

# L'extension `delarray`\*

David Carlisle

28/10/2014

Ce fichier est maintenu par l'équipe du «`LATEX` Project». Les rapports d'anomalie peuvent être envoyés en anglais à <http://latex-project.org/bugs.html> (catégorie `tools`).

## 1 Exemples

Cette extension ajoute à l'extension `array` un système de paires implicites `\left` `\right`. Si vous souhaitez obtenir un tableau entouré de parenthèses, vous pouvez alors saisir :

`\begin{array}{cc}` ...

$$\left( \begin{array}{cc} a & b \\ c & d \end{array} \right)$$

De la même manière, un environnement équivalent à `\cases` de `PLAIN TEX` peut être défini avec :

`\begin{array}\{\{LL\}` ...

$$f(x) = \begin{cases} 0 & \text{si } x = 0 \\ \sin(x)/x & \text{sinon} \end{cases}$$

Ici, `L` désigne une colonne de texte sinistrophe<sup>1</sup> alignée à gauche, qui peut être définie comme suit : `\newcolumnntype{L}{>{\$}l<{\$}}`, comme expliqué dans l'extension `array`. Notez que les délimiteurs doivent toujours être utilisés par paires et que le «`.`» doit être utilisé pour indiquer un «`délimiteur nul`».

Ce dispositif est particulièrement utile si les arguments `[t]` ou `[b]` sont aussi utilisés. Dans ces cas le résultat n'est pas équivalent à celui obtenu en entourant l'environnement par `\left...\right`, comme vous pouvez le voir dans les exemples

---

\*Ce fichier a pour numéro de version v1.01 et a été mis à jour le 28/10/2014. Son titre original est «*The delarray package*». La première traduction, basée la version v1.01, a été publiée par Jean-Pierre Drucbert en 2000.

1. N.D.T. : il s'agit d'un mode où le texte est écrit de gauche à droite.

suivants :

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad \text{et non} \quad \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

```
\begin{array}{t}({c}) 1\\2\\3 \end{array}
\begin{array}{c}({c}) 1\\2\\3 \end{array}
\begin{array}{b}({c}) 1\\2\\3 \end{array}
\quad\mbox{et non}\quad
\left(\begin{array}{t}{c} 1\\2\\3 \end{array}\right)
\left(\begin{array}{c}{c} 1\\2\\3 \end{array}\right)
\left(\begin{array}{b}{c} 1\\2\\3 \end{array}\right)
```

## 2 Les commandes

```
1 \*package
2 \RequirePackage{array}[1994/02/03]

\@tabarray Cette commande teste la présence d'un crochet optionnel puis appelle \@array
ou \@array[c] (par défaut).
3 \def\@tabarray{\ifnextchar[{\@array}{\@array[c]}}

\@array Cette commande teste la présence d'un délimiteur optionnel avant l'accolade
gauche débutant le préambule de la table. S'il n'y a pas de délimiteur, \@arrayleft
et \@arrayright ne sont associés à aucune action et \@array est appelé avec l'ar-
gument de position (entre crochets). Sinon, \@delarray est appelé.
4 \def\@array[#1]{\ifnextchar\bgroup
5   {\let\@arrayleft\relax\let\@arrayright\relax\@array[#1]}%
6   {\@delarray[#1]}}

\@delarray Nous savons maintenant que nous avons un environnement array (ou tabular)
avec des délimiteurs.
7 \def\@delarray[#1]#2#3#4{%

La ligne suivante est complètement redondante mais elle permet de capter des
erreurs associées aux délimiteurs avant que le traitement de l'alignement ne com-
mence. Une erreur courante est ici l'oubli d'un « . » dans une construction similaire
à l'environnement \cases. Ceci provoque l'engloutissement de la première unité
lexicale, ce qui conduit généralement à de nombreux messages d'erreurs infon-
dés masquant la vraie source de l'erreur, le délimiteur manquant, d'autant que
\@arrayright place l'alignement et les délimiteurs ensemble.
8 \setbox\z@\hbox{${\left#2\right#4$}%
```

Dans le cas de l'argument « c », nous n'avons pas besoin de remettre en boîte l'alignement, ce qui fait que nous pouvons définir `\@arrayleft` et `\@arrayright` juste en y insérant les délimiteurs.

```
9 \if#1c\def\@arrayleft{\left#2}\def\@arrayright{\right#4}%
```

Sinon, nous avons (sauf erreur) un argument [t] ou [b] et nous stockons alors l'alignement sans les délimiteurs dans `box0`.

```
10 \else\def\@arrayleft{\setbox\z@}%
```

Puis une fois que l'alignement est traité, `\@arrayright` est traité.

```
11 \def\@arrayright{%
```

L'abaissement que doit subir la boîte (cette mesure est négative dans le cas de [b]) est calculé ici. Il s'agit là d'un peu d'arithmétique, telle que décrite dans le *T<sub>E</sub>XBook*, annexe G, règle 8. Nous effectuons le calcul de cette manière plutôt qu'en prenant la différence entre la profondeur de la boîte `box0` et la profondeur de la boîte définie ci-dessous dans la mesure où la profondeur de cette boîte peut être affectée par les délimiteurs si `\delimitershortfall` ou `\delimiterfactor` ont des valeurs non standards.

```
12 \dimen@=\dp\z@
```

```
13 \advance\dimen@-\ht\z@
```

```
14 \divide \dimen@ by \tw@
```

```
15 \advance\dimen@ by\fontdimen22 \textfont\tw@
```

Maintenant, nous abaissons l'alignement et les délimiteurs à leur place.

```
16 \lower\dimen@\hbox{$\left#2\center{\unvbox\z@}\right#4$}}%
```

Fin du `\if#1c`

```
17 \fi
```

Maintenant que nous avons défini `\@arrayleft` et `\@arrayright`, `\@array` est appelé.

```
18 \@array[#1]{#3}}
```

```
19 \</package>
```

## 2.1 L'extension `newarray`

Toutes les fonctionnalités de l'ancienne extension `newarray` ont été basculées dans les extensions `array` ou `delarray`.