

# The pdfmanagement-firstaid package – temporary patches and package replacements L<sup>A</sup>T<sub>E</sub>X PDF management testphase bundle

The L<sup>A</sup>T<sub>E</sub>X Project\*

Version 0.95u, released 2023-02-07

## 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}{2023-02-07}{0.95u}
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](mailto: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/luatfloat doesn't use the pdfmanagement but write to ExtGState directly To repair this some new lua code is needed. This also improves the color parsing and allows to use cmyk and spot colors as font colors. It is also needed to use a patch version of `fontspec-luatex.sty` to slightly change the way how the Opacity and Color key are handled. The patch is only executed if luatfloat is new enough.

```

31 \sys_if_engine luatex:T
32   {
33     \lua_now:n
34     {
35       if-not~luatfloat.set_transparent_colorstack~
36       then~
37         tex.print("\bool_gset_false:N\g__pdfmanagement_firstaid_fontspec_bool")~
38       end
39     }
40     \bool_if:NT \g__pdfmanagement_firstaid_fontspec_bool
41     {
42       \msg_info:nnn{pdfmanagement}{firstaid}{fontspec}
43       \file_if_exist:nT { luatfloat-l3color-tmp.lua }
44       {
45         \lua_load_module:n { luatfloat-l3color-tmp.lua }
46       }
47
48       \declare@file@substitution{fontspec-luatex.sty}{fontspec-luatex-tmp-ltx.sty}
49       \AddToHook{package/fontspec/after}
50       {
51         \str_if_eq:vnF{ver@fontspec.sty}{2022/01/15-v2.8a-Font-selection-for-XeLaTeX-and-
52           {\msg_warn:nnn{pdfmanagement}{firstaid-changed}{fontspec}}
53       }
54     }
55   }

```

## 1.2 Output routine

The patches for the LaTeX output routine commands for the tagging of the header/footer, the suppressing of links and the patch to multicol for the para tagging by default, but must be requested with the key `testphase=new-or-1`. But until latex-lab has been updated we must ensure that they are loaded with the various phases.

```

56 \bool_if:NT \g__pdfmanagement_firstaid_output_bool
57   {

```

This catches also phase-II and tagpdf.

```

58     \file_if_exist:nF {new-or-1-latex-lab-testphase.ltx}
59     {
60       \AddToHook{package/latex-lab-testphase-new-or/after}
61       {\RequirePackage{output-patches-tmp-ltx}}

```

This is for phase-I

```
62      \AddToHook{package/tagpdf/after}
63      {\RequirePackage{output-patches-tmp-ltx}}
64    }
65  }
```

### 1.3 beamer

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

### 1.4 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.

```
66 \bool_if:NT \g__pdfmanagement_firstaid_color_bool
67 {
68   \declare@file@substitution{color.sty}{color-ltx.sty}
69 }
```

### 1.5 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`.

```
70 \bool_if:NT \g__pdfmanagement_firstaid_xcolor_bool
71 {
72   \AddToHook
73   {
74     package/xcolor/after
75   }
76   {\RequirePackage{xcolor-patches-tmp-ltx}}
77 }
```

