

The pdfcol package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/09/09 v1.0

Abstract

Since version 1.40 pdfTeX supports color stacks. The driver file `pdftex.def` for package `color` defines and uses a main color stack since version v0.04b. Package `pdfcol` is intended for package writers. It defines macros for setting and maintaining new color stacks.

Contents

1	Documentation	1
1.1	Requirements	2
1.2	Interface	2
2	Implementation	3
2.1	Reload check and package identification	3
2.2	Catcodes	4
2.3	Check requirements	4
2.3.1	Check PDF mode	4
2.3.2	Check version of pdfTeX	4
2.3.3	Check <code>pdftex.def</code>	5
2.4	Enabled interface macros	5
2.5	Disabled interface macros	7
3	Test	7
3.1	Catcode checks for loading	7
3.2	Very simple test	8
4	Installation	9
4.1	Download	9
4.2	Bundle installation	9
4.3	Package installation	9
4.4	Refresh file name databases	10
4.5	Some details for the interested	10
5	History	10
	[2007/09/09 v1.0]	10
6	Index	11

1 Documentation

Version 1.40 of pdfTeX adds new primitives `\pdfcolorstackinit` and `\pdfcolorstack`. Now color stacks can be defined and used. A main color stack is maintained by the driver file `pdftex.def` similar to dvips or dvi2pdfm. However the number of color stacks is not limited to one in pdfTeX. Thus further color problems can now be solved, such as footnotes across pages or text that is set in parallel columns

(e.g. packages `parallel` or `parcolumn`). Unlike the main color stack, the support by additional color stacks cannot be done in a transparent manner.

This package `pdfcol` provides an easier interface to additional color stacks without the need to use the low level primitives.

1.1 Requirements

- pdfTeX 1.40 or greater.
- pdfTeX in PDF mode. (I don't know a DVI driver that support several color stacks.)
- `pdftex.def` 2007/01/02 v0.04b.

Package `pdfcol` checks the requirements and sets switch `\ifpdfcolAvailable` accordingly.

1.2 Interface

`\ifpdfcolAvailable`

If the requirements of section 1.1 are met the switch `\ifpdfcolAvailable` behaves as `\iftrue`. Otherwise the other interface macros in this section will be disabled with a message. Also the first use of such a macro will print a message. The messages are print to the `..log` file only if pdfTeX is not used in PDF mode.

`\pdfcolErrorNoStacks`

The first call of `\pdfcolErrorNoStacks` prints an error message, if color stacks are not available.

`\pdfcolInitStack {<name>}`

A new color stack is initialized by `\pdfcolInitStack`. The `<name>` is used for indentifying the stack. It usually consists of letters and digits. (The name must survive a `\csname`.)

The intension of the macro is the definition of an additional color stack. Thus the stack is not page bounded like the main color stack. Black (`0 g 0 G`) is used as initial color value. And colors are written with modifier `direct` that means without setting the current transfer matrix and changing the current point (see documentation of pdfTeX for `\pdfliteral direct{...}`).

`\pdfcolIfStackExists {<name>} {<then>} {<else>}`

Macro `\pdfcolIfStackExists` checks whether color stack `<name>` exists. In case of success argument `<then>` is executed and `<else>` otherwise.

`\pdfcolSwitchStack {<name>}`

Macro `\pdfcolSwitchStack` switches the color stack. The color macros of package `color` (or `xcolor`) now uses the new color stack with name `<name>`.

`\pdfcolSetCurrent {<name>}`

Macro `\pdfcolSetCurrent` sets the color that is read in the top-most entry of color stack `<name>`. If `<name>` is empty, the default color stack is used.

2 Implementation

```
1 (*package)
```

2.1 Reload check and package identification

Reload check, especially if the package is not used with L^AT_EX.

```
2 \begingroup
3 \catcode44 12 % ,
4 \catcode45 12 % -
5 \catcode46 12 % .
6 \catcode58 12 % :
7 \catcode64 11 % @
8 \expandafter\let\expandafter\x\csname ver@pdfcol.sty\endcsname
9 \ifcase 0%
10 \ifx\x\relax % plain
11 \else
12 \ifx\x\empty % LaTeX
13 \else
14 1%
15 \fi
16 \fi
17 \else
18 \expandafter\ifx\csname PackageInfo\endcsname\relax
19 \def\x#1#2{%
20 \immediate\write-1{Package #1 Info: #2.}%
21 }%
22 \else
23 \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
24 \fi
25 \x{pdfcol}{The package is already loaded}%
26 \endgroup
27 \expandafter\endinput
28 \fi
29 \endgroup
```

