

# L'extension appendix\*

Auteur : Peter Wilson, Herries Press

Mainteneur : Will Robertson

will point robertson arobase latex-project point org

02/09/2009

## Résumé

L'extension `appendix` fournit quelques éléments pour modifier la composition des titres des annexes. De plus, les environnements `(sub)appendices` sont mis à disposition et permettent d'obtenir, par exemple, des annexes par chapitre ou par section.

L'extension est conçue pour fonctionner avec les classes de document disposant des commandes `\chapter` et/ou `\section`. Elle n'a pas été testée avec des extensions qui modifient la définition des commandes de sectionnement.

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>L'extension appendix</b>	<b>2</b>
2.1	Problèmes connus . . . . .	5
<b>3</b>	<b>Le code de l'extension</b>	<b>5</b>

## 1 Introduction

Dans les classes de document <sup>1</sup> standards, la commande `\appendix` effectue les actions suivantes :

- pour les classes avec chapitre :
  - remet à zéro les compteurs `chapter` et `section`,
  - force `\@chapapp` à `\appendixname`,
  - redéfinit `\thechapter` pour produire une numérotation alphabétique des annexes,

---

\*Ce fichier (`appendix-fr.dtx`) a pour numéro de version v1.2b, datant du 02/09/2009. La première traduction en français de la version v1.0 est due à Jean-Pierre Drucbert.

1. Par la suite, le terme « classe » sera utilisé systématiquement pour désigner la « classe de document ».

- pour les autres classes :
  - remet à zéro les compteurs de section et de sous-section,
  - redéfinit `\thesection` pour produire une numérotation alphabétique des annexes.

L’extension `appendix` offre des possibilités supplémentaires pour les annexes. Elle est compatible avec l’extension `hyperref`<sup>2</sup> mais peut être source de problèmes quand elle est utilisée avec des extensions qui modifient les définitions des commandes de sectionnement.

Certaines parties de l’extension ont été développées en tant que part d’une classe et d’un ensemble d’extensions traitant de la composition de documents au standard ISO [Wil96]. Ce manuel est réalisé conformément aux conventions de l’utilitaire L<sup>A</sup>T<sub>E</sub>X DOCSTRIP qui permet l’extraction automatique du fichier source contenant les macros L<sup>A</sup>T<sub>E</sub>X [GM05].

La section 2 décrit l’utilisation de l’extension. Son code source est, quant à lui, détaillé dans la section 3.

## 2 L’extension `appendix`

L’extension `appendix` propose quelques commandes qui peuvent être utilisées en complément de la commande `\appendix`. Elle fournit aussi un environnement qui peut être utilisé à la place de la commande `\appendix`. Cet environnement offre quelques possibilités supplémentaires par rapport à la commande `\appendix`. Nous allons présenter d’abord les nouvelles commandes puis étudier le nouvel environnement.

`\appendixpage` La commande `\appendixpage` compose un titre intercalaire à la manière de ce que fait la commande `\part` dans la classe du document. Le titre est ici celui de la variable `\appendixpagename`.

`\addappheadtotoc` La commande `\addappheadtotoc` insère un titre dans la table des matières. Son texte est donné par la valeur de `\appendixtocname`. En cas d’utilisation, cette commande doit être placée avant la première annexe car elle place les titres des différentes annexes dans la table des matières.

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

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

`\noappendicestocpagenum` Par défaut, la commande `\addappheadtotoc` place un numéro de page dans la table des matières. Ceci peut être annulé en utilisant la commande `\noappendicestocpagenum`. Par symétrie, la commande `\appendicestocpagenum` garantit qu’un numéro de page sera bien mis en table des matières.

---

2. Mes remerciements à Hylke W. van Dijk ([hylke@ubicom.tudelft.nl](mailto:hylke@ubicom.tudelft.nl)) qui m’a indiqué que la version 1.1 ne l’était pas et m’a mis sur la piste pour changer cette situation.

**Note :** à moins que `\noappendicestocpagenum` ne soit utilisé, la commande `\addappheadtotoc` utilise le numéro de page courante lorsqu'il crée l'entrée dans la table des matières. La commande `\appendixpage` compose un titre en reprenant le style d'un titre de niveau `\part` dans la classe du document. Pour des classes sans chapitre, cet titre apparaît dans le texte à la manière d'un titre de `\section`; et pour des classes avec chapitre, il apparaît sur une page à part. Autrement dit, dans le second cas, `\appendixpage` exécute une commande `\clear[double]page`, compose le titre des annexes, puis exécute une nouvelle fois `\clear[double]page`. C'est pourquoi l'entrée en table des matières aura le numéro de la page suivant la page de ce titre<sup>3</sup>. Si l'ordre est inversé (par exemple `\addappheadtotoc \appendixname`) alors le numéro de page en table des matières sera celui de la page précédant la page de ce titre. Pour les documents avec chapitre, il est préférable d'utiliser :

```
\clearpage % ou \cleardoublepage
\addappheadtotoc
\appendixpage
```

qui permet d'avoir le bon numéro de la page du titre des annexes en table des matières.

