

First aid for external files and packages that need updating

Frank Mittelbach, L^AT_EX Project

December 16, 2020

Abstract

This file contains some first aid for packages or classes that require updates because of internal changes to L^AT_EX but that aren't yet reflected in the package/class code.

Contents

1	Introduction	1
1.1	Minor kernel fixes	2
2	The Implementation	2
2.1	The <code>filehook</code> package first aid	2
2.2	The <code> bidi </code> package first aid	4
2.3	The <code> dinbrief </code> class first aid	5
2.4	The <code> pgfpages </code> and <code> pgfmorepages </code> first aid	5
2.5	Temporary fixes for the kernel (until the next patch-level release)	6

1 Introduction

Over the years package writers have hooked into various parts of internal L^AT_EX commands (largely because proper interfaces were missing in important places) and if we are now gradually adding such interfaces these internal commands do change and as a result patching into them stops working.

As part of making such internal changes the L^AT_EX Project team attempts to check for such usage in packages, alert the package maintainers and ensures that the packages get updated alongside the core L^AT_EX system. However it is not always possible to get packages that will fail with a new kernel updated in time and if that is the case we try to provide a temporary fix in this file for them. Once the package gets updated the fix will then be removed again.

For that reason, it is put into a separate bundle so that we can update it easily without requiring the CTAN maintainers to install a new full LaTeX system just because we take out (or add) a fix for a package here.

In the best case scenario the file documented here should be empty. In practice it will probably always contain one or the other fix while we are waiting for the package to get updated.

Important notice: The fixes provided here are not meant to be a permanent solution, but are only provided to support the transition period. They are (usually) neither complete nor necessarily the best solution! Furthermore, as they are done from the “outside”, they usually add some burden and slow down L^AT_EX processing, even if the package/class is not used in the document.

We will therefore remove such code as soon as possible again. In practice this means that if some package never gets updated/corrected, then it will eventually fail to work, because after one or at most two L^AT_EX releases we will take out the transition code to ensure that this “first aid patching” doesn’t get out of bounds.

1.1 Minor kernel fixes

If we encounter issues with the kernel code that should get fixed before the next main release we normally generate a patch release for L^AT_EX. However, depending on the complexity of the fix we might first add the fix here and generate a full patch release only when a number of such issues have accumulated. This way we lessen the impact on CTAN maintainers because for each tach release we have to make and distribute also a matching development release.

2 The Implementation

This file is meant to be loaded during format generation which is why we give it the extension `.ltx`.

```

1 (*kernel)
2 \def\LaTeXFirstAidDate{2020/12/14}
3 \def\LaTeXFirstAidVersion{v1.0i}
4 \ProvidesFile{latex2e-first-aid-for-external-files.ltx}
5         [\LaTeXFirstAidDate\space \LaTeXFirstAidVersion\space
6         LaTeX kernel fixes to external files and packages]

```

`\FirstAidNeededT` This is a very simple help to ensure that we only apply first aid to an unmodified package or class. It only works in the case the file has already been loaded and the csname `\ver@#1.#2` got defined (holding the current date, version, and short description info). We then compare its content to a frozen string and make the modification `#3` only if both agree. If they differ we assume that the package/class in question got updated by its maintainer.

```

7 \ExplSyntaxOn
8 \cs_new:Npn\FirstAidNeededT#1#2#3{
9   \exp_args:Nc\str_if_eq:ont{\ver@#1.#2}{#3}
10 }
11 \ExplSyntaxOff
12 </kernel>

```

2.1 The filehook package first aid

The `filehook` package implements hooks into file loading commands. These days this is already provided by the kernel albeit not with the same user interface. Until

that package gets updated (to use the kernel interfaces) we provide a substitution. This does not offer all hooks from `filehook` but all that have been used in packages available in T_EX Live.

Note that this doesn't fix `currfile` because that package uses `filehook` but relies on the internals of the old implementation.

The package has now got an update so we aren't activating the first aid. However, at the moment it basically bypasses the new hook mechanism and puts the old hooks in thereby disabling, for example, the possibility to re-order code through rules.

We therefore keep `filehook-ltx.sty` around as a guideline for further updates.