Package identification:

```
30 \begingroup
31 \catcode40 12 % (
32 \catcode41 12 % )
33 \catcode44 12 % ,
34 \catcode45 12 % -
35 \catcode46 12 % .
36 \catcode47 12 % /
37 \catcode58 12 % :
38 \catcode64 11 % @
39 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
40 \def\x#1#2#3[#4]{\endgroup
41 \immediate\write-1{Package: #3 #4}%
42 \xdef#1{#4}%
43 }%
44 \else
45 \def\x#1#2[#3]{\endgroup
46 #2[#{#3}]%
47 \ifx#1\relax
48 \xdef#1{#3}%
49 \fi
50 }%
51 \fi
52 \expandafter\x\csname ver@pdfcol.sty\endcsname
53 \ProvidesPackage{pdfcol}%
54 [2007/09/09 v1.0 Maintaining additional color stacks for pdfTeX (H0)]
```

2.2 Catcodes

```

55 \expandafter\edef\csname PDFCOL@AtEnd\endcsname{%
56   \catcode64 \the\catcode64\relax
57 }
58 \catcode64 11 % @
59 \def\TMP@EnsureCode#1#2{%
60   \edef\PDFCOL@AtEnd{%
61     \PDFCOL@AtEnd
62     \catcode#1 \the\catcode#1\relax
63   }%
64   \catcode#1 #2\relax
65 }
66 \TMP@EnsureCode{39}{12}% '
67 \TMP@EnsureCode{40}{12}% (
68 \TMP@EnsureCode{41}{12}% )
69 \TMP@EnsureCode{43}{12}% +
70 \TMP@EnsureCode{44}{12}% ,
71 \TMP@EnsureCode{46}{12}% .
72 \TMP@EnsureCode{47}{12}% /
73 \TMP@EnsureCode{61}{12}% =
74 \TMP@EnsureCode{96}{12}% '

```

2.3 Check requirements

ifpdfcolAvailable

```

75 \newif\ifpdfcolAvailable
76 \pdfcolAvailabletrue

```

\PDFCOL@RequirePackage

```

77 \begingroup\expandafter\expandafter\expandafter\endgroup
78 \expandafter\ifx\csname RequirePackage\endcsname\relax
79   \def\PDFCOL@RequirePackage#1[#2]{\input #1.sty\relax}%
80 \else
81   \def\PDFCOL@RequirePackage#1[#2]{%
82     \RequirePackage{#1}[#2]}%
83   }%
84 \fi

```

2.3.1 Check PDF mode

```

85 \PDFCOL@RequirePackage{infwarerr}[2007/09/09]
86 \PDFCOL@RequirePackage{ifpdf}[2007/09/09]
87 \ifpdf
88   \def\PDFCOL@Message{%
89     \@PackageWarningNoLine{pdfcol}%
90   }%
91 \else
92   \pdfcolAvailablefalse
93   \def\PDFCOL@Message{%
94     \@PackageInfoNoLine{pdfcol}%
95   }%
96   \PDFCOL@Message{%
97     Interface disabled because of missing PDF mode of pdfTeX%
98   }%
99 \fi

```

2.3.2 Check version of pdfTeX

```

100 \ifpdfcolAvailable
101   \begingroup\expandafter\expandafter\expandafter\endgroup
102   \expandafter\ifx\csname pdfcolorstack\endcsname\relax
103     \pdfcolAvailablefalse
104     \PDFCOL@Message{%

```

```

105     Interface disabled because of too old pdfTeX.\MessageBreak
106     Required is version 1.40+ for \string\pdfcolorstack
107   }%
108 \fi
109 \fi
110 \ifpdfcolAvailable
111   \begingroup\expandafter\expandafter\expandafter\endgroup
112   \expandafter\ifx\csname pdfcolorstack\endcsname\relax
113     \pdfcolAvailablefalse
114     \PDFCOL@Message{%
115       Interface disabled because of too old pdfTeX.\MessageBreak
116       Required is version 1.40+ for \string\pdfcolorstackinit
117     }%
118   \fi
119 \fi

