

Ateliers L^AT_EX

Premier atelier : Premiers exemples

Yves DELHAYE

19 janvier 2010



Premier exemple : plan

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

Produire un document en utilisant L^AT_EX se fait en deux étapes. La première

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Produire un document en utilisant L^AT_EX se fait en deux étapes. La première consiste à taper le source du document

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Produire un document en utilisant L^AT_EX se fait en deux étapes. La première consiste à taper le source du document et la seconde consiste à le compiler.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

Produire un document en utilisant L^AT_EX se fait en deux étapes. La première consiste à taper le source du document et la seconde consiste à le compiler.

On retrouve donc les deux étapes essentielles de beaucoup de langages de programmation.



Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

Produire un document en utilisant L^AT_EX se fait en deux étapes. La première consiste à taper le source du document et la seconde consiste à le compiler.

On retrouve donc les deux étapes essentielles de beaucoup de langages de programmation.

En théorie, il faut donc un éditeur de texte pour produire le source puis un appel à un programme qui va se charger de traduire le source en un fichier décrivant le document d'une façon facilement compréhensible pour l'ordinateur (donc totalement incompréhensible pour l'humain).



Produire un document en utilisant L^AT_EX se fait en deux étapes. La première consiste à taper le source du document et la seconde consiste à le compiler.

On retrouve donc les deux étapes essentielles de beaucoup de langages de programmation.

En théorie, il faut donc un éditeur de texte pour produire le source puis un appel à un programme qui va se charger de traduire le source en un fichier décrivant le document d'une façon facilement compréhensible pour l'ordinateur (donc totalement incompréhensible pour l'humain).

Avec un EDI comme Kile ou T_EXNicer, les choses sont un peu plus simples car cet éditeur de texte propose, via des menus et boutons, d'automatiser les différentes étapes de compilation.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphe et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Voici le code source d'un document relativement minimal. Les explications suivent :

```
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[frenchb]{babel}
\begin{document}
```

Mes premiers pas avec `\LaTeX{}` sont un peu émouvants. Pour exemples, il faudra taper exactement ce qui est proposé, attention ! Une fois la compilation effectuée, on peut

```
\begin{itemize}
  \item visualiser le document à l'écran ;
  \item imprimer le document ;
  \item convertir le document en fichier PostScript ;
  \item et plein d'autres choses.
\end{itemize}
\end{document}
c'est fini.
```



Le résultat de la compilation est montré page ci-contre.

Mes premiers pas avec $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ sont un peu émouvants. Pour les premiers exemples, il faudra taper exactement ce qui est proposé, en faisant bien attention! Une fois la compilation effectuée, on peut

- visualiser le document à l'écran ;
- imprimer le document ;
- convertir le document en fichier PostScript ;
- et plein d'autres choses.



Etudions en détail ce code :

```
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[frenchb]{babel}
\begin{document}
  Mes premiers pas avec \LaTeX{} sont un peu émouvants. Pour les premiers
  exemples, il faudra taper exactement ce qui est proposé, en faisant bien
  attention ! Une fois la compilation effectuée, on peut
  \begin{itemize}
    \item visualiser le document à l'écran ;
    \item imprimer le document ;
    \item convertir le document en fichier PostScript ;
    \item et plein d'autres choses.
  \end{itemize}
\end{document}
c'est fini.
```



Entête : mettre le décor en place

Ceux d'entre vous qui ont déjà programmé de l'html ou un langage compilé comme le C ou le Pascal seront familiers avec cette notion. Avant de commencer le travail, il faut mettre des choses en place :

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Entête : mettre le décor en place

Ceux d'entre vous qui ont déjà programmé de l'html ou un langage compilé comme le C ou le Pascal seront familiers avec cette notion. Avant de commencer le travail, il faut mettre des choses en place :

- ▶ Dans la déclaration prévenir L^AT_EX que le document est un livre, un article, une lettre une présentation ou un poster.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Entête : mettre le décor en place

Ceux d'entre vous qui ont déjà programmé de l'html ou un langage compilé comme le C ou le Pascal seront familiers avec cette notion. Avant de commencer le travail, il faut mettre des choses en place :

- ▶ Dans la déclaration prévenir L^AT_EX que le document est un livre, un article, une lettre une présentation ou un poster.
- ▶ Les appels aux bibliothèques de commandes servent à étendre les possibilités de L^AT_EX sans devoir réinventer la roue à tout moment.



