

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

The L^AT_EX Project*

Version 0.96h, released 2024-04-09

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}{2024-04-09}{0.96h}
3   {LaTeX PDF management testphase bundle / firstaid-patches}
4
5 <@=pdfmanagement>
6 \clist_map_inline:nn
7   {pgf,xmp,xcolor,color,output,colorspace,luacolor}
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 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.

```

43 \bool_lazy_all:nT
44   {
45     {\sys_if_engine luatex_p:}
46     {\g__pdfmanagement_firstaid_luacolor_bool}
47     {\sys_if_output_pdf_p:}
48   }
49   {
50     \AddToHook{package/luacolor/after}
51     {

```

```

52 \cs_set_protected:Npn \__color_backend_select:nn #1#2
53 {
54   \tl_set:Nn \l__color_backend_fill_tl {#1}
55   \tl_set:Nn \l__color_backend_stroke_tl {#2}
56   \LuaCol@setattribute\LuaCol@Attribute
57   {
58     \directlua
59     {
60       oberdiek.luacolor.get("\luaescapestring{#1~#2}")
61     }
62   }
63 }
64 \cs_set_protected:Npn \__color_backend_fill:n #1
65 {
66   \tl_set:Nn \l__color_backend_fill_tl {#1}
67   \LuaCol@setattribute\LuaCol@Attribute
68   {
69     \directlua
70     {
71       oberdiek.luacolor.get("\luaescapestring{#1}")
72     }
73   }
74 }
75 \cs_set_protected:Npn \__color_backend_stroke:n #1
76 {
77   \tl_set:Nn \l__color_backend_stroke_tl {#1}
78   \LuaCol@setattribute\LuaCol@Attribute
79   {
80     \directlua
81     {
82       oberdiek.luacolor.get("\luaescapestring{#1}")
83     }
84   }
85 }
86 \cs_set_protected:Npn \__color_backend_reset: {}
87 \cs_set_eq:NN \__color_backend_fill_reset: \__color_backend_reset:
88 \cs_set_eq:NN \__color_backend_stroke_reset: \__color_backend_reset:
89 }
90 }

```

1.4 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.

```

91
92 \bool_if:NT \g__pdfmanagement_firstaid_pgf_bool
93   {
94     \msg_info:nnn{pdfmanagement }{firstaid}{pgf}
95     \AddToHook
96       {
97         file/pgfrcs.sty/after
98       }
99     {
100      \cs_set_eq:NN
101        \__pdfmanagement_pgfori_pgfutil@setuppdfresources
102        \pgfutil@setuppdfresources
103      \def\pgfutil@setuppdfresources
104        {
105          \pdfmanagement_if_active:TF
106            {
107              \__pdfmanagement_pgf_sys_setuppdfresources_plain:
108            }
109            {
110              \__pdfmanagement_pgfori_pgfutil@setuppdfresources
111            }
112          }
113        }
114      }
115 %\def\pgfutil@addpdfresource@extgs#1{\pgf@sys@addpdfresource@extgs@plain{#1}}
116 %\def\pgfutil@addpdfresource@colorspaces#1{\pgf@sys@addpdfresource@colorspaces@plain{#1}}
117 %\def\pgfutil@addpdfresource@patterns#1{\pgf@sys@addpdfresource@patterns@plain{#1}}
118 %\def\pgfutil@setuppdfresources{\pgf@sys@setuppdfresources@plain}
119 % \pgf@sys@pdf@possible@resources %used in xform
120 %Trying to patch pgf ..
121 \cs_new_protected:Npn \__pdfmanagement_pgf_sys_setuppdfresources_plain:
122   {
123     %objects are already created ...
124     \def\pgf@sys@pdf@possible@resources
125       {
126         /ColorSpace~\pdf_object_ref:n {__pdf/Page/Resources/ColorSpace}
127         /Pattern   ~\pdf_object_ref:n {__pdf/Page/Resources/Pattern}
128         /ExtGState ~\pdf_object_ref:n {__pdf/Page/Resources/ExtGState}
129       }
130     \let\pgf@sys@pdf@check@resources=\relax%
131     %not sure if needed, but perhaps the lists are used somewhere else ...
132     \let\pgf@sys@pgf@resource@list@extgs=\pgfutil@empty%
133     \let\pgf@sys@pgf@resource@list@patterns=\pgfutil@empty%
134     \let\pgf@sys@pgf@resource@list@colorspaces=\pgfutil@empty%
135     % the commands to add page resources
136     \def\pgf@sys@addpdfresource@extgs@plain##1

