



**IMT Mines Albi-Carmaux**  
École Mines-Télécom

## Le package tabularray

qui permet de réaliser (presque) tous les  
tableaux

**par Paul Gaborit**

Exposé GUTenberg

4 avril 2024

***GUTenberg***



Avant



Maintenant

Paul Gaborit

60 ans sur terre dont plus de la moitié avec  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

<https://perso.mines-albi.fr/~gaborit/latex/>

[paul.gaborit@mines-albi.fr](mailto:paul.gaborit@mines-albi.fr)

[paul.gaborit@gmail.com](mailto:paul.gaborit@gmail.com)



Au naturel



À la Toulouse-Lautrec

Première partie  
**Introduction**



**IMT Mines Albi-Carmaux**  
Ecole Mines-Télécom

- 🔍 Ai-je vraiment besoin d'un tableau ?
- 🔍 N'est-ce pas plutôt une suite d'éléments (donc une liste) nécessitant une présentation régulière afin de pouvoir les comparer rapidement ?



Nom	Description	Prix		
T-Shirt	Coton Bio, Taille L, Cousu main	23,90 €	<b>T-Shirt</b>	23,90 €
Blouse	Polyamide, Taille L	12,95 €	<b>Blouse</b>	12,95 €
			Polyamide, Taille L	

- 👉 Une cellule d'un tableau ne devrait contenir que des informations tabulaires : un nombre, une caractéristique courte... beaucoup moins souvent une phrase ou un paragraphe... et encore moins une liste !

- Ne pas encadrer toutes les cellules ! Un beau tableau contient le moins de filets possibles :
  - En général aucun filet vertical.
  - Un simple filet horizontal entre les titres et les données.
- Choisir soigneusement l'alignement du contenu des cellules pour rendre la lecture évidente.
- Seul ce qui est constant peut être en gras.



Matériau	Densité	
	Europe	U.S.
Papier	1000.00 kg/m <sup>3</sup>	2204.62 lb/m <sup>3</sup>
Chêne	700.00 kg/m <sup>3</sup>	1543.23 lb/m <sup>3</sup>
Or	19400.00 kg/m <sup>3</sup>	42769.67 lb/m <sup>3</sup>



Matériau	Densité	
	Europe	U.S.
	kg/m <sup>3</sup>	lb/m <sup>3</sup>
Papier	1000,00	2204,62
Chêne	700,00	1543,23
Or	19 400,00	42 769,67

- Pour ceux qui ont la flemme... d'apprendre tous les détails de la composition des tableaux en  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ , il existe l'outil en ligne « [LaTeX Tables Editor](#) » qui offre, entre autres :
  - import de tableaux (depuis MS Word, Excel, OpenOffice,  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ...),
  - fusion horizontale/verticale de cellules,
  - lignes/colonnes aisément insérées, supprimées, déplacées ou permutées,
  
- ... et dans lequel on peut spécifier le ou les packages à utiliser.
  - 👉 Mon conseil : choisir `tabularray` comme «  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  environment ».

# Deuxième partie

## Quels sont les packages à utiliser



**IMT Mines Albi-Carmaux**  
Ecole Mines-Télécom

## ■ Avant :

- L<sup>A</sup>T<sub>E</sub>X** offre les environnements **tabular** (en mode texte) et **array** (en mode mathématique) (et, entre autres, la commande `\multicolumn`).
- array** enrichit les environnements **tabular** et **array**; fournit la commande `\newcolumnstype`.
- cellspace** nouveaux types de colonnes améliorant l'espacement autour du contenu des cellules.
- booktabs** améliore la présentation des tableaux (comme dans les livres); fournit les commandes `\toprule`, `\midrule`, `\bottomrule` et `\cmidrule` (en remplacement de `\hline` et `\midline`).
- tabularx** fournit l'environnement **tabularx** avec le nouveau type de colonne **X** (la largeur des colonnes **X** est automatiquement ajustée pour produire un tableau de la largeur demandée).
- arraycols** pour des types de colonnes améliorées (entre autres pour les colonnes **X**).
- siunitx** permet d'aligner des cellules sur le séparateur décimal via les colonnes de type **S**.
- multirow** propose des cellules fusionnées sur plusieurs lignes via la commande `\multirow`.
- makecell** crée des cellules avec des alignements et des orientations spécifiques.
- colortbl** pour créer des cellules à fond coloré (déconseillé pour une bonne impression).
- longtable** pour créer des tableaux sur plusieurs pages.
- tabu** qui tente d'unifier le tout mais dont la syntaxe et la documentation sont très confuses et, surtout, dont le développement est abandonné.