Entête : mettre le décor en place

Ceux d'entre vous qui ont déjà programmé de l'html ou un langage compilé comme le C ou le Pascal seront familiers avec cette notion. Avant de commencer le travail, il faut mettre des choses en place :

- ▶ Dans la déclaration prévenir L^AT_EX que le document est un livre, un article, une lettre une présentation ou un poster.
- ▶ Les appels aux bibliothèques de commandes servent à étendre les possibilités de L^AT_EX sans devoir réinventer la roue à tout moment.
- ▶ Nous verrons plus avant que c'est ici que nous pouvons définir nos propres nouvelles commandes : macros...



Entête : mettre le décor en place

Ceux d'entre vous qui ont déjà programmé de l'html ou un langage compilé comme le C ou le Pascal seront familiers avec cette notion. Avant de commencer le travail, il faut mettre des choses en place :

- ▶ Dans la déclaration prévenir \LaTeX que le document est un livre, un article, une lettre une présentation ou un poster.
- ▶ Les appels aux bibliothèques de commandes servent à étendre les possibilités de \LaTeX sans devoir réinventer la roue à tout moment.
- ▶ Nous verrons plus avant que c'est ici que nous pouvons définir nos propres nouvelles commandes : macros...
- ▶ C'est aussi ici que nous modifierons la mise en page globale du document si nous le désirons.



Déclaration ou classe : livre, rapport, article, ...

Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;



Déclaration ou classe : livre, rapport, article, ...

Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;
- ▶ report ;



Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;
- ▶ report ;
- ▶ article ;



Déclaration ou classe : livre, rapport, article, ...

Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;
- ▶ report ;
- ▶ article ;
- ▶ slides ;



Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;
- ▶ report ;
- ▶ article ;
- ▶ slides ;
- ▶ beamer ;

Déclaration ou classe : livre, rapport, article, ...

Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;
- ▶ report ;
- ▶ article ;
- ▶ slides ;
- ▶ beamer ;
- ▶ letter.



Déclaration ou classe : livre, rapport, article, ...

Un document L^AT_EX commence obligatoirement par la **déclaration**

```
\documentclass{classe_document}
```

où *classe_document* est un des mots-clés suivants :

- ▶ book ;
- ▶ report ;
- ▶ article ;
- ▶ slides ;
- ▶ beamer ;
- ▶ letter.

Il s'agit de la *classe* du document. Théoriquement, book est destiné à taper le source d'un livre, report d'un rapport (ouvrage moins important qu'un livre), article d'un ... article (document encore moins important qu'un rapport), slides ou beamer de transparents et letter de lettres.

bibliothèque de commandes, extension ou package

Les deux lignes suivantes commencent par `\usepackage`.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



bibliothèque de commandes, extension ou package

Les deux lignes suivantes commencent par `\usepackage`.

La syntaxe générale de cette commande est :

```
\usepackage [options] {nom_extension}
```

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



bibliothèque de commandes, extension ou package

Les deux lignes suivantes commencent par `\usepackage`.

La syntaxe générale de cette commande est :

```
\usepackage [options] {nom_extension}
```

Dans notre exemple, on trouve :

```
\usepackage [T1] {fontenc}
```

qui permet de taper directement les lettres accentuées .



bibliothèque de commandes, extension ou package

Les deux lignes suivantes commencent par `\usepackage`.

La syntaxe générale de cette commande est :

```
\usepackage[options]{nom_extension}
```

Dans notre exemple, on trouve :

```
\usepackage[T1]{fontenc}
```

qui permet de taper directement les lettres accentuées .