### 1.6 luacolor

The luacolor package doesn't take colors from l3color into account. We add a fix, but only for pdf mode. It is disabled if luacolor is updated, assuming that the update will then handle it (or expl3 directly)

```
78 \bool_lazy_all:nT
79 {
80   {\sys_if_engine luatex_p:}
81   {\g__pdfmanagement_firstaid_luacolor_bool}
82   {\sys_if_output_pdf_p:}
```

```

83 }
84 {
85   \AddToHook{package/luacolor/after}
86   {
87     \@ifpackagelater{luacolor}{2021-02-18}{}
88     {
89       \cs_set_protected:Npn \__color_backend_select:nn #1#2
90       {
91         \tl_set:Nn \l__color_backend_fill_tl {#1}
92         \tl_set:Nn \l__color_backend_stroke_tl {#2}
93         \LuaCol@setattribute\LuaCol@Attribute
94         {
95           \directlua
96           {
97             oberdiek.luacolor.get("\luaescapestring{#1~#2}")
98           }
99         }
100       }
101       \cs_set_protected:Npn \__color_backend_fill:n #1
102       {
103         \tl_set:Nn \l__color_backend_fill_tl {#1}
104         \LuaCol@setattribute\LuaCol@Attribute
105         {
106           \directlua
107           {
108             oberdiek.luacolor.get("\luaescapestring{#1}")
109           }
110         }
111       }
112       \cs_set_protected:Npn \__color_backend_stroke:n #1
113       {
114         \tl_set:Nn \l__color_backend_stroke_tl {#1}
115         \LuaCol@setattribute\LuaCol@Attribute
116         {
117           \directlua
118           {
119             oberdiek.luacolor.get("\luaescapestring{#1}")
120           }
121         }
122       }
123       \cs_set_protected:Npn \__color_backend_reset: {}
124       \cs_set_eq:NN \__color_backend_fill_reset: \__color_backend_reset:
125       \cs_set_eq:NN \__color_backend_stroke_reset: \__color_backend_reset:
126     }
127   }
128 }

```

## 1.7 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.

```

129
130 \bool_if:NT \g__pdfmanagement_firstaid_pgf_bool
131 {
132   \msg_info:nnn{pdfmanagement }{firstaid}{pgf}
133   \AddToHook
134   {
135     file/pgfrcs.sty/after
136   }
137   {
138     \cs_set_eq:NN
139     \__pdfmanagement_pgfori_pgfutil@setuppdfresources
140     \pgfutil@setuppdfresources
141     \def\pgfutil@setuppdfresources
142     {
143       \pdfmanagement_if_active:TF
144       {
145         \__pdfmanagement_pgf_sys_setuppdfresources_plain:
146       }
147       {
148         \__pdfmanagement_pgfori_pgfutil@setuppdfresources
149       }
150     }
151   }
152 }
153 %\def\pgfutil@addpdfresource@extgs#1{\pgf@sys@addpdfresource@extgs@plain{#1}}
154 %\def\pgfutil@addpdfresource@colorspaces#1{\pgf@sys@addpdfresource@colorspaces@plain{#1}}
155 %\def\pgfutil@addpdfresource@patterns#1{\pgf@sys@addpdfresource@patterns@plain{#1}}
156 %\def\pgfutil@setuppdfresources{\pgf@sys@setuppdfresources@plain}
157 % \pgf@sys@pdf@possible@resources %used in xform
158 %Trying to patch pgf ..
159 \cs_new_protected:Npn \__pdfmanagement_pgf_sys_setuppdfresources_plain:
160 {
161   %objects are already created ...
162   \def\pgf@sys@pdf@possible@resources
163   {
164     /ColorSpace~\pdf_object_ref:n {\__pdf/Page/Resources/ColorSpace}
165     /Pattern ~\pdf_object_ref:n {\__pdf/Page/Resources/Pattern}
166     /ExtGState ~\pdf_object_ref:n {\__pdf/Page/Resources/ExtGState}
167   }
168   \let\pgf@sys@pdf@check@resources=\relax%
169   %not sure if needed, but perhaps the lists are used somewhere else ...