## 👍 Maintenant :

**nicematrix** très puissant et récent pour faire tous les types de tableaux et matrices.

- ☹️ Pas de tableaux sur plusieurs pages.
- ☹️ Syntaxe parfois confuse, voire touffue!

**tabularray** **Le package** qui veut remplacer tous les autres!

- 😊 Syntaxe très régulière et claire!
- ☹️ Pour certaines matrices **nicematrix** reste plus puissant!
- ☹️ Temps de compilation (parfois beaucoup) plus long!



## Avant :

- ☹ Les tableaux très simples sont faciles à faire avec les commandes de base (L<sup>A</sup>T<sub>E</sub>X)...
- ☹ ... mais même avec l'aide de packages supplémentaires (`array`, `cellspace`, `booktabs`, `tabularx`, `arraycols`, `siunitx`, `multirow`, `makecell`, `colortbl`, `longtable` et `tabu`), les tableaux un peu évolués deviennent rapidement complexes à composer.

## Maintenant :

- 😊 Heureusement le package `tabularray` change la donne en offrant une interface complète, puissante et relativement aisée d'utilisation...
- ☹ ... au prix d'un temps de compilation plus long.
- 😊 Pour certains types de matrices pour lesquelles `tabularray` n'est pas encore équipé, `nicematrix` offre ce qui manque.

Troisième partie  
**En pratique**



**IMT Mines Albi-Carmaux**  
École Mines-Télécom

- Le package `tabularray` permet de réaliser quasiment n'importe quel tableau.
  - Il faut une distribution  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  récente pour pouvoir l'utiliser.
  - Sa documentation montre beaucoup d'exemples avec différentes syntaxes pour s'adapter aux anciens usages.
- 👉 Nous ne montrerons ici que la nouvelle syntaxe (qui est plus régulière même si parfois plus verbeuse) mais nous ne montrerons pas toutes les possibilités.

- L'environnement appelé `tblr` fonctionne :
  - en mode texte (comme `tabular`),
  - en mode mathématique (comme `array`).
- Il peut remplacer `tabular` ou `array` (l'argument décrit alors les colonnes) :

```
\begin{tblr}{r|c|l}  
  \hline  
  essai & de          & tableau \\ \\  
  avec  & du             & texte  \\ \\  
  et    & l'ancienne    & syntaxe \\ \\  
  \hline  
\end{tblr}
```

essai	de	tableau
avec	du	texte
et	l'ancienne	syntaxe

- ... mais nous conseillons d'utiliser la syntaxe moderne plus verbeuse mais plus claire et régulière.

- Le package `tabularray` seul permet de faire déjà beaucoup de choses...
  - ... mais on peut le compléter par quelques extensions afin de le rendre encore beaucoup plus puissant.
- 
- Voici la manière dont nous conseillons de le charger (dans le préambule) :

```
\usepackage{tabularray}  
\UseTblrLibrary{amsmath,booktabs,counter,diagbox,nameref,siunitx,varwidth,zref}
```

```

\begin{tblr}{
  colspec={Q[r] Q[c] Q[l]},
  hline{1,4},
  vline{2,3},
}
essai & de & & tableau \\
avec & du & & texte \\
et & la & syntaxe & moderne \\
\end{tblr}

```

essai	de	tableau
avec	du	texte
et	la syntaxe	moderne

- La spécification des colonnes se fait via la clé `colspec` :
  - Une cellule générique s'appelle `Q` et utilise des paramètres entre crochets : alignement à droite (`r` - *right*), centrage (`c` - *center*), alignement à gauche (`l` - *left*), texte justifié avec césure (`j` - *justified*).
- La spécification des lignes séparatrices horizontales se fait via la clé `hline`.