```

2.3.3 Check pdftex.def

```

120 \ifpdfcolAvailable
121   \begingroup\expandafter\expandafter\expandafter\endgroup
122   \expandafter\ifx\csname @pdfcolorstack\endcsname\relax
123     Try to load package color if it is not yet loaded (LATEX case).
124     \begingroup\expandafter\expandafter\expandafter\endgroup
125     \expandafter\ifx\csname ver@color.sty\endcsname\relax
126       \begingroup\expandafter\expandafter\expandafter\endgroup
127       \expandafter\ifx\csname documentclass\endcsname\relax
128       \else
129         \RequirePackage[pdftex]{color}\relax
130       \fi
131     \begingroup\expandafter\expandafter\expandafter\endgroup
132     \expandafter\ifx\csname @pdfcolorstack\endcsname\relax
133       \pdfcolAvailablefalse
134       \PDFCOL@Message{%
135         Interface disabled because ‘pdftex.def’\MessageBreak
136         is not loaded or it is too old.\MessageBreak
137         Required is version 0.04b or greater%
138       }%
139     \fi
140   \fi
141 \fi
142 \let\pdfcolAvailabletrue\relax
143 \let\pdfcolAvailablefalse\relax

```

2.4 Enabled interface macros

```

144 \ifpdfcolAvailable

\pdfcolErrorNoStacks

145   \let\pdfcolErrorNoStacks\relax

\pdfcol@Value

146   \expandafter\ifx\csname pdfcol@Value\endcsname\relax
147   \def\pdfcol@Value{0 g 0 G}%
148   \fi

\pdfcol@LiteralModifier

149   \expandafter\ifx\csname pdfcol@LiteralModifier\endcsname\relax
150   \def\pdfcol@LiteralModifier{direct}%
151   \fi

\pdfcolInitStack

152   \def\pdfcolInitStack#1{%

```

```

153     \expandafter\ifx\csname pdfcol@Stack@#1\endcsname\relax
154     \global\expandafter\chardef\csname pdfcol@Stack@#1\endcsname=%
155     \pdfcolorstackinit\pdfcol@LiteralModifier{\pdfcol@Value}%
156     \relax
157     \@PackageInfo{pdfcol}{%
158       New color stack '#1' = \number\csname pdfcol@Stack@#1\endcsname
159     }%
160   \else
161     \@PackageError{pdfcol}{%
162       Stack '#1' is already defined%
163     }\@ehc
164   \fi
165 }%

\pdfcolIfStackExists

166 \def\pdfcolIfStackExists#1{%
167   \expandafter\ifx\csname pdfcol@Stack@#1\endcsname\relax
168   \expandafter\@secondoftwo
169   \else
170   \expandafter\@firstoftwo
171   \fi
172 }%

\@firstoftwo

173 \expandafter\ifx\csname @firstoftwo\endcsname\relax
174 \long\def\@firstoftwo#1#2{#1}%
175 \fi

\@secondoftwo

176 \expandafter\ifx\csname @secondoftwo\endcsname\relax
177 \long\def\@secondoftwo#1#2{#2}%
178 \fi

\pdfcolSwitchStack

179 \def\pdfcolSwitchStack#1{%
180   \pdfcolIfStackExists{#1}{%
181     \expandafter\let\expandafter\@pdfcolorstack
182     \csname pdfcol@Stack@#1\endcsname
183   }{%
184     \pdfcol@ErrorNoStack{#1}%
185   }%
186 }%

\pdfcolSetCurrent

187 \def\pdfcolSetCurrent#1{%
188   \ifx\#1\%
189     \pdfcolorstack\@pdfcolorstack current\relax
190   \else
191     \pdfcolIfStackExists{#1}{%
192       \pdfcolorstack\csname pdfcol@Stack@#1\endcsname current\relax
193     }{%
194       \pdfcol@ErrorNoStack{#1}%
195     }%
196   \fi
197 }%

\pdfcol@ErrorNoStack

198 \def\pdfcol@ErrorNoStack#1{%
199   \@PackageError{pdfcol}{Stack '#1' does not exists}\@ehc
200 }%