```

```

170 \let\pgf@sys@pgf@resource@list@extgs=\pgfutil@empty%
171 \let\pgf@sys@pgf@resource@list@patterns=\pgfutil@empty%
172 \let\pgf@sys@pgf@resource@list@colorspaces=\pgfutil@empty%
173 % the commands to add page resources
174 \def\pgf@sys@addpdfresource@extgs@plain##1
175 {
176     %\exp_after:wN %for transparent which passes a command
177     \__pdfmanagement_patch_pgftextgs:w ##1\q_stop
178 }
179 \def\pgf@sys@addpdfresource@patterns@plain##1
180 {
181     \__pdfmanagement_patch_pgfpatterns:w ##1\q_stop
182 }
183 \def\pgf@sys@addpdfresource@colorspaces@plain##1
184 {
185     \__pdfmanagement_patch_pgfcolorspaces:w ##1\q_stop
186 }
187 }
188
189 %\AtEndPreamble{\pgfutil@setuppdfresources}
190 % helper commands as pgf doesn't pass resources as two arguments
191 % code to add to the resources existing stuff in the format "/name value":
192 \cs_new:Npn \__pdfmanagement_split_dict_entry_aux:NNw #1 #2 /#3~#4\q_stop
193 {
194     \tl_set:Nn #1 {#3}
195     \tl_set:Nn #2 {#4}
196 }
197
198 \cs_generate_variant:Nn \tl_trim_spaces:n{V}
199 \cs_generate_variant:Nn \pdfmanagement_add:nnn {nex}
200 \cs_new:Npn \__pdfmanagement_patch_pgftextgs:w #1/#2<<#3>>#4\q_stop
201 {
202     \tl_set:Nx\l_tmpa_tl{#2}
203     \pdfmanagement_add:nex
204     {Page/Resources/ExtGState}{\tl_trim_spaces:V\l_tmpa_tl}{<<#3 #4>>}
205 }
206 \cs_new:Npn \__pdfmanagement_patch_pgfpatterns:w #1/#2\space#3\q_stop
207 {
208     \pdfmanagement_add:nex
209     {Page/Resources/Pattern}{\tl_trim_spaces:n{#2}}{#3}
210 }
211 \cs_new:Npn \__pdfmanagement_patch_pgfcolorspaces:w #1/#2[#3]#4\q_stop
212 {
213     \pdfmanagement_add:nex
214     {Page/Resources/ColorSpace}{\tl_trim_spaces:n{#2}}{[#3]}
215 }
216

```

## 1.8 transparent

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

```

217 \bool_if:NT \g__pdfmanagement_firstaid_transparent_bool
218 {
219     \AddToHook{package/transparent/after}

```

```

220 {
221   \@ifpackagelater{transparent}{2022-10-27}{}
222   {
223     \msg_warning:nnnn{pdfmanagement}{firstaid-too-old}
224     {transparent}
225     {1.5~from~2022-10-27}
226   }
227 }
228 }

```

## 1.9 pdfscape

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

```

229 \bool_if:NT \g__pdfmanagement_firstaid_pdfscape_bool
230 {
231   \AddToHook{package/pdfscape/after}
232   {
233     \@ifpackagelater{pdfscape}{2022-10-27}{}
234     {
235       \msg_warning:nnnn{pdfmanagement}{firstaid-too-old}
236       {pdfscape}
237       {0.13~from~2022-10-27}
238     }
239   }
240 }

```

## 1.10 xmp

This handles the new xmp code.

```

241 \bool_if:NT \g__pdfmanagement_firstaid_xmp_bool
242 {
243   \disable@package@load{hyperxmp}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{hyperxmp}}
244   \disable@package@load{pdfx}{\msg_warning:nnn{pdfmanagement}{firstaid-disabled}{pdfx}}
245   \AddToHook{package/doclicense/after}
246   {
247     \AtBeginDocument
248     {
249       \hypersetup
250       {
251         pdfcopyright = {\doclicenseLongTextForHyperref},
252         pdflicenseurl = {\doclicenseURL},
253       }
254     }
255   }
256 }
257 \</package>

