The footnotehyper package

Jean-François Burnol
jfbu (at) free (dot) fr

Package version: v1.1e (2021/08/13)
From source file footnotehyper.dtx of Time-stamp: <13-08-2021 at 11:54:00 CEST>.

Abstract

The footnote package by Mark Wooding (1997/01/28 1.13) allows to gather footnotes (\begin{savenotes}) and later insert them (after \end{savenotes}) at the bottom of the page, even if the intervening material consists of tabulars, minipages or framed contents for example. One can also use the \savenotes/\spewnotes syntax.

Also, footnote.sty provides a footnote environment which allows to insert verbatim material.

Earlier releases of the present footnotehyper package added patches for hyperref compatibility and some bugfixes, addressing in particular the incompatibility with color/xcolor, and with babel-frenchb, and also fixing the footnote environment with optional argument [NUM]. Since v0.99 all macros are defined internally and the footnote package is not loaded at all.

The same user interface is kept. Since v1.0 it is possible to use footnotehyper also in absence of hyperref or when the latter is loaded with its hyperfootnotes=false option. The order of loading of footnotehyper and hyperref is inconsequential.

1 License

2 Changes

v0.9c (2016/04/19) First release: adapt original package to be hyperref and color/xcolor compatible.

v0.9e (2016/04/30) Abort in absence of hyperref. Compatibility with babel-french.

v0.99 (2017/02/16) Do not load package footnote.sty[1] anymore.

From then on **footnotehyper** is incompatible with it at it uses the same user interface.

v1.0 (2017/03/07) Be usable also in absence of hyperref or when the latter was passed hyperfootnotes=false option.

v1.1 (2018/01/23) Fix bug which arose when savenotes environment was used *inside* a minipage: footnotes were disappearing![2] See related remarks at end of section 4.

v1.1a (2019/11/07) Abort under beamer (difficulty with \@makefntext and suspicion beamer does not need **footnotehyper**).

v1.1b (2021/01/26) Fix incompatibility with the combination memoir + babel-french.

v1.1c (2021/01/29) Fix legacy bug of original package interfering with LaTeX2e mechanism to suppress indentation after mid-paragraph lists (when savenotes environment directly wraps the enclosed list environment, mid-paragraph).

v1.1d (2021/02/04) Fix regression at v1.1b which caused a build crash whenever **footnotehyper** decided to raise a warning relative to \@makefntext.[3]

Refactor analysis of \@makefntext for simpler and better support of babel-french.[4] Better support contexts such as presence of package cleveref.

Add \iffootnotehyperparse and \iffootnotehyperwarn booleans.

v1.1e (2021/08/13) Use LaTeX environment hooks if available for \makesavenoteenv, in replacement of the original footnote package code.

3 Usage

The package provides:

- a savenotes environment which re-routes footnotes and delivers them at the end (there is also the \savenotes/\spewnotes syntax; which does create a group like the environment),

[1] http://ctan.org/pkg/footnote

[2] Thanks to François Pantigny for reporting the bug. A later suggestion of the same is to let the package do nothing under Beamer class, and this is what v1.1a 2019/11/07 does.

[3] Thanks to Leon Kiefer for reporting the bug.

[4] Only basic context has been tested with babel-french: standard classes, KOMA-script, memoir. Extra packages may make the **footnotehyper** environments cause breakage.

- footnote and footnotetext environments to allow footnotes with verbatim material,
- a macro to patch environments to let them apply the savenotes mechanism automatically.

The preliminary construction by the package of the footnote and footnotetext environments goes via an automated analysis of the LaTeX macro \@makefntext, as possibly customized by classes and/or packages. This is a rather fragile step, and the next section discusses problems which may arise.

3.1 Potential difficulties with the footnote and footnotetext environments

What is discussed here only affects the *environments* footnote and footnotetext not the macros \footnote and \footnotetext.

footnotehyper inherits from footnote original package the aim to convert \@makefntext into two parts, the first one to be inserted at the start of a footnote in environment form, the second one (usually empty) at its end.[5] It thus hopes that the replacement text of \@makefntext contains only once its parameter token #1, and that it is used there unbraced. This is the case with the article class.

Known bug (may be promoted to feature at some point): the analysis is done only once at begin document, whereas the article.cls's redefines \@makefntext during execution of \maketitle. However, it does not look really urgent to support at all costs usage of the environment footnote in the \author etc... data which contributes to the \maketitle expansion.[6]

Some seemingly innocent redefinitions such as the one of beamer which was last time I checked (that was in 2019):

```
macro:#1->\def \insertfootnotetext {#1}\def \insertfootnotemark {\@makefnmark }
\usebeamertemplate ***{footnote}
```

are not (easily) compatible with environment forms for footnotes allowing verbatim material, as they require fetching the footnote contents.[7]

In case of such a problematic \@makefntext **footnotehyper** raises a warning, to explain that footnotes typeset using the environment forms will use a fall-back layout (inherited from the article document class). Footnotes using \footnote are not impacted by this.

