The pdfcol package

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

2018/11/01 v1.5

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 **2**
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 . . . 5
2.3.1 Check package luacolor . . . 5
2.3.2 Check PDF mode . . . 5
2.3.3 Check version of pdfTeX . . . 6
2.3.4 Check pdftex.def . . . 6
2.4 Enabled interface macros . . . 6
2.5 Disabled interface macros . . . 8

3 Test **9**
3.1 Catcode checks for loading . . . 9
3.2 Very simple test . . . 10
3.3 Detection of package luacolor . . . 11

4 Installation **11**
4.1 Download . . . 11
4.2 Bundle installation . . . 12
4.3 Package installation . . . 12
4.4 Refresh file name databases . . . 12
4.5 Some details for the interested . . . 12

5 Catalogue **13**

6 History **13**
[2007/09/09 v1.0] . . . 13
[2007/12/09 v1.1] . . . 13
[2007/12/12 v1.2] . . . 14
[2016/05/16 v1.3] . . . 14
[2016/05/17 v1.4] . . . 14
[2018/11/01 v1.5] . . . 14

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

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 dvipdfm. 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.
- pdfTEXin 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*⟩.

`\pdfcolSetCurrentColor`

Macro `\pdfcolSetCurrentColor` replaces the topmost entry of the stack by the current color (`\current@color`).

`\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 LaTeX.

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3   \catcode13=5 % ^^M
4   \endlinechar=13 %
5   \catcode35=6 % #
6   \catcode39=12 % '
7   \catcode44=12 % ,
8   \catcode45=12 % -
9   \catcode46=12 % .
10  \catcode58=12 % :
11  \catcode64=11 % @
12  \catcode123=1 % {
13  \catcode125=2 % }
14  \expandafter\let\expandafter\x\csname ver@pdfcol.sty\endcsname
15  \ifx\x\relax % plain-TeX, first loading
16  \else
17    \def\empty{}%
18    \ifx\x\empty % LaTeX, first loading,
19      % variable is initialized, but \ProvidesPackage not yet seen
20    \else
21      \expandafter\ifx\csname PackageInfo\endcsname\relax
22        \def\x#1#2{%
23          \immediate\write-1{Package #1 Info: #2.}%
24        }%
25      \else
26        \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
27      \fi
28      \x{pdfcol}{The package is already loaded}%
29      \aftergroup\endinput
30    \fi
31  \fi
32 \endgroup%
```

Package identification:

```
33 \begingroup\catcode61\catcode48\catcode32=10\relax%
34   \catcode13=5 % ^^M
35   \endlinechar=13 %
36   \catcode35=6 % #
37   \catcode39=12 % '
38   \catcode40=12 % (
```

```
\catcode41=12 % )
\catcode44=12 % ,
\catcode45=12 % -
\catcode46=12 % .
\catcode47=12 % /
\catcode58=12 % :
\catcode64=11 % @
\catcode91=12 % [
\catcode93=12 % ]
\catcode123=1 % {
\catcode125=2 % }
\expandafter\ifx\csname ProvidesPackage\endcsname\relax
  \def\x#1#2#3[#4]{\endgroup
    \immediate\write-1{Package: #3 #4}%
    \xdef#1{#4}%
  }%
\else
  \def\x#1#2[#3]{\endgroup
    #2[{#3}]%
    \ifx#1\@undefined
      \xdef#1{#3}%
    \fi
    \ifx#1\relax
      \xdef#1{#3}%
    \fi
  }%
\fi
\expandafter\x\csname ver@pdfcol.sty\endcsname
\ProvidesPackage{pdfcol}%
  [2018/11/01 v1.5 Handle new color stacks for pdfTeX (HO)]%
```

2.2 Catcodes

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode123=1 % {
  \catcode125=2 % }
  \catcode64=11 % @
  \def\x{\endgroup
    \expandafter\edef\csname PDFCOL@AtEnd\endcsname{%
      \endlinechar=\the\endlinechar\relax
      \catcode13=\the\catcode13\relax
      \catcode32=\the\catcode32\relax
      \catcode35=\the\catcode35\relax
      \catcode61=\the\catcode61\relax
      \catcode64=\the\catcode64\relax
      \catcode123=\the\catcode123\relax
      \catcode125=\the\catcode125\relax
    }%
  }%
\x\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\def\TMP@EnsureCode#1#2{%
  \edef\PDFCOL@AtEnd{%
    \PDFCOL@AtEnd
    \catcode#1=\the\catcode#1\relax
```

```
  }%
  \catcode#1=#2\relax
}
\TMP@EnsureCode{39}{12}% '
\TMP@EnsureCode{40}{12}% (
\TMP@EnsureCode{41}{12}% )
\TMP@EnsureCode{43}{12}% +
\TMP@EnsureCode{44}{12}% ,
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{91}{12}% [
\TMP@EnsureCode{93}{12}% ]
\TMP@EnsureCode{96}{12}% `
\edef\PDFCOL@AtEnd{\PDFCOL@AtEnd\noexpand\endinput}
```

