

# Le package `xr`\*

David Carlisle†

(traduction française: Jean-Pierre Drucbert  
Jean-Pierre.Drucbert@onecert.fr)

1994/05/28

Ce package permet de gérer des Références eXternes

Si un document se réfère à certaines sections d'un autre, par exemple `aaa.tex`, alors charger ce package dans le fichier principal et placer la commande `\externaldocument{aaa}` dans le préambule.

Vous pouvez alors utiliser `\ref` et `\pageref` pour vous référer à tout ce qui a été défini par `\label` dans chaque `aaa.tex` ou dans le document principal. Il est possible de déclarer autant de documents externes que l'on veut.

Si l'un des documents externes ou le document principal utilise le même `\label`, cette étiquette (label) étant définie plusieurs fois, une erreur survient. Afin de contourner ce problème, la commande `\externaldocument` a un argument optionnel. Si vous déclarez

```
\externaldocument[A-]{aaa}
```

alors toutes les références dans le document `aaa` sont préfixées par `A-`. Par exemple, si une section de `aaa` contient `\label{intro}`, alors ce label pourra être référencé par `\ref{A-intro}`. Le préfixe peut être différent de `A-`; il peut être toute chaîne choisie pour que tous les étiquettes importées depuis des fichiers externes soient uniques. Notez cependant que si vos packages déclarent certains caractères actifs (: en français, " en allemand, alors ils ne peuvent pas en général être utilisés dans l'argument de `\label` ni dans l'argument optionnel de `\externaldocument`.

## 1 Les macros (section non traduite)

```
1 <{*package}>
```

Check for the optional argument.

```
2 \def\externaldocument{\@ifnextchar[\XR@{\XR@[]}}
```

Save the optional prefix. Start processing the first aux file.

```
3 \def\xr@#1#2{%
```

---

\*Cette version a comme numéro v5.02, dernière revision 1994/05/28.

†L'auteur des versions 1 à 4 est Jean-Pierre Drucbert (Jean-Pierre.Drucbert@onecert.fr)

```

4 \makeatletter
5 \def\XR@prefix{#1}%
6 \XR@next#2.aux\relax\}}

Process the next aux file in the list and remove it from the head of the list of
files to process.
7 \def\XR@next#1\relax#2\{\%
8 \edef\XR@list{#2}%
9 \XR@loop{#1}}

Check whether the list of aux files is empty.
10 \def\XR@aux{%
11 \ifx\XR@list\@empty\else\expandafter\XR@explist\fi}

Expand the list of aux files, and call \XR@next to process the first one.
12 \def\XR@explist{\expandafter\XR@next\XR@list\}}

If the aux file exists, loop through line by line, looking for \newlabel and
\@input. Otherwise process the next file in the list.
13 \def\XR@loop#1{\openin\@inputcheck#1\relax
14 \ifeof\@inputcheck
15 \PackageWarning{xr}{^^JNo file #1^^JLABELS NOT IMPORTED.^^J}%
16 \expandafter\XR@aux
17 \else
18 \PackageInfo{xr}{IMPORTING LABELS FROM #1}%
19 \expandafter\XR@read\fi}

Read the next line of the aux file.
20 \def\XR@read{%
21 \read\@inputcheck to\XR@line

The ... make sure that \XR@test always has sufficient arguments.
22 \expandafter\XR@test\XR@line...\XR@}

Look at the first token of the line. If it is \newlabel, do the \newlabel. If it
is \@input, add the filename to the list of files to process. Otherwise ignore. Go
around the loop if not at end of file. Finally process the next file in the list.
23 \long\def\XR@test#1#2#3#4\XR@{%
24 \ifx#1\newlabel
25 \newlabel{\XR@prefix#2}{#3}%
26 \else\ifx#1\@input
27 \edef\XR@list{\XR@list#2\relax}%
28 \fi\fi
29 \ifeof\@inputcheck\expandafter\XR@aux
30 \else\expandafter\XR@read\fi}
31 \</package>

```