



# L<sup>A</sup>T<sub>E</sub>X, un peu, beaucoup

Pascal Dupont

Mots clés : L<sup>A</sup>T<sub>E</sub>X, équations



## 4. Les « équations » et leurs styles

Nous avons déjà, dans le troisième numéro de cette rubrique, rencontré quelques formules simples, et les commandes de base pour obtenir les structures les plus courantes de la syntaxe mathématique : fractions, racines, exposants, indices.

Nous allons aujourd'hui en découvrir d'autres, au gré d'exemples, mais surtout, nous allons étudier quelques-uns des grands principes sur lesquels L<sup>A</sup>T<sub>E</sub>X est construit, en mentionnant au passage quelques points de style qu'il est bon de respecter ; autant prendre d'emblée de bonnes habitudes.

Dans les exemples, nous nous permettons quelquefois d'utiliser des commandes qui n'ont pas encore été décrites ; les lecteurs les plus audacieux en profiteront pour les essayer, dans leurs exercices, et en créer des variantes ; les autres patienteront. . .

### Équations dans le texte ou hors-texte

Ci-dessous, nous appellerons « équation » tout passage composé en mode mathématique, même s'il ne s'agit pas, pour le mathématicien que je suis autant que vous, d'une relation d'égalité. Une des caractéristiques les plus visibles en est que bon nombre de lettres (celles qui représentent des variables) sont composées en italiques ; de plus, les règles d'espace sont différentes de celles qui ont cours dans le texte ordinaire.

Les équations insérées dans le texte sont en principe assez brèves (les coupures de ligne y sont souvent peu heureuses) et, surtout, il vaut mieux qu'elles ne contiennent pas d'éléments qui dépassent trop vers le haut ou vers le bas, ce qui obligerait à augmenter l'écartement des lignes de manière fortement inesthétique. Une équation dans le texte est encadrée soit par des symboles « dollar » :  $\$ \dots \$$  (c'est la commande  $T_E X$ ), soit par  $\backslash($  et  $\backslash)$ , soit encore

par  $\backslashbegin{math}$  et  $\backslashend{math}$  (ce sont les commandes L<sup>A</sup>T<sub>E</sub>X<sup>(1)</sup>). Leslie LAMPORT a introduit ces deux paires de commandes par souci d'uniformité, mais très honnêtement, il n'y a aucun bénéfice à les utiliser, et en pratique l'immense majorité des utilisateurs s'en tiennent, à juste titre, aux  $\$$ .

L'écriture  $f(x) = x^2$  est donc obtenue, de manière rigoureusement équivalente, avec le code  $\$f(x)=x^2\$$ , avec  $\backslash(f(x)=x^2\backslash)$  ou encore avec  $\backslashbegin{math}f(x)=x^2 \backslashend{math}$ .

Nous rencontrons ici un premier exemple d'« environnement » : l'environnement *math*. Un environnement est un bloc de texte qui doit être traité de manière particulière ; nous en rencontrerons de nombreux par la suite. L'utilisateur peut d'ailleurs en définir lui-même de nouveaux. On entre dans un environnement par la commande  $\backslashbegin$  suivie du nom de l'environnement entre accolades et on en sort par la commande  $\backslashend$ , elle aussi suivie du nom de l'environnement entre accolades ; ici donc :  $\backslashbegin{math}$  et  $\backslashend{math}$ .

Dès qu'une équation est un peu envahissante, que ce soit horizontalement ou verticalement, il vaut mieux la composer hors texte. Ici encore, trois manières de procéder sont disponibles ; en  $T_E X$  de base, nous encadrons cette équation de « doubles dollars » :  $\$\$ \dots \$\$$ . En syntaxe L<sup>A</sup>T<sub>E</sub>X bien léchée, nous préférons utiliser soit  $\backslash[$  et  $\backslash]$ , soit  $\backslashbegin{displaymath}$  et  $\backslashend{displaymath}$ .