On trouve également :

```
\usepackage[frenchb]{babel}
```

La ligne suivante dit :

```
\begin{document}
```

ce qui indique que le document proprement dit va débuter après cette ligne. Le document prend fin avec la commande :

```
\end{document}
```



La ligne suivante dit :

```
\begin{document}
```

ce qui indique que le document proprement dit va débuter après cette ligne. Le document prend fin avec la commande :

```
\end{document}
```

C'est pourquoi le texte " c'est fini " (dernière ligne du source) n'a pas été pris en compte dans le document final.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

... rien que le document

Mes premiers pas avec \LaTeX sont un peu émouvants. Pour les premiers exemples, il faudra taper exactement ce qui est proposé, en faisant bien attention! Une fois la compilation effectuée, on peut

- visualiser le document à l'écran ;
- imprimer le document ;
- convertir le document en fichier PostScript ;
- et plein d'autres choses.



La partie située entre `\documentclass` et
`\begin{document}` s'appelle

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



La partie située entre `\documentclass` et `\begin{document}` s'appelle la *préambule*.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



La partie située entre `\documentclass` et `\begin{document}` s'appelle la *préambule*.

Et la partie située entre le `\begin{document}` et le `\end{document}` s'appelle le ...

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



La partie située entre `\documentclass` et `\begin{document}` s'appelle la *préambule*.

Et la partie située entre le `\begin{document}` et le `\end{document}` s'appelle le ... *document*.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



La partie située entre `\documentclass` et `\begin{document}` s'appelle la *préambule*.

Et la partie située entre le `\begin{document}` et le `\end{document}` s'appelle le ... *document*.

Jusqu'au chapitre extension,
le préambule sera toujours :

```
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[frenchb]{babel}
```

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



La partie située entre `\documentclass` et `\begin{document}` s'appelle le *préambule*.

Et la partie située entre le `\begin{document}` et le `\end{document}` s'appelle le ... *document*.

Jusqu'au chapitre extension,
le préambule sera toujours :

```
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[frenchb]{babel}
```

et nous ne le spécifierons donc plus, ainsi que le couple :

```
\begin{document}
\end{document}
```

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



L^AT_EX et comment le texte apparaît.

Ateliers L^AT_EX
Premier atelier :
Premiers exemples

Yves DELHAYE

Premiers
documents

Structure d'un fichier
L^AT_EX

Entête, préambule ou
header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et
indentation

Macro commandes

Listes

Typographie

Marges et format de
papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

Regardons à nouveau ce qui a été produit.



Mes premiers pas avec L^AT_EX sont un peu émouvants. Pour les premiers exemples, il faudra taper exactement ce qui est proposé, en faisant bien attention ! Une fois la compilation effectuée, on peut

- ▶ visualiser le document à l'écran ;
- ▶ imprimer le document ;
- ▶ convertir le document en fichier PostScript ;
- ▶ et plein d'autres choses.

```
1 Mes premiers pas avec \LaTeX{} sont
2 un peu \emouvants. Pour les premiers
3 exemples, il faudra taper exactement
4 ce qui est propos\e, en faisant bien
5 attention ! Une
6 fois la compilation effectu\ee,
7 on peut
8 \begin{itemize}
9 \item visualiser le document
10 \a l'\ecran ;
11 \item imprimer le document ;
12 \item convertir le document
13 en fichier PostScript ;
14 \item et plein d'autres choses.
15 \end{itemize}
```



On voit que les coupures de lignes n'ont pas du tout lieu au même endroit que dans le source et que les lignes produites sont justifiées (alignement au niveau des marges gauche et droite).

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



On voit que les coupures de lignes n'ont pas du tout lieu au même endroit que dans le source et que les lignes produites sont justifiées (alignement au niveau des marges gauche et droite).

T_EX calcule automatiquement les endroits où doivent se situer les coupures et procède, le cas échéant, aux césures de mots.



On voit que les coupures de lignes n'ont pas du tout lieu au même endroit que dans le source et que les lignes produites sont justifiées (alignement au niveau des marges gauche et droite).

T_EX calcule automatiquement les endroits où doivent se situer les coupures et procède, le cas échéant, aux césures de mots. D'autre part, une fin de ligne au niveau du source est comprise comme un espace.



