The auxhook package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/04/06 v1.1

Abstract

Package auxhook provides hooks for adding stuff at the begin of .aux files.

Contents

1 User interface

There are two kinds of .aux files, the main .aux file and the .aux file that belongs to an included file, specified by \include.

Some packages write macros in the auxiliary files. If the user stops using the package, these macros will usually cause error messages because of unknown commands. Prominent example is package babel's \select@language.

But such a package could be written more cooperative. It can also provide a definition in the auxiliary file (\providecommand) that silently disables the macros of the package if the package is no longer in use.

In case of the main auxiliary file, \AtBeginDocument can be used for this purpose. Especially if several packages are involved, the order cannot be controlled always (e.g., see package hypdestopt that hooks into hyperref's macros). And there isn't any hook for the auxiliary files of the \include feature.

Thus this package patches LaTeX's macros `\document` and `\@include` to add the hooks where the auxiliary files are opend and the first line with `\relax` is written.

The patching can fail, if these macros are redefined by some other package. If the other package still uses the original definition, then load package auxhook earlier. (With `\RequirePackage` the package also can be loaded before the class). If the redefinition doesn't use the original meaning, then you can try to load package auxhook afterwards, but you need luck that the patch succeeds.

The hooks are macros:

`\@beginmainauxhook`: Start of the main auxiliary file. The hook is called after the first line with `\relax` is written.

`\@beginpartauxhook`: The same for the auxiliary files that belongs to the files that are included by `\include`.

If you want to add something to these hooks, you can use `\g@addto@macro` from LaTeX's kernel. But the package provides macros to add code that adds a line to the auxiliary file:

`\AddLineBeginMainAux` {⟨*line*⟩}
`\AddLineBeginPartAux` {⟨*line*⟩}
`\AddLineBeginAux` {⟨*line*⟩}

The ⟨*line*⟩ is added at the begin of the main auxiliary file by `\AddLineBeginMainAux` and at the begin of the auxiliary files of included files by `\AddLineBeginPartAux`. `\AddLineBeginAux` writes in both kinds of auxiliary files.

Examples, see packages hypdestopt ([1]) and zref ([3]).

2 Implementation

2.1 Identification

```
1 ⟨*package⟩
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{auxhook}%
4   [2007/04/06 v1.1 Hooks for auxiliary files (HO)]%
```

2.2 Hook setup

`\@beginmainauxhook` The hook for the main auxiliary file, initially empty.

```
5 \providecommand*{\@beginmainauxhook}{}
```

`\@beginpartauxhook` The hook for auxiliary files of included files, initially empty.

```
6 \providecommand*{\@beginpartauxhook}{}
```

2.3 User macros

`\AddLineBeginMainAux`

```
7 \newcommand{\AddLineBeginMainAux}[1]{%
8   \g@addto@macro\@beginmainauxhook{%
9     \immediate\write\@mainaux{#1}%
10  }%
11 }
```

`\AtBeginPartAuxLine`

```
12 \newcommand{\AddLineBeginPartAux}[1]{%
13   \g@addto@macro\@beginpartauxhook{%
14     \immediate\write\@partaux{#1}%
```

```
  }%
}
```

\AddLineBeginAux

```
\newcommand{\AddLineBeginAux}[1]{%
  \AddLineBeginMainAux{#1}%
  \AddLineBeginPartAux{#1}%
}
```

2.4 Patches

2.4.1 \document

```
\begingroup
  \@ifundefined{beamer@origdocument}{%
    \def\auxhook@document{\document}%
  }{%
    \def\auxhook@document{\beamer@origdocument}%
  }%
  \long\def\y#1\immediate\write\@mainaux#2#3\auxhook@nil{%
    \toks@{%
      #1\immediate\write\@mainaux{#2}%
      \@beginmainauxhook
      #3%
    }%
    \expandafter\xdef\auxhook@document{\the\toks@}%
    \endgroup
  }%
  \long\def\x#1\immediate\write\@mainaux#2#3\auxhook@nil{%
    \toks@{#3}%
    \edef\x{\the\toks@}%
    \ifx\x\@empty
      \PackageWarningNoLine{auxhook}{%
        Cannot patch \expandafter\string\auxhook@document,%
        \MessageBreak
        using \string\AtBeginDocument\space instead%
      }%
      \endgroup
      \AtBeginDocument{%
        \if@filesw
          \@beginmainauxhook
        \fi
      }%
    \else
      \expandafter\expandafter\expandafter\y\auxhook@document
          \auxhook@nil
    \fi
  }%
\expandafter\expandafter\expandafter\x\auxhook@document
    \immediate\write\@mainaux{}\auxhook@nil
```

2.4.2 \@include

```
\begingroup
  \long\def\y#1\immediate\write\@partaux#2#3\auxhook@nil{%
    \endgroup
    \def\@include##1 {%
      #1\immediate\write\@partaux{#2}%
      \@beginpartauxhook
      #3%
    }%
  }%
  \long\def\x#1\immediate\write\@partaux#2#3\auxhook@nil{%
    \toks@{#3}%
```

```
    \edef\x{\the\toks@}%
    \ifx\x\@empty
      \PackageWarningNoLine{auxhook}{%
        Cannot patch \string\@include,\MessageBreak
        patch dropped%
      }%
      \endgroup
    \else
      \expandafter\y\@include{##1} \auxhook@nil
    \fi
  }%
\expandafter\x\@include{#1} \immediate\write\@partaux{}\auxhook@nil
```

⟨/package⟩

3 Installation

3.1 Download

Package. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/oberdiek/auxhook.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/auxhook.pdf Documentation.

Bundle. All the packages of the bundle 'oberdiek' are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/oberdiek.tds.zip

TDS refers to the standard "A Directory Structure for TEX Files" (CTAN:tds/tds.pdf). Directories with `texmf` in their name are usually organized this way.

3.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package attachfile2 comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

3.3 Package installation

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain-TEX:

```
tex auxhook.dtx
```

[1] ftp://ftp.ctan.org/tex-archive/

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
auxhook.sty → tex/latex/oberdiek/auxhook.sty
auxhook.pdf → doc/latex/oberdiek/auxhook.pdf
auxhook.dtx → source/latex/oberdiek/auxhook.dtx
```

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

3.4 Refresh file name databases

If your TEX distribution (teTEX, mikTEX, ...) relies on file name databases, you must refresh these. For example, teTEX users run `texhash` or `mktexlsr`.

3.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is pdftk, e.g. unpack the file into the current directory:

```
pdftk auxhook.pdf unpack_files output .
```

Unpacking with LATEX. The `.dtx` chooses its action depending on the format:

plain-TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{auxhook.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLATEX:

```
pdflatex auxhook.dtx
makeindex -s gind.ist auxhook.idx
pdflatex auxhook.dtx
makeindex -s gind.ist auxhook.idx
pdflatex auxhook.dtx
```

4 References

[1] Heiko Oberdiek: *The hypdestopt package*; 2006/05/30 v1.0; CTAN:macros/latex/contrib/oberdiek/hypdestopt.pdf.

[2] Sebastian Rahtz, Heiko Oberdiek: *The hyperref package*; 2006/08/16 v6.75c; CTAN:macros/latex/contrib/hyperref/.

[3] Heiko Oberdiek: *The zref package*; 2006/05/25 v1.2; CTAN:macros/latex/contrib/oberdiek/zref.pdf.

5 History

[2006/05/31 v1.0]

- First version.

[2007/04/06 v1.1]

- Fix for class beamer.

6 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.