Le *package* appendix[*†]

Peter Wilson
Catholic University of America[‡]
Maintenant à `peter.r.wilson@boeing.com`

1998/11/29

Résumé

Ce *package* offre quelques possibilités relatives aux appendices. Il est conçu pour fonctionner avec les cinq classes LaTeX standard `book`, `report`, `article`, `proc` et `ltxdoc`. Les résultats avec d'autres classes peuvent être problématiques. Ce *package* n'a pas été testé avec d'autres *packages* qui modifient les définitions des commandes de sectionnement.

Table des matières

1 Introduction

Dans les classes standard, la commande `\appendix` fait les choses suivantes :

- Pour les classes avec chapitres :
  - remet à zéro les compteurs `chapter` et `section`.
  - force `\@chapapp` à `\appendixname`.
  - redéfinit `\thechapter` à produire une numérotation alphabétique des appendices.
- Pour les autres classes :
  - Remet à zéro les compteurs de section et de sous-section.
  - Redéfinit `\thesection` pour produire une numérotation alphabétique des appendices.

Le *package* `appendix` offre des possibilités supplémentaires pour les appendices.

Des portions du *package* ont été développés en tant que partie d'une classe et d'un ensemble de *packages* pour la composition de documents au standard ISO [Wil96]. Ce manuel est réalisé conformément aux conventions de l'utilitaire

*Traduit de l'anglais par Jean-Pierre Drucbert (`drucbert@onecert.fr`).

†Ce fichier est la version v1.0, modifié le 1998/11/29.

‡Ce travail a été à l'origine exécuté en tant que chercheur invité au National Institute of Standards and Technology.

LaTeX docstrip qui permet l'extraction automatique du fichier source contenant les macros LaTeX [GMS94].

La section 2 décrit l'utilisation du *package* et le code source du *package* est détaillé dans la section 3.

2 Le *package* appendix

Ce *package* offre quelques commandes qui peuvent être utilisées en plus de la commande `\appendix`. Il fournit aussi un environnement qui peut être utilisé à la place de la commande `\appendix`. Cet environnement offre quelques action supplémentaires par rapport à la simple commande `\appendix`. Nous allons présenter d'abord les nouvelles commandes puis le nouvel environnement.

`\appendixpage` La commande `\appendixpage` composera un en-tête dans le style d'un entête de `\part` pour la classe. Le texte de l'en-tête est donné par la valeur de `\appendixpagename`.

`\addappheadtotoc` La commande `\addappheadtotoc` insérera un en-tête général dans la table des matières (*ToC*, *table of contents*). Le texte est donnée par la valeur de `\appendixtocname`. Si elle est utilisée, cette commande doit être placée avant le premier appendice, car elle est censée introduire les titres des appendices dans la *ToC*.

Les commandes ci-dessus peuvent être utilisée en conjonction avec la commande traditionnelle `\appendix`, qu'elles doivent suivre immédiatement. Par exemple :

```
\appendix
\addappheadtotoc
\appendixpage
```

`\appendixname` `\appendixtocname` `\appendixpagename` La commande `\appendixname` est définie dans les classes qui définissent des chapitres. Elle est fournie dans ce *package* que la classe l'ait définie ou non. Sa valeur par défaut est « Appendix ». La valeur par défaut de `\appendixtocname` et `\appendixpagename` est « Appendices ». Ces noms peuvent êtres changés via `\renewcommand`. Par exemple,

```
\renewcommand{\appendixtocname}{List of appendices}
```

`appendices` L'environnement `appendices` peut être utilisé à la place de la commande `\appendix`. Il offre davantage de fonctionnalité qu'il n'est possible d'avoir avec des combinaisons de commandes `\appendix`, `\addappheadtotoc` et `\appendixpage`. Les fonctions de l'environnement `appendices` sont accessibles via des options du *package*, qui sont :