```

2.5 Disabled interface macros

```
201 \else

\pdfcolErrorNoStacks

202 \def\pdfcolErrorNoStacks{%
203   \@PackageError{pdfcol}{%
204     Color stacks are not available%
205   }{%
206     Update pdfTeX (1.40) and 'pdftex.def' (0.04b) %
207     if necessary.\MessageBreak
208     Ensure that 'pdftex.def' is loaded %
209     (package 'color' or 'xcolor').\MessageBreak
210     Further messages can be found in TeX's %
211     protocol file '\jobname.log'.\MessageBreak
212     \MessageBreak
213     \@ehc
214   }%
215   \global\let\pdfcolErrorNoStacks\relax
216 }%

\PDFCOL@Disabled

217 \def\PDFCOL@Disabled{%
218   \PDFCOL@Message{%
219     pdfTeX's color stacks are not available%
220   }%
221   \global\let\PDFCOL@Disabled\relax
222 }%

\pdfcolInitStack

223 \def\pdfcolInitStack#1{%
224   \PDFCOL@Disabled
225 }%

\pdfcolIfStackExists

226 \long\def\pdfcolIfStackExists#1#2#3{#3}%

\pdfcolSwitchStack

227 \def\pdfcolSwitchStack#1{%
228   \PDFCOL@Disabled
229 }%

\pdfcolSetCurrent

230 \def\pdfcolSetCurrent#1{%
231   \PDFCOL@Disabled
232 }%

233 \fi
234 \PDFCOL@AtEnd
235 </package>
```

3 Test

3.1 Catcode checks for loading

```
236 <*test1>
237 \catcode'\@=11 %
238 \def\RestoreCatcodes{}
239 \count@=0 %
240 \loop
```

```

241 \edef\RestoreCatcodes{%
242   \RestoreCatcodes
243   \catcode\the\count@=\the\catcode\count@\relax
244 }%
245 \ifnum\count@<255 %
246   \advance\count@\@ne
247 \repeat
248
249 \def\RangeCatcodeInvalid#1#2{%
250   \count@=#1\relax
251   \loop
252     \catcode\count@=15 %
253   \ifnum\count@<#2\relax
254     \advance\count@\@ne
255   \repeat
256 }
257 \def\Test{%
258   \RangeCatcodeInvalid{0}{47}%
259   \RangeCatcodeInvalid{58}{64}%
260   \RangeCatcodeInvalid{91}{96}%
261   \RangeCatcodeInvalid{123}{255}%
262   \catcode'\@=12 %
263   \catcode'\=0 %
264   \catcode'\{=1 %
265   \catcode'\}=2 %
266   \catcode'\#=6 %
267   \catcode'\[=12 %
268   \catcode'\]=12 %
269   \catcode'\%=14 %
270   \catcode'\ =10 %
271   \catcode13=5 %
272   \input pdfcol.sty\relax
273   \RestoreCatcodes
274 }
275 \Test
276 \csname @@end\endcsname
277 \end
278 </test1>

```

3.2 Very simple test

```

279 <*test2 | test3>
280 \NeedsTeXFormat{LaTeX2e}
281 \nofiles
282 \documentclass{article}
283 \usepackage{pdfcol}[2007/09/09]
284 \usepackage{qstest}
285 \IncludeTests{*}
286 \LogTests{log}{*}{*}
287 \begin{document}
288   \begin{qstest}{pdfcol}{}%
289     \makeatletter
290 <*test2>
291   \Expect*{\ifpdfcolAvailable true\else false\fi}{false}%
292 </test2>
293 <*test3>
294   \Expect*{\ifpdfcolAvailable true\else false\fi}{true}%
295   \Expect*{\number\pdfcolorstack}{0}%
296 </test3>
297   \setbox0=\hbox{%
298     \pdfcolInitStack{test}%
299 <*test3>
300     \Expect*{\number\pdfcol@Stack@test}{1}%

```



```

301      \Expect*{\number\@pdfcolorstack}{0}%
302 </test3>
303      \pdfcolSwitchStack{test}%
304 <*test3>
305      \Expect*{\number\@pdfcolorstack}{1}%
306 </test3>
307      \pdfcolSetCurrent{test}%
308      \pdfcolSetCurrent{}%
309    }%
310    \Expect*{\the\wd0}{0.0pt}%
311 <*test3>
312      \Expect*{\number\@pdfcolorstack}{0}%
313      \Expect*{\number\pdfcol@Stack@test}{1}%
314      \Expect*{\pdfcolIfStackExists{test}{true}{false}}{true}%
315 </test3>
316      \Expect*{\pdfcolIfStackExists{dummy}{true}{false}}{false}%
317    \end{qstest}%
318 \end{document}
319 </test2 | test3>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

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

[CTAN:macros/latex/contrib/oberdiek/pdfcol.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:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

TDS refers to the standard “A Directory Structure for T_EX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

4.2 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/
```

4.3 Package installation

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

```
tex pdfcol.dtx
```

¹<http://ftp.ctan.org/tex-archive/>

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

```
pdfcol.sty          → tex/generic/oberdiek/pdfcol.sty
pdfcol.pdf          → doc/latex/oberdiek/pdfcol.pdf
test/pdfcol-test1.tex → doc/latex/oberdiek/test/pdfcol-test1.tex
test/pdfcol-test2.tex → doc/latex/oberdiek/test/pdfcol-test2.tex
test/pdfcol-test3.tex → doc/latex/oberdiek/test/pdfcol-test3.tex
pdfcol.dtx          → source/latex/oberdiek/pdfcol.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`.

4.4 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`.

4.5 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 pdfcol.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{pdfcol.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 pdfcol.dtx
makeindex -s gind.ist pdfcol.idx
pdflatex pdfcol.dtx
makeindex -s gind.ist pdfcol.idx
pdflatex pdfcol.dtx
```

5 History

[2007/09/09 v1.0]

- First version.

