

The hypdestopt package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2006/06/01 v2.0

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	1
1.1	Introduction	1
1.2	Requirements	2
1.3	Use	2
1.4	Limitations	2
1.5	Future	3
2	Implementation	3
2.1	Identification	3
2.2	Options	3
2.2.1	Option <code>verbose</code>	3
2.2.2	Options <code>num</code> and <code>name</code>	3
2.3	Check requirements	3
2.4	Preamble for auxiliary file	4
2.5	Generation of destination names	4
2.6	Assign destination names	5
2.7	Redefinition of <code>hyperref</code> 's hooks	6
2.7.1	Destination setting	7
2.7.2	Links	7
2.7.3	Outlines	7
3	Installation	8
3.1	Some details for the interested	9
4	References	9
5	History	9
	[2006/06/01 v1.0]	9
	[2006/06/01 v2.0]	9
6	Index	10

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 pdfL^AT_EX 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 [2006/06/01 v2.0 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}
17 \ifpdf
18 \else
19   \PackageError{hypdestopt}{%
20     This package requires pdfTeX in PDF mode%
21   }\@ehc
22 \expandafter\endinput
23 \fi
```

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

```

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

```

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

```

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

```

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

```

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

41 \RequirePackage{auxhook}

```

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.

```

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

49 \AddLineBeginAux{%
50   \string\providecommand{\string\HypDest@Use}[1]{}%
51 }

```

2.5 Generation of destination names

Counter HypDest is used for identifying destinations.

```

52 \newcounter{HypDest}

53 \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.

```

54 \def\HypDest@HexChar#1{%
55   \ifcase#1\or

```

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

```

56     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).

```
57      08\or 09\or 0A\or 0B\or 0C\or 0E\or 0F\or
58      10\or 11\or 12\or 13\or 14\or 15\or 16\or 17\or
59      18\or 19\or 1A\or 1B\or 1C\or 1D\or 1E\or 1F\or
60      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.

```
61      2A\or 2B\or 2C\or 2D\or 2E\or 2F\or
62      30\or 31\or 32\or 33\or 34\or 35\or 36\or 37\or
63      38\or 39\or 3A\or 3B\or 3C\or 3D\or 3E\or 3F\or
64      40\or 41\or 42\or 43\or 44\or 45\or 46\or 47\or
65      48\or 49\or 4A\or 4B\or 4C\or 4D\or 4E\or 4F\or
66      50\or 51\or 52\or 53\or 54\or 55\or 56\or 57\or
```

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

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

```
87      F8\or F9\or FA\or FB\or FC\or FD\or FE%
88      \fi
89  }%
```

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

```
90      \newalphalph\HypDest@HexString\HypDest@HexChar{250}%
```

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

```
91      \renewcommand*{\theHypDest}{%
92      \pdfunescapehex{\HypDest@HexString{\value{HypDest}}}%
93      }%
```

With option `num` we use the number directly.

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

2.6 Assign destination names

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

```
99 \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.

```

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

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

```

123 \AtBeginDocument{%
124   \let\HypDest@Use\@gobble
125 }
```

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

```

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

2.7 Redefinition of `hyperref`'s hooks

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

```

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

2.7.1 Destination setting

```
142 \ifHypDest@name
143 \let\HypDest@Org@DestName\Hy@DestName
144 \renewcommand*{\Hy@DestName}[2]{%
145 \@ifundefined{\HypDest@Prefix#1}{%
146 \HypDest@VerboseInfo{%
147 DestName: (#1) unused%
148 }%
149 }%
150 \HypDest@Org@DestName{\csname\HypDest@Prefix#1\endcsname}{#2}%
151 \HypDest@VerboseInfo{%
152 DestName: (#1) %
153 0x\pdfescapehex{\csname\HypDest@Prefix#1\endcsname}%
154 }%
155 }%
156 }%
157 \else
158 \renewcommand*{\Hy@DestName}[2]{%
159 \@ifundefined{\HypDest@Prefix#1}{%
160 \HypDest@VerboseInfo{%
161 DestName: (#1) unused%
162 }%
163 }%
164 \pdfdest num\csname\HypDest@Prefix#1\endcsname#2\relax
165 \HypDest@VerboseInfo{%
166 DestName: (#1) %
167 num \csname\HypDest@Prefix#1\endcsname
168 }%
169 }%
170 }%
171 \fi
```

2.7.2 Links

```
172 \let\HypDest@Org@StartlinkName\Hy@StartlinkName
173 \ifHypDest@name
174 \renewcommand*{\Hy@StartlinkName}[2]{%
175 \HypDest@MarkUsed{#2}%
176 \HypDest@Org@StartlinkName{#1}{%
177 \@ifundefined{\HypDest@Prefix#2}{%
178 #2%
179 }%
180 \csname\HypDest@Prefix#2\endcsname
181 }%
182 }%
183 }%
184 \else
185 \renewcommand*{\Hy@StartlinkName}[2]{%
186 \HypDest@MarkUsed{#2}%
187 \@ifundefined{\HypDest@Prefix#2}{%
188 \HypDest@Org@StartlinkName{#1}{#2}%
189 }%
190 \pdfstartlink attr{#1}%
191 goto num\csname\HypDest@Prefix#2\endcsname
192 \relax
193 }%
194 }%
195 \fi
```

