The hypdestopt package

Heiko Oberdiek

<oberdiek@uni-freiburg.de>

2008/08/08 v2.2

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 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   [2008/08/08 v2.2 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}
```

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.

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

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

2.5 Generation of destination names

Counter HypDest is used for identifying destinations.

```
53 \newcounter{HypDest}
```

```
54 \ifHypDest@name
```

4

\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.

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

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

```
57      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).

```
58      08\or 09\or 0A\or 0B\or 0C\or 0E\or 0F\or
59      10\or 11\or 12\or 13\or 14\or 15\or 16\or 17\or
60      18\or 19\or 1A\or 1B\or 1C\or 1D\or 1E\or 1F\or
61      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.

```
62      2A\or 2B\or 2C\or 2D\or 2E\or 2F\or
63      30\or 31\or 32\or 33\or 34\or 35\or 36\or 37\or
64      38\or 39\or 3A\or 3B\or 3C\or 3D\or 3E\or 3F\or
65      40\or 41\or 42\or 43\or 44\or 45\or 46\or 47\or
66      48\or 49\or 4A\or 4B\or 4C\or 4D\or 4E\or 4F\or
67      50\or 51\or 52\or 53\or 54\or 55\or 56\or 57\or
```

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

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

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

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

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

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

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

With option num we use the number directly.

```
95 \else
96   \renewcommand*{\theHypDest}{%
```

```
97    \number\value{HypDest}%
98   }%
99 \fi
```

2.6   Assign destination names

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

```
100 \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.

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

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

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

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

```
127 \def\HypDest@MarkUsed#1{%
128   \HypDest@VerboseInfo{%
129     MarkUsed: (#1)%
130   }%
131   \if@filesw
132     \immediate\write\@auxout{%
133       \string\HypDest@Use{\pdf@escapehex{#1}}%
134     }%
135   \fi
136 }%
```

2.7   Redefinition of **hyperref**'s hooks

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

```
137 \HypDest@PrependDocument{%
```

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

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

2.7.2 Links

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

```
194        }%
195      }%
196    \fi
```

2.7.3 Outlines of package **hyperref**

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

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

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

2.7.4 Outlines of package **bookmark**

```
228  \@ifpackageloaded{bookmark}{%
229    \@ifpackagelater{bookmark}{2008/08/08}{%
230      \renewcommand*{\BKM@DefGotoNameAction}[2]{%
231        \@ifundefined{\HypDest@Prefix#2}{%
232          \edef#1{goto name{hypdestopt\string :unknown}}%
233        }{%
234          \edef#1{goto num\csname\HypDest@Prefix#2\endcsname}%
235        }%
236      }%
237      \def\BKM@HypDestOptHook{%
238        \ifx\BKM@dest\@empty
239        \else
240          \ifx\BKM@gotor\@empty
241            \HypDest@MarkUsed\BKM@dest
242          \fi
243        \fi
244      }%
245    }{%
246      \@PackageError{hypdestopt}{%
247        Package 'bookmark' is too old.\MessageBreak
248        Version 2008/08/08 or later is needed%
249      }\@ehc
```

```
250      }%
251    }{}%
252 }
253 ⟨/package⟩
```

3 Installation

3.1 Download

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

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

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

---

[1] ftp://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 hypdestopt.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{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 pdfLaTeX:

```
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 LATEX-L.
- Options `name` and `num` added.

[2007/11/11 v2.1]

- Use of package `pdftexcmds` for LuaTeX support.

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.