[5] LaTeX inserts some stuff before and after the footnote text, even before handing it over as argument to \@makefntext. These tokens are currently hardcoded into the **footnotehyper** environments for footnotes.

[6] TeXperts note: \def\FNH@prefntext{\@makefntext{}} would make the footnote environment dynamically adjust to circumstances, when \@makefntext only adds some prefix and no postfix. In fact, this is (in a more complicated form for compatibility with KOMA-script and to obey the FBFrenchFootnotes setting and the additional extra stuff inserted by babel-french before and after) basically what is done by **footnotehyper** to handle babel-french.

As it may cause instability if extra packages fiddle with \@makefntext, or \@makefntext is radically re-defined in some environments provided by the class, **footnotehyper** does not use this when its begin document analysis concluded the argument was used unbraced and at last position in replacement text of \@makefntext, but it freezes the found prefix. However, when it is concluded that probably \@makefntext has been redefined in an <extra tokens>\old@makefntext way (for example, this is the case with cleveref package), then the \def\FNH@prefntext{\@makefntext{}} approach is taken, despite the risks inherent to it.

You can provide your own custom definitions for \FNH@prefntext and \FNH@postfntext. Then add to the preamble \footnotehyperparsefalse.

[7] Since v1.1a, **footnotehyper** simply aborts under beamer class.

Also **footnotehyper** emits some info message if \@makefntext was not as simple as expected but nevertheless there is some hope that the footnote and footnotetext environments will be fully functional. This is currently the case in presence of package cleveref (see the TeXperts footnote 6).

You can turn off these messages by adding \footnotehyperwarnfalse to the document preamble.

3.2 Other potential or actual limitations

It should be recalled that in case of \footnotemark[N] and \footnotetext[N]{...} mark-up hyperref creates no hyperlink. This is not changed by **footnotehyper** and applies also to the \begin{footnotetext}[N] case. Without optional argument the link is created, and the link is created also for \footnote[N] or \begin{footnote}[N].

This package does not handle especially floating environments, except that one can always surround them in the source in a savenotes environment and one knows that the footnotes will be delivered at the \end{savenotes}... which may well be one page earlier than the actual location of the floating material in the produced document !

Environments typesetting multiple times their contents are the most hostile to footnotes. Currently, **footnotehyper** only handles especially the amsmath environments (as in footnote.sty.)

3.3 The \makesavenoteenv macro

Finally there is a \makesavenoteenv macro which takes as argument an environment name and patches it to do the \savenotes/\spewnotes automatically.

The syntax is either \makesavenoteenv{foo} which patches environment foo (since 1.1e, via the hooks provided by LaTeX since October 2020) to do automatically \savenotes/\spewnotes, or \makesavenoteenv[bar]{foo} which defines environment bar as foo inside a savenotes environment.

With LaTeX earlier than October 2020, the macro is the same as in the original footnote package and proceeds in a more brutal way than what is described in previous paragraph. It is safer to avoid it, as one never knows what happens with such patches: for example the [H] specifier provided by the float package overwrites the \end{table} definition during the execution of \begin{table}...![8] As another example, \makesavenoteenv{tcolorbox} with the original footnote package code breaks, but the new version activated with LaTeX from October 2020 or later appears to work.

3.4 Example of output, and of input

Inside[9] a	tabular[10]

[8] By the way I have not checked if this float package feature behaves nicely, or has been updated to be compatible, with the LaTeX hook mechanism of October 2020.

Here is an illustrative example of usage of the `savenotes` environment:

```
\begin{savenotes}
\begin{framed}
  Please refer to the documentation of the |footnote| package.%
  \footnote{\url{http://ctan.org/pkg/footnote}}

  Particularly you may check its |savenotes| environment.%
  \footnote{% here is how to add anchor for hyperlink target:
  \phantomsection\label{fn:floats}% (this % to avoid space at start of paragraph)
  It doesn't bring any
  feature to especially handle the issues related to footnotes in floating
  environments, though.}
\end{framed}
\end{savenotes}
Here is a link to an interesting footnote: \ref{fn:floats}.
```

and the present frame has `\footnote`'s from inside a `tabular` and is inside a `savenotes` environment.[11] Let's test an `amsmath` environment with `\intertext`. As

$$E = mc^2 \ , \tag{1}$$

was too easy[12], let's try:

$$a^n + b^n = c^n \ . \tag{2}$$

And a footnote with some verbatim material[13].

The input for the footnote 13 was coded as:

```
And a footnote with some verbatim material%
\begin{footnote}
  The footnote environment allows verbatim contents: \verb|&$^%\[}$|
\end{footnote}.
```

[9] If the frame extends to the next page, the end of the `savenotes` environment delivers its intercepted footnotes only there.