Exemples :

`hline{1,4}` la 1<sup>re</sup> et la 4<sup>e</sup> ligne.

`hline{3-8}` de la 3<sup>e</sup> à la 8<sup>e</sup> ligne.

`hline{1,3-Z}` la 1<sup>re</sup> ligne et de la 3<sup>e</sup> à la dernière (`Z`) ligne.

`hline{-}` toutes les lignes (raccourci : `hlines`).

- La spécification des lignes séparatrices verticales se fait via la clé `vline` (même règles que pour `hline`).
- 👉 Dans un tableau a 5 lignes, il y a 6 lignes séparatrices horizontales !  
Dans un tableau a 3 colonnes, il y a 4 lignes séparatrices verticales !

```
\begin{tblr}{  
  vlines, hlines  
}  
{Alpha\ \ Alph.} & Beta & Gamma & \\  
Delta & {Epsilon\ \ Eps.} & Zeta & \\  
Eta & Theta & {Iota\ \ Iot.} & \\  
\end{tblr}
```

Alpha Alph.	Beta	Gamma
Delta	Epsilon Eps.	Zeta
Eta	Theta	Iota Iot.

- Il n'est pas nécessaire de spécifier les colonnes (mais c'est conseillé).
  - 👉 faire attention au nombre de `&` dans chaque ligne (un de moins que de colonnes).
- Une colonne non spécifiée est de type `Q[1, t]` (où `t` indique que la première ligne du contenu sert à l'alignement vertical dans la cellule).
- Pour passer à la ligne dans une cellule :
  - écrire le contenu de cette cellule dans un groupe entre accolades `{...}`
  - puis insérer dans le contenu un `\\` pour passer à la ligne.

```
\begin{tblr}{  
  vlines, hlines,  
  colsep=10pt, rowsep=1pt,  
}  
{Alpha\\ Alph.} & Beta & Gamma & \\  
Delta & {Epsilon\\ Eps.} & Zeta & \\  
Eta & Theta & {Iota\\ Iot.} & \\  
\end{tblr}
```

Alpha Alph.	Beta	Gamma
Delta	Epsilon Eps.	Zeta
Eta	Theta	Iota Iot.

- L'écartement entre les lignes du tableau (ou entre une ligne de tableau et la ligne séparatrice horizontale lorsqu'elle existe) est spécifié par la clé `rowsep`.
  - 👉 la valeur par défaut est `rowsep=2pt`.
    - On peut définir des valeurs différentes pour le haut (`aboveseq`) et le bas (`belowsep`).
- Il en est de même pour les colonnes via la clé `colsep`.
  - 👉 la valeur par défaut est `colsep=6pt`.
    - On peut définir des valeurs différentes à gauche (`leftsep`) et à droite (`rightsep`).



## Fixer la largeur des colonnes

```

\begin{tblr}{
  vlines,hlines,
  colspec={Q[1cm,r] Q[1.5cm,c] Q[1cm,l]},
  rowspec={Q[f] Q[m] Q[h]},
}
Alpha Alph. & Beta          & Gamma    \\
Delta      & Epsilon Epsilon & Zeta     \\
Eta       & Theta          & Iota Long \\
\end{tblr}

```

Alpha Alph.	Beta	Gamma
Delta	Epsilon Epsilon	Zeta
Eta	Theta	Iota Long

**colspec** définit les réglages des cellules de chaque colonne. Ici, on spécifie la largeur et l'alignement horizontal de chaque colonne :

- à gauche (**l** – *left*), centré (**c** – *center*), à droite (**r** – *right*), justifié (**j** – *justify*).
- **3cm** pour une colonne de 3 cm de large (si le contenu d'une cellule est trop long, il y a passage à la ligne automatique).