2.3 Check requirements

\PDFCOL@RequirePackage

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname RequirePackage\endcsname\relax
  \def\PDFCOL@RequirePackage#1[#2]{\input #1.sty\relax}%
\else
  \def\PDFCOL@RequirePackage#1[#2]{%
    \RequirePackage{#1}[{#2}]%
  }%
\fi
```

LuaTeX Compatability

```
\ifx\pdfextension\@undefined\else
 \def\pdfcolorstackinit {\pdffeedback colorstackinit}
 \protected\def\pdfcolorstack     {\pdfextension colorstack}
\fi
```

```
\PDFCOL@RequirePackage{ltxcmds}[2010/03/01]
```

ifpdfcolAvailable

```
\ltx@newif\ifpdfcolAvailable
\pdfcolAvailabletrue
```

2.3.1 Check package luacolor

```
\ltx@newif\ifPDFCOL@luacolor
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname ver@luacolor.sty\endcsname\relax
  \PDFCOL@luacolorfalse
\else
  \PDFCOL@luacolortrue
\fi
```

2.3.2 Check PDF mode

```
\PDFCOL@RequirePackage{infwarerr}[2007/09/09]
\PDFCOL@RequirePackage{ifpdf}[2007/09/09]
\ifcase\ifpdf\ifPDFCOL@luacolor 1\fi\else 1\fi0 %
  \def\PDFCOL@Message{%
    \@PackageWarningNoLine{pdfcol}%
  }%
\else
  \pdfcolAvailablefalse
  \def\PDFCOL@Message{%
    \@PackageInfoNoLine{pdfcol}%
  }%
  \PDFCOL@Message{%
    Interface disabled because of %
```

```
    \ifPDFCOL@luacolor
      package `luacolor'%
    \else
      missing PDF mode of pdfTeX%
    \fi
  }%
\fi
```

2.3.3 Check version of pdfTeX

```
\ifpdfcolAvailable
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname pdfcolorstack\endcsname\relax
    \pdfcolAvailablefalse
    \PDFCOL@Message{%
      Interface disabled because of too old pdfTeX.\MessageBreak
      Required is version 1.40+ for \string\pdfcolorstack
    }%
  \fi
\fi
\ifpdfcolAvailable
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname pdfcolorstack\endcsname\relax
    \pdfcolAvailablefalse
    \PDFCOL@Message{%
      Interface disabled because of too old pdfTeX.\MessageBreak
      Required is version 1.40+ for \string\pdfcolorstackinit
    }%
  \fi
\fi
```

2.3.4 Check pdftex.def

```
\ifpdfcolAvailable
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname @pdfcolorstack\endcsname\relax
```

Try to load package color if it is not yet loaded (LaTeX case).

