

L^AT_EX & les tableaux

Première partie : les tableaux simples avec l'environnement `tabular`

Bertrand Masson

Les fiches de Bébert

17 avril 2011

- 1 Ton premier tableau
- 2 Les colonnes
- 3 Les lignes
- 4 1^{er} problème
 - Taille des tableaux
 - La solution : $p\{\text{valeurUnité}\}$
- 5 $*\{\text{nb}\}\{\text{type}\}$
- 6 $@\{\text{texte}\}$
- 7 Traits horizontaux partiels
- 8 Espace entre les colonnes
- 9 Épaisseur des traits
- 10 Espace entre 2 traits
- 11 Espace entre 2 lignes
- 12 Espace entre 2 lignes
- 13 Fusion de colonnes
- 14 Conclusion

Ton premier tableau

La réalisation de tableaux se fait au moyen de l'environnement **tabular**. On n'a pas besoin de charger un package, c'est compris dans L^AT_EX. Un exemple valant mieux qu'un long discours, voici un tableau, les explications suivent :

```
\begin{tabular}{|l|c||rc||}
```

```
\hline
```

```
Première ligne & colonne 2 & colonne 3 & colonne 4 \\\
```

```
\hline \hline
```

```
aligné à gauche & centré & aligné à droite & 4 \\\
```

```
ligne3 &&colonne 3& 4 \\\
```

```
\hline
```

```
\end{tabular}
```

Le résultat

Première ligne	colonne 2	colonne 3	colonne 4
aligné à gauche	centré	aligné à droite	4
ligne3		colonne 3	4

Créer les colonnes

Comme tous les environnements L^AT_EX tu commences par un `\begin{tabular}` et tu finis par un `\end{tabular}`.

Maintenant il te faut définir le nombre et le type de colonnes, c'est le rôle de la commande `{|l|c|r|}` qui suit le `\begin{tabular}`.

Dans cet exemple tu crées 3 colonnes : avec `l`, pour left, donc la première colonne est justifiée gauche, `c` (center) une 2^e colonne centrée, `r` (right) une 3^e colonne justifiée droite.

Les `|` (*pipe* en anglais) obtenus par AltGr 6 dessinent des traits verticaux séparant les colonnes. Ils ne sont pas obligatoire, dans ce cas, pas de lignes verticales et plusieurs accolés donnent des lignes verticales multiples. `{lcr}` donnera un tableau sans filets, `{|l|c||r|}` dans ce cas, les 2^e et 3^e colonnes sont séparées par un double trait vertical.

Créer les lignes

`\hline` trace une ligne horizontale sur toute la largeur du tableau. Il se comporte comme les `|`.

```
textecolonne1 & textecolonne2 & textecolonne3 &  
textecolonne4 \\
```

Permet de remplir une ligne du tableau. `&` sépare les colonnes et le `\\` final indique la fin de la ligne.

Attention il doit y avoir `nombreDeColonne - 1 &` (il n'y en a pas ni au début de la ligne, ni à la fin) sous peine d'erreurs de compilation. Une cellule vide est noté `&&` ou `& &`, comme la 3^e ligne 2^e colonne de notre premier exemple.

Remarques, que les espaces juste avant et juste après les `&` sont purement inutiles, mais ils facilitent la lecture. `&un mot&` et `& un mot &` on le même résultat, les espaces ajoutées, n'ont aucune influence sur l'espacement entre le texte et les filets.

Taille des tableaux

La première difficulté que tu vas rencontrer, concerne la taille du tableau. En effet `tabular` dimensionne la taille de la colonne à la taille de la plus grande ligne et très vite on déborde de la page, d'autant plus que `{lcr}` ne reconnaît pas les sauts de ligne.

```
\begin{tabular}{|l|l|l|}
```

```
\hline
```

```
Première ligne longue, très
```

```
longue, trop longue & colonne 2 \\
```

```
\hline
```

```
ligne2 & rien \\
```

```
\hline
```

```
\end{tabular}
```

Première ligne longue, très longue, trop longue	colonne 2
ligne2	rien

Même la commande `\par` n'est pas reconnue et retourne une erreur de compilation.

La solution : `p{valeurUnité}`

Cette commande permet de fixer la largeur de la colonne. Elle reconnaît bien entendu toutes les unités comprises par L^AT_EX (voir fiche Unités & longueurs). Le même tableau que précédemment avec une première colonne fixée à 3 cm.

```
\begin{tabular}{|p{3cm}|l|}
\hline
Première ligne longue, très longue, trop longue & colonne 2 \\
\hline
ligne2 & rien \\
\hline
\end{tabular}
```