Replacing `filehook` with a leaner version would then work like this:

```
13 <*kernel>
14 %\declare@file@substitution{filehook.sty}{filehook-ltx.sty}
15 </kernel>
```

What follows is a simplified (partial) implementation of the `filehook` interfaces. Not implemented are:

```
\AtBeginOfFiles      \AtEndOfFiles
\AtBeginOfInputs     \AtEndOfInputs
\AtBeginOfInputFile  \AtEndOfInputFile
```

```
16 <*filehook>
17 \newcommand\AtBeginOfEveryFile [1]
18   {\AddToHook{file/before}{#1}}
19 \newcommand\AtEndOfEveryFile [1]
20   {\AddToHook{file/after}{#1}}
21 \newcommand\AtBeginOfIncludes [1]
22   {\AddToHook{include/before}{#1}}
23 \newcommand\AtEndOfIncludes [1]
24   {\AddToHook{include/end}{#1}}
25 \newcommand\AfterIncludes [1]
26   {\AddToHook{include/after}{#1}}
27 \newcommand\AtBeginOfPackages [1]
28   {\AddToHook{package/before}{#1}}
29 \newcommand\AtEndOfPackages [1]
30   {\AddToHook{package/after}{#1}}
31 \newcommand\AtBeginOfClasses [1]
32   {\AddToHook{class/before}{#1}}
33 \newcommand\AtEndOfClasses [1]
34   {\AddToHook{class/after}{#1}}
35 \newcommand\AtBeginOfFile [2]
36   {\AddToHook{file/before/#1}{#2}}
37 \newcommand\AtEndOfFile [2]
38   {\AddToHook{file/after/#1}{#2}}
```

Some commands offered a starred form

```
39 \DeclareDocumentCommand \AtBeginOfPackageFile {smm}
40   {\IfBooleanTF{#1}%
41     {\@ifpackageloaded{#2}%
42       {#3}%
```

```

43     {\AddToHook{package/before/#2}{#3}}}%
44     {\AddToHook{package/before/#2}{#3}}}%
45   }
46 \DeclareDocumentCommand \AtEndOfPackageFile {smm}
47   {\IfBooleanTF{#1}%
48     {\@ifpackageloaded{#2}%
49       {#3}%
50       {\AddToHook{package/after/#2}{#3}}}%
51     {\AddToHook{package/after/#2}{#3}}}%
52   }

```

Are the * forms here of any use? I know they are use 3–4 times on CTAN but I wonder if those are real or mistaken usages.

```

53 \DeclareDocumentCommand \AtBeginOfClassFile {smm}
54   {\IfBooleanTF{#1}%
55     {\@ifclassloaded{#2}%
56       {#3}%
57       {\AddToHook{class/before/#2}{#3}}}%
58     {\AddToHook{class/before/#2}{#3}}}%
59   }
60 \DeclareDocumentCommand \AtEndOfClassFile {smm}
61   {\IfBooleanTF{#1}%
62     {\@ifclassloaded{#2}%
63       {#3}%
64       {\AddToHook{class/after/#2}{#3}}}%
65     {\AddToHook{class/after/#2}{#3}}}%
66   }
67 \newcommand\AtBeginOfIncludeFile [2]
68   {\AddToHook{include/before/#1}{#2}}
69 \newcommand\AtEndOfIncludeFile [2]
70   {\AddToHook{include/end/#1}{#2}}
71 \newcommand\AfterIncludeFile [2]
72   {\AddToHook{include/after/#1}{#2}}
73 </filehook>

```

2.2 The bidi package first aid

The `bidi` package adds a lot of hooks in various places and those added to `\document` and `\enddocument` are now no longer necessary as the kernel already provides the right hooks there.

However, we aren't trying to change that but instead only make sure that the existing patches still work by adding some first aid after `biditytools` has been loaded.

If the package gets updated one can easily take that out simply through

```
\RemoveFromHook{file/after/biditytools.sty}[firstaid]
```

This makes it easy to test new `bidi` code while the first aid code is still in the kernel.

```
74 (*kernel)
```

```

75 \AddToHook{file/after/biditools.sty}[firstaid]{%
76   \FirstAidNeededT{biditools}{sty}%
77   {2020/05/13 v2 Programming tools for bidi package}%
78   {

```

`bidi` adds some code to the beginning of `\document` which contains `\endgroup` and `\begingroup` which is no longer correct.

Patching `\document` using `\bidi@patchcmd` doesn't work so we take the extra groups out by hand:

```

79   \def\firstaid@bidi@document@patch
80     \endgroup#1\begingroup#2\firstaid@bidi@document@patch
81     {\unexpanded{#1#2}}%
82   \edef\document{\expandafter\firstaid@bidi@document@patch\document
83     \firstaid@bidi@document@patch}%

```

There are also some patches into `\enddocument`, some continue to go in but one fails, so we add that now into the right place.

```

84   \AddToHook{enddocument/info}%
85     {\let\bidi@AfterEndDocumentCheckLabelsRerun@\firstofone
86     \bidi@afterenddocumentchecklabelsrerunhook}%
87   }
88 }

```

2.3 The `dinbrief` class first aid

Again a case of a no longer correct `\endgroup` in `document`. Here the fix is simply though.

```

89 \AddToHook{file/after/dinbrief.cls}[firstaid]{%
90   \FirstAidNeededT{dinbrief}{cls}{2000/03/02 LaTeX2e class}%
91   {\AddToHook{env/document/begin}{\begingroup}}%
92 }