```
    \begingroup\expandafter\expandafter\expandafter\endgroup
    \expandafter\ifx\csname ver@color.sty\endcsname\relax
      \begingroup\expandafter\expandafter\expandafter\endgroup
      \expandafter\ifx\csname documentclass\endcsname\relax
      \else
        \RequirePackage[pdftex]{color}\relax
      \fi
    \fi
    \begingroup\expandafter\expandafter\expandafter\endgroup
    \expandafter\ifx\csname @pdfcolorstack\endcsname\relax
      \pdfcolAvailablefalse
      \PDFCOL@Message{%
        Interface disabled because `pdftex.def'\MessageBreak
        is not loaded or it is too old.\MessageBreak
        Required is version 0.04b or greater%
      }%
    \fi
  \fi
\fi
\let\pdfcolAvailabletrue\relax
\let\pdfcolAvailablefalse\relax
```

2.4 Enabled interface macros

```
\ifpdfcolAvailable
```

\pdfcolErrorNoStacks

```
  \let\pdfcolErrorNoStacks\relax
```

\pdfcol@Value

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

\pdfcol@LiteralModifier

```
\expandafter\ifx\csname pdfcol@LiteralModifier\endcsname\relax
  \def\pdfcol@LiteralModifier{direct}%
\fi
```

\pdfcolInitStack

```
\def\pdfcolInitStack#1{%
  \expandafter\ifx\csname pdfcol@Stack@#1\endcsname\relax
    \global\expandafter\chardef\csname pdfcol@Stack@#1\endcsname=%
       \pdfcolorstackinit\pdfcol@LiteralModifier{\pdfcol@Value}%
       \relax
    \@PackageInfo{pdfcol}{%
      New color stack `#1' = \number\csname pdfcol@Stack@#1\endcsname
    }%
  \else
    \@PackageError{pdfcol}{%
      Stack `#1' is already defined%
    }\@ehc
  \fi
}%
```

\pdfcolIfStackExists

```
\def\pdfcolIfStackExists#1{%
  \expandafter\ifx\csname pdfcol@Stack@#1\endcsname\relax
    \expandafter\@secondoftwo
  \else
    \expandafter\@firstoftwo
  \fi
}%
```

\@firstoftwo

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

\@secondoftwo

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

\pdfcolSwitchStack

```
\def\pdfcolSwitchStack#1{%
  \pdfcolIfStackExists{#1}{%
    \expandafter\let\expandafter\@pdfcolorstack
                \csname pdfcol@Stack@#1\endcsname
  }{%
    \pdfcol@ErrorNoStack{#1}%
  }%
}%
```

\pdfcolSetCurrentColor

```
\def\pdfcolSetCurrentColor{%
  \pdfcolorstack\@pdfcolorstack set{\current@color}%
}%
```

\pdfcolSetCurrent

```
\def\pdfcolSetCurrent#1{%
  \ifx\\#1\\%
    \pdfcolorstack\@pdfcolorstack current\relax
  \else
    \pdfcolIfStackExists{#1}{%
      \pdfcolorstack\csname pdfcol@Stack@#1\endcsname current\relax
    }{%
      \pdfcol@ErrorNoStack{#1}%
    }%
  \fi
}%
```

\pdfcol@ErrorNoStack

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

2.5 Disabled interface macros

```
\else
```

\pdfcolErrorNoStacks

```
\def\pdfcolErrorNoStacks{%
  \@PackageError{pdfcol}{%
    Color stacks are not available%
  }{%
    Update pdfTeX (1.40) and `pdftex.def' (0.04b) %
       if necessary.\MessageBreak
    Ensure that `pdftex.def' is loaded %
       (package `color' or `xcolor').\MessageBreak
    Further messages can be found in TeX's %
       protocol file `\jobname.log'.\MessageBreak
    \MessageBreak
    \@ehc
  }%
  \global\let\pdfcolErrorNoStacks\relax
}%
```

\PDFCOL@Disabled

```
\def\PDFCOL@Disabled{%
  \PDFCOL@Message{%
    pdfTeX's color stacks are not available%
  }%
  \global\let\PDFCOL@Disabled\relax
}%
```

\pdfcolInitStack

```
\def\pdfcolInitStack#1{%
  \PDFCOL@Disabled
}%
```

\pdfcolIfStackExists

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

\pdfcolSwitchStack

```
\def\pdfcolSwitchStack#1{%
  \PDFCOL@Disabled
}%
```

\pdfcolSetCurrentColor

```
  \def\pdfcolSetCurrentColor{%
    \PDFCOL@Disabled
  }%
```

\pdfcolSetCurrent