`\appendixname`      La commande `\appendixname` est définie dans les classes avec chapitre Elle  
`\appendixtocname`    est fournie dans cette extension, que la classe l'ait définie ou pas. Sa valeur  
`\appendixpagename` par défaut est « *Appendix* ». La valeur par défaut de `\appendixtocname` et  
`\appendixpagename` `\appendixpagename` est « *Appendices* ». Ces noms peuvent étre changés par le  
biais de `\renewcommand`. Par exemple, pour avoir les différents titres en français :

```
\renewcommand{\appendixname}{Annexe}
\renewcommand{\appendixpagename}{Annexes}
\renewcommand{\appendixtocname}{Annexes}
```

**appendices**      L'environnement `appendices` peut être utilisé à la place de la commande `\appendix`. Il offre plus de possibilités que celles issues des combinaisons de commandes `\appendix`, `\addappheadtotoc` et `\appendixpage`. L'accès aux fonctions de l'environnement `appendices` se fait normalement par des options de l'extension, mais des déclarations peuvent les remplacer. Les options sont :

- `toc` qui place un titre (par défaut « *Appendices* ») dans la table des matières avant de lister les annexes (ce qui s'obtient aussi avec la commande `\addappheadtotoc`).
- `page` qui place un titre (par défaut « *Appendices* ») dans le document au point où l'environnement `appendices` débute (ce qui s'obtient avec la commande `\appendixpage`).
- `title` qui ajoute un terme (par défaut « *Appendix* ») avant chaque titre d'annexe dans le corps du document. Ce terme est donné par la valeur de la variable `\appendixname`. Notez qu'il s'agit du comportement par défaut des classes avec chapitre.

---

3. Merci à Eduardo Jacob ([edu@kender.es](mailto:edu@kender.es)) d'avoir relevé ce point.

- `titletoc` qui ajoute un terme (par défaut « *Appendix* ») avant chaque titre d'annexe listé dans la table des matières. Ce terme est donné par la valeur de la variable `\appendixname`.
- `header` qui ajoute un terme (par défaut « *Appendix* ») avant chaque titre d'annexe apparaissant dans l'en-tête de page. Ce terme est donné par la valeur de la variable `\appendixname`. Notez que ceci est le comportement par défaut des classes avec chapitre.

Selon les options d'extension et la classe choisies, l'environnement `appendices` peut changer la définition d'éléments des commandes de sectionnement (`\chapter` ou `\section`). Ceci peut être problématique si l'environnement est utilisé en conjonction avec une extension modifiant ces commandes. Si c'est le cas, il vous faut alors examiner le code de l'environnement `appendices` et faire les modifications nécessaires à l'extension de votre choix (dans votre version de fichier de l'extension). Les modifications effectuées sur les commandes de sectionnement sont supprimées à la fin de l'environnement `appendices`.

<code>\appendixtocon</code>	La déclaration <code>\appendixtocon</code> est équivalente à l'option <code>toc</code> . Inversement,
<code>\appendixtocoff</code>	<code>\appendixtocoff</code> est équivalente à ne pas utiliser cette option.
<code>\appendixpageon</code>	La déclaration <code>\appendixpageon</code> est équivalente à l'option <code>page</code> . Inversement,
<code>\appendixpageoff</code>	<code>\appendixpageoff</code> est équivalente à ne pas utiliser cette option.
<code>\appendixtitleon</code>	La déclaration <code>\appendixtitleon</code> est équivalente à l'option <code>title</code> . Inversement,
<code>\appendixtitleoff</code>	<code>\appendixtitleoff</code> est équivalente à ne pas utiliser cette option.
<code>\appendixtitletocon</code>	La déclaration <code>\appendixtitletocon</code> est équivalente à l'option <code>titletoc</code> . Inversement,
<code>\appendixtitletocoff</code>	<code>\appendixtitletocoff</code> est équivalente à ne pas utiliser cette option.
<code>\appendixheaderon</code>	La déclaration <code>\appendixheaderon</code> est équivalente à l'option <code>header</code> . Inversement,
<code>\appendixheaderoff</code>	<code>\appendixheaderoff</code> est équivalente à ne pas utiliser cette option.
<code>\restoreapp</code>	Lorsqu'il finit, l'environnement <code>appendices</code> restitue aux compteurs de chapitres et sections la valeur qu'ils avaient au moment où l'environnement débutait, ceci afin qu'il puisse être utilisé entre de grandes divisions du document. Par défaut, la valeur du compteur d'annexes est sauvegardée et récupérée par l'environnement. Ceci signifie que les annexes dans une série d'environnements <code>appendices</code> seront numérotées par des lettres qui se suivent. Pour pouvoir repartir de la lettre A pour chaque environnement, il faut utiliser en préambule de document la commande suivante :

```
\renewcommand{\restoreapp}{{}
```