**rowspec** définit les réglages des cellules de chaque ligne. Ici, on spécifie l'alignement vertical des cellules :

- alignée en milieu (**m** – *middle*), alignée au pied (**f** – *foot*), alignée en tête (**h** – *head*).
- Il existe aussi : alignée sur la première ligne du contenu (**t** – *top*) et alignée sur la dernière ligne du contenu (**b** – *bottom*).

👉 Généralement, on utilise uniquement la clé `colspec...` Mais si on utilise les deux clés, les réglages sont mis en commun et, en cas de conflit, le dernier l'emporte.

```

\begin{tblr}{
  colspec={Q[c,mode=math] Q[1,cmd=\unit] Q[1,co=1]},
  row{odd}={bg=orange!10!white},
  row{1}={1,cmd=\textbf,mode=text,fg=white,bg=orange},
  rowsep=2pt, colsep=3pt,
}
Sym. & Unité & Signification & \\
v & \mper\s & & Vitesse & \\
\omega & \radian\per\s & & Vitesse angulaire & \\
a & \mper\square\s & & Accélération & \\
\alpha & \radian\per\square\s & & Accélération angulaire & \\
\end{tblr}

```

Sym.	Unité	Signification
$v$	$\text{m s}^{-1}$	Vitesse
$\omega$	$\text{rad s}^{-1}$	Vitesse angulaire
$a$	$\text{m s}^{-2}$	Accélération
$\alpha$	$\text{rad s}^{-2}$	Accélération angulaire

- La 1<sup>re</sup> colonne est en mode mathématique (`mode=math`).
- La 2<sup>e</sup> colonne contient des unités : on applique automatiquement la commande `\unit` (du package `siunitx`) au contenu de chaque cellule (`cmd=\unit`).
- La 3<sup>e</sup> colonne utilise le coefficient 1 (`co=1`) pour élargir la largeur de la colonne afin d'occuper toute la largeur disponible (comme la colonne X de `tabularx`).
- Les lignes impaires (`row{odd}`) ont un fond orange clair (`bg=orange!10!white`).
- La 1<sup>re</sup> ligne (`row{1}`) utilise un réglage spécifique pour ses cellules : alignement à gauche (1), en gras (`cmd=\textbf`), en mode texte (`mode=text`), écrit en blanc (`fg=white`) sur fond orange (`bg=orange`).

```
\def\mytext{Du texte assez long sur plusieurs
lignes et colonnes}
\begin{tblr}{
  vspan=even, hspan=minimal,
  hlines={blue}, vline{2-3}={red},
  row{odd}={orange!10}, row{even}={lime!20},
  colspec={Q[1] Q[1,co=1] Q[1,co=2]},
  cell{2}{2}={c=2,r=2}{1,cyan!10,font=\itshape},
  cell{4}{2}={c=2}{c}, cell{5}{1}={c=2}{1},
  colsep=3pt,
}
C1 & C2 (co=1) & C3 (co=2) \\
L2 & {\mytext. \mytext. \mytext.} & \\
L3 & 13-c2 & \\
L4 & Texte centré sur 2 colonnes & 14-c3 \\
L5 & 15-c2 & 15-C3 \\
\end{tblr}
```

C1	C2 (co=1)	C3 (co=2)
L2	<i>Du texte assez long sur plusieurs lignes et colonnes. Du texte assez long sur plusieurs lignes et colonnes. Du texte assez long sur plusieurs lignes et colonnes.</i>	
L3		
L4	Texte centré sur 2 colonnes	
L5		L5-C3

