

The hypdestopt package

Heiko Oberdiek*
<heiko.oberdiek at googlemail.com>

2016/05/21 v2.5

Abstract

Package hypdestopt supports hyperref's pdftex driver. It removes unnecessary destinations and shortens the destination names or uses numbered destinations to get smaller PDF files.

Contents

1	User interface	2
1.1	Introduction	2
1.2	Requirements	2
1.3	Use	2
1.4	Limitations	3
1.5	Future	3
2	Implementation	3
2.1	Identification	3
2.2	Options	3
2.2.1	Option verbose	3
2.2.2	Options num and name	4
2.3	Check requirements	4
2.4	Preamble for auxiliary file	4
2.5	Generation of destination names	5
2.6	Assign destination names	6
2.7	Redefinition of hyperref's hooks	7
2.7.1	Destination setting	7
2.7.2	Links	8
2.7.3	Outlines of package hyperref	8
2.7.4	Outlines of package bookmark	9
3	Installation	9
3.1	Download	9
3.2	Bundle installation	9
3.3	Package installation	10
3.4	Refresh file name databases	10
3.5	Some details for the interested	10
4	Catalogue	11
5	References	11

*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

6 History	11
[2006/06/01 v1.0]	11
[2006/06/01 v2.0]	11
[2007/11/11 v2.1]	11
[2008/08/08 v2.2]	12
[2011/05/13 v2.3]	12
[2016/05/16 v2.4]	12
[2016/05/21 v2.5]	12
7 Index	12

1 User interface

1.1 Introduction

Before PDF-1.5 annotations and destinations cannot be compressed. If the destination names are not needed for external use, the file size can be decreased by the following means:

- Unused destinations are removed.
- The destination names are shortened (option `name`).
- Using numbered destinations (option `num`).

1.2 Requirements

- Package `hyperref` 2006/06/01 v6.75a or newer ([2]).
- Package `alphalph` 2006/05/30 v1.4 or newer ([1]), if option `name` is used.
- Package `ifpdf` ([3]).
- pdfTeX 1.30.0 or newer.
- pdfTeX in PDF mode.
- ϵ -TeX extensions enabled.
- Probably an additional compile run of pdfLaTeX is necessary.

In the first compile runs you can get warnings such as:

```
! pdfTeX warning (dest): name{...} has been referenced ...
```

These warnings should vanish in later compile runs. However these warnings also can occur without this package. The package does not cure them, thus these warnings will remain, but the destination name can be different. In such cases test without package, too.

1.3 Use

If the requirements are met, load the package:

```
\usepackage{hypdestopt}
```

The following options are supported:

verbose: Verbose debug output is enabled and written in the protocol file.

num: Numbered destinations are used. The file size is smaller, because names are no longer used. This is the default.

name: Destinations are identified by names.

1.4 Limitations

- Forget this package, if you need preserved destination names.
- Destination name strings use all bytes (0..255) except the carriage return (13), left parenthesis (40), right parenthesis (41), and backslash (92), because they must be quoted in general and therefore occupy two bytes instead of one.

Further the zero byte (0) is avoided for programs that implement strings using zero terminated C strings. And 255 (0xFF) is avoided to get rid of a possible unicode marker at the begin.

So far I have not seen problems with:

- AcrobatReader 5.08/Linux
- AcrobatReader 7.0/Linux
- xpdf 3.00
- Ghostscript 8.50
- gv 3.5.8
- GSview 4.6

But I have not tested all and all possible PDF viewers.

- Use of named destinations (`\pdfdest`, `\pdfoutline`, `\pdfstartlink`, ...) that are not supported by this package.
- Currently only `hyperref` with `pdfTeX` in PDF mode is supported.

1.5 Future

A more general approach is a PDF postprocessor that takes a PDF file, performs some transformations and writes the result in a more optimized PDF file. Then it does not depend, how the original PDF file was generated and further improvements are easier to apply. For example, the destination names could be sorted: often used destination names would then be shorter than seldom used ones.

