

14. Remplacer et enregistrer les fichiers avec scrwfile

Un des problèmes qui n'ont pas été résolus par l'introduction de ϵ -TeX est le fait que seulement 18 fichiers TeX peuvent être ouverts simultanément pour l'écriture, quantité qui, au premier abord, paraît assez grande.

Toutefois, il convient de noter que LaTeX lui-même occupe une partie de ces fichiers : `\@mainaux`, `\@partaux`, `\tableofcontents`, `\listoffigures`, `\listoftables`, `\makeindex` et pour chaque répertoire généré à partir de LaTeX. A cet effet, les fichiers d'aide proviennent de packs comme `hyperref` ou `minitoc`.

En fin de compte, il peut donc arriver que, à un moment donné, le message

```
! Pas de place pour une nouvelle \écriture.  
  \ch@ck ... \else \errmessage {pas de place pour un nouveau 3#}  
                                          \fi
```

apparaisse.

Cette version de LaTeX ouvre toujours immédiatement un nouveau fichier pour écrire dans les répertoires tels que table des matières, liste des tables et liste des figures, ce qui a également un autre inconvénient.

Ces répertoires sont non seulement définis par les commandes correspondantes, mais peuvent également ne pas être fixés une fois de plus, parce que leurs fichiers auxiliaires restent vides jusqu'à la fin du document. Le pack `scrwfile` fournit un changement fondamental du noyau de LaTeX, qui peut résoudre les deux problèmes

14.1. Modifications fondamentales du noyau de LaTeX

Les classes LaTeX utilisent la commande du noyau LaTeX `\starttoc` pour ouvrir, par exemple, un répertoire `\tableofcontents` ou `\listoffigure`, liste de flottants ou similaire. Ainsi LaTeX lit le fichier d'aide avec le contenu du répertoire, mais l'ouvre aussi en écriture.

Néanmoins, si par la suite, de nouvelles entrées sont effectuées en utilisant, par exemple `\addtocontents` ou `\addcontentsline`, elles ne seront pas écrites immédiatement dans le fichier d'aide. LaTeX `\@writefile` les enregistre dans un fichier auxiliaire et c'est seulement lors de la relecture des fichiers `.aux`, à la fin du document, que `\@writefile` écrit les instructions dans des fichiers auxiliaires réels.

Les fichiers d'aide ne sont pas fermés explicitement par LaTeX qui compte sur TeX pour les fermer de toute façon à la fin.

Cette procédure garantit que les fichiers auxiliaires seront effectivement décrits avec le `\end{document}`, mais sont également ouverts simultanément pendant toute l'utilisation de LaTeX, de la première page jusqu'à la fin du document, bien que leur contenu soit écrit après cette fin. L'approche de `scrwfile` est exactement celle-ci: redéfinition des `\starttoc` et `\writefile`.

Certes, des changements du noyau de LaTeX peuvent potentiellement entraîner des incompatibilités avec d'autres packs. Dans le cas de `scrwfile` cela peut arriver avec tous les packs qui redéfinissent (trop) `\starttoc` ou `\writefile`.

Parfois, changer l'ordre de chargement des packs peut aider.

En fait, de tels problèmes avec `scrwfile` n'ont pas été rapportés récemment, bien que plusieurs utilisateurs ont testé le pack depuis la première version, pendant plus d'un an. Si cela, néanmoins, vous arrivait, merci de contacter l'auteur de KOMA-Script.

14.2. Le système de fichiers

Pendant le chargement du pack en utilisant `\usepackage{scrwfile}` `scrwfile` redéfinira `\starttoc` de sorte qu'il n'est plus nécessaire d'ouvrir en écriture un fichier ou de lui allouer un descripteur, ces procédures devenant automatiques.

Immédiatement avant la fermeture `\end{document}` du fichier.aux, `\writefile` redéfinit cette déclaration non plus dans les fichiers d'aide habituels, mais dans un nouveau fichier auxiliaire de support réel, unique, avec extension `.wrt`. Après avoir lu le fichier `.aux`, ce dossier sera traité une fois par le fichier d'aide.