On notera également que la première ligne du paragraphe est indentée (retrait vers la droite de la première ligne) alors que le source ne demandait rien. Il s'agit de la présentation par défaut des paragraphes (en typographie française). On verra au chapitre “ commandes de base : Fontes, ... ” comment spécifier les formes que devront avoir les paragraphes.



Le terme " $\LaTeX\{\}$ ", dans le source, a été traduit par " L^AT_EX " dans le document. De façon générale, tout ce qui commence par une *contre-oblique* est une *macro* (macro-instruction, une sorte de commande en fait). Ici, le nom de cette commande est LaTeX (notez l'emploi des majuscules et minuscules, par exemple, la commande latex n'existe pas). Il s'agit d'une suite d'instructions permettant d'obtenir le curieux logo avec ses lettres qui ne sont pas alignées verticalement.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Le gros bloc :

```
\begin{itemize}
  \item visualiser le document à l'écran ;
  \item imprimer le document ;
  \item convertir le document en fichier PostScript ;
  \item et plein d'autres choses.
\end{itemize}
```

Premiers
documents

Structure d'un fichier
L^AT_EX

Entête, préambule ou
header

Le document

En résumé

Etude du résultat

Coûres

Paragraphes et
indentation

Macro commandes

Listes

Typographie

Marges et format de
papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Le gros bloc :

```
\begin{itemize}
  \item visualiser le document à l'écran ;
  \item imprimer le document ;
  \item convertir le document en fichier PostScript ;
  \item et plein d'autres choses.
\end{itemize}
```

a été traduit par une liste d'items.

Premiers
documents

Structure d'un fichier
L^AT_EX

Entête, préambule ou
header

Le document

En résumé

Etude du résultat

Coûres

Paragraphes et
indentation

Macro commandes

Listes

Typographie

Marges et format de
papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Le gros bloc :

```
\begin{itemize}
  \item visualiser le document à l'écran ;
  \item imprimer le document ;
  \item convertir le document en fichier PostScript ;
  \item et plein d'autres choses.
\end{itemize}
```

a été traduit par une liste d'items.

On notera la syntaxe qui permet de déclarer où commence la liste d'items (`\begin{itemize}`) et où elle se termine (`\end{itemize}`).

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Le gros bloc :

```
\begin{itemize}
  \item visualiser le document à l'écran ;
  \item imprimer le document ;
  \item convertir le document en fichier PostScript ;
  \item et plein d'autres choses.
\end{itemize}
```

a été traduit par une liste d'items.

On notera la syntaxe qui permet de déclarer où commence la liste d'items (`\begin{itemize}`) et où elle se termine (`\end{itemize}`).

Ces blocs qui commencent par un `\begin{bidule}` et qui se terminent par un `\end{bidule}` s'appellent des *environnements*.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Voilà pour la partie visible. Maintenant, regardez avec beaucoup d'attention le point d'exclamation et les points virgules. Avec de bons yeux, on remarque que l'espace située¹ avant la double ponctuation est plus étroite que celle mise après. Il s'agit d'une règle de typographie française, l'espace située avant étant en fait une espace fine insécable. C'est l'appel de `\usepackage[frenchb]{babel}` qui a déclenché automatiquement ce comportement.



¹Les espaces horizontales sont des mots féminins en typographie.

Dans le même registre, chaque item de la liste débute par un tiret long (un tiret demi cadratin pour être précis). Sans la précision `\usepackage[frenchb]{babel}`, le signe au début de chaque item aurait été un rond noir (\bullet)².

²Il s'agit d'une règle beaucoup plus controversée. L'auteur du package est un bon de la mettre !