Attention cependant : il n'y a pas, cette fois, équivalence entre la commande  $T_E X$  et les commandes L<sup>A</sup>T<sub>E</sub>X ; en effet, dans certains styles de mise en page prévus par L<sup>A</sup>T<sub>E</sub>X, les équations hors-texte sont non pas centrées horizontalement, mais calées à une distance fixe de la marge de gauche ; assez logiquement, ces styles opéreront de la manière attendue sur les équations déclarées à la manière de L<sup>A</sup>T<sub>E</sub>X, mais laisseront centrées celles qui

<sup>(1)</sup> Rappelons que L<sup>A</sup>T<sub>E</sub>X est un ensemble de commandes prédéfinies qui s'ajoutent à celles de  $T_E X$  pour en rendre l'utilisation plus aisée.



sont balisées par des doubles dollars. Nous préconisons donc, de manière assez ferme, l'utilisation des commandes `\[` et `\]` ; seuls les puristes utiliseront `\begin{displaymath}` et `\end{displaymath}`, inutilement longues à encoder — mais qui ont tout de même le mérite de nous rappeler qu'une équation hors-texte est contenue dans un environnement *displaymath*. Ainsi, pour obtenir :

### Exemple

La matrice

$$A = \begin{pmatrix} 0 & -1 & 2 \\ 1 & 0 & -3 \\ -2 & 3 & 0 \end{pmatrix}$$

est antisymétrique.

codons-nous :

La matrice

```
\[A=
\left(\begin{array}{rrr}
0&-1&2\\
1&0&-3\\
-2&3&0
\end{array}\right)\]
est antisymétrique.
```

Parfois, référence doit être faite à une équation dans la suite du texte. Il est alors nécessaire de l'équiper d'un « repère » : par exemple un astérisque, ou un numéro, entre parenthèses. La commande `\eqno{...}` est prévue pour cela. Pour composer :

### Exemple

Soit

$$f(x, y) = \frac{y}{x} + \left(\frac{y}{x}\right)^2 ; \quad (*)$$

nous avons évidemment aussi

$$f(x, y) = \frac{xy + y^2}{x^2} ;$$

cependant, l'expression (\*) a l'avantage de montrer plus nettement qu'il existe une fonction *g* telle que, pour tous  $x \neq 0$  et  $y$ ,  $f(x, y) = g(y/x)$ , à savoir  $g : t \mapsto t + t^2$ .

### Exemple

nous écrirons, dans le document-source :

```
Soit
\[f(x,y)=
\frac{y}{x}+\left(\frac{y}{x}\right)^2
\text{;}
\eqno{(*)}\]
```

nous avons évidemment aussi

```
\[f(x,y)=\frac{xy+y^2}{x^2}\text{;}
\]
cependant, l'expression $(*)$ a l'avantage
de montrer plus nettement qu'il existe
une fonction $g$ telle que, pour tous
$x\neq 0$ et $y$, $f(x,y)=g(y/x)$, à savoir
$g :t\mapsto t+t^2$.
```

Deux remarques au sujet de cet exemple. D'abord, la ponctuation. Les règles de style exigent que les équations hors-texte comportent la ponctuation normale, au même titre que si elles étaient dans le texte. Or, si nous plaçons « `\]` ; » dans le document-source, le point-virgule serait renvoyé au début de la ligne qui suit l'équation, ce qui est évidemment inacceptable. La ponctuation doit donc bien se trouver *dans* l'équation ; mais si nous l'y plaçons sans précaution, elle serait régie par les règles de la ponctuation mathématique, ce qui n'est pas souhaitable ; voilà pourquoi le point-virgule est encodé bardé d'une commande « `\text` », pour qu'il soit traité... comme du texte (notons toutefois que pour un point ou une virgule, il n'y aurait pas eu de différence).

Ensuite, l'étiquette. S'il n'y a, dans le document, que peu d'équations à signaler, utiliser l'astérisque comme nous l'avons fait convient très bien. En revanche, si les équations auxquelles il faut faire référence se multiplient, le plus commode est de les numéroter. Ceci peut bien sûr se faire à la main, avec la même commande « `\eqno` », mais cela signifie qu'en cas de remaniement du texte, si nous introduisons ou supprimons une équation numérotée, toute la suite des numéros est à revoir. Il vaut donc beaucoup mieux confier à L<sup>A</sup>T<sub>E</sub>X la gestion de cette numérotation : cela fait partie de ses compétences. Pour cela, au lieu d'utiliser pour nos équations l'environnement *displaymath* (ou son synonyme `\[... \]`), nous les placerons dans un environnement *equation*. Ainsi,

### Exemple

$$E = m(a^2 + b^2) \quad (1)$$

### Exemple

est obtenu à l'aide de

```
\begin{equation}
E=m(a^2+b^2)
\end{equation}
```

et le numéro de l'équation est créé par L<sup>A</sup>T<sub>E</sub>X, sans que nous ne devions nous en préoccuper. Selon le



style du document, le numéro pourra être simple, comme ici, ou formé du numéro de chapitre suivi d'un numéro d'équation dont le compteur est remis à zéro au début de chaque chapitre. Nous verrons dans un article ultérieur comment faire référence à ces numéros d'équations automatiques... et donc fluctuants !