Après avoir lu les fichiers `.wrt` le fichier est toujours en cours de traitement et sera en effet écrit une fois pour chacun des fichiers auxiliaires avec `\writefile`. Toutefois, il ne doit pas être ouvert en même temps que chacun de ces fichiers auxiliaires. Il ne le sera donc que pour l'écriture et, à chaque fois, de nouveau explicitement fermés.

Depuis qu'un traitement de l'écriture interne à LaTeX est réutilisé, `scrwfile` ne nécessite aucune écriture distincte du fichier pour ce type de répertoires. Même si une seule table des matières doit être générée, après le chargement du pack `scrwfile` un descripteur de fichier d'écriture sera disponible pour glossaires, index, bibliographies, et autres répertoires qui n'utilisent pas `\starttoc`.

En outre, le nombre de tables des matières et des listes de tout ce qui utilise `\starttoc` ne sera pas limité.

14.3. Le clonage des entrées avec `CloneFileWrite`

Il est parfois utile d'accéder à un fichier non pas une, mais plusieurs fois. Comme `\starttoc` ne permet pas d'ouvrir les fichiers pour écrire plus longtemps, cela peut être fait en utilisant simplement `\starttoc` plusieurs fois avec la même extension.

Après que `\@writefile` pour les fichiers système de la section précédente ait été modifié de sorte qu'il n'écrit pas directement le fichier d'aide correspondant, une autre idée est venue. La copie de l'instruction `\@writefile` dans le fichier `.wrt` peut être transférée à d'autres entrées de cibles, ce qui permet de copier l'entrée d'un fichier dans un autre fichier. Ainsi, toutes les entrées du fichier `\jobname.source` peuvent être ajoutées à `\jobname.destination`. Si la destination d'extension est une nouvelle destination, elle sera ajoutée à la liste des extensions connues utilisant le pack KOMA-Script `tocbasic`.

Nous appelons cette opération de copie clonage.

```
\TOCclone[titre]{source}{cible}
```

active la fonction de clonage pour les fichiers titre avec des extensions source et destination (cible), `\listofcible`. Ce clonage d'entrée de fichier clone des répertoires

entiers. A cet effet, il est nécessaire de préciser la fin d'un fichier cible dont les enregistrements doivent être copiés. Dans ceux-ci, les entrées seront copiées. Bien sûr, vous pouvez écrire dans ce répertoire des entrées supplémentaires dans une partie seulement des répertoires de contenu également clonées.

L'objectif de fin du fichier cible est géré administrativement en utilisant tocbasic (voir le chapitre 14). Si un tel fichier est déjà sous contrôle de tocbasic un avertissement est émis. Sinon, tocbasic crée un nouveau répertoire pour cette extension. Le titre de ce répertoire peut être déterminé avec l'arguments en option de l'en-tête de liste. Ensuite, vous pouvez sortir ce nouveau répertoire, par exemple via la commande \listofdestination (liste des extensions de cible), et une nouvelle liste de macro est définie. La rubrique sera utilisée comme section (ou chapitre) de cette liste.

Le niveau de répertoire des propriétés numérotées ainsi que onecolumn totoc (voir la commande \setuptoc à la section 14.2) seront automatiquement inclus dans le répertoire cible, s'ils étaient déjà définis pour le répertoire source. Dans ce cas, plusieurs caractéristiques tocbasic de la source seront copiées à destination, si et seulement si elles ont été mises en place lorsque \TOCclone a été utilisé. Les attributs nobabel sont toujours réglées, car les commandes de sélection de la langue font partie du fichier d'aide, donc sont également clonées.

