

© 2015 Université Laval



Cette création est mise à disposition selon le contrat **Attribution-Partage dans les mêmes conditions 4.0 International** de Creative Commons. En vertu de ce contrat, vous êtes libre de :

- ▶ **partager** — reproduire, distribuer et communiquer l'œuvre ;
- ▶ **remixer** — adapter l'œuvre ;
- ▶ utiliser cette œuvre à des fins commerciales.

Selon les conditions suivantes :



Attribution — Vous devez créditer l'œuvre, intégrer un lien vers le contrat et indiquer si des modifications ont été effectuées à l'œuvre. Vous devez indiquer ces informations par tous les moyens possibles, mais vous ne pouvez suggérer que l'Offrant vous soutient ou soutient la façon dont vous avez utilisé son œuvre.



Partage dans les mêmes conditions — Dans le cas où vous modifiez, transformez ou créez à partir du matériel composant l'œuvre originale, vous devez diffuser l'œuvre modifiée dans les même conditions, c'est à dire avec le même contrat avec lequel l'œuvre originale a été diffusée.

Notes de cours et exercices développés par Vincent Goulet, professeur titulaire, avec la contribution financière de la Bibliothèque de l'Université Laval.

Table des matières

1	Document contenu dans plusieurs fichiers	1
1.1	Insertion de contenu avec la commande <code>\input</code>	1
1.2	Insertion de parties avec la commande <code>\include</code>	2

1 Document contenu dans plusieurs fichiers

Un document \LaTeX comporte toujours un préambule suivi du corps du texte. Lorsque ceux-ci sont relativement courts (peu de commandes spéciales et moins d'une vingtaine de pages de texte), il demeure assez simple et convivial d'en faire l'édition dans un seul fichier à l'aide de son éditeur de texte favori.

Cependant, si le préambule devient long et complexe ou, surtout, lorsque l'ampleur du document augmente jusqu'à compter un grand nombre de pages sur plusieurs chapitres, il convient de répartir les divers éléments du document dans des fichiers séparés.

La segmentation en plusieurs fichiers rend l'édition du texte plus simple et plus efficace. De plus, elle peut significativement accélérer la compilation des documents très longs ou comportant plusieurs images.

1.1 Insertion de contenu avec la commande `\input`

La commande `\input` permet d'insérer le contenu d'un autre fichier dans un document \LaTeX .

- La syntaxe de la commande est

```
\input{fichier}
```

où le nom du fichier à insérer est *fichier.tex*. On laisse donc tomber l'extension `.tex` qui est implicite.

- Le contenu du fichier est inséré tel quel dans le document, comme s'il avait été tapé dans le fichier qui contient l'appel à `\input`.
- Le procédé est surtout utile pour sauvegarder séparément des bouts de code qui pourraient nuire à l'édition du texte (figures, longs tableaux) ou

qui sont communs entre plusieurs documents (licence d'utilisation, auteur et affiliation).

- La commande peut aussi être utilisée dans le préambule pour charger une partie ou l'ensemble de celui-ci. Cela permet de composer un même préambule pour plusieurs documents. Il suffit alors de faire d'éventuelles modifications à un seul endroit pour les voir prendre effet dans tous les documents.

1.2 Insertion de parties avec la commande `\include`

Extrait de la documentation de la classe **ulthese** :

«Il est recommandé de segmenter tout document d'une certaine ampleur dans des fichiers `.tex` distincts pour chaque partie — habituellement un fichier par chapitre. Le document complet est composé à l'aide d'un fichier maître qui contient le préambule \LaTeX et un ensemble de commandes `\include` pour réunir les parties dans un tout.»

Comme `\input`, la commande `\include` insère le contenu d'un autre fichier dans un document \LaTeX . Son effet est cependant différent et c'est son utilisation qui permet d'accélérer la compilation d'un long document.