Avec ceci, avons-nous fait le tour du problème ? Non pas : il nous reste à traiter notamment le cas des équations trop longues pour tenir sur la largeur de la page (ou de la colonne...), comme

**Exemple**

$$30! = 30 \times 29 \times \cdots \times 2 \times 1 = \\ = 265\,252\,859\,812\,191\,058\,636\,308\,480\,000\,000$$

**Exemple**

C'est l'environnement `multline*` qui nous permet de nous en sortir ; attention, il n'est pas disponible « d'origine » : il ne le sera que si nous avons, dans le préambule, chargé le module `amsmath` ; cet environnement justifie la première ligne à gauche (moyennant un petit retrait) et la dernière à droite (avec le même retrait) ; s'il y a plus de deux lignes, les autres sont centrées. Le séparateur de lignes est une double controblique : « `\` ». Le code de l'exemple précédent est

```
\begin{multline*}
30 !=
30\times29\times\cdots\times2\times1=\!
=265\,252\,859\,812\,191\,058\,636\,308\,
480\,000\,000
\end{multline*}
```

Nous avons voulu, dans le développement décimal du grand nombre entier, séparer par un petit espace les groupes de trois chiffres, selon l'usage. Comme, en mode mathématique, les blancs sont ignorés, il faut utiliser une commande pour ce faire : « `\,` » (au passage, mentionnons les commandes « `\ :` » et « `\ ;` », qui insèrent des espaces un peu plus grand et encore plus grand (2)).

Notons aussi les points de suspension centrés, produits par `\cdots`, avec `c` pour *centered*.

Si, au lieu de `multline*`, nous avons employé l'environnement `multline`, notre équation de deux lignes

aurait été numérotée, automatiquement, dans la même séquence que les équations contenues dans des environnements `equation`. La plupart des environnements qui gèrent les équations (nous en découvrirons encore quelques-uns) ont une variante étoilée, produisant des équations non numérotées et une variante non étoilée, produisant des équations numérotées. Ainsi, il existe un environnement `equation*` : c'est simplement un autre nom de l'environnement `displaymath`.

Une autre situation assez fréquente est celle des « tableaux d'équations » : plusieurs équations qui se suivent et dans lesquelles, pour l'esthétique autant que pour la lisibilité, des symboles doivent être alignés verticalement. Voici deux cas.

**Exemple**

$$f(x, y, z) = xe^{xyz^2} ; \\ \partial_1 f(x, y, z) = (1 + xyz^2) e^{xyz^2}, \\ \partial_2 f(x, y, z) = x^2 z^2 e^{xyz^2}, \\ \partial_3 f(x, y, z) = 2x^2 yz e^{xyz^2}.$$

**Exemple**

**Exemple**

$$(a + b)^2 = (a + b) \cdot (a + b) \\ = a^2 + ab + ba + b^2 \\ = a^2 + 2ab + b^2.$$

**Exemple**

En principe, l'environnement (de base) `eqnarray*` produit ce genre de dispositions ; mais il a le mauvais goût de mettre trop d'espace autour des symboles « pivots », ceux que l'on veut aligner ; il vaut donc mieux l'oublier une fois pour toutes et s'en remettre à `align*` (qui nécessite le module `amsmath`).

Voici comment est codé le premier exemple :

```
\begin{align*}
f(x,y,z)&=xe^{\{xyz^2\}}\text{ ;}\! \\
\partial_1 f(x,y,z)&= \\
\left(1+xyz^2\right)e^{\{xyz^2\}}\! \\
\partial_2 f(x,y,z)&= \\
x^2z^2e^{\{xyz^2\}}\! \\
\partial_3 f(x,y,z)&= \\
2x^2yze^{\{xyz^2\}}. \\
\end{align*}
```

(2) De quelles dimensions, me demandera le lecteur ? Cette question n'a pas de vraie réponse, car en T<sub>E</sub>X la plupart des espaces sont élastiques : ils se compriment ou s'étirent, dans une certaine mesure, pour permettre la justification des lignes à gauche et à droite.