```
  \def\pdfcolSetCurrent#1{%
    \PDFCOL@Disabled
  }%
\fi
\PDFCOL@AtEnd%
⟨/package⟩
```

3 Test

3.1 Catcode checks for loading

```
⟨*test1⟩
\catcode`\{=1 %
\catcode`\}=2 %
\catcode`\#=6 %
\catcode`\@=11 %
\expandafter\ifx\csname count@\endcsname\relax
  \countdef\count@=255 %
\fi
\expandafter\ifx\csname @gobble\endcsname\relax
  \long\def\@gobble#1{}%
\fi
\expandafter\ifx\csname @firstofone\endcsname\relax
  \long\def\@firstofone#1{#1}%
\fi
\expandafter\ifx\csname loop\endcsname\relax
  \expandafter\@firstofone
\else
  \expandafter\@gobble
\fi
{%
  \def\loop#1\repeat{%
    \def\body{#1}%
    \iterate
  }%
  \def\iterate{%
    \body
      \let\next\iterate
    \else
      \let\next\relax
    \fi
    \next
  }%
  \let\repeat=\fi
}%
\def\RestoreCatcodes{}
\count@=0 %
\loop
  \edef\RestoreCatcodes{%
    \RestoreCatcodes
    \catcode\the\count@=\the\catcode\count@\relax
  }%
\ifnum\count@<255 %
  \advance\count@ 1 %
```

```
\repeat

\def\RangeCatcodeInvalid#1#2{%
  \count@=#1\relax
  \loop
    \catcode\count@=15 %
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\def\RangeCatcodeCheck#1#2#3{%
  \count@=#1\relax
  \loop
    \ifnum#3=\catcode\count@
    \else
      \errmessage{%
        Character \the\count@\space
        with wrong catcode \the\catcode\count@\space
        instead of \number#3%
      }%
    \fi
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\def\space{ }
\expandafter\ifx\csname LoadCommand\endcsname\relax
  \def\LoadCommand{\input pdfcol.sty\relax}%
\fi
\def\Test{%
  \RangeCatcodeInvalid{0}{47}%
  \RangeCatcodeInvalid{58}{64}%
  \RangeCatcodeInvalid{91}{96}%
  \RangeCatcodeInvalid{123}{255}%
  \catcode`\@=12 %
  \catcode`\\=0 %
  \catcode`\%=14 %
  \LoadCommand
  \RangeCatcodeCheck{0}{36}{15}%
  \RangeCatcodeCheck{37}{37}{14}%
  \RangeCatcodeCheck{38}{47}{15}%
  \RangeCatcodeCheck{48}{57}{12}%
  \RangeCatcodeCheck{58}{63}{15}%
  \RangeCatcodeCheck{64}{64}{12}%
  \RangeCatcodeCheck{65}{90}{11}%
  \RangeCatcodeCheck{91}{91}{15}%
  \RangeCatcodeCheck{92}{92}{0}%
  \RangeCatcodeCheck{93}{96}{15}%
  \RangeCatcodeCheck{97}{122}{11}%
  \RangeCatcodeCheck{123}{255}{15}%
  \RestoreCatcodes
}
\Test
\csname @@end\endcsname
\end
```

⟨/test1⟩

3.2 Very simple test

```
⟨*test2 | test3⟩
\NeedsTeXFormat{LaTeX2e}
\nofiles
\documentclass{article}
```

```
\usepackage{pdfcol}[2018/11/01]
\usepackage{qstest}
\IncludeTests{*}
\LogTests{log}{*}{*}
\begin{document}
  \begin{qstest}{pdfcol}{}%
    \makeatletter
⟨*test2⟩
    \Expect*{\ifpdfcolAvailable true\else false\fi}{false}%
⟨/test2⟩
⟨*test3⟩
    \Expect*{\ifpdfcolAvailable true\else false\fi}{true}%
    \Expect*{\number\@pdfcolorstack}{0}%
⟨/test3⟩
    \setbox0=\hbox{%
      \pdfcolInitStack{test}%
⟨*test3⟩
      \Expect*{\number\pdfcol@Stack@test}{1}%
      \Expect*{\number\@pdfcolorstack}{0}%
⟨/test3⟩
      \pdfcolSwitchStack{test}%
⟨*test3⟩
      \Expect*{\number\@pdfcolorstack}{1}%
⟨/test3⟩
      \pdfcolSetCurrent{test}%
      \pdfcolSetCurrent{}%
    }%
    \Expect*{\the\wd0}{0.0pt}%
⟨*test3⟩
    \Expect*{\number\@pdfcolorstack}{0}%
    \Expect*{\number\pdfcol@Stack@test}{1}%
    \Expect*{\pdfcolIfStackExists{test}{true}{false}}{true}%
⟨/test3⟩
    \Expect*{\pdfcolIfStackExists{dummy}{true}{false}}{false}%
  \end{qstest}%
\end{document}
⟨/test2 | test3⟩
```

3.3 Detection of package **luacolor**