Exemple :

Supposons que vous voulez un aperçu de répartition, dans lequel seuls les chapitres sont affichés, en plus de la table des matières habituelle:

```
\usepackage{scrwfile}
\TOCclone[vue d'ensemble des grandes lignes ]{toc}{stoc}
```

Cela crée un nouveau répertoire avec la rubrique «Vue d'ensemble Outline» qui est d'abord appliquée. Le nouveau répertoire utilise le stoc extension de fichier. Toutes les entrées dans le fichier avec l'extension toc devront également être copiées dans ce répertoire. Pour activer ce nouveau répertoire qui retourne désormais seulement le niveau de chapitre, nous utilisons:

```
\addtocontents{stoc}{\protect\value{tocdepth}=0}
```

Normalement une entrée dans un répertoire peut être prise avant, jusqu'au lancement de \begin{document}, cela fonctionne déjà en chargeant scrwfile dans le préambule du document.

Représenté ici de façon non conventionnelle, tocdepth change le compteur dans le fichier de répertoire, changement qui reste en vigueur uniquement pour ce répertoire. Plus tard dans le document, la liste d'entrées sera alors stoc, gérée par \listofstoc, ne montrant que parties et sections du document, ce qui est assez difficile lorsqu'un aperçu de la table des matières doit être affiché.

Nous désirons créer une nouvelle table des matières avec la rubrique «Short Contents». La nouvelle table des matières utilise un fichier d'aide avec une extension stoc. Toutes les entrées dans le fichier d'aide avec l'extension toc seront également copiées dans ce nouveau fichier d'aide. La nouvelle table des matières short ne doit avoir d'entrée de chapitre.

Ce peut être fait en utilisant:

```
\addtocontents{toc}{%
\protect\addcontentsline
{stoc}{chapter}{\protect\contentsname}%
}
```

Cependant, toutes les entrées sont copiées dans toc après stoc, et seraient donc également acceptées par l'aperçu du format de cette entrée. Donc, cet article ne peut pas être généré à partir du fichier répertoire. Depuis le pack tocbasic est utilisé, et

```
\AfterStartingTOC[toc]{%
\addcontentsline{stoc}{chapter}
{\protect\contentsname}}
```

peut être employé. Bien sûr, cela suppose que le fichier avec l'extension toc est également sous le contrôle de tocbasic. Ceci est valable pour toutes les classes de KOMA-Script. Pour plus de détails sur la commande `\AfterStartingTOC` voir la section 14.2.

Pour afficher les nouveaux contenus courts des fichiers aide de l'extension stoc nous allons utiliser `\listofstoc` quelque part après `\begin{document}`.

Si nous voulons aussi une entrée pour la table des matières des contenus courts, nous ne pouvons pas utiliser `\addtocontents {toc}{% écriture du contenu \protect\addcontentslinedefault{stoc}% écriture du contenu court {chapter}% une entrée de chapitre avec {\contentsname}% le nom des matières}` parce que la commande `\addcontentsline` serait trop copiée dans stoc et donc nous ne pouvons pas ajouter la commande au toc-fichier. Alors, le pack tocbasic sera utilisé:

14.4. Note sur le niveau de développement

Bien que le pack a déjà été testé par plusieurs utilisateurs et qu'il est en cours d'utilisation, son développement n'est pas encore terminé. Par conséquent, il est théoriquement possible que des modifications lui soient apportées, notamment dans le fonctionnement interne du pack, qui contient déjà une partie du code prévu pour les extensions. Aucun usager n'ayant fait de demande en ce sens, il n'existe pas actuellement de documentation utilisateur pour cela.

14.5. Options incompatibles de pack connu

Comme mentionné dans le chapitre 14.1, `scrwfile` redéfinit quelques instructions du noyau LaTeX. Cela se produit non seulement pendant le chargement de le pack, mais beaucoup plus à divers moments au cours de l'exécution d'un document, par exemple, avant de lire le fichier aux. Cela signifie que `scrwfile` n'est pas compatible avec les autres packs qui redéfinissent également ces instructions lors de l'exécution.

Un exemple d'une telle incompatibilité est `titletoc`. L'exécution de ce pack est définie, dans certaines circonstances, par `\writefile`. Utiliser `scrwfile` et `titletoc` ensemble et la fonction des deux packs n'est plus garantie. Cette erreur `titletoc` ne se produit qu'avec `scrwfile`.