et le second :

```
\begin{align*}
(a+b)^2&=(a+b)\cdot(a+b)\\
&=a^2+ab+ba+b^2\\
&=a^2+2ab+b^2.
\end{align*}
```

Comme dans *multiline\**, les lignes sont séparées par une double controblique ; et les points de chaque ligne qui doivent être alignés verticalement sont définis par les esperluettes (&) ; pour être précis, ce ne sont donc pas les signes d'égalité en tant que tels qui sont alignés, mais leurs extrémités gauches ; comme ces symboles sont de même largeur (et il en va de même pour les différents symboles d'inégalité), cela revient au même. Dans le dernier exemple, les deuxième et troisième lignes commencent par l'esperluette : les membres de gauche sont vides !

L'environnement *align* fait de même, en numérotant chaque ligne du tableau d'équations.

Remarquons encore, au passage, à la première ligne du dernier exemple, le point multiplicatif, produit avec la commande `\cdot` (sans *s* !) ; un simple point de ponctuation se serait trouvé sur la ligne d'écriture plutôt que sur l'axe de la formule (l'axe de symétrie horizontal des symboles +, =, etc.), et n'aurait pas été entouré d'espaces : le résultat aurait été tout simplement hideux !

## Les quatre styles mathématiques

Lorsque nous codons des maths pour T<sub>E</sub>X, quatre styles sont disponibles : *displaystyle*, *textstyle*, *scriptstyle* et *scriptscriptstyle*. En principe, nous n'avons pas à nous en soucier : le logiciel gère pour nous le choix de l'un ou de l'autre. Mais il peut arriver que nous souhaitions outrepasser les choix par défaut ; il importe donc de savoir comment faire et quelles en sont les conséquences.

En principe, le style *hors-texte* (*displaystyle*) est celui des équations hors-texte et le style *du texte* (*mathématique*) (*textstyle*) est celui que nous obtenons lorsque nous basculons vers le mode mathématique au moyen de dollars simples. Viennent ensuite le style *d'indice* (*scriptstyle*), dans lequel apparaissent les indices et les exposants, et le style *d'indice d'indice* (*scriptscriptstyle*), qui s'applique aux indices ou exposants « de second niveau » ; les indices et exposants qui seraient d'un « niveau » strictement supérieur à 2 sont toujours composés en *scriptscriptstyle*. Si nous construisons une frac-

tion alors que nous sommes en *displaystyle* (resp. *textstyle*, *scriptstyle*, *scriptscriptstyle*), son numérateur et son dénominateur seront en *textstyle* (resp. *scriptstyle*, *scriptscriptstyle*, *scriptscriptstyle*). Remarquons que, si le module *amsmath* a été inclus, il existe une commande `\dfrac{...}{...}` qui compose la fraction comme si le style était hors-texte (c'est-à-dire avec de grands numérateur et dénominateur) et une commande `\tfrac{...}{...}` qui la compose comme si le style était texte (c'est-à-dire avec de petits numérateur et dénominateur, en taille d'indice).

L'indice d'un radical est en *scriptscriptstyle*.

Le premier effet du style, le plus visible, est de régler la taille des caractères : en *displaystyle* et en *textstyle*, ils sont du même corps que le texte qui entoure l'équation ; en *scriptstyle*, ils sont plus petits et en *scriptscriptstyle*, plus petits encore. Mais un examen attentif révèle d'autres subtilités : les espacements ne sont pas simplement augmentés ou réduits en proportion du corps du lettrage ; la hauteur des exposants ou des indices diffère ; etc.

Une autre différence importante entre le style hors-texte et les autres est la taille des « grands opérateurs » (principalement : le symbole de sommation, le symbole de produit et l'intégrale) et le placement de leurs bornes (qui, techniquement, sont des indices ou des exposants).

Lorsque nous souhaitons modifier le style que T<sub>E</sub>X a choisi pour nous, nous utilisons une des *déclarations* `\displaystyle`, `\textstyle`, `\scriptstyle` et `\scriptscriptstyle`. Le terme « déclaration » désigne une commande dont les effets vont rester en vigueur soit jusqu'à un contrordre (une nouvelle déclaration qui annule les effets de la première), soit jusqu'à la fin du groupe qui la contient (les environnements, délimités par leurs balises `\begin{...}` et `\end{...}`), sont de tels groupes ; mais est aussi un groupe tout ce qui se trouve à l'intérieur d'accolades appariées).

Voici quelques exemples commentés.