2 Implementation

2.1 Identification

```
1 (*package)
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{hypdestopt}%
4 [2016/05/21 v2.5 Hyperref destination optimizer (HO)]%
```

2.2 Options

2.2.1 Option `verbose`

```
5 \newif\ifHypDest@Verbose
6 \DeclareOption{verbose}{\HypDest@Verbosetrue}
```

`\HypDest@VerboseInfo` Wrapper for verbose messages.

```
7 \def\HypDest@VerboseInfo#1{%
8   \ifHypDest@Verbose
9     \PackageInfo{hypdestopt}{#1}%
10  \fi
11 }
```

2.2.2 Options num and name

The options `num` or `name` specify the method, how destinations are referenced (by name or number). Default is option `num`.

```
12 \newif\ifHypDest@name
13 \DeclareOption{num}{\HypDest@namefalse}
14 \DeclareOption{name}{\HypDest@nametrue}
15 \ProcessOptions*\relax
```

2.3 Check requirements

First pdfTeX must running in PDF mode.

```
16 \RequirePackage{ifpdf}[2007/09/09]
17 \RequirePackage{pdftexcmds}[2007/11/11]
18 \ifpdf
19 \else
20 \PackageError{hypdestopt}{%
21   This package requires pdfTeX in PDF mode%
22 }{\@ehc}
23 \expandafter\endinput
24 \fi
```

The version of pdfTeX must not be too old, because `\pdfescapehex` and `\pdfunescapehex` are used.

```
25 \begingroup\expandafter\expandafter\expandafter\endgroup
26 \expandafter\ifx\csname pdf@escapehex\endcsname\relax
27 \PackageError{hypdestopt}{%
28   This pdfTeX is too old, at least 1.30.0 is required%
29 }{\@ehc}
30 \expandafter\endinput
31 \fi
```

Features of ϵ -TeX are used, e.g. `\numexpr`.

```
32 \begingroup\expandafter\expandafter\expandafter\endgroup
33 \expandafter\ifx\csname numexpr\endcsname\relax
34 \PackageError{hypdestopt}{%
35   e-TeX features are missing%
36 }{\@ehc}
37 \expandafter\endinput
38 \fi
```

Package `alphalph` provides `\newalphalph` since version 2006/05/30 v1.4.

```
39 \ifHypDest@name
40 \RequirePackage{alphalph}[2006/05/30]%
41 \fi

42 \RequirePackage{auxhook}[2009/12/14]
43 \RequirePackage{pdfescape}[2007/04/21]
```

2.4 Preamble for auxiliary file

Provide dummy definitions for the macros that are used in the auxiliary files. If the package is used no longer, then these commands will not generate errors.

`\HypDest@PrependDocument` We add our stuff in front of the `\AtBeginDocument` hook to ensure that we are before `hyperref`'s stuff.

```
44 \long\def\HypDest@PrependDocument#1{%
45   \begingroup
46     \toks\z@{#1}%
47     \toks\tw@\expandafter{\@begindocumenthook}%
48     \xdef\@begindocumenthook{\the\toks\z@\the\toks\tw@}%
49   \endgroup
50 }
```

```

51 \AddLineBeginAux{%
52 \string\providecommand{\string\HypDest@Use}[1]{}%
53 }

```

2.5 Generation of destination names

Counter `HypDest` is used for identifying destinations.

```

54 \newcounter{HypDest}
55 \ifHypDest@name

```

`\HypDest@HexChar` Destination names are generated by automatically numbering with the help of package `alphalph`. `\HypDest@HexChar` converts a number of the range 1 until 252 into the hexadecimal representation of the string character.

```

56 \def\HypDest@HexChar#1{%
57 \ifcase#1\or

```

Avoid zero byte because of C strings in PDF viewer applications.

```

58 01\or 02\or 03\or 04\or 05\or 06\or 07\or