- La fusion de cellules pour être « naturelle » nécessite quelques réglages (`vspan=even` et `hspan=minimal`).
- `cell{i}{j}` règle la cellule de la *i*-ème ligne et de la *j*-ème colonne.
- La cellule `cell{2}{2}` fusionne 2 colonnes (`c=2`) et 2 lignes (`r=2`) et contient un texte aligné à gauche (1) composé en italique (`font=\itshape`) sur fond bleu clair (`cyan!10`).
- Les cellules `cell{4}{2}` et `cell{5}{1}` fusionnent 2 colonnes (`c=2`) et contiennent un texte centré (c).
- La fusion se fait toujours vers la droite et vers le bas. Le contenu des cellules consommées par la fusion n'est pas pris en compte (ex : 13-c2, 14-c3 et 15-c2).

```

\newcommand\intertitre[1]{%
  \SetRow{rowsep=1pt,bg=gray!10}
  \SetCell[c=2]{mode=text,c,font=\itshape\footnotesize}
  #1 &
}
\begin{tblr}[expand=\intertitre]{
  colspec={Q[r,mode=math] S}, vspan=even, hspan=minimal,
  row{1}={font=\bfseries,mode=text,c},
}
Tems  $t$  (\unit{\micro\second})
& {{{Distance  $d$  (\unit{\micro\meter})}}} \ \backslash
\intertitre{Au début: petites variations} \ \backslash
t = 1 \quad & 2.3456 \ \backslash
t = 2 \quad & 6.7835 \ \backslash
\intertitre{Plus tard : les variations augmentent...} \ \backslash
t = 10 \quad & 90.473 \ \backslash
t = 20 \quad & 642.5 \ \backslash
\intertitre{Puis ça explose !} \ \backslash
t = 100 \quad & 1.2e6 \ \backslash
\end{tblr}

```

Temps $t$ ( $\mu\text{s}$ )	Distance $d$ ( $\mu\text{m}$ )
-----------------------------	--------------------------------

*Au début : petites variations*

$t = 1$	2,3456
---------	--------

$t = 2$	6,7835
---------	--------


*Plus tard : les variations augmentent...*

$t = 10$	90,473
----------	--------

$t = 20$	642,5
----------	-------

*Puis ça explose!*

$t = 100$	$1,2 \times 10^6$
-----------	-------------------

- La commande `\intertitre` compose une ligne complète (avec un `&`). L'environnement `tblr` doit développer cette commande (`[expand=\intertitre]`) avant l'analyse du contenu du tableau.
- Dans `\intertitre`, `\SetRow` modifie les réglages de la ligne courante et `\SetCell` fusionne deux colonnes (`[c=2]`) en définissant la présentation du contenu (`mode=text,c,font=\itshape\scriptsize`).
  -  `\SetRow` et `\SetCell` permettent de modifier les réglages d'une ligne ou d'une cellule, comme les clés `row` et `cell`, mais directement dans le tableau, donc sans devoir compter les lignes.
- La 2<sup>e</sup> colonne utilise le type `S` (provenant de `siunitx`) pour aligner correctement les valeurs numériques. Dans cette colonne, une valeur entre triple accolades (`{{{Distance ...}}}`) n'est pas considérée comme une valeur numérique.

■ Version de base de `amsmath` :

```
\begin{equation*}
\begin{pmatrix}
\frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\
\frac{2}{3} & -\frac{1}{3} & -\frac{2}{3} \\
\frac{1}{3} & -\frac{2}{3} & \frac{2}{3}
\end{pmatrix}
\end{equation*}
```

$$\begin{pmatrix} \frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & -\frac{1}{3} & -\frac{2}{3} \\ \frac{1}{3} & -\frac{2}{3} & \frac{2}{3} \end{pmatrix}$$

■ Version améliorée de `tabularray` :

```
\begin{equation*}
\begin{+pmatrix}[cells={r},row{2}={orange!10}]
\frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\
\frac{2}{3} & -\frac{1}{3} & -\frac{2}{3} \\
\frac{1}{3} & -\frac{2}{3} & \frac{2}{3}
\end{+pmatrix}
\end{equation*}
```

$$\begin{pmatrix} \frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & -\frac{1}{3} & -\frac{2}{3} \\ \frac{1}{3} & -\frac{2}{3} & \frac{2}{3} \end{pmatrix}$$

👉 `tabularray` définit les environnements améliorés `+array`, `+matrix`, `+bmatrix`, `+Bmatrix`, `+pmatrix`, `+vmatrix`, `+Vmatrix` et `+cases`.