<code>subappendices</code>	Dans l'environnement <code>subappendices</code> , une « sous-annexe » est introduite par la commande <code>\section</code> pour les classes avec chapitre et par la commande <code>\subsection</code> pour les autres classes. Ceci fournit un moyen efficace d'avoir des annexes comme partie intégrante d'une division du document principal, à la fin de cette division. L'environnement <code>subappendices</code> autorise uniquement les options <code>title</code> et <code>titletoc</code> .
----------------------------	--

<code>\setthesection</code>	Par défaut, les « sous-annexes » sont numérotées comme des (sous-)sections normales, à ceci près que le numéro de la « sous-annexe » est composé par une lettre majuscule. Ce comportement peut être modifié en redéfinissant les commandes
<code>\setthesubsection</code>	

`\setthe(...)`. Par exemple, pour obtenir uniquement une lettre non précédée du numéro de la division principale, saisissez selon le contexte :

```
\renewcommand{\setthesection}{\Alph{section}}| % ou
\renewcommand{\setthesubsection}{\Alph{subsection}}
```

## 2.1 Problèmes connus

Il existe une interaction malencontreuse entre les commandes du noyau  $\text{\LaTeX}$  `\include` et `\addcontentsline`. Si elles sont utilisées comme suit :

```
\addcontentsline{toc}{...}{Entrée particulière}
\include{import}
```

alors le texte de la commande `\addcontentsline` (« Entrée particulière » dans l'exemple) n'est pas écrit dans le fichier approprié (d'extension « toc ») tant que le fichier importé par `\include` n'a pas écrit toutes ses entrées dans le fichier « toc ». Pour autant que je puisse en juger, il n'y a pas de méthode de contournement sans réécriture de parties du code du noyau  $\text{\LaTeX}$ .

Il revient donc à l'auteur d'éviter d'utiliser la commande `\addcontentsline` (ou toute commande qui utilise en interne la commande `\addcontentsline`, comme la commande `\addappheadtotoc`) avant qu'un fichier intégré par `\include` n'ait écrit ses entrées dans le fichier « toc ». Par ailleurs, tout fonctionne normalement si les commandes `\addcontentsline` sont mises dans le fichier importé par `\include` ou si le fichier est importé par `\input` plutôt que par `\include`.

## 3 Le code de l'extension

Sont donnés le nom et la version de l'extension (qui nécessite  $\text{\LaTeX} 2_{\epsilon}$ ).

```
1 \*usc
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{appendix}[2009/09/02 v1.2b extra appendix facilities]
4
```

Afin de tenter d'éviter les conflits de noms avec d'autres extensions, chaque nom interne de l'extension inclut les caractères `@pp`.

```
\if@knownclass@pp Ces commandes sont utilisées pour déterminer quel style d'annexes est retenu dans
\if@chapter@pp le document. La classe est supposée être par défaut article (ou toute classe sans
chapitre).
```

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

Vérification de la présence de commandes `\chapter` et `\section`.

```
7 \@ifundefined{chapter}{%
8 \@ifundefined{section}{\@knownclass@pptrue}}{%
9 \@chapter@pptrue\@knownclass@pptrue}
```

