

L^AT_EX, les unités & les longueurs

Ou comment manipuler les distances sous L^AT_EX

Bertrand Masson

23 janvier 2010

- 1 Introduction
- 2 Unités : définition
 - Unités absolues
 - Unités relatives
 - Le cadratin = em
 - L'œil = ex
- 3 Système propre à L^AT_EX
 - `\textwidth`
 - `\textheight`
 - `\parindent`
 - Longueurs diverses
- 4 Utilisation
- 5 Manipuler les longueurs
 - Modifier les longueurs
 - Augmenter les longueurs
 - Créer ses propres longueurs
- 6 Les longueurs élastiques
- 7 Conclusion

Introduction

Pourquoi vouloir manipuler les longueurs sous L^AT_EX ?

Je pense qu'il faut le faire le moins souvent possible. Ajouter ou modifier les espaces entre les paragraphes ou les illustrations diminuer la taille des marges, . . . est très souvent une mauvaise idées héritée des traitements de texte WYSIWYG. L^AT_EX se débrouille très bien tout seul, avec la mise en page, après tout c'est son boulot.

M'enfin on ne se refait pas et il arrive toujours un moment où l'on veut mettre son grain de sel dans cette belle organisation, ne serait-ce que pour tracer un trait, mettre une illustration au centre de la page, espacer des colonnes, supprimer une indentation, que sais-je encore ?

Cette fiche te donnera quelque renseignements pour manipuler les longueurs sous L^AT_EX.

Attention les commandes de longueur sont très nombreuses sous L^AT_EX. Il en existe pour gérer la mise en page, les tableaux, les boîtes, les illustrations, . . . Ces longueurs ne seront pas abordées ici mais dans des fiches spécifiques, consacrés à ces objets.

Définitions

Sous L^AT_EX il existe deux types d'unité :

Les unités absolues : ces unités ont toujours la même valeur quelque soit la forme du document et sont indépendantes de L^AT_EX. Par exemple le point, le centimètre, le millimètre.

Les unités relative : ces unités dépendent des caractéristiques de ton documents. Par exemple le cadratin, qui représente la largeur de la lettre M dans la police courante, dépend bien évidemment la police utilisée. `\textwidth` qui représente la longueur du texte dépend de la mise en page de ton document.

Unités absolues ou unités relatives, *that the question?*

Je ne sais pas si il y a une réponse définitive, tout ce que je peux te répondre, c'est ce que je fais.

En gros pour tout ce qui concerne le texte, comme la taille d'un saut de ligne ou d'une indentation, la taille d'un tableau et de ses colonnes, j'utilise les distances relatives, car si, par exemple, je change le type de police ou la dimension des marges, les distances ainsi définies vont évoluer en fonction de ces nouvelles données et les proportions recherchées vont être conservées.

Par contre pour tout ce qui concerne la page, comme placement d'objet, la taille des marges, . . . , j'utilise les unités absolues. Le format de page A4 étant défini en centimètre (21x29,7cm) je me repère mieux dans la page en utilisant les centimètres. De même pour la taille des photos, un format de 9x13cm est plus facile à se représenter que `0.8\textwidth`.

Unités absolues

Je ne vais pas te décrire la totalité des unités prise en charge par L^AT_EX, car elles sont nombreuses et je doute, que par exemple, tu aies besoin d'utiliser le cicéro qui vaut 12 didots, lui-même valant 1,07pt. Mais si tu as besoin d'unités exotiques ou typographiques, une recherche sur internet, devrait répondre à tes questions.

L'unité de base de L^AT_EX est une unité typographique le point noté pt et qui vaut 0,35 mm. Moi je préfère utiliser le centimètre (cm) et le millimètre (mm), beaucoup plus parlant pour nous enfants de la révolution de 1789.
 $2,54 \text{ cm} = 72,27 \text{ pt}$.

Unités relatives

Même remarque que pour les unités absolues elles sont nombreuses et je ne vais te décrire que les principales que j'utilise.

Le cadratin

Le cadratin (em), qui représente la largeur de la lettre M dans la police courante, mais qui, dans les faits, équivaut à la largeur des corps. Par exemple dans un texte tapé en corps 11, un cadratin mesurera 11 points.

Le tiret long ou tiret cadratin — que l'on écrit en L^AT_EX « --- » et que tu utilises si tu veux retranscrire un dialogue, vaut 1em.

Le tiret court (demi-cadratin) – que l'on écrit en L^AT_EX « -- » et qui est utilisé dans les listes, vaut 0.5em.

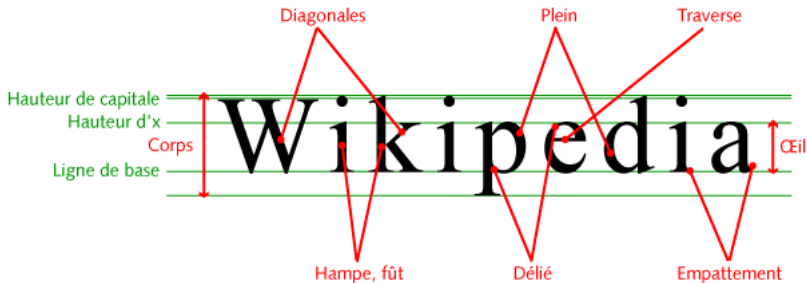
Le quart de cadratin (0.25em) - que l'on écrit en L^AT_EX « - » qui est utilisé pour les césure ou les mots ou noms composés.

