

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

The L^AT_EX Project*

Version 0.96a, released 2023-10-09

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}{2023-10-09}{0.96a}
3   {LaTeX PDF management testphase bundle / firstaid-patches}
4
5 <@@=pdfmanagement>
6 \clist_map_inline:nn
7   {pgf,transparent,xmp,pdflscape,xcolor,color,
8    beamer,output,colorspace,fontspec,luacolor}
9   {
10     \bool_new:c          { g__pdfmanagement_firstaid_#1_bool }
11     \bool_gset_true:c { g__pdfmanagement_firstaid_#1_bool }
12   }
13 \clist_map_inline:Nn \g__pdfmanagement_firstaidoff_clist
14   {
15     \bool_if_exist:cT { g__pdfmanagement_firstaid_#1_bool }
16     {
17       \bool_gset_false:c { g__pdfmanagement_firstaid_#1_bool }
18     }
19   }
20 \msg_new:nnn { pdfmanagement } { firstaid }
21             { loading-pdfmanagement-firstaid-code-for-#1 }
22 \msg_new:nnn { pdfmanagement } { firstaid-changed }
23             { package-#1-has-changed.-Check-if-patch-is-still-valid! }
24 \msg_new:nnn { pdfmanagement } { firstaid-disabled }
25             { The-loading-of-package-#1-is-disabled.\
26             It-is-not-compatible-with-the-PDF-management. }
```

*E-mail: latex-team@latex-project.org

```

27 \msg_new:nnn { pdfmanagement } { firstaid-too-old }
28           { Package-#1-is-too-old-and-not-compatible.\\
29             Get-at-least-version-#2.}
30

```

1.1 Fontspec

When using luaLaTeX opacity is broken (issue #30) as fontspec/luatload doesn't use the pdfmanagement but write to ExtGState directly To repair this some new lua code is needed. It also needs a new fontspec and a change in the color key.

```

31 \sys_if_engine luatex:T
32   {
33     \bool_if:NT \g__pdfmanagement_firstaid_fontspec_bool
34       {
35         \@ifpackagelater{fontspec}{2023-03-10}
36         {
37           \__fontspec_keys_define_code:nnn {fontspec} {Color}
38           {
39             \tl_set:Nn \l__fontspec_hexcol_tl {#1}
40           }
41         }{}
42       }
43   }

```

1.2 beamer

beamer made use of the now unsupported syntax `linkbordercolor={.5□.5□.5}`. This has been changed so the patch has been removed.

1.3 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.

```

44 \bool_if:NT \g__pdfmanagement_firstaid_color_bool
45   {
46     \declare@file@substitution{color.sty}{color-ltx.sty}
47   }

```

1.4 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`.

```

48 \bool_if:NT \g__pdfmanagement_firstaid_xcolor_bool

```

```

49 {
50   \AddToHook
51   {
52     package/xcolor/after
53   }
54   {\RequirePackage{xcolor-patches-tmp-ltx}}

```

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

```

55   \DeclareHookRule{package/xcolor/after}{pdfmanagement-firstaid}{before}{xcolor}
56 }

```

1.5 luacolor

The luacolor package doesn't take colors from l3color into account. We add a fix, but only for pdf mode. luacolor can disable the code by clearing the hook if needed.

```

57 \bool_lazy_all:nT
58 {
59   {\sys_if_engine_luatex_p:}
60   {\g_pdfmanagement_firstaid_luacolor_bool}
61   {\sys_if_output_pdf_p:}
62 }
63 {
64   \AddToHook{package/luacolor/after}
65   {
66     \cs_set_protected:Npn \__color_backend_select:nn #1#2
67     {
68       \tl_set:Nn \l__color_backend_fill_tl {#1}
69       \tl_set:Nn \l__color_backend_stroke_tl {#2}
70       \LuaCol@setattribute\LuaCol@Attribute
71       {
72         \directlua
73         {
74           oberdiek.luacolor.get("\luaescapestring{#1~#2}")
75         }
76       }
77     }
78     \cs_set_protected:Npn \__color_backend_fill:n #1
79     {
80       \tl_set:Nn \l__color_backend_fill_tl {#1}
81       \LuaCol@setattribute\LuaCol@Attribute
82       {
83         \directlua
84         {
85           oberdiek.luacolor.get("\luaescapestring{#1}")
86         }
87       }
88     }
89     \cs_set_protected:Npn \__color_backend_stroke:n #1
90     {
91       \tl_set:Nn \l__color_backend_stroke_tl {#1}
92       \LuaCol@setattribute\LuaCol@Attribute
93       {

```

```

94         \directlua
95         {
96             oberdiek.luacolor.get("\luaescapestring{#1}")
97         }
98     }
99 }
100 \cs_set_protected:Npn \__color_backend_reset: {}
101 \cs_set_eq:NN \__color_backend_fill_reset: \__color_backend_reset:
102 \cs_set_eq:NN \__color_backend_stroke_reset: \__color_backend_reset:
103 }
104 }

```

1.6 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.