```

## 1.11 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.

```

258 <*package>
259 \bool_if:NT \g__pdfmanagement_firstaid_colorspace_bool
260 {
261     \AddToHook
262     {
263         package/colorspace/after
264     }
265     {\RequirePackage{colorspace-patches-tmp-ltx}}
266 }
267 </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.

<b>Symbols</b>	
<code>\\</code> .....	25, 28, 37
<b>A</b>	
<code>\AddToHook</code> .....	49,
	60, 62, 72, 85, 133, 219, 231, 245, 261
<code>\AtBeginDocument</code> .....	247
<code>\AtEndPreamble</code> .....	189
<b>B</b>	
bool commands:	
<code>\bool_gset_false:N</code> .....	17
<code>\bool_gset_true:N</code> .....	11
<code>\bool_if:NTF</code> .....	40,
	56, 66, 70, 130, 217, 229, 241, 259
<code>\bool_if_exist:NTF</code> .....	15
<code>\bool_lazy_all:nTF</code> .....	78
<code>\bool_new:N</code> .....	10
<b>C</b>	
clist commands:	
<code>\clist_map_inline:Nn</code> .....	13
<code>\clist_map_inline:nn</code> .....	6
color commands:	
<code>\color_set:nn</code> .....	3
color internal commands:	
<code>\__color_backend_fill:n</code> .....	101
<code>\__color_backend_fill_reset:</code> ..	124
<code>\l_color_backend_fill_tl</code> ...	91, 103
<code>\__color_backend_reset:</code> .....	123, 124, 125
<code>\__color_backend_select:nn</code> .....	89
<code>\__color_backend_stroke:n</code> .....	112
<code>\__color_backend_stroke_reset:</code> ..	125
<code>\l_color_backend_stroke_tl</code> ..	92, 114
cs commands:	
<code>\cs_generate_variant:Nn</code> ....	198, 199
<code>\cs_new:Npn</code> .....	192, 200, 206, 211
<code>\cs_new_protected:Npn</code> .....	159
<code>\cs_set_eq:NN</code> .....	124, 125, 138
<code>\cs_set_protected:Npn</code> .....	89, 101, 112, 123
<b>D</b>	
<code>\def</code> .....	141,
	153, 154, 155, 156, 162, 174, 179, 183
<code>\definecolor</code> .....	3
<code>\directlua</code> .....	95, 106, 117
<code>\doclicenseLongTextForHyperref</code> ...	251
<code>\doclicenseURL</code> .....	252
<code>\DocumentMetadata</code> .....	1
<b>E</b>	
exp commands:	
<code>\exp_after:wN</code> .....	176
<b>F</b>	
file commands:	
<code>\file_if_exist:nTF</code> .....	43, 58
<b>H</b>	
<code>\hypersetup</code> .....	249
<b>L</b>	
<code>\let</code> .....	168, 170, 171, 172
lua commands:	
<code>\lua_load_module:n</code> .....	45
<code>\lua_now:n</code> .....	33
<code>\luaescapestring</code> .....	97, 108, 119
<b>M</b>	
msg commands:	
<code>\msg_info:nnn</code> .....	42, 132
<code>\msg_new:nnn</code> .....	20, 22, 24, 27



<code>\msg_warn:nnn</code> . . . . .	52		<b>R</b>
<code>\msg_warning:nnn</code> . . . . .	243, 244	<code>\relax</code> . . . . .	168
<code>\msg_warning:nnnn</code> . . . . .	223, 235	<code>\RequirePackage</code> . . . . .	61, 63, 76, 265
<b>P</b>			
pdf commands:		<b>S</b>	
<code>\pdf_object_ref:n</code> . . . . .	164, 165, 166	<code>\space</code> . . . . .	206
pdfmanagement commands:		str commands:	
<code>\pdfmanagement_add:nnn</code> . . . . .		<code>\str_if_eq:nnTF</code> . . . . .	51
. . . . .	199, 203, 208, 213	sys commands:	
<code>\pdfmanagement_if_active:TF</code> . . . . .	143	<code>\sys_if_engine luatex:TF</code> . . . . .	31
