

Documentation tikz-tabular

Version 0.1 – 18 mai 2025

Version française

Stéphane Pasquet

Sommaire

1	Introduction	2
1.1	Objet de ce package	2
1.2	Python	2
2	Installation	2
3	Utilisation	3
3.1	Un exemple basique	3
3.2	Le code \LaTeX correspondant	3
3.3	La syntaxe	4
3.3.1	La variable <code>\globalpath</code>	4
3.3.2	L'environnement <code>tikztabular</code>	4
3.3.3	Le titre du tableau (facultatif)	4
3.3.4	Les en-têtes des colonnes du tableau (facultatif)	4
3.3.5	Les pieds des colonnes (facultatif)	5
3.3.6	Les premières colonnes	5
3.3.7	Les dernières colonnes	5
4	Les options	6
5	F.A.Q.	7
5.1	Pourquoi avoir défini l'environnement dans un scope ?	7
5.2	Pourquoi les options sont précédées d'un chevron ?	7
5.3	Pourquoi devoir déclarer la variable <code>\globalpath</code> ?	7

1 Introduction

1.1 Objet de ce package

Ce package permet de créer un tableau en mode graphique (basé sur TiKZ) à l'aide d'une syntaxe allégée.

1.2 Python

Il est nécessaire que Python soit installé.

Si cela n'est déjà fait, il suffit d'aller sur la page :

<https://www.python.org/downloads/>

et de télécharger la dernière version.

Utilisateurs de Windows : Veillez à ce que le chemin qui va vers `python.exe` figure dans le PATH du système de windows (si vous utilisez ce système d'exploitation).

variables d'environnement > variables systèmes > chemin vers `python.exe`

Le chemin est de la forme :

`C:\Users\<vous>\AppData\Local\Programs\Python\Python3xx`

2 Installation

Si vous effectuez une installation manuelle de ce package, vous devez respecter l'arborescence \LaTeX . Si par exemple vos packages installés manuellement sont dans :

`c:\texmf\`

vous devrez veiller à avoir :

`c:\texmf\doc\tikz-tabular\`

`c:\texmf\latex\tikz-tabular\`

Note importante : après avoir mis les fichiers dans les bons dossiers, il est nécessaire de changer une ligne du fichier `tikz-tabular.sty` :

```
22 \immediate\write18{python "c:/texmf/tex/latex/tikz-tabular
/tikztabular.py" num-\thetikz@tabular-dir-\globalpath}%
```

Cette ligne doit contenir le chemin exact vers le fichier `python` qui est dans le même dossier.

Utilisateurs de Windows et MikTeX : rafraichissez la base de données une fois le package installé.

Dans la MikTeX Console,

Tasks > Refresh file name database

3 Utilisation

3.1 Un exemple basique

Mon tableau				
Tête 1	Tête 2	Tête 3	Tête 4	
Cellule 1.1	Cellule 1.2	Cellule 1.3	Cellule 1.4	Cellule 1.5
Cellule 2.1	Cellule 2.2	Cellule 2.3	Cellule 2.4	Cellule 2.5
Pied 1	Pied 2	Pied 3	Pied 4	

3.2 Le code \LaTeX correspondant

Code minimal

```
\documentclass{article}
\usepackage{tikz-tabular}
\newcommand{\globalpath}{"c:/texmf/doc/tikz-
    tabular/"}

\setlength{\parindent}{0pt}

\begin{document}

\begin{tikzpicture}
\begin{tikztabular}
<columns=5
[tabulartitle]
\fontfamily{lmss}\selectfont\Large\textbf{Mon tableau}
[endtabulartitle]

[tabularhead]
\textbf{Tête 1} & \textbf{Tête 2} & \textbf{Tête 3} &
\textbf{Tête 4} &
[endtabularhead]

Cellule 1.1 & Cellule 1.2 & Cellule 1.3 & Cellule 1.4
& Cellule 1.5
Cellule 2.1 & Cellule 2.2 & Cellule 2.3 & Cellule 2.4
& Cellule 2.5

[tabularfoot]
\textbf{Pied 1} & \textbf{Pied 2} & \textbf{Pied 3} &
\textbf{Pied 4} &
\end{tikztabular}
\end{tikzpicture}

\end{document}
```

3.3 La syntaxe

3.3.1 La variable `\globalpath`

Elle est nécessaire pour déclarer le chemin du fichier courant.

Sans cela, les fichiers auxiliaires construits pour les tableaux ne seront pas créés.

Code minimal

```
\newcommand{\globalpath}{ "c:/texmf/doc/tikz-tabular/" }
```

3.3.2 L'environnement `tikztabular`

Code minimal

```
\begin{tikzpicture}  
  \begin{tikztabular}  
    ...  
  \end{tikztabular}  
\end{tikzpicture}
```

Il doit être dans un environnement `tikzpicture`. En effet, il est construit dans un environnement `scope`.

3.3.3 Le titre du tableau (facultatif)

Code minimal

```
[tabulartitle]  
Mon titre  
[endtabulartitle]
```

Il permet de déclarer que le tableau a un titre. Il est affiché sur toute la largeur du tableau.