```

2.4 The `pgfpages` and `pgfmorepages` first aid

`pgfpages` alters the `\shipout` primitive to support multiple page up scenarios. If used together with `atbegshi` that worked because the alterations done by `atbegshi` came later and so used the new definition provide by `pgfpages`. Now that the code from `atbegshi` is already in the kernel this further redefinition doesn't happen with the result that the change to `\shipout` comes to late and breaks the kernel processes.

```

93 \ExplSyntaxOn
94 \AddToHook{file/after/pgfpages.sty}[firstaid]{%

```

Undo overwriting `\shipout`:

```

95   \cs_gset_eq:NN \shipout \pgfpages@originalshipout

```

Instead overwrite the L3 programming layer name of the primitive. This is really an absolute no-go, but for now the simplest solution to keep the original code running.

It will be replaced when the “configuration points” interface for L^AT_EX becomes available. At that point the package will be able to set up a different strategy for doing shipouts and without the need to overwrite a primitive (which it did in the past and which we do below) and then this code here can be taken out again.

```

96 \cs_set_eq:NN \pgfpages@originalshipout \tex_shipout:D
97 \cs_set_eq:NN \tex_shipout:D \pgfpages@interceptshipout
98 }
99 \ExplSyntaxOff

```

Same issue with `pgfmorepages` but slightly different implementation (sigh).

```

100 \ExplSyntaxOn
101 \AddToHook{file/after/pgfmorepages.sty}[firstaid]{
102   \cs_set_nopar:Npn \pgfhookintoshipout {
103     \cs_set_eq:NN \pgfpages@originalshipout \tex_shipout:D
104     \cs_set_eq:NN \tex_shipout:D \pgfpages@interceptshipout
105   }
106 }
107 \ExplSyntaxOff

```

2.5 Temporary fixes for the kernel (until the next patch-level release)

```
108 </kernel>
```

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; numbers in roman refer to the code lines where the entry is used.

	Symbols				
<code>\@firstofone</code> 85	<code>\AtEndOfClassFile</code>	. 60	E	
<code>\@ifclassloaded</code>	. 55, 62	<code>\AtEndOfEveryFile</code>	. 19	<code>\edef</code> 82
<code>\@ifpackageloaded</code>	41, 48	<code>\AtEndOfFile</code> 37	<code>\endgroup</code> 80
		<code>\AtEndOfIncludeFile</code>	69	<code>\exp</code> 9
A		<code>\AtEndOfIncludes</code>	.. 23	<code>\expandafter</code> 82
<code>\AddToHook</code>	18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 43, 44, 50, 51, 57, 58, 64, 65, 68, 70, 72, 75, 84, 89, 91, 94, 101	<code>\AtEndOfPackageFile</code>	46	<code>\ExplSyntaxOff</code>
<code>\AfterIncludeFile</code>	. 71	<code>\AtEndOfPackages</code>	.. 29	 11, 99, 107
<code>\AfterIncludes</code> 25			<code>\ExplSyntaxOn</code>	7, 93, 100
<code>\AtBeginOfClasses</code>	. 31	B			
<code>\AtBeginOfClassFile</code>	53	<code>\begingroup</code> 80, 91	F	
<code>\AtBeginOfEveryFile</code>	17	<code>\bidi@AfterEndDocumentCheckLabelsRun</code> 85	<code>\firstaid@bidi@document@patch</code> 79, 80, 82, 83
<code>\AtBeginOfFile</code> 35	<code>\bidi@afterenddocumentchecklabelsrerrunhook</code> 86	<code>\FirstAidNeededT</code>	..
<code>\AtBeginOfIncludeFile</code> 67			<code>\LaTeXFirstAidDate</code>	2, 5
<code>\AtBeginOfIncludes</code>	. 21	C		<code>\IfBooleanTF</code>
<code>\AtBeginOfPackageFile</code> 39	<code>\cs</code> 8, 95, 96, 97, 102, 103, 104	 40, 47, 54, 61
<code>\AtBeginOfPackages</code>	. 27			L	
<code>\AtEndOfClasses</code>	... 33	D		<code>\LaTeXFirstAidVersion</code> 3, 5
		<code>\declare@file@substitution</code> 14	<code>\let</code> 85
		<code>\DeclareDocumentCommand</code> 39, 46, 53, 60		
		<code>\def</code> 2, 3, 79	N	
		<code>\document</code> 82	<code>\newcommand</code> 17,

19, 21, 23, 25,	\pgfpages@originalshipout	T
27, 29, 31, 33, 95, 96, 103	
35, 37, 67, 69, 71	\ProvidesFile 4	\tex 96, 97, 103, 104
P		S
\pgfhookintoshipout 102	\shipout 95	U
\pgfpages@interceptshipout\space 5	
..... 97, 104	\str 9	\unexpanded 81