pdfmanagement internal commands:		<code>\sys_if_engine luatex_p:</code> . . . . .	80
<code>\g__pdfmanagement_firstaid_-</code>		<code>\sys_if_output_pdf_p:</code> . . . . .	82
<code>color_bool</code> . . . . .	66		
<code>\g__pdfmanagement_firstaid_-</code>		<b>T</b>	
<code>colorspace_bool</code> . . . . .	259	TeX and L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> commands:	
<code>\g__pdfmanagement_firstaid_-</code>		<code>\ifpackagelater</code> . . . . .	87, 221, 233
<code>fontspec_bool</code> . . . . .	40	<code>\declare@file@substitution</code> . . . . .	48, 68
<code>\g__pdfmanagement_firstaid_-</code>		<code>\disable@package@load</code> . . . . .	243, 244
<code>luacolor_bool</code> . . . . .	81	<code>\LuaCol@Attribute</code> . . . . .	93, 104, 115
<code>\g__pdfmanagement_firstaid_-</code>		<code>\LuaCol@setattribute</code> . . . . .	93, 104, 115
<code>output_bool</code> . . . . .	56	<code>\pgf@sys@addpdfresource@colorspaces@plain</code>	
<code>\g__pdfmanagement_firstaid_-</code>		. . . . .	154, 183
<code>pdflscape_bool</code> . . . . .	229	<code>\pgf@sys@addpdfresource@extgs@plain</code>	
<code>\g__pdfmanagement_firstaid_pgf_-</code>		. . . . .	153, 174
<code>bool</code> . . . . .	130	<code>\pgf@sys@addpdfresource@patterns@plain</code>	
<code>\g__pdfmanagement_firstaid_-</code>		. . . . .	155, 179
<code>transparent_bool</code> . . . . .	217	<code>\pgf@sys@pdf@check@resources</code> . . . . .	168
<code>\g__pdfmanagement_firstaid_-</code>		<code>\pgf@sys@pdf@possible@resources</code> . . . . .	
<code>xcolor_bool</code> . . . . .	70	. . . . .	157, 162
<code>\g__pdfmanagement_firstaid_xmp_-</code>		<code>\pgf@sys@pgf@resource@list@colorspaces</code>	
<code>bool</code> . . . . .	241	. . . . .	172
<code>\g__pdfmanagement_firstaidoff_-</code>		<code>\pgf@sys@pgf@resource@list@extgs</code> . . . . .	170
<code>clist</code> . . . . .	13	<code>\pgf@sys@pgf@resource@list@patterns</code>	
<code>\__pdfmanagement_patch_pgfcOLORSPACES:w</code>		. . . . .	171
. . . . .	185, 211	<code>\pgf@sys@setuppdfresources@plain</code> . . . . .	156
<code>\__pdfmanagement_patch_pgfextgs:w</code>		<code>\pgfutil@addpdfresource@colorspaces</code>	
. . . . .	177, 200	. . . . .	4, 154
<code>\__pdfmanagement_patch_pgfpatterns:w</code>		<code>\pgfutil@addpdfresource@extgs</code> . . . . .	4, 153
. . . . .	181, 206	<code>\pgfutil@addpdfresource@patterns</code>	
<code>\__pdfmanagement_pgf_sys_-</code>		. . . . .	5, 155
<code>setuppdfresources_plain:</code> . . . . .	145, 159	<code>\pgfutil@empty</code> . . . . .	170, 171, 172
<code>\__pdfmanagement_pgfori_pgfutil@setuppdfresources</code>		<code>\pgfutil@setuppdfresources</code> . . . . .	
. . . . .	139, 148	. . . . .	140, 141, 156, 189
<code>\__pdfmanagement_split_dict_-</code>		tl commands:	
<code>entry_aux:NNw</code> . . . . .	192	<code>\tl_set:Nn</code> . . . . .	91, 92, 103, 114, 194, 195, 202
<code>\ProvidesExplPackage</code> . . . . .	2	<code>\tl_trim_spaces:n</code> . . . . .	198, 204, 209, 214
		<code>\l_tmpa_tl</code> . . . . .	202, 204
<b>Q</b>			
quark commands:			
<code>\q_stop</code> . . . . .	177, 181, 185, 192, 200, 206, 211		