[10] Alternatively a `\savenotes`/`\spewnotes` pair could have been used.

[11] Here is an issue which has nothing (as I finally figured out) to do with `footnote`, and only indirectly with LaTeX: if you embed a *full-width* minipage (with initial `\noindent`) in any environment not doing `\ignorespacesafterend`, be careful to add a % either immediately after the `\end{minipage}` (or a `\relax` or a `\par`) or after the surrounding environment `\end{foo}` or use `\end{minipage}\end{foo}` else the output may have an extra blank line if the source has a blank line after the `foo` environment. Here is a typical example, with a `tabular` rather:

Now some use of `\footnotemark` followed by a `footnotetext` (here is the mark: [14]) environment. And use of `\footnotemark[99]` in association with a `footnotetext` environment using the same optional argument `[99]` (here is the mark: [99], and you can see it is not an hyperlink). And a final footnote, done with `\begin{footnote}[57]`[57]. There is no problem with the hyperlink, then.[15]

4 Notes

A few items worth of mention:

- the `footnote` package patches the LaTeX kernel `\parbox`. **footnotehyper** doesn't (but the code can be found commented-out at the end of the present file).
- the `footnote` package defines a `minipage*` environment which is `minipage` patched by `\makesavenoteenv`, **footnotehyper** doesn't.
- the `footnote` environment from `footnote.sty` does a `\leavevmode\unskip` which **footnotehyper** doesn't: hence if one locates `\begin{footnote}` at start of a line in the LaTeX source, one will typically need a % at end of text on previous line to avoid the end-of-line space.
- the `hyperref` package inserts no hyperlink in case of `\footnotemark[N]/\footnotetext[N]`. This is not modified by **footnotehyper**.
- side-note: there is an interference between `hyperref` and `frenchb` regarding the footnote marker when using the syntax `\footnotemark[NUM]`. For the record here is a patch (last tested briefly with `hyperref 2016/06/24 v6.83q` and `frenchb 2017/01/30 v3.2g`):

```
\AtBeginDocument{%
```

```
\newenvironment{foo}{}{}

\noindent\begin{tabular}{p{\dimexpr\linewidth-2\tabcolsep\relax}}
  A\dotfill B
\end{tabular}

C

\begin{foo}
\noindent\begin{tabular}{p{\dimexpr\linewidth-2\tabcolsep\relax}}
  A\dotfill B
\end{tabular}
\end{foo}

C
```

If you try it out you will see an extra blank line in PDF output above the second C. Starting with `v0.99` the `\end{savenotes}` emits an `\ignorespacesafterend` which avoids this generic TeX/LaTeX problem. For good measure there is now an `\ignorespaces` in `\begin{savenotes}`.

[12] There is also $E = h\nu$.

[13] The footnote environment allows verbatim contents: `&$^%\[}$`

[14] Notice that the hyperlinking works for `\footnotemark` associated to the environment `footnotetext`.

[99] `hyperref` creates no hyperlink in this case, or in the `\footnotemark[N]/\footnotetext[N]{<foo>}` case. It does when the `[N]` is absent or when it is used with a `\footnote` command (or a `footnote` environment.)

[57] **footnotehyper** works since `v1.0` also in absence of `hyperref` or when the latter was passed `hyperfootnotes=false` option.

[15] Oh, and don't forget to read this interesting footnote: 11 (just in case you skipped on first reading).

```
   \let\@xfootnotemarkORIFB \@xfootnotemark
   \def\@xfootnotemarkFB {\leavevmode\unskip\unkern\,\@xfootnotemarkORIFB }%
   \ifHy@hyperfootnotes\ifFBAutoSpaceFootnotes
       \let\@xfootnotemark\@xfootnotemarkFB
   \fi\fi
  }%
```

On 2021/01/29 the interference (lost of some babel-french customization) is still there, as I checked now. This has nothing to do with **footnotehyper**.

- some environments typeset multiple times their contents, which causes issues; **footnotehyper** takes provisions only to handle the amsmath measuring step.
- LaTeX2e has some "features" when using footnotes in minipage's which are themselves in a minipage which may also have footnotes externally to the internal minipages... try it out with some \fboxes around the sub-minipages, to see.

  footnotehyper behaves like original package footnote when the savenotes environment is used *inside* a minipage. Only reasonable usage in case of nested minipages seems to use only a single top level (i.e. external) savenotes environment. But there will anyhow be collisions of the alphabetic enumerations. These collisions are there with or without **footnotehyper** (or footnote.sty.) I did not make any attempt, nor intend to in future, to address in an automatized manner these problematic contexts.

5 Implementation

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{footnotehyper}%
 [2021/08/13 v1.1e hyperref aware footnote.sty (JFB)]
```

`no options` The package has no options. I am too lazy.