Première longue, très longue, trop longue	colonne 2
ligne2	rien

La solution : `p{valeurUnité}`

Cette commande ne produit que des colonnes justifiées, ce qui peut avoir comme dans l'exemple précédent des effets désastreux. Tu as plusieurs façons de remédier à cela : l'utilisation de la commande `\par` (saut de paragraphe), ou les commandes `\centering` `\raggedright` et `\raggedleft`.

```
\begin{tabular}{|p{3cm}|l|}
\hline
Première ligne longue,\par très longue,
\par trop longue& colonne 2 \\
\hline
ligne2 &rien \\
\hline
\end{tabular}
```

Première ligne longue, très longue, trop longue	colonne 2
ligne2	rien

```
\begin{tabular}{|p{3cm}|l|}
\hline
\centering Première ligne longue, très longue,
trop longue& colonne 2 \\
\hline
ligne2 &rien \\
\hline
\end{tabular}
```

Première ligne longue, très longue, trop longue	colonne 2
ligne2	rien

Je te laisse essayer les commandes `\raggedright` et `\raggedleft`.

Un raccourci

Pour éviter de taper plusieurs fois la même définition de colonne, il existe la commande `*{nombreColonnes}{formatColonne}`.

`{|*{5}{p{2cm}}|}` est équivalent à

`{|p{2cm}|p{2cm}|p{2cm}|p{2cm}|p{2cm}|}`

`{|*{3}{c}|*{2}{p{4cm}}|1|*{3}{p{2cm}}|}` est équivalent à

`{|c|c|c|p{4cm}|p{4cm}|1|p{2cm}|p{2cm}|p{2cm}|}`

Remplacer les traits verticaux : @{texte}

La commande `@{texte}` permet de remplacer les lignes verticales par la valeur de `texte`. Voici un exemple où la séparation entre les 2^e et 3^e colonne est constituée d'une flèche `\rightarrow` = \rightarrow et la séparation entre les 3^e et 4^e est un téléphone `\ding{37}` = ☎ (`\usepackage{pifont}`).

```
\begin{tabular}{|l@{ $\rightarrow$ }c@{\ding{37}}l|}
\hline
Robert Bidochon&RB&03.03.03.01\\\hline
Ginette Lacaille&GL&02.02.02.01\\\hline
\end{tabular}
```

Robert Bidochon	\rightarrow	RB	☎	03.03.03.01
-----------------	---------------	----	---	-------------

Ginette Lacaille	\rightarrow	GL	☎	02.02.02.01
------------------	---------------	----	---	-------------

Remplacer les traits verticaux : @{\texte}

Attention comme tu peux le remarquer sur l'exemple précédent, la commande @\texte supprime les espaces avant et après le trait de séparation.

```
\begin{tabular}{|l|l|l|}
\hline
Robert Bidochon&RB&03.03.03.03.01\\\hline
Ginette Lacaille&GL&02.02.02.02.01\\\hline
\end{tabular}
```

Robert Bidochon	RB	03.03.03.03.01
Ginette Lacaille	GL	02.02.02.02.01

```
\begin{tabular}{|l@{}c@{}l|}
\hline
Robert Bidochon&RB&03.03.03.03.01\\\hline
Ginette Lacaille&GL&02.02.02.02.01\\\hline
\end{tabular}
```

Robert Bidochon	RB	03.03.03.03.01
Ginette Lacaille	GL	02.02.02.02.01

L'avantage c'est que tu peux utiliser cette méthode pour aligner des chiffres sur la décimale :

```
\begin{tabular}{r@{,}l}
```

```
4&14159\\
```

```
+96&2\\
```

```
\hline
```

```
100&34159\\
```

```
\end{tabular}
```

```

      4,14159
    +96,2
  -----
   100,34159

```

Des traits horizontaux partiels = `\cline`

`\cline{colonneDébut-colonneFin}`, trace des lignes partielles :
`\cline{1-1}` trace une ligne uniquement dans la 1^{re} colonne,
`\cline{2-4}` du début de la colonne 2 à la fin de la colonne 4.

```
\begin{tabular}{|1|1|1|1|}
\hline
case \no1&case \no2&case \no3&case \no4\\
\cline{1-1}
case \no5&case \no6&case \no7&case \no8\\
\cline{2-4}
case \no9&case \no10&case \no11&case \no12\\
\cline{1-2}\cline{4-4}
case \no13&case \no14&case \no15&case \no16\\
\hline
\end{tabular}
```

case n° 1	case n° 2	case n° 3	case n° 4
case n° 5	case n° 6	case n° 7	case n° 8
case n° 9	case n° 10	case n° 11	case n° 12
case n° 13	case n° 14	case n° 15	case n° 16

Modifier l'espace entre les colonnes

`\tabcolsep` correspond à la valeur de l'espace avant ou après un | (donc la moitié de l'espace inter-colonne). Tu le modifies à l'aide des commandes `\setlength` ou `\addtolength` (voir la fiche sur « L^AT_EX, les unités & les longueurs »). Sa valeur par défaut est de 6 pt.

case 1	case 2	case 3
case 4	case 5	case 6

le même tableau avec une valeur de `\tabcolsep` de 12pt.