```

```

137     {
138         %\exp_after:wN %for transparent which passes a command
139         \_pdfmanagement_patch_pgftxtgs:w ##1\q_stop
140     }
141     \def\pgf@sys@addpdfresource@patterns@plain##1
142     {
143         \_pdfmanagement_patch_pgfpatterns:w ##1\q_stop
144     }
145     \def\pgf@sys@addpdfresource@colorspaces@plain##1
146     {
147         \_pdfmanagement_patch_pgfcolorspaces:w ##1\q_stop
148     }
149 }
150
151 %\AtEndPreamble{\pgfutil@setuppdfresources}
152 % helper commands as pgf doesn't pass resources as two arguments
153 % code to add to the resources existing stuff in the format "/name value":
154 \cs_new:Npn \_pdfmanagement_split_dict_entry_aux:NNw #1 #2 /#3-#4\q_stop
155 {
156     \tl_set:Nn #1 {#3}
157     \tl_set:Nn #2 {#4}
158 }
159
160 \cs_generate_variant:Nn \tl_trim_spaces:n{V}
161 \cs_generate_variant:Nn \pdfmanagement_add:nnn {nee}
162 \cs_new:Npn \_pdfmanagement_patch_pgftxtgs:w #1/#2<<#3>>#4\q_stop
163 {
164     \tl_set:Ne\l_tmpa_tl{#2}
165     \pdfmanagement_add:nee
166     {Page/Resources/ExtGState}{\tl_trim_spaces:V\l_tmpa_tl}{<<#3 #4>>}
167 }
168 \cs_new:Npn \_pdfmanagement_patch_pgfpatterns:w #1/#2\space#3\q_stop
169 {
170     \pdfmanagement_add:nee
171     {Page/Resources/Pattern}{\tl_trim_spaces:n{#2}}{#3}
172 }
173 \cs_new:Npn \_pdfmanagement_patch_pgfcolorspaces:w #1/#2[#3]#4\q_stop
174 {
175     \pdfmanagement_add:nee
176     {Page/Resources/ColorSpace}{\tl_trim_spaces:n{#2}}{[#3]}
177 }
178

```

1.5 xmp

This handles the new xmp code.

```

179 \bool_if:NT \g__pdfmanagement_firstaid_xmp_bool
180 {
181     \disable@package@load{hyperxmp}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{hyperxmp}}
182     \disable@package@load{pdfx} {\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{pdfx}}
183     \AddToHook{package/doclicense/after}
184     {
185         \AtBeginDocument
186         {

```

```

187         \hypersetup
188         {
189             pdfcopyright = {\doclicenseLongTextForHyperref},
190             pdflicenseurl = {\doclicenseURL},
191         }
192     }
193 }
194 }
195 \end{package}