```

105
106 \bool_if:NT \g__pdfmanagement_firstaid_pgf_bool
107 {
108     \msg_info:nnn{pdfmanagement}{firstaid}{pgf}
109     \AddToHook
110     {
111         file/pgfrcs.sty/after
112     }
113     {
114         \cs_set_eq:NN
115         \__pdfmanagement_pgfori_pgfutil@setuppdfresources
116         \pgfutil@setuppdfresources
117         \def\pgfutil@setuppdfresources
118         {
119             \pdfmanagement_if_active:TF
120             {
121                 \__pdfmanagement_pgf_sys_setuppdfresources_plain:
122             }
123             {
124                 \__pdfmanagement_pgfori_pgfutil@setuppdfresources
125             }
126         }
127     }
128 }

```

```

126     }
127   }
128 }
129 %\def\pgfutil@addpdfresource@extgs#1{\pgf@sys@addpdfresource@extgs@plain{#1}}
130 %\def\pgfutil@addpdfresource@colorspaces#1{\pgf@sys@addpdfresource@colorspaces@plain{#1}}
131 %\def\pgfutil@addpdfresource@patterns#1{\pgf@sys@addpdfresource@patterns@plain{#1}}
132 %\def\pgfutil@setuppdfresources{\pgf@sys@setuppdfresources@plain}
133 % \pgf@sys@pdf@possible@resources %used in xform
134 %Trying to patch pgf ..
135 \cs_new_protected:Npn \__pdfmanagement_pgf_sys_setuppdfresources_plain:
136 {
137   %objects are already created ...
138   \def\pgf@sys@pdf@possible@resources
139   {
140     /ColorSpace~\pdf_object_ref:n {__pdf/Page/Resources/ColorSpace}
141     /Pattern ~\pdf_object_ref:n {__pdf/Page/Resources/Pattern}
142     /ExtGState ~\pdf_object_ref:n {__pdf/Page/Resources/ExtGState}
143   }
144   \let\pgf@sys@pdf@check@resources=\relax%
145   %not sure if needed, but perhaps the lists are used somewhere else ...
146   \let\pgf@sys@pgf@resource@list@extgs=\pgfutil@empty%
147   \let\pgf@sys@pgf@resource@list@patterns=\pgfutil@empty%
148   \let\pgf@sys@pgf@resource@list@colorspaces=\pgfutil@empty%
149   % the commands to add page resources
150   \def\pgf@sys@addpdfresource@extgs@plain##1
151   {
152     %\exp_after:wN %for transparent which passes a command
153     \__pdfmanagement_patch_pgfextgs:w ##1\q_stop
154   }
155   \def\pgf@sys@addpdfresource@patterns@plain##1
156   {
157     \__pdfmanagement_patch_pgfpatterns:w ##1\q_stop
158   }
159   \def\pgf@sys@addpdfresource@colorspaces@plain##1
160   {
161     \__pdfmanagement_patch_pgfcolorspaces:w ##1\q_stop
162   }
163 }
164
165 %\AtEndPreamble{\pgfutil@setuppdfresources}
166 % helper commands as pgf doesn't pass resources as two arguments
167 % code to add to the resources existing stuff in the format "/name value":
168 \cs_new:Npn \__pdfmanagement_split_dict_entry_aux:NNw #1 #2 /#3~#4\q_stop
169 {
170   \tl_set:Nn #1 {#3}
171   \tl_set:Nn #2 {#4}
172 }
173
174 \cs_generate_variant:Nn \tl_trim_spaces:n{V}
175 \cs_generate_variant:Nn \pdfmanagement_add:nnn {nex}
176 \cs_new:Npn \__pdfmanagement_patch_pgfextgs:w #1/#2<<#3>>#4\q_stop
177 {
178   \tl_set:Nx\l_tmpa_tl{#2}
179   \pdfmanagement_add:nex

```