Le signe moins – s'écrit en L^AT_EX « \$-\$ ». Le \$ permet de passer en mode mathématique.

L'espace fine qui ce met devant les signes doubles comme : ; ! ou ? s'écrit « \, » en LaTeX et vaut un quart de quadratin.

L'œil

Cette unité, notée « ex » correspond à la hauteur du x bas-de-casse. Voici un petit schéma emprunté à Wikipédia pour mieux comprendre.



Je l'utilise essentiellement pour définir des hauteurs, comme par exemple l'espace entre deux lignes `\vspace{1.5ex}`, c'est à dire, égal à 150% de la hauteur du x bas-de-casse, ou l'épaisseur d'un trait `\rule{5cm}{0.5ex}`.

Système propre à L^AT_EX

`\textheight`, qui représente la hauteur du texte dans la page. J'utilise cette unité pour calculer la taille des images pleine page pour qu'elle ne déborde pas de la page de la façon suivante :

```
\includegraphics[height=0.8\textheight]{monImage}
```

Le trait ci-dessous (`\rule{\textheight}{0.5ex}`) représente la valeur de `\textheight` pour ce document.



Tu t'attendais peut-être à un trait vertical, mais `\textheight` représente une longueur pas un vecteur. Nous apprendrons à tracer des trait verticaux dans une autre fiche.

Système propre à L^AT_EX

Trois autres longueurs de L^AT_EX peuvent peut-être t'être utile :

- `\parindent` qui est la dimension de l'alinéa (indentation en début de paragraphe), prédéfinie à 15pt
`\rule{\parindent}{0.5ex} =`
Tu ne vois aucun trait ! c'est normal dans ce document il n'y a pas d'alinéa, la valeur de `\parindent = 0pt`
- `\baselineskip` représente la distance entre la base de la ligne et la base de la ligne suivante (10.95pt dans ce document)
`\rule{\baselineskip}{0.5ex} =` **—**
- `\parskip` qui correspond à la distance séparant deux paragraphes.

Longueurs diverses

Voici un tableau récapitulant les longueurs utilisées pour la mise en page :

`\oddsidemargin` Marge pour les pages impaires

`\evensidemargin` Marge pour les pages paires

`\topmargin` Marge de haut de page

`\headheight` Hauteur de l'entête de page

`\headsep` Distance entre l'entête de page et le corps de page


`\marginparsep` Distance entre la colonne des notes de marge et le corps de texte

`\marginparwidth` Largeur de la colonne des notes de marge


`\footskip` Distance entre la fin du corps de texte et le bas de page


Utilisation des longueurs L^AT_EX


Toutes les longueurs que l'on vient de voir peuvent servir à exprimer une dimension dans toutes les commandes qui utilisent des dimensions comme la taille d'un trait, la taille d'une image, la largeur de la colonne d'un tableau, la longueur d'un espace,...

Tu peux l'utiliser telle quelle : `\rule{\baselineskip}{0.5ex}` = 

ou à l'aide de multiple

la moitié `\rule{0.5\baselineskip}{0.5ex}` = 

le double `\rule{2\baselineskip}{0.5ex}` = 

n'importe quoi `\rule{3.78\baselineskip}{0.5ex}` = 

Tu peux utiliser des valeurs négatives. Ces valeurs servent pour déporter un objet dans la marge, à l'aide par exemple de la commande `\hspace` qui crée un espace horizontal. Observe ci-dessous l'effet de la commande suivante :

`abc\hspace{-5\baselineskip}def`

abc

Tu ne peux pas additionner, par cette méthode, des longueurs

`\hspace{\baselineskip+3cm}` produit une erreur. Il existe une commande pour faire cela que nous verrons dans les pages suivantes.

Modifier les longueurs

Tu peux modifier les longueurs pré-définies par L^AT_EX par la commande `\setlength{\nomLongueur}{valeurUnité}`

Par exemple `\setlength{\parskip}{1cm}` séparera chaque paragraphe d'un centimètre.

`\setlength{\baselineskip}{2\baselineskip}` produira un double interligne.

`\setlength{\parindent}{0pt}` supprimera tous les alinéas suivant la commande. **Attention même pour une valeur nulle il faut préciser l'unité.**

Petit rappel sur la portée d'une commande

L'exécution d'une commande commence à partir de l'endroit où est placée cette commande, et se poursuit jusqu'à la fin du document ou une autre commande annulant ou modifiant son action.

Si la commande est placée avant le `\begin{document}` elle affectera tout le document (`\setlength{\parindent}{0pt}` avant `\begin{document}` supprimera tous les alinéas dans l'ensemble du document).