```

Omit carriage return (`13/^^0d`). It needs quoting, otherwise it would be converted to line feed (`10/^^0a`).

```

59 08\or 09\or 0A\or 0B\or 0C\or 0E\or 0F\or
60 10\or 11\or 12\or 13\or 14\or 15\or 16\or 17\or
61 18\or 19\or 1A\or 1B\or 1C\or 1D\or 1E\or 1F\or
62 20\or 21\or 22\or 23\or 24\or 25\or 26\or 27\or

```

Omit left and right parentheses (`40/^^28`, `41/^^39`), they need quoting in general.

```

63 2A\or 2B\or 2C\or 2D\or 2E\or 2F\or
64 30\or 31\or 32\or 33\or 34\or 35\or 36\or 37\or
65 38\or 39\or 3A\or 3B\or 3C\or 3D\or 3E\or 3F\or
66 40\or 41\or 42\or 43\or 44\or 45\or 46\or 47\or
67 48\or 49\or 4A\or 4B\or 4C\or 4D\or 4E\or 4F\or
68 50\or 51\or 52\or 53\or 54\or 55\or 56\or 57\or

```

Omit backslash (`92/^^5C`), it needs quoting.

```

69 58\or 59\or 5A\or 5B\or 5D\or 5E\or 5F\or
70 60\or 61\or 62\or 63\or 64\or 65\or 66\or 67\or
71 68\or 69\or 6A\or 6B\or 6C\or 6D\or 6E\or 6F\or
72 70\or 71\or 72\or 73\or 74\or 75\or 76\or 77\or
73 78\or 79\or 7A\or 7B\or 7C\or 7D\or 7E\or 7F\or
74 80\or 81\or 82\or 83\or 84\or 85\or 86\or 87\or
75 88\or 89\or 8A\or 8B\or 8C\or 8D\or 8E\or 8F\or
76 90\or 91\or 92\or 93\or 94\or 95\or 96\or 97\or
77 98\or 99\or 9A\or 9B\or 9C\or 9D\or 9E\or 9F\or
78 A0\or A1\or A2\or A3\or A4\or A5\or A6\or A7\or
79 A8\or A9\or AA\or AB\or AC\or AD\or AE\or AF\or
80 B0\or B1\or B2\or B3\or B4\or B5\or B6\or B7\or
81 B8\or B9\or BA\or BB\or BC\or BD\or BE\or BF\or
82 C0\or C1\or C2\or C3\or C4\or C5\or C6\or C7\or
83 C8\or C9\or CA\or CB\or CC\or CD\or CE\or CF\or
84 D0\or D1\or D2\or D3\or D4\or D5\or D6\or D7\or
85 D8\or D9\or DA\or DB\or DC\or DD\or DE\or DF\or
86 E0\or E1\or E2\or E3\or E4\or E5\or E6\or E7\or
87 E8\or E9\or EA\or EB\or EC\or ED\or EE\or EF\or
88 F0\or F1\or F2\or F3\or F4\or F5\or F6\or F7\or

```

Avoid 255 (`0xFF`) to get rid of a possible unicode marker at the begin of the string.

```

89 F8\or F9\or FA\or FB\or FC\or FD\or FE%
90 \fi
91 }%

```

`HypDest@HexString` Now package `alphalph` comes into play. `\HypDest@HexString` is defined and converts a positive number into a string, given in hexadecimal representation.

```

92 \newalphalph\HypDest@HexString\HypDest@HexChar{250}%

```

`\theHypDest` For use, the hexadecimal string is converted back.

```
93 \renewcommand*{\theHypDest}{%
94 \pdf@unescapehex{\HypDest@HexString{\value{HypDest}}}%
95 }%
```

With option `num` we use the number directly.

```
96 \else
97 \renewcommand*{\theHypDest}{%
98 \number\value{HypDest}%
99 }%
100 \fi
```