- toc Met un en-tête (par exemple « Appendices ») dans la table des matières (*ToC*) avant de lister les appendices (en appelant la commande `\addappheadtotoc`.) ;
- page Met un titre (par exemple « Appendices ») dans le document au point où l'environnement `appendices` a commencé (en appelant la commande `\appendixpage`.) ;
- title Ajoute un nom (par exemple « Appendix ») avant chaque titre d'appendice dans le corps du document. Le nom est donné par la valeur de `\appendixname`. Notez que c'est le comportement par défaut dans les classes qui ont des chapitres ;

- titletoc Ajoute un nom (par exemple, « Appendice ») avant chaque appendice listé dans la *ToC*. Le nom est donné par la valeur de `\appendixname` ;
- header Ajoute un nom (par exemple « Appendix ») avant chaque appendice dans les en-têtes de page. Le nom est donné par la valeur de `\appendixname`. Notez que c'est le comportement par défaut pour les classes qui ont des chapitres.

Selon les options particulières du *package* qui sont choisies et la classe de document, l'environnement `appendices` peut changer la définition d'éléments des commandes de sectionnement (par exemple `\chapter` ou `\section`). Ceci peut être un problème si l'environnement est utilisé en conjonction avec un autre *package* qui modifie ces commandes. Si c'est le cas, il vous faut examiner le code de l'environnement `appendices` et faire les modifications nécessaires à celui ou ceux des autres *packages* (dans votre propre version de ce ou ces *packages*). Les modifications effectuées sur les commandes de sectionnement sont éliminées à la fin de l'environnement `appendices`.

3 Le code du *package*

On annonce le nom et la version du *package*, qui nécessite LaTeX 2_ε et le *package* ifthen.

```
⟨*usc⟩
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{appendix}[1998/11/29 v1.0 extra appendix facilities]
\RequirePackage{ifthen}
```

Dans l'optique d'éviter les conflits de noms avec d'autres *packages*, chaque nom interne inclut les caractères `@pp`.

`\if@knownclass@pp` Ceci est utilisé lorsque nous avons besoin de décider quel style d'appendices est
`\if@chapter@pp` utilisé dans le document. On suppose que la classe est article (ou ses dérivés).

```
\newif\if@chapter@pp\@chapter@ppfalse
\newif\if@knownclass@pp\@knownclass@ppfalse
```

On teste la classe actuelle.

```
\@ifclassloaded{book}{\@chapter@pptrue\@knownclass@pptrue}{}
\@ifclassloaded{report}{\@chapter@pptrue\@knownclass@pptrue}{}
\@ifclassloaded{article}{\@knownclass@pptrue}{}
\@ifclassloaded{proc}{\@knownclass@pptrue}{}
\@ifclassloaded{ltxdoc}{\@knownclass@pptrue}{}
```

`\if@dotoc@pp` Un ensemble de booléens pour les options. Par défaut, l'environnement `appendices`
`\if@dotitle@pp` ne fait rien de plus que la commande `\appendix` à moins que une ou plusieurs op-
`\if@dotitletoc@pp` tions ait été modifiée.
`\if@dohead@pp`
`\if@dopage@pp`

```
\newif\if@dotoc@pp\@dotoc@ppfalse
\newif\if@dotitle@pp\@dotitle@ppfalse
\newif\if@dotitletoc@pp\@dotitletoc@ppfalse
\newif\if@dohead@pp\@dohead@ppfalse
\newif\if@dopage@pp\@dopage@ppfalse
```

Maintenant, nous pouvons nous occuper des options.

```
\DeclareOption{toc}{\@dotoc@pptrue}
\DeclareOption{title}{\@dotitle@pptrue}
\DeclareOption{titletoc}{\@dotitletoc@pptrue}
```

```
\DeclareOption{header}{\@dohead@pptrue}
\DeclareOption{page}{\@dopage@pptrue}
```