```
⟨*test4⟩
\NeedsTeXFormat{LaTeX2e}
\documentclass{article}
\usepackage{luacolor}
\usepackage{pdfcol}
\makeatletter
\ifpdfcolAvailable
  \@latex@error{Detection of package luacolor failed}%
\fi
\csname @@end\endcsname
⟨/test4⟩
```

4 Installation

4.1 Download

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

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

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

[1] http://ctan.org/pkg/pdfcol

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.

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 TEX:

```
tex pdfcol.dtx
```

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
test/pdfcol-test4.tex → doc/latex/oberdiek/test/pdfcol-test4.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 (teTEX, mikTEX, ...) relies on file name databases, you must refresh these. For example, teTEX users run texhash or mktexlsr.

4.5 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{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 pdfLaTeX:

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

5 Catalogue

The following XML file can be used as source for the TeX 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 pdfcol.xml.

```
⟨*catalogue⟩
<?xml version='1.0' encoding='us-ascii'?>
<!DOCTYPE entry SYSTEM 'catalogue.dtd'>
<entry datestamp='$Date$' modifier='$Author$' id='pdfcol'>
  <name>pdfcol</name>
  <caption>Defines macros fpr maintaining color stacks under pdfTeX.</caption>
  <authorref id='auth:oberdiek'/>
  <copyright owner='Heiko Oberdiek' year='2007'/>
  <license type='lppl1.3'/>
  <version number='1.5'/>
  <description>
    Since version 1.40 pdfTeX supports color stacks.
    The driver file <tt>pdftex.def</tt> for package
    <xref refid='color'>color</xref> defines and uses a main color
    stack since version v0.04b.
    <p/>
    This package is intended for package writers.
    It defines macros for setting and maintaining new color stacks.
    <p/>
    The package is part of the <xref refid='oberdiek'>oberdiek</xref>
    bundle.
  </description>
  <documentation details='Package documentation'
      href='ctan:/macros/latex/contrib/oberdiek/pdfcol.pdf'/>
  <ctan file='true' path='/macros/latex/contrib/oberdiek/pdfcol.dtx'/>
  <miktex location='oberdiek'/>
  <texlive location='oberdiek'/>
  <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
</entry>
⟨/catalogue⟩
```

6 History

[2007/09/09 v1.0]

- First version.

[2007/12/09 v1.1]

- \pdfcolSetCurrentColor added.

[2007/12/12 v1.2]

- Detection for package luacolor added.

[2016/05/16 v1.3]

- Documentation updates.

[2016/05/17 v1.4]

- Use luatex85 package for new luatex compatibility

[2018/11/01 v1.5]

- Remove luatex85 dependency

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

\# 299
\% 375
\@ 300, 373
\@PackageError 215, 256, 260
\@PackageInfo 211
\@PackageInfoNoLine 143
\@PackageWarningNoLine 138
\@ehc 217, 256, 270
\@firstofone 308, 311
\@firstoftwo 224, 227
\@gobble 305, 313
\@latex@error 443
\@pdfcolorstack 235, 242, 246, 411, 417, 421, 428