2.6 Assign destination names

`\HypDest@Prefix` The new destination names are remembered in macros whose names start with prefix `\HypDest@Prefix`.

```
101 \edef\HypDest@Prefix{\HypDest\string:}
```

`\HypDest@Use` During the first read of the auxiliary files, the used destinations get fresh generated short destination names. Also for the old destination names we use the hexadecimal representation. That avoid problems with arbitrary names.

```
102 \def\HypDest@Use#1{%
103 \begingroup
104 \edef\x{%
105 \expandafter\noexpand
106 \csname\HypDest@Prefix\pdf@unescapehex{#1}\endcsname
107 }%
108 \expandafter\ifx\x\relax
109 \stepcounter{HypDest}%
110 \expandafter\xdef\x{\theHypDest}%
111 \let\on@line\@empty
112 \ifHypDest@name
113 \HypDest@VerboseInfo{%
114 Use: (\pdf@unescapehex{#1}) -\string> %
115 0x\pdf@escapehex{\x} (\number\value{HypDest})%
116 }%
117 \else
118 \HypDest@VerboseInfo{%
119 Use: (\pdf@unescapehex{#1}) -\string> num \x
120 }%
121 \fi
122 \fi
123 \endgroup
124 }
```

After the first `.aux` file processing the destination names are assigned and we can disable `\HypDest@Use`.

```
125 \AtBeginDocument{%
126 \let\HypDest@Use\@gobble
127 }
```

`\HypDest@MarkUsed` Destinations that are actually used are marked by `\HypDest@MarkUsed`. `\nofiles` is respected.

```
128 \def\HypDest@MarkUsed#1{%
129 \HypDest@VerboseInfo{%
130 MarkUsed: (#1)%
131 }%
132 \if@files
133 \immediate\write\@auxout{%
134 \string\HypDest@Use{\pdf@escapehex{#1}}%
135 }%
```

```

136 \fi
137 }%

```

2.7 Redefinition of hyperref's hooks

Package hyperref can be loaded later, therefore we redefine hyperref's macros at `\begin{document}`.

```

138 \HypDest@PrependDocument{%
    Check hyperref version.
139 \ifpackagelater{hyperref}{2006/06/01}{-}{%
140 \PackageError{hypdestopt}{%
141 hyperref 2006/06/01 v6.75a or later is required%
142 }\@ehc
143 }%

```

2.7.1 Destination setting

luatex compatibility

```

144 \ifx\pdfextension\@undefined\else
145 \protected\def\pdfdest{\pdfextension dest }
146 \fi

147 \ifHypDest@name
148 \let\HypDest@Org@DestName\Hy@DestName
149 \renewcommand*{\Hy@DestName}[2]{%
150 \EdefUnescapeString\HypDest@temp{#1}%
151 \@ifundefined{\HypDest@Prefix\HypDest@temp}{%
152 \HypDest@VerboseInfo{%
153 DestName: (\HypDest@temp) unused%
154 }%
155 }{%
156 \HypDest@Org@DestName{%
157 \csname\HypDest@Prefix\HypDest@temp\endcsname
158 }{#2}%
159 \HypDest@VerboseInfo{%
160 DestName: (\HypDest@temp) %
161 0x\pdf@escapehex{%
162 \csname\HypDest@Prefix\HypDest@temp\endcsname
163 }%
164 }%
165 }%
166 }%
167 \else
168 \renewcommand*{\Hy@DestName}[2]{%
169 \EdefUnescapeString\HypDest@temp{#1}%
170 \@ifundefined{\HypDest@Prefix\HypDest@temp}{%
171 \HypDest@VerboseInfo{%
172 DestName: (\HypDest@temp) unused%
173 }%
174 }{%
175 \pdfdest num%
176 \csname\HypDest@Prefix\HypDest@temp\endcsname#2\relax
177 \HypDest@VerboseInfo{%
178 DestName: (\HypDest@temp) %
179 num \csname\HypDest@Prefix\HypDest@temp\endcsname
180 }%
181 }%
182 }%
183 \fi