On traite les options maintenant.

```
\ProcessOptions\relax
```

Provoque une alerte si la classe n'est pas reconnue. Utilise le style chapitre si `\chapter` est défini, sinon utilise le style section.

```
\if@knownclass@pp\else
  \PackageWarning{appendix}%
    {I don't recognize the class but I'll do my best}
  \ifx\chapter\undefined\else\@chapter@pptrue\fi
\fi
```

\appendixname
\appendixtocname
\appendixpagename

Ces commandes contiennent les noms qui vont être utilisés. `\appendixname` peut être déjà défini dans la classe. Les autres sont nouveaux.

```
\providecommand{\appendixname}{Appendix}
\newcommand{\appendixtocname}{Appendices}
\newcommand{\appendixpagename}{Appendices}
```

\if@openright

Nous avons besoin de ceci. Ceci est défini dans les classes de type book et pas dans celles de type article.

```
% \ifx\if@openright\undefined
%   \newif\if@openright
% \fi
```

\appendixpage

La commande pour composer une page annonçant le début des appendices. Cela est basé sur la définition de `\part` (soit à partir de la classe book ou de la classe article). Notez que cela *ne crée pas* d'entrée dans la *ToC*.

```
\newcommand{\appendixpage}{%
  \if@chapter@pp \@chap@pppage \else \@sec@pppage \fi
}
```

\@chap@pppage

Crée une page d'appendice en style chapitre. Copie du code de `\part` à partir de la classe book, mais on utilise `\appendixpagename` comme titre.

```
\newcommand{\@chap@pppage}{%
  \if@openright\cleardoublepage\else\clearpage\fi
  \thispagestyle{plain}%
  \if@twocolumn\onecolumn\@tempswatrue\else\@tempswafalse\fi
  \null\vfil
  \markboth{}{}%
  {\centering
   \interlinepenalty \@M
   \normalfont
   \Huge \bfseries \appendixpagename\par}%
```

Dans la classe book, la commande `\part` se termine par l'appel à `\@endpart`. Cela pose deux problèmes ici : (1) `\@endpart` n'est pas défini dans la classe article et (2) cela ajoute une page blanche qui n'est pas très esthétique si l'option openany est utilisée. Donc, le code est décrit ici :

```
  \vfil\newpage
  \if@twoside
    \if@openright
      \null
      \thispagestyle{empty}%
```

```
      \newpage
    \fi
  \fi
  \if@tempswa
    \twocolumn
  \fi
}
```

`\@sec@pppage` Copie du code de la commande `\part` à partir de la classe article, mais on utilise `\appendixpagename` comme titre.

```
\newcommand{\@sec@pppage}{%
  \par
  \addvspace{4ex}%
  \@afterindentfalse
  {\parindent \z@ \raggedright
   \interlinepenalty \@M
   \normalfont
   \huge \bfseries \appendixpagename%
   \markboth{}{}\par}%
  \nobreak
  \vskip 3ex
  \@afterheading
}
```

`\addappheadtotoc` Cette commande ajoute une ligne « appendice » à la *ToC*. Le style est le même que pour la « Liste des figures » dans tocbibind, *i.e.* comme un en-tête de chapitre ou de section.

```
\newcommand{\addappheadtotoc}{%
  \if@chapter@pp
    \addcontentsline{toc}{chapter}{\appendixtocname}
  \else
    \addcontentsline{toc}{section}{\appendixtocname}
  \fi
}
```

`\@resets@pp` Réinitialisation des compteurs de structure et des noms appropriés. Cela fait exactement la même chose que la commande `\appendix` standard.

```
\newcommand{\@resets@pp}{
  \par
  \setcounter{section}{0}
  \if@chapter@pp
    \setcounter{chapter}{0}
    \renewcommand\@chapapp{\appendixname}
    \renewcommand\thechapter{\@Alph\c@chapter}
  \else
    \setcounter{subsection}{0}
    \renewcommand\thesection{\@Alph\c@section}
  \fi
}
```

