark \hbs@getbookmark reads an outline entry in parentheses.

```
30 \def\hbs@getbookmark(#1){%
31   \hbs@bmdef{#1}%
32   \hbs@checkarg
33 }
```

\hbs@bmdef The command \hbs@bmdef save the bookmark entry in parameter #1 in the macro \hbs@bmstring and catches the case, if the user has given several outline strings.

```
34 \def\hbs@bmdef#1{%
35   \ifx\hbs@bmstring\relax
36     \def\hbs@bmstring{#1}%
37   \else
38     \PackageError{hypbmsec}{%
39       Sectioning command with too many parameters%
40     }{%
41       You can only give one outline entry.%
42     }%
43   \fi
44 }
```

\hbs@process The parameter #1 is the mandatory argument of the sectioning commands. \hbs@process calls the original sectioning command stored in \hbs@seccmd with arguments that depend of which optional argument are used previously.

```
45 \long\def\hbs@process#1{%
46   \ifx\hbs@tocstring\relax
47     \ifx\hbs@bmstring\relax
48       \hbs@seccmd{#1}%
49     \else
50       \hbs@seccmd{\texorpdfstring{#1}\hbs@bmstring}%
51     \fi
52   \else
53     \ifx\hbs@bmstring\relax
54       \hbs@seccmd[\hbs@tocstring]{#1}%
55     \else
56       \hbs@seccmd[\texorpdfstring\hbs@tocstring\hbs@bmstring]{#1}%
57     \fi
58   \fi
59 }
```

We have to check, whether package hyperref is loaded and have to provide a definition for \texorpdfstring. Because hyperref can be loaded after this package, we do the work later (\AtBeginDocument).

This code only checks versions of hyperref that define \ifbookmark (v6.4x until v6.53) or \texorpdfstring (v6.54 and above). Older versions aren't supported.

```
60 \AtBeginDocument{%
61   \@ifundefined{texorpdfstring}{%
62     \@ifundefined{ifbookmark}{%
63       \let\texorpdfstring\@firstoftwo
64       \@ifpackageloaded{hyperref}{%
```

```
65        \PackageInfo{hypbmsec}{%
66          \ifx\hy@driver\@empty
67            Default driver
68          \else
69            '\hy@driver'
70          \fi
71          of hyperref not supported,\MessageBreak
72          bookmark parameters will be ignored%
73        }%
74      }{%
75        \PackageInfo{hypbmsec}{%
76          Package hyperref not loaded,\MessageBreak
77          bookmark parameters will be ignored%
78        }%
79      }%
80    }%
81    {%
82      \newcommand\texorpdfstring[2]{\ifbookmark{#2}{#1}}%
83      \PackageWarningNoLine{hypbmsec}{%
84        Old hyperref version found,\MessageBreak
85        update of hyperref recommended%
86      }%
87    }%
88  }{}%
```

Other packages are allowed to redefine the sectioning commands, if they does not change the syntax. Therefore the redefinitons of this package should be done after the other packages.

```
89  \let\hbs@part          \part
90  \let\hbs@section       \section
91  \let\hbs@subsection    \subsection
92  \let\hbs@subsubsection\subsubsection
93  \let\hbs@paragraph     \paragraph
94  \let\hbs@subparagraph \subparagraph
95  \renewcommand\part          {\hbs@do\hbs@part}
96  \renewcommand\section       {\hbs@do\hbs@section}
97  \renewcommand\subsection    {\hbs@do\hbs@subsection}
98  \renewcommand\subsubsection{\hbs@do\hbs@subsubsection}
99  \renewcommand\paragraph     {\hbs@do\hbs@paragraph}
100 \renewcommand\subparagraph {\hbs@do\hbs@subparagraph}
101 \begingroup\expandafter\expandafter\expandafter\endgroup
102 \expandafter\ifx\csname chapter\endcsname\relax\else
103   \let\hbs@chapter    \chapter
104   \renewcommand\chapter     {\hbs@do\hbs@chapter}
105 \fi
106 }
```

```
107 ⟨/package⟩
```

3 Installation

CTAN. This package is available on CTAN[3]:

CTAN:macros/latex/contrib/oberdiek/hypbmsec.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/hypbmsec.pdf Documentation.

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

    tex hypbmsec.dtx

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

```
hypbmsec.sty  →  tex/latex/oberdiek/hypbmsec.sty
hypbmsec.pdf  →  doc/latex/oberdiek/hypbmsec.pdf
hypbmsec.dtx  →  source/latex/oberdiek/hypbmsec.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.

Refresh file databases. If your TEX distribution (teTEX, mikTEX, . . .) rely on file databases, you must refresh these. For example, teTEX users run `texhash` or `mktexlsr`.

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 hypbmsec.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{hypbmsec.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 hypbmsec.dtx
makeindex -s gind.ist hypbmsec.idx
pdflatex hypbmsec.dtx
makeindex -s gind.ist hypbmsec.idx
pdflatex hypbmsec.dtx
```

4 History

[1998/11/20 v1.0]

- First version.

- It merges package hysecopt and

- package hypbmpar.

- Published for the DANTE'99 meating[4].

---

[4]Url: http://dante99.cs.uni-dortmund.de/handouts/oberdiek/hypbmsec.sty

[1999/04/12 v2.0]

- Adaptation to hyperref version 6.54.

- Documentation in dtx format.

- Copyright: LPPL (CTAN:macros/latex/base/lppl.txt)

- First CTAN release.

[2000/03/22 v2.1]

- Bug fix in redefinition of \chapter.

- Copyright: LPPL 1.2

[2006/02/20 v2.2]

- Code is not changed.

- New DTX framework.

- LPPL 1.3

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.