```

180         {Page/Resources/ExtGState}{\tl_trim_spaces:V\l_tmpa_tl}{<<#3 #4>>}
181     }
182 \cs_new:Npn \__pdfmanagement_patch_pgfpatterns:w #1/#2\space#3\q_stop
183 {
184     \pdfmanagement_add:nex
185     {Page/Resources/Pattern}{\tl_trim_spaces:n{#2}}{#3}
186 }
187 \cs_new:Npn \__pdfmanagement_patch_pgfcolorspaces:w #1/#2[#3]#4\q_stop
188 {
189     \pdfmanagement_add:nex
190     {Page/Resources/ColorSpace}{\tl_trim_spaces:n{#2}}{[#3]}
191 }
192

```

1.7 transparent

We check if the new version is used and issue a warning otherwise

```

193 \bool_if:NT \g__pdfmanagement_firstaid_transparent_bool
194 {
195     \AddToHook{package/transparent/after}
196     {
197         \@ifpackagelater{transparent}{2022-10-27}{}
198         {
199             \msg_warning:nnnn{pdfmanagement}{firstaid-too-old}
200             {transparent}
201             {1.5~from~2022-10-27}
202         }
203     }
204 }

```

1.8 pdfscape

We check if the new version is used and issue a warning otherwise

```

205 \bool_if:NT \g__pdfmanagement_firstaid_pdfscape_bool
206 {
207     \AddToHook{package/pdfscape/after}
208     {
209         \@ifpackagelater{pdfscape}{2022-10-27}{}
210         {
211             \msg_warning:nnnn{pdfmanagement}{firstaid-too-old}
212             {pdfscape}
213             {0.13~from~2022-10-27}
214         }
215     }
216 }

```

1.9 xmp

This handles the new xmp code.

```

217 \bool_if:NT \g__pdfmanagement_firstaid_xmp_bool
218 {
219     \disable@package@load{hyperxmp}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{hyperxmp}}
220     \disable@package@load{pdfx}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{pdfx}}

```

```

221 \AddToHook{package/doclicense/after}
222 {
223     \AtBeginDocument
224     {
225         \hypersetup
226         {
227             pdfcopyright = {\doclicenseLongTextForHyperref},
228             pdflicenseurl = {\doclicenseURL},
229         }
230     }
231 }
232 }
233 \end{package}

```

1.10 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.

```

234 \begin{package}
235 \bool_if:NT \g__pdfmanagement_firstaid_colorspace_bool
236 {
237     \AddToHook
238     {
239         package/colorspace/after
240     }
241     {\RequirePackage{colorspace-patches-tmp-ltx}}
242 }
243 \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		\bool_new:N 10	
\ \ 25, 28		C	
A		clist commands:	
\AddToHook	50, 64, 109, 195, 207, 221, 237	\clist_map_inline:Nn	13
\AtBeginDocument	223	\clist_map_inline:nn	6
\AtEndPreamble	165	color commands:	
B		\color_set:nn	2
bool commands:		color internal commands:	
\bool_gset_false:N	17	__color_backend_fill:n	78
\bool_gset_true:N	11	__color_backend_fill_reset:	101
\bool_if:NTF	33, 44, 48, 106, 193, 205, 217, 235	l_color_backend_fill_tl	68, 80
\bool_if_exist:NTF	15	__color_backend_reset:	100, 101, 102
\bool_lazy_all:nTF	57	__color_backend_select:nn	66
		__color_backend_stroke:n	89
		__color_backend_stroke_reset:	102