<code>\phantomsection</code> <code>\the@pps</code> <code>\if@pphyper</code>	<p>La commande <code>\phantomsection</code> doit être fournie si l’extension <code>hyperref</code> n’est pas utilisée. De plus, que l’extension <code>hyperref</code> soit utilisée ou pas, un compteur doit être défini pour supporter d’éventuels hyperliens (ce qui sert à lever toute ambiguïté sur les (sous-)annexes). <code>\if@pphyper</code> vaut <code>true</code> si l’extension <code>hyperref</code> est chargée.</p> <pre> 10 \providecommand{\phantomsection}{} 11 \newcounter{@pps} 12 \renewcommand{\the@pps}{\alph{@pps}} 13 \newif\if@pphyper 14 \@pphyperfalse 15 \AtBeginDocument{% 16   \ifpackageloaded{hyperref}{\@pphypertrue}{} 17 }</pre>
<code>\if@dotoc@pp</code> <code>\if@dotitle@pp</code> <code>\if@dotitletoc@pp</code> <code>\if@dohead@pp</code> <code>\if@dopage@pp</code>	<p>Un ensemble de booléens pour les options. Par défaut, l’environnement <code>appendices</code> ne fait rien de plus que la commande <code>\appendix</code> à moins que des options ne soient utilisées.</p> <pre> 18 \newif\if@dotoc@pp\@dotoc@ppfalse 19 \newif\if@dotitle@pp\@dotitle@ppfalse 20 \newif\if@dotitletoc@pp\@dotitletoc@ppfalse 21 \newif\if@dohead@pp\@dohead@ppfalse 22 \newif\if@dopage@pp\@dopage@ppfalse</pre> <p>Les cinq options sont déclarées.</p> <pre> 23 \DeclareOption{toc}{\@dotoc@pptrue} 24 \DeclareOption{title}{\@dotitle@pptrue} 25 \DeclareOption{titletoc}{\@dotitletoc@pptrue} 26 \DeclareOption{header}{\@dohead@pptrue} 27 \DeclareOption{page}{\@dopage@pptrue}</pre> <p>Elles sont ensuite traitées.</p> <pre> 28 \ProcessOptions\relax</pre> <p>Émission d’un avertissement si les commandes <code>\chapter</code> et <code>\section</code> ne sont pas définies (« Il n’y a pas de commande <code>\chapter</code> ou <code>\section</code>. L’extension <code>appendix</code> ne sera pas utilisée. »), puis sortie de l’extension.</p> <pre> 29 \newcommand{\@ppendinput}{} 30 \if@knownclass@pp\else 31   \PackageWarningNoLine{appendix}% 32     {There is no \protect\chapter\space or \protect\section\space command.\MessageBreak 33     The appendix package will not be used} 34   \renewcommand{\@ppendinput}{\endinput} 35 \fi 36 \@ppendinput 37</pre>
<code>\appendixtocon</code> <code>\appendixtocoff</code>	<p>Formes déclaratives de l’option <code>toc</code>.</p> <pre> 38 \newcommand{\appendixtocon}{\@dotoc@pptrue} 39 \newcommand{\appendixtocoff}{\@dotoc@ppfalse}</pre>

<code>\appendixpageon</code>	Formes déclaratives de l'option page.
<code>\appendixpageoff</code>	40 <code>\newcommand{\appendixpageon}{\@dopage@pptrue}</code> 41 <code>\newcommand{\appendixpageoff}{\@dopage@ppfalse}</code>
<code>\appendixtitleon</code>	Formes déclaratives de l'option title.
<code>\appendixtitleoff</code>	42 <code>\newcommand{\appendixtitleon}{\@dotitle@pptrue}</code> 43 <code>\newcommand{\appendixtitleoff}{\@dotitle@ppfalse}</code>
<code>\appendixtitletocon</code>	Formes déclaratives de l'option titletoc.
<code>\appendixtitletocoff</code>	44 <code>\newcommand{\appendixtitletocon}{\@dotitletoc@pptrue}</code> 45 <code>\newcommand{\appendixtitletocoff}{\@dotitletoc@ppfalse}</code>
<code>\appendixheaderon</code>	Formes déclaratives de l'option header.
<code>\appendixheaderoff</code>	46 <code>\newcommand{\appendixheaderon}{\@dohead@pptrue}</code> 47 <code>\newcommand{\appendixheaderoff}{\@dohead@ppfalse}</code>
<code>\@ppsavesec</code>	Lors de l'utilisation de l'environnement <code>appendices</code> le numéro de division (le chapitre ou la section) du document principal et le numéro d'annexe doivent pouvoir être sauvegardés comme restaurés. La commande <code>\restoreapp</code> est à la main de l'utilisateur.
<code>\@pprestoresec</code>	
<code>\@ppsaveapp</code>	
<code>\restoreapp</code>	
	48 <code>\newcounter{@ppsavesec}</code> 49 <code>\newcounter{@ppsaveapp}</code> 50 <code>\setcounter{@ppsaveapp}{0}</code> 51 <code>\newcommand{\@ppsavesec}{%</code> 52 <code>\ifchapter@pp \setcounter{@ppsavesec}{\value{chapter}} \else</code> 53 <code>\setcounter{@ppsavesec}{\value{section}} \fi}</code> 54 <code>\newcommand{\@pprestoresec}{%</code> 55 <code>\ifchapter@pp \setcounter{chapter}{\value{@ppsavesec}} \else</code> 56 <code>\setcounter{section}{\value{@ppsavesec}} \fi}</code> 57 <code>\newcommand{\@ppsaveapp}{%</code> 58 <code>\ifchapter@pp \setcounter{@ppsaveapp}{\value{chapter}} \else</code> 59 <code>\setcounter{@ppsaveapp}{\value{section}} \fi}</code> 60 <code>\newcommand{\restoreapp}{%</code> 61 <code>\ifchapter@pp \setcounter{chapter}{\value{@ppsaveapp}} \else</code> 62 <code>\setcounter{section}{\value{@ppsaveapp}} \fi}</code>
<code>\appendixname</code>	Ces commandes contiennent les noms à utiliser. La commande <code>\appendixname</code> peut avoir déjà été définie par la classe. Les autres commandes sont nouvelles.
<code>\appendixtocname</code>	
<code>\appendixpagename</code>	63 <code>\providecommand{\appendixname}{Appendix}</code> 64 <code>\newcommand{\appendixtocname}{Appendices}</code> 65 <code>\newcommand{\appendixpagename}{Appendices}</code>
<code>\appendixpage</code>	La commande compose un titre dans le corps du document annonçant le début des annexes. Elle est basée sur la définition de <code>\part</code> , soit sur la base de la classe <code>book</code> (avec <code>\@chap@pppage</code> ), soit sur celle de la classe <code>article</code> (avec <code>\@sec@pppage</code> ). 66 <code>\newcommand{\appendixpage}{%</code> 67 <code>\ifchapter@pp \@chap@pppage \else \@sec@pppage \fi</code> 68 <code>}</code>