3.3.4 Les en-têtes des colonnes du tableau (facultatif)

Code minimal

```
[tabularhead]  
En-têtes séparées par "&"  
[endtabularhead]
```

Cela permet d'afficher les en-têtes des colonnes de manière différente des autres cellules.

3.3.5 Les pieds des colonnes (facultatif)

Code minimal

```
[tabularfoot]  
Pieds séparées par "&"
```

Cela permet d'afficher les pieds des colonnes de manière différente des autres cellules.

Dans la mesure où les pieds sont sensés être en fin de tableau, il n'y a pas de commande [endtabularfoot].

3.3.6 Les premières colonnes

Mon tableau

Cellule 1.1

Cellule 1.2

Cellule 1.3

Cellule 1.4

Cellule 1.5

Cellule 2.1

Cellule 2.2

Cellule 2.3

Cellule 2.4

Cellule 2.5

Code minimal

```
\begin{tikztabular}  
<columns=5  
<firstcolumncolor=orange!75!black  
[tabulartitle]  
Mon tableau  
[endtabulartitle]  
Cellule 1.1 & Cellule 1.2 & Cellule 1.3 & Cellule 1.4  
  & Cellule 1.5  
Cellule 2.1 & Cellule 2.2 & Cellule 2.3 & Cellule 2.4  
  & Cellule 2.5  
\end{tikztabular}
```

3.3.7 Les dernières colonnes

Cellule 1.1

Cellule 1.2

Cellule 1.3

Cellule 1.4

Cellule 1.5

Cellule 2.1

Cellule 2.2

Cellule 2.3

Cellule 2.4

Cellule 2.5

Code minimal

```
\begin{tikztabular}  
<columns=5  
<lastcolumncolor=orange!75!black  
Cellule 1.1 & Cellule 1.2 & Cellule 1.3 & Cellule 1.4  
  & Cellule 1.5  
Cellule 2.1 & Cellule 2.2 & Cellule 2.3 & Cellule 2.4  
  & Cellule 2.5  
\end{tikztabular}
```

4 Les options

Les options sont à mettre dans l'environnement `tikztabular` *avant* le contenu des cellules.

Elles doivent être mises *nécessairement* sous la forme :

`<option = valeur` (avec ou sans espace mais avec le chevron)

Les options		
Nom de l'option	Désignation	Exemple
<code>columns</code>	nombre de colonnes	<code>columns = 5</code>
<code>corners</code>	taille des coins arrondis	<code>corners = 1mm</code>
<code>colorcell</code>	couleur des cellules	<code>colorcell = red</code>
<code>tabularwidth</code>	largeur du tableau	<code>tabularwidth = 20cm</code>
<code>cellheight</code>	hauteur des cellules	<code>cellheight = 2em</code>
<code>marginh</code>	marge horizontale entre deux colonnes	<code>marginh = 1em</code>
<code>marginv</code>	marge verticale entre deux lignes	<code>marginv = 1em</code>
<code>marginvtitle</code>	marge verticale entre le titre et la ligne suivante	<code>marginvtitle = 1em</code>
<code>tabulartitlecolor</code>	couleur de fond du titre	<code>tabulartitlecolor = black</code>
<code>texttitlecolor</code>	couleur du titre	<code>texttitlecolor = white</code>
<code>firstcolumncolor</code>	couleur de la première colonne	<code>firstcolumncolor = black</code>
<code>lastcolumncolor</code>	couleur de la dernière colonne	<code>lastcolumncolor = black</code>
<code>tabularheadcolor</code>	couleur de fond des en-têtes	<code>tabularheadcolor = orange</code>
<code>textheadcolor</code>	couleur du texte des en-têtes	<code>textheadcolor = white</code>
<code>tabularfootcolor</code>	couleur de fond des pieds	<code>tabularfootcolor = white</code>
<code>textfootcolor</code>	couleur du texte des pieds	<code>textfootcolor = white</code>
<code>textcolor</code>	couleur du texte des cellules	<code>textcolor = black</code>

5 F.A.Q.

5.1 Pourquoi avoir défini l'environnement dans un scope ?

Je souhaitais que l'on puisse agir sur le tableau, par exemple en y ajoutant des flèches. Chaque cellule porte un nom sous la forme :

```
(cell-<ligne>-<colonne>)
```

Ainsi, pour joindre deux cellules entre elles par une flèche, on pourra écrire :

```
\draw[->,>=latex] (cell-1-1) to[bend left=10] (cell-3-4);
```

5.2 Pourquoi les options sont précédées d'un chevron ?

Une fois le traitement des options par Python terminé, le programme supprime toutes les lignes contenant le noms des options.

Imaginons que les options soient inscrites sans symbole particulier (comme le chevron). Si une cellule du tableau contient un mot comme « columns » (qui est une option), la ligne contenant le mot est supprimée...

J'aurais pu adopter une autre logique et mettre par exemple `[start]` pour indiquer que le tableau commence, mais cela m'a semblé plus contraignant.

5.3 Pourquoi devoir déclarer la variable `\globalpath` ?

TeX ne peut pas transmettre (à ma connaissance) le chemin du fichier compilé.

Or, ce chemin est nécessaire au programme Python.

Il faut donc définir cette variable manuellement.