```
\newif\iffootnotehyperparse\footnotehyperparsetrue
\newif\iffootnotehyperwarn \footnotehyperwarntrue
\def\FNH@msgbk{^^J(footnotehyper)\@spaces}% make room for message lines
\DeclareOption*%
    {\PackageWarning{footnotehyper}{Option `\CurrentOption' is unknown}}%
\ProcessOptions\relax
```

v1.1a lets the package abort under Beamer class and warn user.

```
\@ifclassloaded{beamer}
    {\PackageWarningNoLine{footnotehyper}{This package is
        incompatible with the beamer class. Aborting input..}%
     \endinput}
    {}%
```

Versions up to v0.9f loaded footnote.sty, with lots of patching afterwards. Starting with v0.99, **footnotehyper** does everything by itself with FNH@ prefix. Brief overview of some of the fixed issues:

- there was incompatibility with hyperref,
- and with color,
- if the \@makefntext at the time of loading of footnote.sty does not have its argument visible at top level in its meaning, or is used multiple times there, then the footnote environment will lead to low level TEX error,
- footnote.sty modifies \parbox,
- footnote.sty does some too early \let,
- the footnote environment from footnote.sty does not work if used with optional argument [N].

Starting with v1.0, **footnotehyper** may be used also in absence of hyperref.

```
\newbox\FNH@notes
\newdimen\FNH@width
\newtoks\FNH@toks % 1.1c
\let\FNH@colwidth\columnwidth
\newif\ifFNH@savingnotes
\AtBeginDocument {%
    \let\FNH@latex@footnote    \footnote
    \let\FNH@latex@footnotetext\footnotetext
    \let\FNH@H@@footnotetext   \@footnotetext
    \let\FNH@H@@mpfootnotetext \@mpfootnotetext
    \newenvironment{savenotes}
        {\FNH@savenotes\ignorespaces}{\FNH@spewnotes\ignorespacesafterend}%
    \let\spewnotes      \FNH@spewnotes
    \let\footnote       \FNH@footnote
    \let\footnotetext   \FNH@footnotetext
    \let\endfootnote    \FNH@endfntext
    \let\endfootnotetext\FNH@endfntext
    \@ifpackageloaded{hyperref}
     {\ifHy@hyperfootnotes
         \let\FNH@H@@footnotetext\H@@footnotetext
         \let\FNH@H@@mpfootnotetext\H@@mpfootnotetext
      \else
         \let\FNH@hyper@fntext\FNH@nohyp@fntext
      \fi}%
     {\let\FNH@hyper@fntext\FNH@nohyp@fntext}%
}%
```

\FNH@hyper@fntext
\FNH@nohyp@fntext
\FNH@fntext

These are the **footnotehyper** replacement for `\@footnotetext` inside the `savenotes` environment. There is a version creating an hyperlink and another one not creating an hyperlink. The `\FNH@fntext` macro serves as general dispatch. This may be a place to customize if one wants to handle environments doing multiple passes: but the footnote counter must have been taken care of elsewhere. The code currently handles only the case of `amsmath` environments.

```
\def\FNH@hyper@fntext{\FNH@fntext\FNH@hyper@fntext@i}%
\def\FNH@nohyp@fntext{\FNH@fntext\FNH@nohyp@fntext@i}%
\def\FNH@fntext #1{\ifx\ifmeasuring@\@undefined
    \expandafter\@secondoftwo\else\expandafter\@firstofone\fi
    {\ifmeasuring@\expandafter\@gobbletwo\fi}#1%
}%
```

\FNH@hyper@fntext@i

We do the `\ifHy@nesting` test although hyperref's manual says "Allows links to be nested; no drivers currently support this."

```
\long\def\FNH@hyper@fntext@i#1{%
  \global\setbox\FNH@notes\vbox
  {\unvbox\FNH@notes
   \FNH@startnote
   \@makefntext
    {\rule\z@\footnotesep\ignorespaces
     \ifHy@nesting\expandafter\ltx@firstoftwo
             \else\expandafter\ltx@secondoftwo
     \fi
     {\expandafter\hyper@@anchor\expandafter{\Hy@footnote@currentHref}{#1}}%
     {\Hy@raisedlink
       {\expandafter\hyper@@anchor\expandafter{\Hy@footnote@currentHref}%
       {\relax}}%
      \let\@currentHref\Hy@footnote@currentHref
      \let\@currentlabelname\@empty
      #1}%
     \@finalstrut\strutbox
    }%
   \FNH@endnote
  }%
}%
```

\FNH@nohyp@fntext@i

The original `\fn@fntext` had no `\long`.