`\clear@ppage` Les classes sans chapitre ne définissent pas la commande `\if@openright` mais cette dernière est ici nécessaire pour les classes avec chapitre pour générer les sauts de page appropriés. La commande `\clear@ppage` effectue le travail souhaité mais ne peut être utilisée que dans un code gérant des chapitres sous peine d’obtenir des messages d’erreur comme `extra \else` (« `\else` en trop ») ou `extra \fi` (« `\fi` en trop »).

```
69 \newcommand{\clear@ppage}{%
70   \if@openright\cleardoublepage\else\clearpage\fi}
71
```

`\@chap@ppage` Cette commande produit une page d’annexe à la manière de ce qui est fait pour une partie dans une classe avec chapitre. Elle copie du code de la commande `\part` de la classe `book` mais utilise `\appendixpagename` comme titre.

```
72 \newcommand{\@chap@ppage}{%
73   \clear@ppage
74   \thispagestyle{plain}%
75   \if@twocolumn\onecolumn\@tempwattrue\else\@tempwafalse\fi
76   \null\vfil
77   \markboth{}{}%
78   {\centering
79     \interlinepenalty \OM
80     \normalfont
81     \Huge \bfseries \appendixpagename\par}%

```

Ajout d’une entrée en table des matières si besoin est.

```
82   \if@dotoc@pp
83     \addappheadtotoc
84   \fi

```

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

```
85   \vfil\newpage
86   \if@twoside
87     \if@openright
88       \null
89       \thispagestyle{empty}%
90       \newpage
91     \fi
92   \fi
93   \if@tempwa
94     \twocolumn
95   \fi
96 }
97
```

`\@sec@ppage` Cette commande produit un titre pour les annexes à la manière d’une section non numérotée dans une classe sans chapitre. Elle copie du code de la commande `\part` de la classe `article` mais utilise `\appendixpagename` comme titre.



```

98 \newcommand{\@sec@pppage}{%
99   \par
100   \addvspace{4ex}%
101   \@afterindentfalse
102   {\parindent \z@ \raggedright
103     \interlinepenalty \@M
104     \normalfont
105     \huge \bfseries \appendixpagename%
106     \markboth{}{}\par}%

```

Ajout d’une entrée en table des matières si besoin est.

```

107 \if@dotoc@pp
108   \addappheadtotoc
109 \fi
110 \nobreak
111 \vskip 3ex
112 \@afterheading
113 }
114

```

`\if@pptocpage` La commande `\addappheadtotoc` ajoute une ligne « *Appendices* » à la table des  
`\noappendicestocpagenum` matières. Le style retenu est le même que pour la « *List of figures*<sup>4</sup> » dans l’ex-  
`\appendicestocpagenum` tension `tocbibind`, autrement dit comme un titre de chapitre ou de section selon la  
`\addappheadtotoc` classe du document. `\if@pptocpage` contrôle si le numéro de page est mis ou pas  
dans la table des matières.

```

115 \newif\if@pptocpage
116 \@pptocpagetrue
117 \newcommand{\noappendicestocpagenum}{\@pptocpagefalse}
118 \newcommand{\appendicestocpagenum}{\@pptocpagetrue}
119 \newcommand{\addappheadtotoc}{%
120   \phantomsection
121   \if@chapter@pp
122     \if@pptocpage
123       \addcontentsline{toc}{chapter}{\appendixtocname}%
124     \else
125       \if@pphyper
126         \addtocontents{toc}%
127           {\protect\contentsline{chapter}{\appendixtocname}{\@currentHref}}%
128       \else
129         \addtocontents{toc}%
130           {\protect\contentsline{chapter}{\appendixtocname}{}}%
131       \fi
132     \fi
133   \else