```

2.7.2 Links

```
184 \let\HypDest@Org@StartlinkName\Hy@StartlinkName
185 \ifHypDest@name
186   \renewcommand*{\Hy@StartlinkName}[2]{%
187     \HypDest@MarkUsed{#2}%
188     \HypDest@Org@StartlinkName{#1}{%
189       \@ifundefined{\HypDest@Prefix#2}{%
190         #2%
191       }{%
192         \csname\HypDest@Prefix#2\endcsname
193       }%
194     }%
195   }%
196 \else
197   \renewcommand*{\Hy@StartlinkName}[2]{%
198     \HypDest@MarkUsed{#2}%
199     \@ifundefined{\HypDest@Prefix#2}{%
200       \HypDest@Org@StartlinkName{#1}{#2}%
201     }{%
202       \pdfstartlink attr{#1}%
203         goto num\csname\HypDest@Prefix#2\endcsname
204       \relax
205     }%
206   }%
207 \fi
```

2.7.3 Outlines of package hyperref

```
208 \let\HypDest@Org@OutlineName\Hy@OutlineName
209 \ifHypDest@name
210   \renewcommand*{\Hy@OutlineName}[4]{%
211     \HypDest@Org@OutlineName{#1}{%
212       \@ifundefined{\HypDest@Prefix#2}{%
213         #2%
214       }{%
215         \csname\HypDest@Prefix#2\endcsname
216       }%
217     }{#3}{#4}%
218   }%
219 \else
220   \renewcommand*{\Hy@OutlineName}[4]{%
221     \@ifundefined{\HypDest@Prefix#2}{%
222       \HypDest@Org@OutlineName{#1}{#2}{#3}{#4}%
223     }{%
224       \pdfoutline goto num\csname\HypDest@Prefix#2\endcsname
225         count#3{#4}%
226     }%
227   }%
228 \fi
```

Because `\Hy@OutlineName` is called after the `.out` file is written in the previous run. Therefore we mark the destination earlier in `\@@writetorep`.

```
229 \let\HypDest@Org@@writetorep\@@writetorep
230 \renewcommand*{\@@writetorep}[5]{%
231   \begingroup
232     \edef\Hy@tempa{#5}%
233     \ifx\Hy@tempa\Hy@bookmarkstype
234       \HypDest@MarkUsed{#3}%
235     \fi
236   \endgroup
237   \HypDest@Org@@writetorep{#1}{#2}{#3}{#4}{#5}%
238 }
```


2.7.4 Outlines of package bookmark

```
239 \@ifpackageloaded{bookmark}{}%
240 \@ifpackagelater{bookmark}{2008/08/08}{}%
241 \renewcommand*{\BKM@DefGotoNameAction}[2]{%
242   \@ifundefined{\HypDest@Prefix#2}{%
243     \edef#1{goto name{hypdestopt\string :unknown}}}%
244   }{%
245     \ifHypDest@name
246     \edef#1{goto name{\csname\HypDest@Prefix#2\endcsname}}}%
247   \else
248     \edef#1{goto num\csname\HypDest@Prefix#2\endcsname}%
249   \fi
250 }%
251 }%
252 \def\BKM@HypDestOptHook{%
253   \ifx\BKM@dest\@empty
254   \else
255     \ifx\BKM@gotor\@empty
256     \HypDest@MarkUsed\BKM@dest
257     \fi
258   \fi
259 }%
260 }{%
261   \@PackageError{hypdestopt}{%
262     Package `bookmark' is too old.\MessageBreak
263     Version 2008/08/08 or later is needed%
264   }\@ehc
265 }%
266 }{}%
267 }
268 \end{package}
```

3 Installation

3.1 Download

Package. This package is available on CTAN¹:

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

[CTAN:macros/latex/contrib/oberdiek/hypdestopt.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<http://ctan.org/pkg/hypdestopt>

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 hypdestopt.dtx
```

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

```
hypdestopt.sty → tex/latex/oberdiek/hypdestopt.sty
hypdestopt.pdf → doc/latex/oberdiek/hypdestopt.pdf
hypdestopt.dtx → source/latex/oberdiek/hypdestopt.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` , `mik \TeX` , ...) relies on file name databases, you must refresh these. For example, `te \TeX` users run `texhash` or `mktextlsr`.