```

1.6 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.

```

196 \begin{package}
197 \bool_if:NT \g__pdfmanagement_firstaid_colorspace_bool
198 {
199     \AddToHook
200     {
201         package/colorspace/after
202     }
203     {\RequirePackage{colorspace-patches-tmp-ltx}}
204 }
205 \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>\clist_map_inline:nn</code> 6
<code>\</code> 24, 27	color commands:	
	A	<code>\color_set:nn</code> 2	
<code>\AddToHook</code> 36, 50, 95, 183, 199	color internal commands:	
<code>\AtBeginDocument</code> 185	<code>_color_backend_fill:n</code> 64	
<code>\AtEndPreamble</code> 151	<code>_color_backend_fill_reset:</code> . . . 87	
	B	<code>\l_color_backend_fill_tl</code> . . . 54, 66	
bool commands:		<code>_color_backend_reset:</code> . . . 86, 87, 88	
<code>\bool_gset_false:N</code> 16	<code>_color_backend_select:nn</code> 52	
<code>\bool_gset_true:N</code> 10	<code>_color_backend_stroke:n</code> 75	
<code>\bool_if:NTF</code> 30, 34, 92, 179, 197	<code>_color_backend_stroke_reset:</code> . . 88	
<code>\bool_if_exist:NTF</code> 14	<code>\l_color_backend_stroke_tl</code> . . 55, 77	
<code>\bool_lazy_all:nTF</code> 43	cs commands:	
<code>\bool_new:N</code> 9	<code>\cs_generate_variant:Nn</code> 160, 161	
	C	<code>\cs_new:Npn</code> 154, 162, 168, 173	
clist commands:		<code>\cs_new_protected:Npn</code> 121	
<code>\clist_map_inline:Nn</code> 12	<code>\cs_set_eq:NN</code> 87, 88, 100	
		<code>\cs_set_protected:Npn</code> . . 52, 64, 75, 86	

D	
<code>\DeclareHookRule</code>	41
<code>\def</code>	103,
115, 116, 117, 118, 124, 136, 141, 145	
<code>\definecolor</code>	2
<code>\directlua</code>	58, 69, 80
<code>\doclicenseLongTextForHyperref</code> ...	189
<code>\doclicenseURL</code>	190
<code>\DocumentMetadata</code>	1
E	
exp commands:	
<code>\exp_after:wN</code>	138
H	
<code>\hypersetup</code>	187
L	
<code>\let</code>	130, 132, 133, 134
<code>\luaescapestring</code>	60, 71, 82
M	
msg commands:	
<code>\msg_info:nnn</code>	94
<code>\msg_new:nnn</code>	19, 21, 23, 26
<code>\msg_warning:nnn</code>	181, 182
P	
pdf commands:	
<code>\pdf_object_ref:n</code>	126, 127, 128
pdfmanagement commands:	
<code>\pdfmanagement_add:nnn</code>	
.....	161, 165, 170, 175
<code>\pdfmanagement_if_active:TF</code> ...	105
pdfmanagement internal commands:	
<code>\g__pdfmanagement_firstaid_-</code> color_bool	30
<code>\g__pdfmanagement_firstaid_-</code> colorspace_bool	197
<code>\g__pdfmanagement_firstaid_-</code> luacolor_bool	46
<code>\g__pdfmanagement_firstaid_pgf_-</code> bool	92
<code>\g__pdfmanagement_firstaid_-</code> xcolor_bool	34
<code>\g__pdfmanagement_firstaid_xmp_-</code> bool	179
<code>\g__pdfmanagement_firstaidoff_-</code> clist	12
<code>__pdfmanagement_patch_pgfcolorspaces:w</code>	147, 173
<code>__pdfmanagement_patch_pgfextgs:w</code>	139, 162
<code>__pdfmanagement_patch_pgfpatterns:w</code>	143, 168
<code>__pdfmanagement_pgf_sys_-</code> setuppdfresources_plain:	107, 121
<code>__pdfmanagement_pgfori_pgful@setuppdfresources</code>	101, 110
<code>__pdfmanagement_split_dict_-</code> entry_aux:NNw	154
<code>\ProvidesExplPackage</code>	2
Q	
quark commands:	
<code>\q_stop</code>	139, 143, 147, 154, 162, 168, 173
R	
<code>\relax</code>	130
<code>\RequirePackage</code>	40, 203
S	
<code>\space</code>	168
sys commands:	
<code>\sys_if_engine luatex_p:</code>	45
<code>\sys_if_output_pdf_p:</code>	47
T	
\TeX and \LaTeX 2 ϵ commands:	
<code>\declare@file@substitution</code>	32
<code>\disable@package@load</code>	181, 182
<code>\LuaCol@Attribute</code>	56, 67, 78
<code>\LuaCol@setattribute</code>	56, 67, 78
<code>\pgf@sys@addpdfresource@colorspaces@plain</code>	116, 145
<code>\pgf@sys@addpdfresource@extgs@plain</code>	115, 136
<code>\pgf@sys@addpdfresource@patterns@plain</code>	117, 141
<code>\pgf@sys@pdf@check@resources</code> ..	130
<code>\pgf@sys@pdf@possible@resources</code>	119, 124
<code>\pgf@sys@pgf@resource@list@colorspaces</code>	134
<code>\pgf@sys@pgf@resource@list@extgs</code> ..	132
<code>\pgf@sys@pgf@resource@list@patterns</code>	133
<code>\pgf@sys@setuppdfresources@plain</code> ..	118
<code>\pgfutil@addpdfresource@colorspaces</code>	3, 116
<code>\pgfutil@addpdfresource@extgs</code> ..	3, 115
<code>\pgfutil@addpdfresource@patterns</code>	3, 117
<code>\pgfutil@empty</code>	132, 133, 134
<code>\pgfutil@setuppdfresources</code>	102, 103, 118, 151
tl commands:	
<code>\tl_set:Nn</code>	54, 55, 66, 77, 156, 157, 164
<code>\tl_trim_spaces:n</code> ..	160, 166, 171, 176
<code>\l_tmpa_tl</code>	164, 166