```

Cas d’une classe sans chapitre.

```

134 \if@pptocpage

```

---

4. La « table des figures » avec l’option `frenchb` de l’extension `babel`.

```

135     \addcontentsline{toc}{section}{\appendixtocname}%
136   \else
137     \if@pphyper
138       \addtocontents{toc}%
139       {\protect\contentsline{section}{\appendixtocname}{\@currentHref}}%
140     \else
141       \addtocontents{toc}%
142       {\protect\contentsline{section}{\appendixtocname}}%
143     \fi
144   \fi
145 \fi
146 }
147

```

Pour référence interne, voici la version standard de la commande `\appendix` mais modifiée à la fois pour les classes avec et sans chapitre.

```

\newcommand{\appendix}{\par
  \if@chapter@pp
    \setcounter{chapter}{0}%
    \setcounter{section}{0}%
    \gdef\@chapapp{\appendixname}%
    \gdef\thechapter{\@Alph{c}{chapter}}
  \else
    \setcounter{section}{0}%
    \setcounter{subsection}{0}%
    \gdef\thesection{\@Alph{c}{section}}
  \fi
}

```

Et, de manière équivalente, voici ce que fait l'extension `hyperref`.

```

\def\Hy@chapterstring{chapter}
\def\Hy@appendixstring{appendix}
\def\Hy@chapapp{\Hy@chapterstring}
\let\Hy@org@appendix\appendix
\def\appendix{%
  \Hy@org@appendix
  \if@chapter@pp
    \gdef\theHchapter{\Alph{chapter}}%
  \else
    \gdef\theHsection{\Alph{section}}%
  \fi
  \xdef\Hy@chapapp{\Hy@appendixstring}%
}

```

`\theH@pps` La commande `\theH@pps` est utilisée pour lever l'ambiguïté sur le contenu d'annexes pouvant partager les mêmes marques hypertextes. Cette commande est définie avec `\providecommand` au cas où les extensions `appendix` et `hyperref` seraient

chargées dans le « mauvais » ordre, ce qui amène alors `hyperref` à définir la commande avant que `appendix` ne puisse y accéder.

```
148 \providecommand{\theH@pps}{\alph{@pps}}
149
```

`\@resets@pp` La commande réinitialise les compteurs de sectionnement appropriés et les noms associés. Elle fait ainsi à peu près la même chose que la commande `\appendix` standard, à ceci près qu'elle sauvegarde et restaure la numérotation de certaines divisions : la sauvegarde du numéro de division (la section ou le chapitre selon la classe) est faite au début de l'utilisation, la restauration du numéro d'annexe en fin d'utilisation.

```
150 \newcommand{\@resets@pp}{\par
151   \@ppsavesec
152   \stepcounter{@pps}
153   \setcounter{section}{0}%
154   \if@chapter@pp
155     \setcounter{chapter}{0}%
156     \renewcommand\@chapapp{\appendixname}%
157     \renewcommand\thechapter{\@Alph@c@chapter}%
158   \else
159     \setcounter{subsection}{0}%
160     \renewcommand\thesection{\@Alph@c@section}%
161   \fi
162   \if@pphyper
```

Le code traite ici des points associés à l'extension `hyperref`.

```
163   \if@chapter@pp
164     \renewcommand{\theHchapter}{\theH@pps.\Alph{chapter}}%
165   \else
166     \renewcommand{\theHsection}{\theH@pps.\Alph{section}}%
167   \fi
168   \def\Hy@chapapp{\appendixname}%
169 \fi
170 \restoreapp
171 }
172
```

`appendices` Cette partie est au cœur de l'extension. Elle commence par les réinitialisations faites par la commande `\appendix`. Ensuite elle traite les options simples avant d'entrer dans les difficultés liées aux redéfinitions. Pensez bien à faire attention à l'interaction entre `\addappheadtotoc` et `\appendixpage`.

```
173 \newenvironment{appendices}{%
174   \@resets@pp
175   \if@dotoc@pp
176     \if@dopage@pp           % page et table des matières à la fois
177     \if@chapter@pp         % présence de chapitre
178     \clear@ppage
179   \fi
180   \appendixpage
```

```

181 \else % table des matières uniquement
182 \if@chapter@pp % présence de chapitre
183 \clear@ppage
184 \fi
185 \addappheadtotoc
186 \fi
187 \else
188 \if@dopage@pp % page uniquement
189 \appendixpage
190 \fi
191 \fi

```

Il y a une seule autre option se rapportant au style des chapitres : elle est traitée ici, ce qui prépare le traitement du style des sections. Pour implémenter l'option `\titletoc`, la commande `\addcontentsline` est redéfinie.

```

192 \if@chapter@pp
193 \if@dotitletoc@pp \@redotocentry@pp{chapter} \fi
194 \else

```

Le reste du code est spécifique au style des sections. Nous en profitons pour finir de traiter l'option `\titletoc` par la même occasion.

```

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