2.7.3 Outlines

```
196 \let\HypDest@Org@OutlineName\Hy@OutlineName
197 \ifHypDest@name
198 \renewcommand*{\Hy@OutlineName}[4]{%
```

```

199     \HypDest@Org@OutlineName{#1}-{%
200     \@ifundefined{\HypDest@Prefix#2}-{%
201     #2%
202     }{%
203     \csname\HypDest@Prefix#2\endcsname
204     }%
205     }{#3}{#4}%
206     }%
207 \else
208     \renewcommand*{\Hy@OutlineName}[4]{%
209     \@ifundefined{\HypDest@Prefix#2}-{%
210     \HypDest@Org@OutlineName{#1}{#2}{#3}{#4}%
211     }{%
212     \pdfoutline goto num\csname\HypDest@Prefix#2\endcsname
213     count#3{#4}%
214     }%
215     }%
216 \fi

```

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

```

217 \let\HypDest@Org@@writetorep\@@writetorep
218 \renewcommand*{\@@writetorep}[5]{%
219     \begingroup
220     \edef\Hy@tempa{#5}%
221     \ifx\Hy@tempa\Hy@bookmarkstype
222     \HypDest@MarkUsed{#3}%
223     \fi
224     \endgroup
225     \HypDest@Org@@writetorep{#1}{#2}{#3}{#4}{#5}%
226 }%
227 }
228 </package>

```

3 Installation

CTAN. This package is available on CTAN¹:

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

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

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.

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

¹<http://ftp.ctan.org/tex-archive/>

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 hypdestopt.pdf unpack_files output .
```

Unpacking with L^AT_EX. The `.dtx` chooses its action depending on the format:

plain-T_EX: Run `docstrip` and extract the files.

L^AT_EX: Generate the documentation.

If you insist on using L^AT_EX for `docstrip` (really, `docstrip` does not need L^AT_EX), 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 pdfL^AT_EX:

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

4 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](#).

5 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 L^AT_EX-L.
- Options `name` and `num` added.

6 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>\@@writetorep</code>	217, 218
<code>\@auxout</code>	131
<code>\@begindocumenthook</code>	45, 46
<code>\@ehc</code>	21, 28, 35, 140
<code>\@empty</code>	109
<code>\@gobble</code>	124
<code>\@ifpackagelater</code>	137
<code>\@ifundefined</code>	145, 159, 177, 187, 200, 209
A	
<code>\AddLineBeginAux</code>	49
<code>\AtBeginDocument</code>	123
C	
<code>\csname</code>	25, 32, 104, 150, 153, 164, 167, 180, 191, 203, 212
D	
<code>\DeclareOption</code>	6, 13, 14
E	
<code>\endcsname</code>	25, 32, 104, 150, 153, 164, 167, 180, 191, 203, 212
<code>\endinput</code>	22, 29, 36
H	
<code>\Hy@bookmarkstype</code>	221
<code>\Hy@DestName</code>	143, 144, 158
<code>\Hy@OutlineName</code>	196, 198, 208
<code>\Hy@StartlinkName</code>	172, 174, 185
<code>\Hy@tempa</code>	220, 221
<code>\HypDest@HexChar</code>	54, 90
<code>\HypDest@HexString</code>	90, 90, 92
<code>\HypDest@MarkUsed</code>	126, 175, 186, 222
<code>\HypDest@namefalse</code>	13
<code>\HypDest@nametrue</code>	14
<code>\HypDest@Org@@writetorep</code>	217, 225
<code>\HypDest@Org@DestName</code>	143, 150
<code>\HypDest@Org@OutlineName</code>	196, 199, 210
<code>\HypDest@Org@StartlinkName</code>	172, 176, 188
<code>\HypDest@Prefix</code>	99, 104, 145, 150, 153, 159, 164, 167, 177, 180, 187, 191, 200, 203, 209, 212
<code>\HypDest@PrependDocument</code>	42, 136
<code>\HypDest@Use</code>	50, 100, 124, 132
<code>\HypDest@VerboseInfo</code>	7, 111, 116, 127, 146, 151, 160, 165
<code>\HypDest@Verbosetrue</code>	6
I	
<code>\if@files</code>	130
N	
<code>\NeedsTeXFormat</code>	2
<code>\newalphalph</code>	90
<code>\newcounter</code>	52
<code>\newif</code>	5, 12
<code>\number</code>	96, 113
O	
<code>\on@line</code>	109
P	
<code>\PackageError</code>	19, 26, 33, 138
<code>\PackageInfo</code>	9
<code>\pdfdest</code>	164
<code>\pdfescapehex</code>	113, 132, 153
<code>\pdfoutline</code>	212
<code>\pdfstartlink</code>	190
<code>\pdfunescapehex</code>	92, 104, 112, 117
<code>\ProcessOptions</code>	15
<code>\providecommand</code>	50
<code>\ProvidesPackage</code>	3
R	
<code>\renewcommand</code>	91, 95, 144, 158, 174, 185, 198, 208, 218
<code>\RequirePackage</code>	16, 39, 41
S	
<code>\stepcounter</code>	107
T	
<code>\the</code>	46
<code>\theHypDest</code>	91, 95, 108
<code>\toks</code>	44, 45, 46
<code>\tw@</code>	45, 46
V	
<code>\value</code>	92, 96, 113
W	
<code>\write</code>	131
X	
<code>\x</code>	102, 106, 108, 113, 117
Z	
<code>\z@</code>	44, 46
<code>\ifcase</code>	55
<code>\ifHypDest@name</code>	12, 38, 53, 110, 142, 173, 197
<code>\ifHypDest@Verbose</code>	5, 8
<code>\ifpdf</code>	17
<code>\ifx</code>	25, 32, 106, 221
<code>\immediate</code>	131