```
\long\def\FNH@nohyp@fntext@i#1{%
  \global\setbox\FNH@notes\vbox
  {\unvbox\FNH@notes
   \FNH@startnote
   \@makefntext{\rule\z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
   \FNH@endnote
  }%
}%
```

\FNH@startnote Same as original (the code comment is kept from original.)

```
\def\FNH@startnote{%
  \hsize\FNH@colwidth
  \interlinepenalty\interfootnotelinepenalty
  \reset@font\footnotesize
  \floatingpenalty\@MM% Is this right???
  \@parboxrestore
  \protected@edef\@currentlabel{\csname p@\@mpfn\endcsname\@thefnmark}%
  \color@begingroup
}%
```

\FNH@endnote Fixed from original.

```
\def\FNH@endnote{\color@endgroup}%
```

\FNH@savenotes Same as original apart from using hyperref-aware \FNH@hyper@fntext, and taking into account hyperref's custom \@xfootnotenext. This was missed by v0.9f hence \footnotetext[N]{..} did not work inside savenotes environment. Fixed for v0.99.

Maybe I should change the way \@minipagerestore is handled.

```
\def\FNH@savenotes{%
  \begingroup
  \ifFNH@savingnotes\else
    \FNH@savingnotestrue
    \let\@footnotetext    \FNH@hyper@fntext
    \let\@mpfootnotetext  \FNH@hyper@fntext
    \let\H@@mpfootnotetext\FNH@nohyp@fntext % fool hyperref's \@xfootnotenext
    \FNH@width\columnwidth
    \let\FNH@colwidth\FNH@width
    \global\setbox\FNH@notes\box\voidb@x
    \let\FNH@thempfn\thempfn
    \let\FNH@mpfn\@mpfn
    \ifx\@minipagerestore\relax\let\@minipagerestore\@empty\fi
    \expandafter\def\expandafter\@minipagerestore\expandafter{%
      \@minipagerestore
      \let\thempfn\FNH@thempfn
      \let\@mpfn\FNH@mpfn
    }%
  \fi
}%
```

\FNH@spewnotes This uses \FNH@H@@footnotetext which is the \H@@footnotetext hyperref's preserved original meaning of \@footnotetext not creating a link target.

v1.1 fixes the bug about disappearing footnotes if savenotes environment is used inside a minipage. I had never really considered such usage, hence missed realizing there was a bug.

v1.1c 2021/01/29 fixes a legacy bug from footnote package: if used to enclose a list environment inside a paragraph, it broke the mechanism which suppresses indentation following the list.

Now, situation would be far simpler here if we did not have this extra \begingroup \endgroup pair in \FNH@savenotes/\FNH@spewnotes.

A priori, as far as I understand, testing the \if@endpe flag should be enough, but let's be extra cautious and check that \par is not \@@par. Attention here that this is not necessarily followed by \end{savenotes} and we have to support the \savenotes/\spewnotes syntax. The complication is added from it creating a group without being a genuine LaTeX2e environment.

```
\def\FNH@spewnotes {%
  \if@endpe\ifx\par\@@par\FNH@toks{}\else
     \FNH@toks\expandafter{\expandafter
              \def\expandafter\par\expandafter{\par}\@endpetrue}%
     \expandafter\expandafter\expandafter
     \FNH@toks
     \expandafter\expandafter\expandafter
     {\expandafter\the\expandafter\FNH@toks
      \expandafter\def\expandafter\@par\expandafter{\@par}}%
     \expandafter\expandafter\expandafter
     \FNH@toks
     \expandafter\expandafter\expandafter
     {\expandafter\the\expandafter\FNH@toks
      \expandafter\everypar\expandafter{\the\everypar}}\fi
  \else\FNH@toks{}\fi
  \expandafter
  \endgroup\the\FNH@toks
  \ifFNH@savingnotes\else
   \ifvoid\FNH@notes\else
    \begingroup
     \let\@makefntext\@empty
     \let\@finalstrut\@gobble
     \let\rule\@gobbletwo
     \ifx\@footnotetext\@mpfootnotetext
        \expandafter\FNH@H@@mpfootnotetext
     \else
        \expandafter\FNH@H@@footnotetext
     \fi{\unvbox\FNH@notes}%
    \endgroup
   \fi
  \fi
}%
```

\FNH@footnote
\FNH@footnotetext

We now take care of footnote.sty's footnote environment. The original \fn@endfntext is lacking a \fn@endnote, and this meant that footnote.sty was incompatible with color/xcolor packages. Also this \fn@endnote was \let to \color@endgroup which is wrong.

Furthermore, independently of presence of the \color/xcolor issue, the footnote.sty's footnote environment raised an error if used with an optional argument. v0.9f addresses this issue.

The footnotetext environment adds a complication, in case of optional argument we should not try to set up a link due to the fact that hyperref does not support it for the \footnotemark[N]/\footnotetext[N] syntax. And we need to make sure that the footnote and footnotetext environments obey the \savenotes/\spewnotes mechanism.

To handle all of this we code things completely differently from footnote.sty.

The v0.9f \begin{footnotetext}[N] inside savenotes tried to create an hyperref target. Fixed for v0.99.

Note: the footnote.sty code did a \leavevmode\unskip at entrance of footnote environment, which **footnotehyper** has not kept.