```

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

```

196 \if@dohead@pp
197 \def\sectionmark##1{%
198 \if@twoside
199 \markboth{\@formatsecmark@pp{##1}}{}
200 \else
201 \markright{\@formatsecmark@pp{##1}}{}
202 \fi}
203 \fi

```

Le code suivant implémente l'option `\title` en traitant astucieusement la commande `\@secntformat` <sup>5</sup>.

```

204 \if@dotitle@pp
205 \def\sectionname{\appendixname}
206 \def\@secntformat##1{\@ifundefined{##1name}{}{\csname ##1name\endcsname\ }}%
207 \csname the##1\endcsname\quad}
208 \fi
209 \fi}%

```

À la fin de l'environnement, le numéro d'annexe est sauvé et le numéro de section est restauré.

```

210 \@ppsaveapp\@ppstoresec}
211

```

`\setthesection` Ces commandes permettent à l'utilisateur de spécifier le style de numérotation pour les « sous-annexes ».

---

5. d'après une contribution de Donald ARSENAU dans `compt.text.tex` le 13 août 1998.

```

212 \newcommand{\setthesection}{\thechapter.\Alph{section}}
213 \newcommand{\setthesubsection}{\thesection.\Alph{subsection}}
214

```

`\@resets@ppsub` La commande est similaire à `\@resets@pp` à ceci près qu'elle est utilisée pour l'environnement `subappendices` ; de fait, elle est un peu plus simple.

```

215 \newcommand{\@resets@ppsub}{\par
216   \stepcounter{@pps}
217   \if@chapter@pp
218     \setcounter{section}{0}
219     \renewcommand{\thesection}{\setthesection}
220   \else
221     \setcounter{subsection}{0}
222     \renewcommand{\thesubsection}{\setthesubsection}
223   \fi
224   \if@pphyper
    Le code traite ici des points associés à l'extension hyperref.
225     \if@chapter@pp
226       \renewcommand{\theHsection}{\theH@pps.\setthesection}%
227     \else
228       \renewcommand{\theHsubsection}{\theH@pps.\setthesubsection}%
229     \fi
230     \def\Hy@chapapp{\appendixname}%
231   \fi
232 }
233

```

`subappendices` Cet environnement gère les « sous-annexes ». Il commence par reparamétrer la commande `\(sub)section`.

```

234 \newenvironment{subappendices}{%
235   \@resets@ppsub

```

Il y a deux options utilisables pour le style des chapitres. Pour implémenter l'option `titletoc`, la commande `\addcontentsline` est redéfinie.

```

236   \if@chapter@pp
237     \if@dotitletoc@pp \@redotocentry@pp{section} \fi

```

L'option `title` est implémentée en traitant astucieusement la commande `\@secntformat`

```

238     \if@dotitle@pp
239       \def\sectionname{\appendixname}
240       \def\@secntformat##1{\@ifundefined{##1name}{\csname ##1name\endcsname\ }%
241         \csname the##1\endcsname\quad}
242     \fi
243   \else

```

Le reste du code gère ici le style des sections.

```

244     \if@dotitletoc@pp \@redotocentry@pp{subsection} \fi
245     \if@dotitle@pp
246       \def\subsectionname{\appendixname}
247       \def\@secntformat##1{\@ifundefined{##1name}{\csname ##1name\endcsname\ }%

```

```

248      \csname the##1\endcsname\quad}
249      \fi
250      \fi}{}}
251

```

`\@formatsecmark@pp` La commande `format` l'en-tête pour la commande `\sectionmark` redéfinie.

```

252 \newcommand{\@formatsecmark@pp}[1]{%
253   \MakeUppercase{\appendixname\space
254     \ifnum \c@secnumdepth >\z@
255       \thesection\quad
256     \fi
257   #1}}

```

`\@redotocentry@pp` Pour implémenter l'option `titletoc`, la commande `\addcontentsline`, qui ajoute des entrées en table des matières, est redéfinie. `\@redotocentry@pp{<sect>}` réalise la redéfinition, avec `<sect>` le nom de la division souhaitée (soit le chapitre avec `chapter`, soit la section avec `section`).

```

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

```

La définition classique de `\addcontentsline` est sauvegardée et la redéfinition commence.

```

259   \let\oldacl@pp=\addcontentsline
260   \def\addcontentsline##1##2##3{%

```

Il faut vérifier si l'écriture dans le fichier de table des matières est demandée.

```

261     \def\@pptempa{##1}\def\@pptempb{toc}%
262     \ifx\@pptempa\@pptempb

```

Si c'est bien le cas, le niveau de sectionnement est vérifié.

```

263       \def\@pptempa{##2}\def\@pptempb{#1}%
264       \ifx\@pptempa\@pptempb