On peut remarquer également, et ce n'est pas très esthétique, que les marges sont très importantes et que le numéro de page est situé très haut dans la page. Ce comportement malheureux a deux origines. La taille des marges par défaut est de un pouce (environ 2,54 cm) ; c'est K_NUTH qui en a décidé ainsi et K_NUTH aime bien les marges importantes. D'autre part, T_EX a été créé par un américain qui devait écrire des textes sur du papier américain. Or, le format standard outre-atlantique est " letter " alors qu'en France, il s'agit du format A4. Le document a été créé en supposant que le papier était du type letter ce qui explique la position bizarre du numéro de page (il suffit de couper la feuille au format letter et on retrouve quelque chose de plus propre !) Bien évidemment, ce comportement peut être réglé.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

Pour le moment, on indiquera qu'on travaille sur du papier au format A4 en spécifiant un paramètre optionnel au niveau de la commande `\documentclass`. En fait, j'avais menti précédemment en disant que tous nos documents commenceraient par cette commande ! En réalité, ils commenceront par la commande :

```
\documentclass[a4paper]{classe_de_document}
```



Petite cerise sur le gâteau : examinez avec attention le mot " effectuée " comme il apparaît sur le document et le mot " effectuée " tel qu'il apparaîtrait sur des traitements de texte peu scrupuleux. Dans le premier cas, les deux " f " ont été liés : il s'agit d'une ligature. Un texte correctement composé doit faire les ligatures ff, fi, fl pour ff, fi et fl afin d'éviter les chocs disgracieux entre ces lettres. T_EX gère également d'autres ligatures permettant d'obtenir les différents types de tirets :

" - " (trait d'union) obtenu avec " - " ;

" – " (tiret demi cadratin) obtenu avec " -- " ;

" — " (tiret cadratin) obtenu avec " --- " .



Ce premier exemple a permis de présenter le processus général pour obtenir un document et de voir les notions de classe de document, d'extension (ou package), de préambule, de macro (ou commande) et d'environnement.



Un dernier mot pour spécifier ce qu'est exactement un nom de macro et pourquoi les accolades qui suivent `\LaTeX` n'ont pas été traduites par des accolades au niveau du document final.



Un dernier mot pour spécifier ce qu'est exactement un nom de macro et pourquoi les accolades qui suivent `\LaTeX` n'ont pas été traduites par des accolades au niveau du document final. L^AT_EX permet d'écrire n'importe quel caractère au niveau du source (les caractères accentués nécessitant quand même la présence d'un `\usepackage[T1]{fontenc}` au niveau du préambule) sauf les dix caractères suivants qui ont chacun un rôle très particulier.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Un dernier mot pour spécifier ce qu'est exactement un nom de macro et pourquoi les accolades qui suivent `\LaTeX` n'ont pas été traduites par des accolades au niveau du document final. L^AT_EX permet d'écrire n'importe quel caractère au niveau du source (les caractères accentués nécessitant quand même la présence d'un `\usepackage[T1]{fontenc}` au niveau du préambule) sauf les dix caractères suivants qui ont chacun un rôle très particulier.

Une utilisation erronée de ces caractères produira invariablement des erreurs lors de la compilation et des résultats surprenants au niveau de la sortie.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



Caractères réservés

\	début de nom de macro
%	commentaire
~	espace insécable
{	début de groupe
}	fin de groupe
\$	mode mathématique
_	indice
^	exposant
#	paramètre de macro
&	colonne de tableau

Les cinq symboles du bas ne seront vus que plus tard. Nous allons terminer cette présentation rapide en expliquant sommairement la signification des cinq symboles supérieurs.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

`\` sert à indiquer une macro, les caractères qui suivent donnant le nom de cette macro.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragrophes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



`\` sert à indiquer une macro, les caractères qui suivent donnant le nom de cette macro.

T_EX a des règles qui peuvent sembler bizarres pour déterminer ce qu'est un nom de macro.

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple



`\` sert à indiquer une macro, les caractères qui suivent donnant le nom de cette macro.

T_EX a des règles qui peuvent sembler bizarres pour déterminer ce qu'est un nom de macro.

Une série de lettres (majuscules et/ou minuscules) qui se suivent déterminent un nom de macro,



`\` sert à indiquer une macro, les caractères qui suivent donnant le nom de cette macro.

T_EX a des règles qui peuvent sembler bizarres pour déterminer ce qu'est un nom de macro.

