

The pdfmanagement-firstaid package – temporary patches and package replacements L^AT_EX PDF management bundle

The L^AT_EX Project^{*}

Version 0.97d, released 2026-10-06

1 pdfmanagement-firstaid documentation

This code is temporary! It tries to patch commands of other packages or even replace package which are incompatible with the pdfmanagement, to remove clashes and test if everything works as expected. This code should disappear when packages adapt to the central interfaces.

The package contains an number of sections for various packages. Every section can be disabled in (the first) `\DocumentMetadata` with `debug={firstaidoff={name1,name2,...},...}`.

```
1 <*package>
2 \ProvidesExplPackage{pdfmanagement-firstaid}{2026-10-06}{0.97d}
3   {LaTeX PDF management bundle / firstaid-patches}
4
5 <@@=pdfmanagement>
6 \clist_map_inline:nn
7   {pgf,xmp,xcolor,color,output,colorspace}
8   {
9     \bool_new:c          { g__pdfmanagement_firstaid_#1_bool }
10    \bool_gset_true:c { g__pdfmanagement_firstaid_#1_bool }
11  }
12 \clist_map_inline:Nn \g__pdfmanagement_firstaidoff_clist
13 {
14   \bool_if_exist:cT { g__pdfmanagement_firstaid_#1_bool }
15   {
16     \bool_gset_false:c { g__pdfmanagement_firstaid_#1_bool }
17   }
18 }
19 \msg_new:nnn { pdfmanagement } { firstaid }
20   { loading~pdfmanagement~firstaid~code~for~#1 }
21 \msg_new:nnn { pdfmanagement } { firstaid-changed }
22   { package~#1~has~changed.~Check~if~patch~is~still~valid! }
23 \msg_new:nnn { pdfmanagement } { firstaid-disabled }
24   { The~loading~of~package~#1~is~disabled.\\
25     It~is~not~compatible~with~the~PDF~management. }
26 \msg_new:nnn { pdfmanagement } { firstaid-too-old }
```

^{*}E-mail: latex-team@latex-project.org

```

27         { Package~#1~is~too~old~and~not~compatible.\\
28           Get~at~least~version~#2.}
29

```

1.1 color

`color` is not incompatible, but the new `hyperref` driver makes use of `l3color` to set the colors. It is therefore necessary to patch some internal `color` commands, so that colors defined with its `\definecolor` command are known to `l3color` and so `hyperref`. This only supports the color models from `l3color` (which covers all standard model of the `color` package). The `named` model is mapped to `\color_set:nn`.

This patch serves also as test to check if this change can be safely added to `color` later.

```

30 \bool_if:NT \g__pdfmanagement_firstaid_color_bool
31 {
32   \declare@file@substitution{color.sty}{color-ltx.sty}
33 }

```

1.2 xcolor

`xcolor` is not incompatible, but the new `hyperref` driver makes use of `l3color` to set the colors. It is therefore necessary to patch `xcolor`, so that colors defined with its `\definecolor` command are known to `l3color` and so `hyperref`. This only supports the color model from `l3color`. Colors defined with the models `cm` and `thsb` are silently ignored.

The `named` model is mapped to `\color_set:nn`.

```

34 \bool_if:NT \g__pdfmanagement_firstaid_xcolor_bool
35 {
36   \AddToHook
37   {
38     package/xcolor/after
39   }
40   {\RequirePackage{xcolor-patches-tmp-ltx}}

```

The patch must be before color definitions are loaded, which will happen in hooks in a newer `xcolor`:

```

41   \DeclareHookRule{package/xcolor/after}{pdfmanagement-firstaid}{before}{xcolor}
42 }

```

1.3 pgf

In `pgf`, resource management is set up in the file `pgfutil-common.tex`. This then provides three functions for adding to the resources, all of which are objects:

- `\pgfutil@addpdfresource@extgs`: Extended graphics state
- `\pgfutil@addpdfresource@colorspaces`: Color spaces
- `\pgfutil@addpdfresource@patterns`: Patterns

These resource dictionaries are used by adding entries in a cumulative sense; the macro layer deals with ensuring that each entry is only given once. Note that the objects themselves must be given only once for each page.

To support these functions, there are a series of set-up macros which install these resources. That has to take place for every page: the exact route therefore depends on the driver.

For the pdfmanagement project we need to avoid that pgf interferes in ExtGState, ColorSpace and Pattern (Shadings are added to the xform resources and so probably unproblematic for now). The actual patch is in a file hook guarded by the boolean, the rest of the code is always defined.

```

43 \bool_if:NT \g__pdfmanagement_firstaid_pgf_bool
44 {
45   \msg_info:nnn{pdfmanagement}{firstaid}{pgf}
46   \AddToHook
47   {
48     file/pgfrcs.sty/after
49   }
50   {
51     \cs_set_eq:NN
52     \__pdfmanagement_pgfori_pgful@setuppdfresources
53     \pgful@setuppdfresources
54     \def\pgful@setuppdfresources
55     {
56       \__pdfmanagement_pgf_sys_setuppdfresources_plain:
57     }
58   }
59 }
60 %\def\pgful@addpdfresource@extgs#1{\pgf@sys@addpdfresource@extgs@plain{#1}}
61 %\def\pgful@addpdfresource@colorspaces#1{\pgf@sys@addpdfresource@colorspaces@plain{#1}}
62 %\def\pgful@addpdfresource@patterns#1{\pgf@sys@addpdfresource@patterns@plain{#1}}
63 %\def\pgful@setuppdfresources{\pgf@sys@setuppdfresources@plain}
64 % \pgf@sys@pdf@possible@resources %used in xform
65 %Trying to patch pgf ..
66 \cs_new_protected:Npn \__pdfmanagement_pgf_sys_setuppdfresources_plain:
67 {
68   %objects are already created ...
69   \def\pgf@sys@pdf@possible@resources
70   {
71     /ColorSpace~\pdf_object_ref:n {\__pdf/Page/Resources/ColorSpace}
72     /Pattern ~\pdf_object_ref:n {\__pdf/Page/Resources/Pattern}
73     /ExtGState ~\pdf_object_ref:n {\__pdf/Page/Resources/ExtGState}
74   }
75   \let\pgf@sys@pdf@check@resources=\relax%
76   %not sure if needed, but perhaps the lists are used somewhere else ...
77   \let\pgf@sys@pgf@resource@list@extgs=\pgful@empty%
78   \let\pgf@sys@pgf@resource@list@patterns=\pgful@empty%
79   \let\pgf@sys@pgf@resource@list@colorspaces=\pgful@empty%
80   % the commands to add page resources
81   \def\pgf@sys@addpdfresource@extgs@plain##1
82   {
83     %\exp_after:wN %for transparent which passes a command
84     \__pdfmanagement_patch_pgfextgs:w ##1\q_stop
85   }