```

Le niveau de sectionnement est le même que celui spécifié par l'argument de `\@redotocentry@pp` ; la redéfinition se poursuit donc.

```

265 \oldacl@pp{##1}{##2}{\appendixname\space ##3}%
266   \else

```

Le niveau de sectionnement est différent : la redéfinition n'est donc pas nécessaire, la commande `\addcontentsline` classique est utilisée.

```

267     \oldacl@pp{##1}{##2}{##3}%
268     \fi
269   \else

```

Le fichier ciblé n'était pas celui de la table des matières : la redéfinition n'est donc pas nécessaire, la commande `\addcontentsline` classique est utilisée.

```

270     \oldacl@pp{##1}{##2}{##3}%
271     \fi}
272 }

```

Fin de l'extension.

```

273 </usc>

```

## Références

- [GM05] Michel Goossens et Frank Mittelbach. *LaTeX Companion*, 2<sup>e</sup> éd., Pearson, 2005.
- [Wil96] Peter R. Wilson. *LaTeX for standards : The LaTeX package files user manual*. NIST Report NISTIR, juin 1996.

## Index

Les nombres en italique renvoient à la page où se trouve l'entrée correspondante; les numéros soulignés renvoient à la ligne de code de la définition; les numéros en caractères romains renvoient aux lignes de code où l'entrée est utilisée.

[illegible]

environnements :	<code>\interlinepenalty</code> .	<b>S</b>
appendices ..... <a href="#">173</a>	..... 79, 103	<code>\section</code> ..... 32
subappendices .. <a href="#">234</a>		<code>\sectionmark</code> ..... 197
	<b>M</b>	<code>\sectionname</code> .. 205, 239
<b>H</b>	<code>\MakeUppercase</code> .... 253	<code>\setthesection</code> ....
<code>\Huge</code> ..... 81	<code>\MessageBreak</code> ..... 32	..... <a href="#">212</a> , 219, 226
<code>\huge</code> ..... 105		<code>\setthesubsection</code> .
<code>\Hy@chapapp</code> ... 168, 230	<b>N</b>	..... <a href="#">212</a> , 222, 228
	<code>\newif</code> 5, 6, 13, 18–22, 115	<code>\stepcounter</code> .. 152, 216
	<code>\noappendicestocpagenum</code>	subappendices (envi-
	..... <a href="#">115</a>	ronnement) .. <a href="#">234</a>
<b>I</b>	<code>\normalfont</code> .... 80, 104	<code>\subsectionname</code> ... 246
<code>\if@chapter@pp</code> ....		
.. <a href="#">5</a> , 52, 55, 58,	<b>O</b>	<b>T</b>
61, 67, 121, 154,	<code>\oldacl@pp</code> ..... . 259, 265, 267, 270	<code>\the@pps</code> ..... <a href="#">10</a>
163, 177, 182,	<code>\onecolumn</code> ..... 75	<code>\thechapter</code> ... 157, 212
192, 217, 225, 236	<b>P</b>	<code>\theH@pps</code> .... <a href="#">148</a> ,
<code>\if@dohead@pp</code> .. <a href="#">18</a> , 196	<code>\PackageWarningNoLine</code>	164, 166, 226, 228
<code>\if@dopage@pp</code> ....	..... 31	<code>\theHchapter</code> ..... 164
..... <a href="#">18</a> , 176, 188	<code>\parindent</code> ..... 102	<code>\theHsection</code> .. 166, 226
<code>\if@dotitle@pp</code> ....	<code>\phantomsection</code> <a href="#">10</a> , 120	<code>\theHsubsection</code> ... 228
. <a href="#">18</a> , 204, 238, 245	<code>\providecommand</code> ...	<code>\thesection</code> ..... . 160, 213, 219, 255
<code>\if@dotitletoc@pp</code> <a href="#">18</a> ,	..... 10, 63, 148	<code>\thesubsection</code> .... 222
193, 195, 237, 244	<code>\ProvidesPackage</code> ... 3	<code>\thispagestyle</code> .. 74, 89
<code>\if@dotoc@pp</code> ....	<b>Q</b>	<code>\twocolumn</code> ..... 94
.. <a href="#">18</a> , 82, 107, 175	<code>\quad</code> . 207, 241, 248, 255	
<code>\if@knownclass@pp</code> <a href="#">5</a> , 30	<b>R</b>	<b>V</b>
<code>\if@openright</code> ... 70, 87	<code>\restoreapp</code> .... <a href="#">48</a> , 170	<code>\value</code> ... 52, 53, 55,
<code>\if@pphyper</code> .... <a href="#">10</a> ,		56, 58, 59, 61, 62
125, 137, 162, 224		
<code>\if@pptocpage</code> ..... <a href="#">115</a>		
<code>\if@twocolumn</code> ..... 75		
<code>\if@twoside</code> .... 86, 198		