3.5 Some details for the interested

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

4 Catalogue

The following XML file can be used as source for the [T_EX Catalogue](#). The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `hypdestopt.xml`.

```
269 (*catalogue)
270 <?xml version='1.0' encoding='us-ascii'?>
271 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
272 <entry datestamp='$Date$' modifier='$Author$' id='hypdestopt'>
273   <name>hypdestopt</name>
274   <caption>Hyperref destination optimizer.</caption>
275   <authorref id='auth:oberdiek' />
276   <copyright owner='Heiko Oberdiek' year='2006-2008,2011' />
277   <license type='lppl1.3' />
278   <version number='2.5' />
279   <description>
280     This package supports <xref refid='hyperref'>hyperref</xref>'s
281     pdftex driver. It removes unnecessary destinations
282     and shortens the destination names or uses numbered destinations
283     to get smaller PDF files.
284   <p />
285   The package is part of the <xref refid='oberdiek'>oberdiek</xref>
286   bundle.
287 </description>
288 <documentation details='Package documentation'
289   href='ctan:/macros/latex/contrib/oberdiek/hypdestopt.pdf' />
290 <ctan file='true' path='/macros/latex/contrib/oberdiek/hypdestopt.dtx' />
291 <miktex location='oberdiek' />
292 <texlive location='oberdiek' />
293 <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip' />
294 </entry>
295 </catalogue>
```

5 References

- [1] Heiko Oberdiek: *The alphalph package*; 2006/05/30 v1.4; [CTAN:macros/latex/contrib/oberdiek/alphalph.pdf](#).
- [2] Sebastian Rahtz, Heiko Oberdiek: *The hyperref package*; 2006/06/01 v6.75a; [CTAN:macros/latex/contrib/hyperref/](#).
- [3] Heiko Oberdiek: *The ifpdf package*; 2006/02/20 v1.4; [CTAN:macros/latex/contrib/oberdiek/ifpdf.pdf](#).

6 History

[2006/06/01 v1.0]

- First version.

[2006/06/01 v2.0]

- New method for referencing destinations by number; an idea proposed by Lars Hellström in the mailing list LATEX-L.
- Options name and num added.

[2007/11/11 v2.1]

- Use of package `pdftexcmds` for LuaT_EX support.

[2008/08/08 v2.2]

- Support for package bookmark added.

[2011/05/13 v2.3]

- Fix for `\Hy@DestName` if the destination name contains special characters.
- Fix for option name and package bookmark.

[2016/05/16 v2.4]

- Documentation updates.

[2016/05/21 v2.5]

- LuaTeX compatibility

7 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.