Une série de lettres (majuscules et/ou minuscules) qui se suivent déterminent un nom de macro, un caractère autre qu'une lettre (y compris un des 10 symboles spéciaux) détermine *seul* un nom de macro.



Voici quelques exemples pour bien comprendre :

`\bonjour` est la macro `bonjour` ;

`\\bonjour` est la macro `\` suivie du texte " `bonjour` " ;

`\\\bonjour` est la macro `\` suivie de la macro `bonjour` ;

`_` où `_` désigne un espace est la macro `_` (le nom est
constitué par un espace : bizarre n'est-ce pas ?) ;

`\formatA4` est la macro `formatA` suivie du texte " `4` ".

On notera le piège du dernier exemple (les chiffres ne sont pas
des lettres).



Le caractère % introduit un commentaire, c'est-à-dire un texte apparaissant dans le source mais n'ayant aucune influence au niveau du document³. Un commentaire se termine à la fin d'une ligne du source (voir l'exemple suivant pour mieux comprendre). Les commentaires sont particulièrement utiles pour mettre des notes explicatives au niveau du source sans que celles-ci n'influencent le résultat.

³Ce n'est pas tout à fait vrai mais c'est une bonne approximation de la vérité.

Le caractère `~` permet de placer une espace insécable. Les espaces insécables sont obligatoires entre certains mots. Par exemple entre un prénom et un nom de famille (Yves~Delhaye), entre certaines abréviations et le numéro qui va avec (page~4, Cf.~6.4, etc.).



Groupes

Les caractères `{` et `}` sont très importants sous `TeX`. Ils délimitent des *groupes*. Expliquer en détails ce qu'est un groupe sortirait du cadre de ce stage. Pour simplifier à outrance, disons qu'un groupe délimite une zone à l'intérieur de laquelle on peut obtenir des comportements qui n'existeront plus à l'extérieur de ce groupe. Dans le source on a employé la construction `\LaTeX{}`, c'est-à-dire la macro `LaTeX` suivie d'un groupe vide. Sans ce groupe vide, le source :

Mes premiers pas avec `\LaTeX` sont un peu émouvants
aurait produit la sortie suivante :

Mes premiers pas avec `LATEX` sont un peu émouvants
car l'espace qui suit la macro `LaTeX` ne sert qu'à indiquer que le nom de la macro se termine et cet espace disparaît une fois ce rôle terminé. C'est heureux, sinon il serait difficile d'écrire quelque chose comme " Le `LATEX`book " qui a bien le droit d'exister après tout ! La première idée serait de mettre deux

Voici le second exemple qui va tenter d'illustrer tout ce qui vient d'être vu. À partir de maintenant, les sources ne présenteront pas le préambule mais il faudra absolument le spécifier. Le source sera présenté sur la droite de la page, le résultat correspondant étant présenté à gauche en vis-à-vis. Pour faciliter les explications, les lignes du source seront numérotées.



Le second exemple de source \LaTeX va permettre d'illustrer les points que nous venons de voir. Dans les exemples, nous avons vu la macro dont le nom était constitué uniquement d'un espace. Ce source \LaTeX en donne une utilisation possible.

Cet exemple va présenter les caractéristiques telles que la gestion des espaces, les commentaires, les espaces insécables et une toute petite illustration de ce qu'il est possible de faire avec les groupes.

1 Le second exemple de
2 source $\{\text{\LaTeX}\}$ va
3 permettre d' illustrer
4 les points que nous
5 venons de voir . Dans
6 les exemples, nous
7 avons vu la macro dont le
8 nom \backslash 'était constitué \backslash 'e uniquement
9 d'un espace. Ce source $\backslash\text{\LaTeX}\backslash$ e
10 donne une utilisation possible .

11
12 Cet exemple va pr \backslash 'esenter les
13 caract \backslash 'eristiques telles que
14 la
gestion des
espaces,

15 les commentaires, *% comme celu*
16 les espaces ins \backslash 'ecables et
17 une toute petite illustration de
18 qu'il est possible de faire avec



On notera les deux façons supplémentaires d'écrire le mot " L^AT_EX " en mettant une espace à la suite, le fait que le commentaire " comme celui-ci " (ligne 15) n'apparaît pas au niveau de la sortie, et les ribambelles d'espaces au niveau du source qui n'ont donné qu'une seule espace au niveau du document final (lignes 14 à 19).