\@secondoftwo 222, 230
\@undefined 58, 120
\\ 245, 374
\{ 297
\} 298

A

\advance 338, 346, 361
\aftergroup 29

B

\begin 403, 404
\body 317, 321

C

\catcode 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 69, 70, 72, 73, 74, 78, 79, 80, 81, 82, 83, 84, 87, 88, 90, 91, 92, 93, 97, 99, 297, 298, 299, 300, 335, 344, 352, 356, 373, 374, 375
\chardef 208
\count@ 302, 331, 335, 337, 338, 342, 344, 345, 346, 350, 352, 355, 356, 360, 361
\countdef 302
\csname 14, 21, 50, 66, 76, 113, 129, 156, 166, 176, 178, 180, 186, 200, 203, 207, 208, 212, 221, 227, 230, 236, 249, 301, 304, 307, 310, 365, 392, 445
\current@color 242

D

\documentclass 398, 438

E

\empty 17, 18
\end 393, 433, 434
\endcsname 14, 21, 50, 66, 76, 113, 129, 156, 166, 176, 178, 180, 186, 200, 203, 207, 208, 212, 221, 227, 230, 236, 249, 301, 304, 307, 310, 365, 392, 445
\endinput 29, 111
\endlinechar 4, 35, 71, 77, 89
\errmessage 354
\Expect 407, 410, 411, 416, 417, 421, 426, 428, 429, 430, 432

H

\hbox 413

I

\ifcase 136
\ifnum 337, 345, 352, 360
\ifpdf 136
\ifPDFCOL@luacolor 127, 136, 147
\ifpdfcolAvailable 2, 125, 125, 154, 164, 174, 198, 407, 410, 442
\ifx . 15, 18, 21, 50, 58, 61, 113, 120, 129, 156, 166, 176, 178, 180,

186, 200, 203, 207, 221, 227, 230, 245, 301, 304, 307, 310, 365
\immediate 23, 52
\IncludeTests 401
\input 114, 366
\iterate 318, 320, 322

J

\jobname 268

L

\LoadCommand 366, 376
\LogTests 402
\loop 316, 332, 343, 351
\ltx@newif 125, 127

M

\makeatletter 405, 441
\MessageBreak 159, 169, 189, 190, 264, 266, 268, 269

N

\NeedsTeXFormat 396, 437
\next 322, 324, 326
\nofiles 397
\number 212, 357, 411, 416, 417, 421, 428, 429

P

\PackageInfo 26
\PDFCOL@AtEnd 95, 96, 111, 294
\PDFCOL@Disabled 274, 281, 285, 288, 291
\pdfcol@ErrorNoStack . . . 238, 251, 255
\pdfcol@LiteralModifier 203, 209
\PDFCOL@luacolorfalse 130
\PDFCOL@luacolortrue 132
\PDFCOL@Message 137, 142, 145, 158, 168, 188, 275
\PDFCOL@RequirePackage 112, 124, 134, 135
\pdfcol@Stack@test 416, 429
\pdfcol@Value 200, 209
\pdfcolAvailablefalse 141, 157, 167, 187, 197
\pdfcolAvailabletrue 126, 196
\pdfcolErrorNoStacks *2*, 199, 259
\pdfcolIfStackExists *2*, 220, 234, 248, 283, 430, 432
\pdfcolInitStack *2*, 206, 280, 414
\pdfcolorstack . . 122, 160, 242, 246, 249
\pdfcolorstackinit 121, 170, 209
\pdfcolSetCurrent . *3*, 244, 290, 423, 424
\pdfcolSetCurrentColor *3*, 241, 287
\pdfcolSwitchStack . . . *3*, 233, 284, 419
\pdfextension 120, 122
\pdffeedback 121
\protected 122
\ProvidesPackage 19, 67

R

\RangeCatcodeCheck 349, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388
\RangeCatcodeInvalid 341, 369, 370, 371, 372
\repeat 316, 328, 339, 347, 362
\RequirePackage 117, 182
\RestoreCatcodes . . 330, 333, 334, 389

S

\setbox 413
\space 355, 356, 364

T

\Test 368, 391
\the 77, 78, 79, 80, 81, 82, 83, 84, 97, 335, 355, 356, 426
\TMP@EnsureCode 94, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110

U

\usepackage 399, 400, 439, 440

W

\wd 426
\write 23, 52

X

\x 14, 15, 18, 22, 26, 28, 51, 56, 66, 75, 87