Symbols	H
<code>\@@writetorep</code> 229, 230	<code>\Hy@bookmarkstype</code> 233
<code>\@PackageError</code> 261	<code>\Hy@DestName</code> 148, 149, 168
<code>\@auxout</code> 133	<code>\Hy@OutlineName</code> 208, 210, 220
<code>\@begindocumenthook</code> 47, 48	<code>\Hy@StartlinkName</code> 184, 186, 197
<code>\@ehc</code> 22, 29, 36, 142, 264	<code>\Hy@tempa</code> 232, 233
<code>\@empty</code> 111, 253, 255	<code>\HypDest@HexChar</code> 56, 92
<code>\@gobble</code> 126	<code>\HypDest@HexString</code> 92, 92, 94
<code>\@ifpackagelater</code> 139, 240	<code>\HypDest@MarkUsed</code>
<code>\@ifpackageloaded</code> 239 128, 187, 198, 234, 256
<code>\@ifundefined</code>	<code>\HypDest@namefalse</code> 13
. 151, 170, 189, 199, 212, 221, 242	<code>\HypDest@nametrue</code> 14
<code>\@undefined</code> 144	<code>\HypDest@Org@@writetorep</code> . . 229, 237
	<code>\HypDest@Org@DestName</code> . . . 148, 156
	<code>\HypDest@Org@OutlineName</code>
 208, 211, 222
	<code>\HypDest@Org@StartlinkName</code>
 184, 188, 200
	<code>\HypDest@Prefix</code>
	. 101, 106, 151, 157, 162, 170,
	176, 179, 189, 192, 199, 203,
	212, 215, 221, 224, 242, 246, 248
	<code>\HypDest@PrependDocument</code> . . 44, 138
	<code>\HypDest@temp</code>
 150, 151, 153, 157, 160,
	162, 169, 170, 172, 176, 178, 179
	<code>\HypDest@Use</code> 52, 102, 126, 134
	<code>\HypDest@VerboseInfo</code> 7,
	113, 118, 129, 152, 159, 171, 177
	<code>\HypDest@Verbosetrue</code> 6
	I
	<code>\if@filesw</code> 132
	<code>\ifcase</code> 57
	<code>\ifHypDest@name</code>
	12, 39, 55, 112, 147, 185, 209, 245
	<code>\ifHypDest@Verbose</code> 5, 8
A	
<code>\AddLineBeginAux</code> 51	
<code>\AtBeginDocument</code> 125	
B	
<code>\BKM@DefGotoNameAction</code> 241	
<code>\BKM@dest</code> 253, 256	
<code>\BKM@gotor</code> 255	
<code>\BKM@HypDestOptHook</code> 252	
C	
<code>\csname</code> . . 26, 33, 106, 157, 162, 176,	
179, 192, 203, 215, 224, 246, 248	
D	
<code>\DeclareOption</code> 6, 13, 14	
E	
<code>\EdefUnescapeString</code> 150, 169	
<code>\endscname</code> 26, 33, 106, 157, 162, 176,	
179, 192, 203, 215, 224, 246, 248	
<code>\endinput</code> 23, 30, 37	

<code>\ifpdf</code>	18	<code>\providecommand</code>	52
<code>\ifx</code>	26, 33, 108, 144, 233, 253, 255	<code>\ProvidesPackage</code>	3
<code>\immediate</code>	133		
M			
<code>\MessageBreak</code>	262	<code>\renewcommand</code>	93, 97, 149, 168, 186, 197, 210, 220, 230, 241
N			
<code>\NeedsTeXFormat</code>	2	<code>\RequirePackage</code>	16, 17, 40, 42, 43
<code>\newalphalph</code>	92	S	
<code>\newcounter</code>	54	<code>\stepcounter</code>	109
<code>\newif</code>	5, 12	T	
<code>\number</code>	98, 115	<code>\the</code>	48
O			
<code>\on@line</code>	111	<code>\theHypDest</code>	93, 97, 110
P			
<code>\PackageError</code>	20, 27, 34, 140	<code>\toks</code>	46, 47, 48
<code>\PackageInfo</code>	9	<code>\tw@</code>	47, 48
<code>\pdf@escapehex</code>	115, 134, 161	V	
<code>\pdf@unescapehex</code> ...	94, 106, 114, 119	<code>\value</code>	94, 98, 115
<code>\pdfdest</code>	145, 175	W	
<code>\pdfextension</code>	144, 145	<code>\write</code>	133
<code>\pdfoutline</code>	224	X	
<code>\pdfstartlink</code>	202	<code>\x</code>	104, 108, 110, 115, 119
<code>\ProcessOptions</code>	15	Z	
<code>\protected</code>	145	<code>\z@</code>	46, 48