```
\def\FNH@footnote@envname    {footnote}%
\def\FNH@footnotetext@envname{footnotetext}%
\def\FNH@footnote{%
    \ifx\@currenvir\FNH@footnote@envname
        \expandafter\FNH@footnoteenv
    \else
        \expandafter\FNH@latex@footnote
    \fi
}%
\def\FNH@footnoteenv{%
    \@ifnextchar[%
      \FNH@footnoteenv@i %]
      {\stepcounter\@mpfn
       \protected@xdef\@thefnmark{\thempfn}%
       \@footnotemark
       \def\FNH@endfntext@fntext{\@footnotetext}%
       \FNH@startfntext}%
}%
\def\FNH@footnoteenv@i[#1]{%
    \begingroup
     \csname c@\@mpfn\endcsname #1\relax
     \unrestored@protected@xdef\@thefnmark{\thempfn}%
    \endgroup
    \@footnotemark
    \def\FNH@endfntext@fntext{\@footnotetext}%
    \FNH@startfntext
}%
\def\FNH@footnotetext{%
    \ifx\@currenvir\FNH@footnotetext@envname
        \expandafter\FNH@footnotetextenv
    \else
        \expandafter\FNH@latex@footnotetext
    \fi
}%
\def\FNH@footnotetextenv{%
    \@ifnextchar[%
      \FNH@footnotetextenv@i %]
      {\protected@xdef\@thefnmark{\thempfn}%
       \def\FNH@endfntext@fntext{\@footnotetext}%
       \FNH@startfntext}%
}%
\def\FNH@footnotetextenv@i[#1]{%
    \begingroup
     \csname c@\@mpfn\endcsname #1\relax
     \unrestored@protected@xdef\@thefnmark{\thempfn}%
    \endgroup
    \ifFNH@savingnotes
      \def\FNH@endfntext@fntext{\FNH@nohyp@fntext}%
    \else
      \def\FNH@endfntext@fntext{\FNH@H@@footnotetext}%
    \fi
    \FNH@startfntext
}%
```

\FNH@startfntext
\FNH@endfntext
\FNH@endfntext@fntext

This is used for the environmental form of the footnote environments. The use of \box\z@ originates in footnote.sty, should I change that ?

Both of \endfootnote and \endfootnotetext are aliases for \FNH@endfntext.

The \FNH@endfntext@fntext may be \@footnotetext (which will be \FNH@hyper@fntext in savenotes environment), or \FNH@H@@footnotetext, or \FNH@nohyp@fntext if in savenotes scope.

```
\def\FNH@startfntext{%
  \setbox\z@\vbox\bgroup
    \FNH@startnote
    \FNH@prefntext
    \rule\z@\footnotesep\ignorespaces
}%
\def\FNH@endfntext {%
    \@finalstrut\strutbox
    \FNH@postfntext
    \FNH@endnote
    \egroup
  \begingroup
    \let\@makefntext\@empty\let\@finalstrut\@gobble\let\rule\@gobbletwo
    \FNH@endfntext@fntext {\unvbox\z@}%
  \endgroup
}%
```

\@makefntext
\FNH@prefntext
\FNH@postfntext
\FNH@check

The definitions of \FNH@prefntext and \FNH@postfntext (which are needed for the footnote environment, \FNH@startfntext and \FNH@endfntext) are extracted from a somewhat daring analysis of \@makefntext. Contrarily to footnote.sty's original code (which may result in low level TEX errors when the footnote environment is executed) the method here will alert the user if the argument of \@makefntext is not visible at top level in its meaning or is used there multiple times. We also insert here some code to handle especially the case of babel-frenchb.

Refactoring at v1.1d. This will make **footnotehyper** compatible with cleveref for example, if nothing else interferes. Not all combinations of classes and packages can be handled and we can not hardcode a pre-analysis for all possible cases. Of course, one can not expect **footnotehyper** to be compatible with other footnote dedicated packages, but at best only with slight modifications of LATEX's defaults. At v1.1d the babel-french context is handled especially (to support it better with KOMAScript classes and simplify handling the memoir situation); there was no real other way than hardcode it more or less, but it can possibly break in presence of additional footnote packages.

Also the \iffootnotehyperparse and \iffootnotehyperwarn booleans were added.

Provide at least some definitions for \FNH@prefntext and \FNH@postfntext in case of \footnotehyperparsefalse in preamble.