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		I	
\#	266	\ifcase	9
\%	269	\ifnum	245, 253
\@	237, 262	\ifpdf	87
\@PackageError	161, 199, 203	\ifpdfcolAvailable	2, 75, 75, 100, 110, 120, 144, 291, 294
\@PackageInfo	157	\ifx	10, 12, 18, 39, 47, 78, 102, 112, 122, 124, 126, 132, 146, 149, 153, 167, 173, 176, 188
\@PackageInfoNoLine	94	\immediate	20, 41
\@PackageWarningNoLine	89	\IncludeTests	285
\@ehc	163, 199, 213	\input	79, 272
\@firstoftwo	170, 173	J	
\@ne	246, 254	\jobname	211
\@pdfcolorstack	181, 189, 295, 301, 305, 312	L	
\@secondoftwo	168, 176	\LogTests	286
\[267	\loop	240, 251
\]	188, 263	M	
\{	264	\makeatletter	289
\}	265	\MessageBreak	105, 115, 135, 136, 207, 209, 211, 212
\]	268	N	
_	270	\NeedsTeXFormat	280
A		\newif	75
\advance	246, 254	\nofiles	281
B		\number	158, 295, 300, 301, 305, 312, 313
\begin	287, 288	P	
C		\PackageInfo	23
\catcode	3, 4, 5, 6, 7, 31, 32, 33, 34, 35, 36, 37, 38, 56, 58, 62, 64, 237, 243, 252, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271	\PDFCOL@AtEnd	60, 61, 234
\chardef	154	\PDFCOL@Disabled	217, 224, 228, 231
\count@	239, 243, 245, 246, 250, 252, 253, 254	\pdfcol@ErrorNoStack	184, 194, 198
\csname	8, 18, 39, 52, 55, 78, 102, 112, 122, 124, 126, 132, 146, 149, 153, 154, 158, 167, 173, 176, 182, 192, 276	\pdfcol@LiteralModifier	149, 155
D		\PDFCOL@Message	88, 93, 96, 104, 114, 134, 218
\documentclass	282	\PDFCOL@RequirePackage	77, 85, 86
E		\pdfcol@Stack@test	300, 313
\empty	12	\pdfcol@Value	146, 155
\end	277, 317, 318	\pdfcolAvailablefalse	92, 103, 113, 133, 143
\endcsname	8, 18, 39, 52, 55, 78, 102, 112, 122, 124, 126, 132, 146, 149, 153, 154, 158, 167, 173, 176, 182, 192, 276	\pdfcolAvailabletrue	76, 142
\endinginput	27	\pdfcolErrorNoStacks	2, 145, 202
\Expect	291, 294, 295, 300, 301, 305, 310, 312, 313, 314, 316	\pdfcolIfStackExists	2, 166, 180, 191, 226, 314, 316
H		\pdfcolInitStack	2, 152, 223, 298
\hbox	297	\pdfcolorstack	106, 189, 192
		\pdfcolorstackinit	116, 155
		\pdfcolSetCurrent	2, 187, 230, 307, 308
		\pdfcolSwitchStack	2, 179, 227, 303
		\ProvidesPackage	53
		R	
		\RangeCatcodeInvalid	249, 258, 259, 260, 261

<code>\repeat</code>	247, 255	U
<code>\RequirePackage</code>	82, 128	<code>\usepackage</code>
<code>\RestoreCatcodes</code> ..	238, 241, 242, 273	283, 284
S		W
<code>\setbox</code>	297	<code>\wd</code>
T		<code>\write</code>
<code>\Test</code>	257, 275	20, 41
<code>\the</code>	56, 62, 243, 310	
<code>\TMP@EnsureCode</code>	59,	X
66, 67, 68, 69, 70, 71, 72, 73, 74		<code>\x</code>
		8, 10, 12, 19, 23, 25, 40, 45, 52