`appendices` Ceci est le cœur du *package*. on commence par les réinitialisations faites par la commande `\appendix`. Ensuite on s'occupe des options faciles avant de passer aux difficultés : les redéfinitions.

```
\newenvironment{appendices}{%
```

```
\@resets@pp
\if@dotoc@pp \addappheadtotoc \fi
\if@dopage@pp \appendixpage \fi
```

Il y a une seule autre option se rapportant au style chapitre, on s'en occupe maintenant et on laisse la voie libre pour faire le style section. Pour l'option titletoc, on redéfinit la commande `\addcontentsline`.

```
\if@chapter@pp
  \if@dotitletoc@pp \@redotocentry@pp{chapter} \fi
\else
```

Le reste du code est spécifique au style section. Tant que l'on y est, nous pouvons aussi bien finir de traiter l'option titletoc.

```
  \if@dotitletoc@pp \@redotocentry@pp{section} \fi
```

Le code suivant implémente l'option header en créant une version spéciale de `\sectionmark`.

```
  \if@dohead@pp
    \def\sectionmark##1{%
      \if@twoside
        \markboth{\@formatsecmark@pp{##1}}{}
      \else
        \markright{\@formatsecmark@pp{##1}}{}
      \fi}
  \fi
```

Le code suivant implémente l'option title en rusant avec la commande `\@seccntformat`[1].

```
  \if@dotitle@pp
    \def\sectionname{\appendixname}
    \def\@seccntformat##1{\@ifundefined{##1name}{}{\csname ##1name\endcsname\ }%
      \csname the##1\endcsname\quad}
  \fi
\fi}{}
```

`\@formatsecmark@pp` Formate l'en-tête pour redéfinir `\sectionmark`.

```
\newcommand{\@formatsecmark@pp}[1]{%
  \MakeUppercase{\appendixname\space
  \ifnum \c@secnumdepth >\z@
    \thesection\quad
  \fi
  #1}}
```

`\@redotocentry@pp` Pour implémenter l'option titletoc, on redéfinit la commande `\addcontentsline` qui est utilisée pour ajouter des entrées dans la *ToC*. `\@redotocentry@pp{`⟨*sect*⟩`}` réalise la redéfinition, avec ⟨*sect*⟩ le nom de l'en-tête particulier (*i.e.* soit le chapitre soit la section).

```
\newcommand{\@redotocentry@pp}[1]{%
```

Enregistre la définition originelle de `\addcontentsline`, puis commence la redéfinition.

```
  \let\oldacl@pp=\addcontentsline
  \def\addcontentsline##1##2##3{%
    \ifthenelse{\equal{##1}{toc}}{%
```

[1] d'après une contribution de Donald Arsenau dans `compt.text.tex` le 13 août 1998.

Ajout au fichier *ToC*, donc contrôle de la commande de sectionnement.

```
123       \ifthenelse{\equal{##2}{#1}}{%
```

La commande de sectionnement est la même que celle passée en argument à `\@redotocentry@pp`, on continue alors la redéfinition.

```
124         \addtocontents{##1}{\protect\contentsline{##2}%
125                                   {\appendixname\space ##3}{\thepage}}}%
```

L'en-tête était différent de l'argument : la redéfinition n'est pas nécessaire, on appelle donc la commande `\addcontentsline` originale.

```
126       {\oldacl@pp{##1}{##2}{##3}}}%
```

On ajoute à un fichier qui n'est pas la *ToC* : aucune redéfinition n'est requise, on appelle donc la commande `\addcontentsline` originale.

```
127     {\oldacl@pp{##1}{##2}{##3}}}}
```

La fin de ce *package*.

```
128 ⟨/usc⟩
```

Références

[GMS94] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion.* Addison-Wesley Publishing Company, 1994.

[Wil96] Peter R. Wilson. *LaTeX for standards : The LaTeX package files user manual.* NIST Report NISTIR, June 1996.

Index

Numbers written in italic refer to the page where the corresponding entry is described, the ones underlined to the code line of the definition, the rest to the code lines where the entry is used.