```
\let\FNH@prefntext\@empty
\let\FNH@postfntext\@empty
\AtBeginDocument{\iffootnotehyperparse\expandafter\FNH@check\fi}%
```

As \ifFBFrenchFootnotes is not a TEX boolean if babel-frenchb isn't loaded, we have to work around this for \if..\fi pairs.

v1.1d fixes a v1.1b bug: any situation in \FNH@check@a causing the \FNH@bad@makefntext@alert branch to be chosen crashed the build due precisely to this problem with \ifFBFrenchFootnotes status. I had taken precautions for the \else branch but not for the "warning" branch.

So let's fix this, and do it in such a way (with \FNH@safeif) that the TEX \if..\fi balancing count does not perturbate enclosing the package loading in a TEX conditional. Why I am bothering, I don't know.

Finally, I refactored entirely the way frenchb context is handled, (using multiple times \FNH@safeif although now only for the artistic aim of balanced conditionals, as all frenchb-related stuff being in their dedicated macro, direct usage of \ifFB... is possible).

As long as nothing else interferes babel-french should be ok with standard classes, KOMA and memoir.

```
\def\FNH@safeif#1{%
   \iftrue\csname if#1\endcsname\csname fi\endcsname\expandafter\@firstoftwo
   \else\csname fi\endcsname\expandafter\@secondoftwo
   \fi
}%
\def\FNH@check{%
   \ifx\@makefntextFB\@undefined\expandafter\FNH@check@
                           \else\expandafter\FNH@frenchb@
   \fi
}%
\def\FNH@frenchb@{%
   \def\FNH@prefntext{%
     \localleftbox{}%
     \let\FBeverypar@save\FBeverypar@quote
     \let\FBeverypar@quote\relax
     \FNH@safeif{FB@koma}%
       {\FNH@safeif{FBFrenchFootnotes}%
          {\ifx\footnote\thanks
             \let\@@makefnmark\@@makefnmarkTH
             \@makefntextTH{} % space as in french.ldf
           \else
             \let\@@makefnmark\@@makefnmarkFB
             \@makefntextFB{} % space as in french.ldf
             \fi
          }{\let\@@makefnmark\@@makefnmarkORI
             \@makefntextORI{}% no space as in french.ldf
          }%
       }%
       {\FNH@safeif{FBFrenchFootnotes}%
          {\@makefntextFB{}}%
          {\@makefntextORI{}}%
       }%
   }%
   \def\FNH@postfntext{%
     \let\FBeverypar@quote\FBeverypar@save
     \localleftbox{\FBeveryline@quote}%
   }%
   \iffootnotehyperwarn
    \PackageInfo{footnotehyper}%
      {babel-french compatibility patch activated.\FNH@msgbk
       It may not work with all combinations of classes/packages.\FNH@msgbk
       In case of problems with the `footnote' or `footnotetext'\FNH@msgbk
       environments, you may try sending to the author a small\FNH@msgbk
       file demonstrating the problem.\FNH@msgbk
       To turn off this message, add \string\footnotehyperwarnfalse\FNH@msgbk
       to the preamble\@gobble}%
   \fi
}%
```

“Daring analysis” is un understatement. At v1.1b we added a dangerous twist to fix a memoir + frenchb triggered issue: if the \@makefntext, as in memoir + frenchb situation, uses \def syntax to define a macro with parameter we had a problem with the # token not being doubled in the replacement fetched by \FNH@check@a. As expedient work-around we fixed this by adding a \scantokens wrapper.

At v1.1d I refactored the babel-french situation, moving it to an entirely dedicated \FNH@frenchb@, and dropped the v1.1b usage of \scantokens.

v1.1d adds \FNH@checkagain@ which will identify circumstances likely to be safe for the approach via \def\FNH@prefntext{\@makefntext{}}. For example this is what will happen with cleveref (if not modified by other packages).

```
\def\FNH@check@{%
    \expandafter\FNH@check@a\@makefntext{1.2!3?4,}%
                \FNH@@@1.2!3?4,\FNH@@@\relax
}%
\long\def\FNH@check@a #11.2!3?4,#2\FNH@@@#3{%
    \ifx\relax#3\expandafter\FNH@checkagain@
    \else
      \def\FNH@prefntext{#1}\def\FNH@postfntext{#2}%
      \expandafter\FNH@check@b
    \fi
}%
```

The argument was not seen unbraced at top. Maybe it is not fetched, or it was but was left at the end, braced. If this is the case we use the fallback \def\FNH@prefntext{\@makefntext{}}, which may work.