```

```

86 \def\pgf@sys@addpdfresource@patterns@plain##1
87 {
88   \_pdfmanagement_patch_pgfpatterns:w ##1\q_stop
89 }
90 \def\pgf@sys@addpdfresource@colorspaces@plain##1
91 {
92   \_pdfmanagement_patch_pgfcolorspaces:w ##1\q_stop
93 }
94 }
95
96 %\AtEndPreamble{\pgfutil@setuppdfresources}
97 % helper commands as pgf doesn't pass resources as two arguments
98 % code to add to the resources existing stuff in the format "/name value":
99 \cs_new:Npn \_pdfmanagement_split_dict_entry_aux:NNw #1 #2 /#3~#4\q_stop
100 {
101   \tl_set:Nn #1 {#3}
102   \tl_set:Nn #2 {#4}
103 }
104
105 \cs_generate_variant:Nn \tl_trim_spaces:n{V}
106 \cs_generate_variant:Nn \pdfmanagement_add:nnn {nee}
107 \cs_new:Npn \_pdfmanagement_patch_pgftxtgs:w #1/#2<<#3>>#4\q_stop
108 {
109   \tl_set:Nn \l_tmpa_tl{#2}
110   \pdfmanagement_add:nee
111     {Page/Resources/ExtGState}{\tl_trim_spaces:V\l_tmpa_tl}{<<#3 #4>>}
112 }
113 \cs_new:Npn \_pdfmanagement_patch_pgfpatterns:w #1/#2\space#3\q_stop
114 {
115   \pdfmanagement_add:nee
116     {Page/Resources/Pattern}{\tl_trim_spaces:n{#2}}{#3}
117 }
118 \cs_new:Npn \_pdfmanagement_patch_pgfcolorspaces:w #1/#2[#3]#4\q_stop
119 {
120   \pdfmanagement_add:nee
121     {Page/Resources/ColorSpace}{\tl_trim_spaces:n{#2}}{[#3]}
122 }
123

```

1.4 xmp

This handles the new xmp code.

```

124 \bool_if:NT \g__pdfmanagement_firstaid_xmp_bool
125 {
126   \disable@package@load{hyperxmp}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{hyperxmp}}
127   \disable@package@load{pdfx}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{pdfx}}
128   \AddToHook{package/doclicense/after}
129   {
130     \AtBeginDocument
131     {
132       \hypersetup
133       {
134         pdfcopyright = {\doclicenseLongTextForHyperref},
135         pdflicenseurl = {\doclicenseURL},

```

```

136         }
137     }
138 }
139 }
140 \end{package}

```

1.5 colorspace

This is rather difficult as no real places to inject patches at first a try to avoid that its ExtGState is missing: it can not be avoided to recreate the objects (and so to get duplicates) as colorspace uses temporary macros whose contents is lost.

```

141 \begin{package}
142 \bool_if:NT \g__pdfmanagement_firstaid_colorspace_bool
143 {
144     \AddToHook
145     {
146         package/colorspace/after
147     }
148     {\RequirePackage{colorspace-patches-tmp-ltx}}
149 }
150 \end{package}

```

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

Symbols		<code>\cs_new:Npn</code> 99, 107, 113, 118
<code>\</code> 24, 27		<code>\cs_new_protected:Npn</code> 66
		<code>\cs_set_eq:NN</code> 51
A		D
<code>\AddToHook</code> 36, 46, 128, 144		<code>\DeclareHookRule</code> 41
<code>\AtBeginDocument</code> 130		<code>\def</code> 54, 60, 61, 62, 63, 69, 81, 86, 90
<code>\AtEndPreamble</code> 96		<code>\definecolor</code> 2
B		<code>\doclicenseLongTextForHyperref</code> 134
bool commands:		<code>\doclicenseURL</code> 135
<code>\bool_gset_false:N</code> 16		<code>\DocumentMetadata</code> 1
<code>\bool_gset_true:N</code> 10		E
<code>\bool_if:NTF</code> 30, 34, 43, 124, 142		exp commands:
<code>\bool_if_exist:NTF</code> 14		<code>\exp_after:wN</code> 83
<code>\bool_new:N</code> 9		H
C		<code>\hypersetup</code> 132
clist commands:		L
<code>\clist_map_inline:Nn</code> 12		<code>\let</code> 75, 77, 78, 79
<code>\clist_map_inline:nn</code> 6		M
color commands:		msg commands:
<code>\color_set:nn</code> 2		<code>\msg_info:nnn</code> 45
cs commands:		
<code>\cs_generate_variant:Nn</code> ... 105, 106		