- L'insertion d'un fichier avec `\include` débute toujours une nouvelle page. On utilisera donc `\include` principalement pour insérer des chapitres entiers plutôt que pour des portions de texte.
- Un fichier inséré avec `\include` peut contenir des appels à `\input`, mais pas à `\include`.
- La syntaxe de la commande `\include` est

```
\include{fichier}
```

où le nom du fichier à insérer est *fichier.tex*. Ici aussi on laisse tomber l'extension `.tex` qui est implicite.

- La structure type d'un fichier maître est la suivante :

```
\documentclass{ulthese}  
[...]  
  
\begin{document}  
  
\frontmatter
```

```
\include{introduction}
\tableofcontents*

\mainmatter
\include{historique}      % premier chapitre
\include{modele}         % deuxième chapitre
[...]

\end{document}
```

- ▶ Le principal avantage de `\include` par rapport à `\input` réside dans le fait que \LaTeX peut préserver entre les compilations les informations telles que les numéros de pages, de sections ou d'équations, ainsi que les références. Cela permet, par exemple, de compiler le texte d'un seul chapitre — plutôt que du document entier — et de néanmoins en obtenir une image représentative.
- ▶ La commande `\includeonly`, que l'on utilise exclusivement dans le préambule, sert à spécifier le ou les fichiers à compiler tout en préservant la numérotation et les références. Sa syntaxe est

```
\includeonly{liste_fichiers}
```

où *liste_fichiers* contient les noms des fichiers à compiler séparés par des virgules et sans l'extension `.tex`.

- ▶ Lors de l'utilisation de la commande `\includeonly`, toute la numérotation dans les fichiers *liste_fichiers* suivra celle établie lors de la compilation précédente.

Supposons qu'un document est composé des fichiers `chapitre1.tex`, `chapitre2.tex`, `chapitre3.tex` et que les chapitres débutent respectivement aux pages 1, 23 et 41.

- ▶ Si seul le fichier `chapitre2.tex` est inclus dans la compilation, le numéro du chapitre sera toujours 2 et le folio de la première page sera toujours 23, même si les 22 pages précédentes ne se trouvent pas dans le document.
- ▶ Si l'on modifie le fichier `chapitre2` de telle sorte que le chapitre se termine maintenant à la page 46, il faudra recompiler le document avec au moins les fichiers `chapitre2` et `chapitre3` pour que les pages du chapitre 3 soient renumérotées à partir de 47.

L'[exercice 1.1](#) illustre mieux le cycle typique d'utilisation des commandes `\include` et `\includeonly`.

- 1.1 Cet exercice fait appel au fichier maître `exercice_include.tex` et à plusieurs fichiers auxiliaires. Schématiquement, le document est composé ainsi :

```
exercice_include.tex
├── \input pagetitre.tex
├── \include presentation.tex
│   └── \includegraphics console-screenshot.pdf
└── \include emacs.tex
```

La commande `\includegraphics` provient du paquetage **graphicx**.

- a) Étudier le code source du fichier maître `exercice_include.tex`, puis le compiler deux à trois fois jusqu'à ce que toutes les références internes soient à jour. Il est normal à ce stade que la figure 1 du document soit vide.
- b) Ajouter dans le préambule du fichier maître la commande

```
\includeonly{emacs}
```

puis compiler le document.

Observer que malgré l'absence du chapitre 1, la numérotation et les références demeurent à jour, notamment la table des matières.

- c) Remplacer la commande ajoutée en b) dans le préambule du fichier maître par la commande

```
\includeonly{presentation}
```

Vers la fin du fichier `presentation.tex`, activer la commande `\includegraphics` en supprimant le symbole `%` au début de la ligne. Compiler de nouveau le document deux fois.

Les modifications ont eu pour effet d'ajouter une page au chapitre 1. Observer que selon la table des matières, le chapitre 2 débute toujours à la page 3 alors que celle-ci est maintenant occupée par la figure 1.

- d) Afin de corriger la table des matières, désactiver la commande `\includeonly` dans le préambule du fichier maître, puis compiler de nouveau le document quelques fois.