```
\def\FNH@checkagain@{%
    \expandafter\FNH@checkagain@a
    \detokenize\expandafter{\@makefntext{1.2!3?4,}}\relax\FNH@@@
}%
\edef\FNH@temp{\noexpand\FNH@checkagain@a ##1\string{1.2!3?4,\string}}%
\expandafter\def\FNH@temp#2#3\FNH@@@{%
    \ifx\relax#2%
      \def\FNH@prefntext{\@makefntext{}}%
      \iffootnotehyperwarn
        \PackageInfo{footnotehyper}%
        {Using the \string\@makefntext{} approach (see doc).\FNH@msgbk
         In case of dysfunctional footnote environments, you can\FNH@msgbk
         try sending the author a small illustrative document.\FNH@msgbk
         To turn off this message, add \string\footnotehyperwarnfalse\FNH@msgbk
         to the preamble\@gobble}% "noline"
      \fi
    \else\FNH@bad@makefntext@alert
    \fi
}%
```

Let's check that pre and post do not contain some weird stuff caused from an original \@makefntext{#1} containing #1 multiple times.

```
\def\FNH@check@b #1\relax{%
    \expandafter\expandafter\expandafter\FNH@check@c
    \expandafter\meaning\expandafter\FNH@prefntext
    \meaning\FNH@postfntext1.2!3?4,\FNH@check@c\relax
}%
\def\FNH@check@c #11.2!3?4,#2#3\relax{%
    \ifx\FNH@check@c#2\else\FNH@bad@makefntext@alert\fi
}%
```

Hard to decipher \@makefntext, so warn user and (1.1e) use as fall-back the code from the article class with a safety \@nameuse layer.

```
\def\FNH@bad@makefntext@alert{%
    \iffootnotehyperwarn
    \PackageWarningNoLine{footnotehyper}%
      {\FNH@msgbk
```

```
       Failed to analyse \string\@makefntext\space into something usable.\FNH@msgbk
       Using as fall-back the article class code.\FNH@msgbk
       You may try to email the author this problematic meaning:\FNH@msgbk
       \meaning\@makefntext\FNH@msgbk
       together with the document preamble (in particular if\FNH@msgbk
       footnote numbers do not show at bottom of page)}%
    \fi
    \let\FNH@prefntext \FNH@prefntext@fallback
    \let\FNH@postfntext\FNH@postfntext@fallback
}%
\def\FNH@prefntext@fallback{% from article class code with \@nameuse added
    \parindent 1em\noindent
    \hb@xt@1.8em{\hss\@textsuperscript{\normalfont\@nameuse{@thefnmark}}}%
}%
\let\FNH@postfntext@fallback\@empty
```

\makesavenoteenv

- with LaTeX prior to October 2020 release, this is the same as original. Not recommended. Safer to use explicitely savenotes environment.
- with LaTeX from October 2020 or later, a safer approach is applied which goes either via the hook mechanism (for the use case with no optional argument), or via a \newenvironment and cautious use of the public \begin and \end interface (for the use case with an optional argument), rather than fiddling with \foo and \endfoo macros.

```
\def\makesavenoteenv{\@ifnextchar[\FNH@msne@ii\FNH@msne@i}%]
\def\FNH@msne@i #1{%
  \expandafter\let\csname FNH$#1\expandafter\endcsname %$
                 \csname #1\endcsname
  \expandafter\let\csname endFNH$#1\expandafter\endcsname %$
                 \csname end#1\endcsname
  \FNH@msne@ii[#1]{FNH$#1}%$
}%
\def\FNH@msne@ii[#1]#2{%
  \expandafter\edef\csname#1\endcsname{%
    \noexpand\savenotes
    \expandafter\noexpand\csname#2\endcsname
  }%
  \expandafter\edef\csname end#1\endcsname{%
    \expandafter\noexpand\csname end#2\endcsname
    \noexpand\expandafter
    \noexpand\spewnotes
    \noexpand\if@endpe\noexpand\@endpetrue\noexpand\fi
  }%
}%
\@ifl@t@r\fmtversion{2020/10/01}{%
    \def\FNH@msne@i#1{%
        \AddToHook{env/#1/before}{\savenotes}%
        \AddToHook{env/#1/after}{\spewnotes}%
    }%
    \def\FNH@msne@ii[#1]#2{%
        \newenvironment{#1}{\begin{savenotes}\begin{#2}}%
                           {\end{#2}\end{savenotes}}%
    }%
  }%
  {}%
```

Original footnote.sty patches `\parbox`, we don't touch it. Also it defines a `minipage*` environment, we don't do it.

```
% \makesavenoteenv[minipage*]{minipage}
% \let\fn@parbox\parbox
% \def\parbox{\@ifnextchar[{\fn@parbox@i{}}{\fn@parbox@ii{}}}
% \def\fn@parbox@i#1[#2]{%
%   \@ifnextchar[{\fn@parbox@i{#1[#2]}}{\fn@parbox@ii{#1[#2]}}%
% }
% \long\def\fn@parbox@ii#1#2#3{\savenotes\fn@parbox#1{#2}{#3}\spewnotes}
\endinput
```