<code>\l__color_backend_stroke_tl</code> .	69, 91	<code>\g__pdfmanagement_firstaid_-</code>	
cs commands:		<code>luacolor_bool</code>	60
<code>\cs_generate_variant:Nn</code>	174, 175	<code>\g__pdfmanagement_firstaid_-</code>	
<code>\cs_new:Npn</code>	168, 176, 182, 187	<code>pdfscape_bool</code>	205
<code>\cs_new_protected:Npn</code>	135	<code>\g__pdfmanagement_firstaid_pgf_-</code>	
<code>\cs_set_eq:NN</code>	101, 102, 114	<code>bool</code>	106
<code>\cs_set_protected:Npn</code>	66, 78, 89, 100	<code>\g__pdfmanagement_firstaid_-</code>	
		<code>transparent_bool</code>	193
D		<code>\g__pdfmanagement_firstaid_-</code>	
<code>\DeclareHookRule</code>	55	<code>xcolor_bool</code>	48
<code>\def</code>	117,	<code>\g__pdfmanagement_firstaid_xmp_-</code>	
	129, 130, 131, 132, 138, 150, 155, 159	<code>bool</code>	217
<code>\definecolor</code>	2	<code>\g__pdfmanagement_firstaidoff_-</code>	
<code>\directlua</code>	72, 83, 94	<code>clist</code>	13
<code>\doclicenseLongTextForHyperref</code> ...	227	<code>__pdfmanagement_patch_pgfcOLORSPACES:w</code>	
<code>\doclicenseURL</code>	228	161, 187
<code>\DocumentMetadata</code>	1	<code>__pdfmanagement_patch_pgftxtgs:w</code>	
		153, 176
E		<code>__pdfmanagement_patch_pgfpatterns:w</code>	
exp commands:		157, 182
<code>\exp_after:wN</code>	152	<code>__pdfmanagement_pgf_sys_-</code>	
		<code>setuppdfresources_plain:</code>	121, 135
F		<code>__pdfmanagement_pgfori_pgful@setuppdfresources</code>	
fontspec internal commands:		115, 124
<code>\l__fontspec_hexcol_tl</code>	39	<code>__pdfmanagement_split_dict_-</code>	
<code>__fontspec_keys_define_code:nnn</code>	37	<code>entry_aux:NNw</code>	168
		<code>\ProvidesExplPackage</code>	2
H			
<code>\hypersetup</code>	225	Q	
		quark commands:	
L		<code>\q_stop</code>	153, 157, 161, 168, 176, 182, 187
<code>\let</code>	144, 146, 147, 148		
<code>\luaescapestring</code>	74, 85, 96	R	
		<code>\relax</code>	144
M		<code>\RequirePackage</code>	54, 241
msg commands:			
<code>\msg_info:nnn</code>	108	S	
<code>\msg_new:nnn</code>	20, 22, 24, 27	<code>\space</code>	182
<code>\msg_warning:nnn</code>	219, 220	sys commands:	
<code>\msg_warning:nnnn</code>	199, 211	<code>\sys_if_engine luatex:TF</code>	31
		<code>\sys_if_engine luatex:p:</code>	59
P		<code>\sys_if_output_pdf_p:</code>	61
pdf commands:			
<code>\pdf_object_ref:n</code>	140, 141, 142	T	
pdfmanagement commands:		TeX and L ^A T _E X 2 _ε commands:	
<code>\pdfmanagement_add:nnn</code>		<code>\@ifpackagelater</code>	35, 197, 209
.....	175, 179, 184, 189	<code>\declare@file@substitution</code>	46
<code>\pdfmanagement_if_active:TF</code> ...	119	<code>\disable@package@load</code>	219, 220
pdfmanagement internal commands:		<code>\LuaCol@Attribute</code>	70, 81, 92
<code>\g__pdfmanagement_firstaid_-</code>		<code>\LuaCol@setattribute</code>	70, 81, 92
<code>color_bool</code>	44	<code>\pgf@sys@addpdfresource@colorspaces@plain</code>	
<code>\g__pdfmanagement_firstaid_-</code>		130, 159
<code>colorspace_bool</code>	235	<code>\pgf@sys@addpdfresource@extgs@plain</code>	
<code>\g__pdfmanagement_firstaid_-</code>		129, 150
<code>fontspec_bool</code>	33		

\pgf@sys@addpdfresource@patterns@plain	4, 130
.....	131, 155	
\pgf@sys@pdf@check@resources ..	144	
\pgf@sys@pdf@possible@resources .		
.....	133, 138	
\pgf@sys@pgf@resource@list@colorspaces	148
\pgf@sys@pgf@resource@list@extgs	146	
\pgf@sys@pgf@resource@list@patterns	147
\pgf@sys@setuppdfresources@plain	132	
\pgfutil@addpdfresource@colorspaces		
	4, 129
\pgfutil@addpdfresource@patterns	4, 131
\pgfutil@empty	146, 147, 148
\pgfutil@setuppdfresources	
	116, 117, 132, 165
tl commands:		
\tl_set:Nn	
....	39, 68, 69, 80, 91, 170, 171, 178	
\tl_trim_spaces:n ..	174, 180, 185, 190	
\l_tmpa_tl	178, 180