Une ligne vide au niveau du source indique un saut de paragraphe. Ici, pour des raisons de commodités, les paragraphes ne présentent ni indentation, ni espaces verticaux entre eux : dans un exemple réel, ils seraient présents.



Le troisième exemple de source \LaTeX va permettre d'illustrer les points que nous venons de voir. Dans les exemples, nous avons vu la macro dont le nom était constitué uniquement d'un espace. Ce source \LaTeX en donne une utilisation possible.

1 Le troisi \ 'eme exemple de
2 source $\{\LaTeX\}$ va
3 permettre d' illustrer les points
4 que nous venons de voir . Dans
5 les exemples, nous avons vu
6 la macro dont le nom \ 'etait
7 constitu \ 'e uniquement d'un espace
8 source \LaTeX en donne une utilis
9 possible .



Le troisième exemple montre la puissance de T_EX. Le premier paragraphe a été repris tel quel sauf qu'on a placé une espace insécable entre deux mots (`source~\LaTeX` en ligne 8). T_EX a alors repris tous les calculs de coupure de ligne afin de présenter un résultat correct. Si les mots " source L^AT_EX " avaient été déplacés sur la quatrième ligne du paragraphe, les lignes précédentes auraient été trop serrées (elles auraient présenté des espaces trop étroites entre les mots). T_EX a alors complètement revu les coupures de lignes, dès la troisième ligne dans l'exemple, en élargissant un peu plus les lignes précédentes. Pour T_EX, il était préférable de rendre les lignes un peu plus lâches que de les serrer. Les calculs entrant en jeu sont très compliqués et assurent d'avoir un résultat proche de celui obtenu de la part d'un typographe professionnel.



Ces deux derniers exemples montrent qu'un source \LaTeX n'a rien de vraiment compliqué lorsque la présentation voulue n'appelle pas de constructions spéciales. Pour obtenir des résultats non standard, il va falloir faire appel à des macros et des environnements. La suite de ce manuel se charge de cette présentation. Le logiciel \TeX connaît environ 300 commandes qu'on appelle des *primitives* car elles sont construites à l'intérieur même du logiciel. Le format Plain \TeX (une bibliothèque de macros incontournables) en déclare environ 300 autres : ces 600 commandes sont la base d'un travail sous \TeX . \LaTeX , en tant que surcouche de \TeX , en définit très approximativement autant que \TeX et Plain \TeX réunis ce qui fait qu'au final, un utilisateur dispose de plus de 1 000 commandes pour indiquer ce qu'il veut faire. Ce nombre est assez colossal et apprendre par cœur la syntaxe et le rôle de ce millier de commandes est une tâche aussi ardue qu'inutile (sauf si vous vous destinez à devenir gourou de \LaTeX , et encore ...).

En premier lieu, un utilisateur lambda n'a pas besoin, loin s'en faut, de toutes ces commandes (ou macros). La plupart du temps, un document n'en nécessite qu'un petit nombre et, normalement, est constitué pour l'essentiel de texte brut. Il est clair que, pour l'essentiel, les macros et environnements utilisés sont très usuels et qu'ils sont connus après peu de temps de pratique. Il n'en reste pas moins que leur apprentissage nécessite un petit effort et qu'il faut avoir de la documentation en permanence lorsqu'on travaille sous L^AT_EX afin de gérer les situations exceptionnelles. Nous allons voir que les macros ont des noms qui permettent souvent de bien s'en souvenir (pour peu que l'anglais ne soit pas complètement inconnu).

Premiers documents

Structure d'un fichier L^AT_EX

Entête, préambule ou header

Le document

En résumé

Etude du résultat

Césures

Paragraphes et indentation

Macro commandes

Listes

Typographie

Marges et format de papier

Ligature

En résumé

Caractères réservés

Deuxième exemple

Troisième exemple