Tu peux limiter la portée d'une commande en l'enserrant entre `{}`. Par exemple

```
{\setlength{\baselineskip}{2\baselineskip} le texte qui doit être affecté par le double interligne}
```

Augmenter les longueurs

La commande `\addtolength{\nomLongueur}{valeurUnité}` modifie une longueur en y ajoutant ou retranchant (si nombre négatif) une valeur.

Par exemple `\addtolength{\parskip}{20pt}` ajoute 20pt au saut de paragraphe.

`\addtolength{\parindent}{\baselineskip}` ajoute la valeur de l'interligne à l'alinéa.

Attention ce n'est pas une véritable addition car on ne peut obtenir une nouvelle dimension correspondant à la somme de 2 autres.

Créer ses propres longueurs

C'est très facile il suffit de placer dans le préambule de ton document (avant le `\begin{document}`) les lignes suivantes.

`\newlength{\nomLongueur}` qui crée la nouvelle longueur

`\setlength{\nomLongueur}{valeurUnité}` qui lui attribue une valeur

Voici deux exemples une première longueur appelée « échelle » qui vaut 5 cm et une seconde, appelée « truc » qui vaut 2 fois la taille de l'interligne

(**Attention tu ne peux pas mettre de nombre dans un nom de commande « échelle5cm » est à proscrire**) :

```
\newlength{\echelle}
```


```
\setlength{\echelle}{5cm}
```

```
\newlength{\truc}
```

```
\setlength{\truc}{2\baselineskip}
```

Tu les utilises de la même façon que les autres longueurs :

```
\rule{\echelle}{0.5ex} = 
```

```
\rule{2\truc}{0.5ex} = 
```

Attention

La distance de `\echelle` sur l'écran, n'est pas bonne et dépend de la définition de celui-ci. Pour rappel la dimension d'une frame dans beamer est de 128x96mm.

Une astuce pour additionner des longueurs

Il suffit de créer une nouvelle longueur, de lui attribuer la première valeur puis de l'augmenter de la seconde. Par exemple pour avoir une longueur valant la somme des marges de droite et de gauche.

```
\newlength{\marges}
```

```
\setlength{\marges}{\oddsidemargin}
```

```
\addtolength{\marges}{\evensidemargin}
```

Je ne sais pas si c'est très utile, mais tu peux le faire.

Les longueurs élastiques

Ce sont des longueurs qui peuvent augmenter ou diminuer selon les cas, mais qui prendront toute la place disponible. Un petit exemple sera plus clair. Par exemple `\hfill` introduit un espace horizontal jusqu'à la fin de la ligne. Voici deux exemples pour te montrer le comportement de `\hfill`

Un texte `\hfill` un autre texte.

Un texte `\hfill` un autre texte beaucoup plus long.

Et leur résultat :

Un texte un autre texte.

Un texte un autre texte beaucoup plus long.

`\hrulefill` a le même rôle que `\hfill`, mais il remplace les blancs par une ligne. Un exemple : Un texte `\hrulefill` un autre texte.

Un texte _____un autre texte.

Un autre exemple :

`\hrulefill` Texte entouré de lignes horizontales et centré`\hrulefill`

_____Texte entouré de lignes horizontales et centré_____

Les longueurs élastiques, suite

Dans le même esprit on trouve :

- des pointillés $A \dotfill B = A \dots\dots\dots B$
- une flèche à droite $A \rightarrowfill B = A \longrightarrow B$
- une flèche à gauche $A \leftarrowfill B = A \longleftarrow B$
- des symboles $A \dingfill{47} B = A \text{ ☎ ☎ ☎ ☎ ☎ ☎ ☎ ☎ } B$

`\dingfill` provient du package `pifont` que l'on a déjà évoqué dans la fiche sur les listes, en conséquence je t'y renvoie pour la liste des symboles. Pour l'utiliser tu dois charger le package `pifont` (`\usepackage{pifont}`)

Tu peux utiliser les longueurs élastiques pour diviser une ligne en parties égales :

`A \rightarrowfill B \dotfill C \leftarrowfill D \dingfill{49} C`
 $A \longrightarrow B \dots\dots\dots C \longleftarrow D \text{ ☎ ☎ ☎ } C$
`A \hrulefill{} B \hrulefill{} C`
 $A \text{-----} B \text{-----} C$

Attention à l'espace

Les deux commandes suivantes ne sont pas exactement identiques

A `\hrulefill` B et

A `\hrulefill` B comme tu peux le constater ci-dessous :

A _____ B

A _____ B

La différence est due à la présence ou non de l'espace entre le A et la commande `\hrulefill`. Je te rappelle que l'espace après une commande est mangé par L^AT_EX car elle est considérée comme un marqueur de fin de commande, alors que celle du début entre le A et la commande est prise en compte. Si tu ne veux pas que le trait colle au B il faut écrire :

A `\hrulefill{}` B

A _____ B

C'est fini

Voilà, j'espère que cette fiche te permettra de te sentir à l'aise avec les longueurs. Un petit conseil, n'en profite pas pour ajouter des espaces un peu partout et de modifier trop les longueurs par défaut de L^AT_EX tu risquerais d'enlaidir ou de déséquilibrer ton document.

Bébert