- On peut définir un environnement qui reproduit le comportement de `tblr` mais en modifiant les réglages par défaut :

```
\NewTblrEnviron{my-own-tblr}  
\SetTblrInner[my-own-tblr]{  
  hlines={1}{-}{gray,dotted}, hline{2-Y}={2}{-}{gray,dotted}, vlines={blue,dashed},  
  row{odd}={lime!20}, colspec={Q[r] Q[c] Q[l]}  
}
```

- Le nouvel environnement `my-own-tblr` est créé via la commande `\NewTblrEnviron`.
- La macro `SetTblrInner` permet de définir ses options.

- Et voici le résultat de son utilisation :

```
\begin{my-own-tblr}{verb, column{2}={fg=blue}}  
Alpha & Beta & Gamma & \\  
Nu & Xi & Pi & \\  
\verb+a+ & \verb+ee{ee}+ & \verb+z{z}+  
\end{my-own-tblr}
```

Alpha	Beta	Gamma
Nu	Xi	Pi
\a	\ee{ee}	\z{z}

- 👉 La clé `verb` permet d'utiliser quasiment n'importe quelles commandes `verbatim` dans les tableaux créés par `tblr`.

- Le package `tabularray` fournit l'environnement `longtblr` qui permet de composer des tableaux sur plusieurs pages. Cela permet par exemple de répéter automatiquement un entête et un pied de tableau à chaque nouvelle page.
- L'extension `diagbox` permet d'utiliser la commande `\diagbox` :

```
\begin{equation*}
\begin{tblr}{
  vline{2}, hline{2},
  colsep=2pt, rowsep=1pt,
  colspec={Q[c] Q[c] Q[c]},
}
\diagbox{a}{b} & x & y \\
0 & x_0 & y_0 \\
1 & x_1 & y_1
\end{tblr}
\end{equation*}
```

$a$	$b$	$x$	$y$
$0$		$x_0$	$y_0$
$1$		$x_1$	$y_1$

- Les compteurs `rownum` et `colnum` sont prédéfinis par `tabularray`. L'extension `counter` permet d'utiliser ses propres compteurs  $\LaTeX$  dans un tableau.

```
\newcounter{mycnt}
\newcommand{\xij}{
  \stepcounter{mycnt}
  $\x_{\arabic{rownum}, \arabic{colnum}}$
  = \alph{mycnt}$
}
\begin{tblr}{colsep=2pt}
\xij & \xij & \xij \\
\xij & \xij & \xij \\
\xij & \xij & \xij
\end{tblr}
```

$$\begin{array}{lll} x_{1,1} = a & x_{1,2} = b & x_{1,3} = c \\ x_{2,1} = d & x_{2,2} = e & x_{2,3} = f \\ x_{3,1} = g & x_{3,2} = h & x_{3,3} = i \end{array}$$

```

\begin{tblr}{
  colspec={Q[r] Q[r] Q[r]},
  rowsep=1pt,colsep=6pt,
  row{1}={font=\bfseries},
  row{2}={c,font=\bfseries},
  cell{4-Z}{2-3}={cmd=\num},
}
\SetCell[r=3]{r,m} Matériau & \SetCell[c=2]{c} Densité & \\
\cmidrule[1r,green!50!black]{2-3}
& Europe & U.S. \\
\SetRow{rowsep=0pt,c,font=\footnotesize}
& kg/m$^3$ & lb/m$^3$ \\
\toprule[green!50!black]
Papier & 1000.00 & 2204.62 \\
Chêne & 700.00 & 1543.23 \\
Or & 19400.00 & 42769.67 \\
\end{tblr}

```

Matériau	Densité	
	Europe	U.S.
	kg/m <sup>3</sup>	lb/m <sup>3</sup>
Papier	1000,00	2204,62
Chêne	700,00	1543,23
Or	19 400,00	42 769,67





**Merci de votre  
attention**

**Questions ?**