case 1	case 2	case 3
case 4	case 5	case 6

```
{\setlength{\tabcolsep}{12pt}
\begin{tabular}{|c|c|c|}
\hline
case 1&case 2&case 3\\ \hline
case 4&case 5&case 6\\ \hline
\end{tabular}}
```

Modifier l'épaisseur des traits

`\arrayrulewidth` correspond à l'épaisseur un `|`, d'un `\hline` ou d'un `\cline`. Tu le modifies à l'aide des commandes `\setlength` ou `\addtolength`. Sa valeur par défaut est de 0.4 pt. **Attention cette commande affecte toutes les lignes d'un tableau on verra dans une autre fiche comment ne modifier qu'une seule ligne.**

case 1	case 2	case 3
case 4	case 5	case 6

Le même tableau avec une valeur de `\arrayrulewidth` de 2pt.

case 1	case 2	case 3
case 4	case 5	case 6

```
{\setlength{\arrayrulewidth}{2pt}
\begin{tabular}{|c|c|c|}
\hline
case 1&case 2&case 3\\ \hline
case 4&case 5&case 6\\ \hline
\end{tabular}}
```

Modifier l'espacement entre 2 traits

Il te faut modifier la valeur de `\doublerulesep` pour écarter 2 `||` ou 2 `\hline` consécutifs. Sa valeur par défaut est de 2 pt. **Attention cette commande affecte toutes les lignes d'un tableau on verra dans une autre fiche comment ne modifier qu'une seule ligne.**

case 1		case 2		case 3
case 4		case 5		case 6

Le même tableau avec une valeur de `\doublerulesep` de 10pt.

case 1		case 2		case 3
case 4		case 5		case 6

```
{\setlength{\doublerulesep}{10pt}
\begin{tabular}{|c||c||c|}
\hline
case 1&case 2&case 3\\ \hline\hline
case 4&case 5&case 6\\ \hline
\end{tabular}}
```

Modifier l'espacement entre 2 lignes

Il te faut modifier la valeur de `\arraystretch` pour augmenter l'espacement entre 2 lignes. La valeur de `\arraystretch` est un coefficient multiplicateur : 0.5 diminue l'espace de moitié, 2.0 le double. Sa valeur par défaut est de 1.0. **Attention cette commande se modifie à l'aide de `renewcommand`.**

case 1	case 2	case 3
case 4	case 5	case 6

Le même tableau avec une valeur de `\arraystretch` de 2.0.

case 1	case 2	case 3
case 4	case 5	case 6

```
{\renewcommand{\arraystretch}{2}
\begin{tabular}{|c|c|c|}
\hline
case 1&case 2&case 3\\ \hline\hline
case 4&case 5&case 6\\ \hline
\end{tabular}}
```

Des espaces différents entre les lignes

Il suffit de mettre la dimension voulue entre [] à la fin des lignes concernées.

case 1	case 2	case 3
case 4	case 5	case 6
case 7	case 8	case 9
case 10	case 11	case 12

```
\begin{tabular}{|c|c|c|}  
\hline  
case 1&case 2&case 3\\ \hline  
case 4&case 5&case 6\\[0.5cm] \hline  
case 7&case 8&case 9\\[2em] \hline  
case 10&case 11&case 12\\ \hline  
\end{tabular}
```

Fusion de colonnes

La fusion de colonnes se fait avec la commande `\multicolumn{nombreCol}{alignement}{contenu}` où alignement vaut `l,c,r`. Tu n'es pas obligé de respecter le choix que tu as fait à l'initialisation des colonnes ; regarde le comportement des cellules n° 1 et n° 10.

```
\begin{tabular}{|l|l|l|l|}
\hline
\multicolumn{2}{|r|}{Cellules fusionnées}&case \no3&case \no4\\
\hline
case \no5&\multicolumn{2}{|l|}{Cellules fusionnées}&case \no8\\
\hline
case \no9&\multicolumn{3}{|c|}{Cellules fusionnées}\\
\hline
case \no13&case \no14&case \no15&case \no16\\
\hline
\end{tabular}
```

	Cellules fusionnées	case n° 3	case n° 4
case n° 5	Cellules fusionnées		case n° 8
case n° 9	Cellules fusionnées		
case n° 13	case n° 14	case n° 15	case n° 16

Attention aux `|` dans `multicolumn` qui peuvent tracer 2 traits comme dans la case n° 10, il vaut mieux procéder comme pour la case n° 6.

Conclusion

Voilà on vient de voir toutes les commandes associées à l'environnement `tabular`. Il permet de réaliser pas mal de tableaux. Mais bien sûr c'est pas suffisant, et de nombreux packages existent pour réaliser des tableaux plus complexes. Dans d'autres fiches nous aborderons les packages `multirow` pour fusionner les lignes, `supertabular` ou `longtable` pour faire des tableaux qui s'étendent sur plusieurs pages, `colortbl` pour colorier les cellules. . .