### Exemple

$$2^{2^{2^2}}$$

Exemple

Le codage est

```
\[2^{2^{2^{2^2}}}\]
```



Nous observons que le premier « 2 » est de la taille du texte, que le suivant est plus petit, le troisième encore plus petit ; à partir de là, les tailles ne diminuent plus.

### Exemple

$$\frac{2a}{3b}, \frac{2a}{3b}, \frac{2a}{3b}, \frac{2a}{3b}, \frac{2a + \frac{4c}{d}}{3b}$$

### Exemple

Code :

```
\frac{2a}{3b},
{\displaystyle\frac{2a}{3b}},
\frac{2a}{3b},\dfrac{2a}{3b},
\dfrac{2a+\frac{4c}{d}}{3b}
```

Nous sommes ici en style texte, donc la première fraction est « petite » (ses numérateur et dénominateur sont en style d'indice); la suivante se trouve après la déclaration `\displaystyle`, donc elle est « grande » ; mais la deuxième ligne du code est contenue dans une paire d'accolades, donc elle constitue un groupe, et la portée de la déclaration ne dépasse pas le bout de cette ligne ; la troisième fraction est donc à nouveau « petite » ; et si l'avant-dernière est « grande », c'est parce qu'elle est produite par la commande `\dfrac` plutôt que simplement par `\frac` ; enfin, la dernière est « grande » grâce à la commande `\dfrac` ; cela signifie, entre autres, que son numérateur est en style texte ; la fraction qui se trouve dans ce numérateur est donc une « petite » fraction.

C'est une question de goût, mais je n'apprécie guère ces « petites » fractions lorsqu'elles contiennent des variables littérales (tant que c'est numérique, cela passe) ; donc, personnellement, je choisirais plutôt d'écrire :

$$\frac{2a + 4c/d}{3b}.$$

### Exemple

Nous savons évidemment, depuis bien longtemps, que  $\iint_{(x,y) \in [0;1]^2} 1/(1-x^2y^2) = \sum_{n \geq 0} 1/(2n+1)^2$  ; insistons :

$$\iint_{(x,y) \in [0;1]^2} \frac{1}{1-x^2y^2} = \sum_{n \geq 0} \frac{1}{(2n+1)^2}.$$

### Exemple

Le but de cet exemple est de bien mettre en évidence la différence de comportement des grands

opérateurs que sont `\sum` et `\iint` (l'intégrale double) selon qu'ils sont en style texte ou hors-texte. Indiquons le code.

Nous savons évidemment,

depuis bien longtemps, que

$$\iint_{(x,y) \in [0;1]^2} 1/(1-x^2y^2) = \sum_{n \geq 0} 1/(2n+1)^2 ;$$

insistons :

```
[\iint\limits_{(x,y) \in [0;1]^2}
\frac{1}{1-x^2y^2}=
\sum_{n \geq 0} \frac{1}{(2n+1)^2}.\]
```

Quelques commentaires.

- Nous avons écrit plus haut que la position des bornes des grands opérateurs dépendait du style, hors-texte ou pas ; pour la somme (et le produit), c'est bien le cas, comme le montre cet exemple ; mais pas pour l'intégrale (simple, double ou triple, la question n'est pas là) ; en codant `[\int_{x=a}^{x=b} f(x), \]` nous obtenons

$$\int_{x=a}^{x=b} f(x),$$

conformément à l'usage. Mais ici, pour cette intégrale double, nous souhaitons que le domaine d'intégration apparaisse *sous* le « double portemanteau » ; nous l'avons forcé à le faire en intercalant la commande `\limits` entre `\iint` et le trait de soulignement qui introduit l'indice ; pour forcer, au contraire, l'indice à venir à droite du symbole plutôt qu'en dessous, nous utiliserions, de même, `\nolimits`.

- La commande `\geq` produit l'inégalité  $\geq$ , fréquente sans doute dans le monde anglosaxon, mais à laquelle on est en droit de préférer  $\gtrsim$  ; ce symbole-ci s'obtient avec la commande `\geqslant`, pourvu que le module `amssymb` ait été chargé ; la même chose, bien sûr, fonctionne avec `\leq` ( $\leq$ ) et `\leqslant` ( $\lesssim$ ). Si vous trouvez ces nouvelles commandes trop longues à taper, rien ne vous empêche de leur donner de nouveaux noms plus courts (p. ex. `\geq` et `\leq`, si vous êtes certains de ne plus vouloir utiliser  $\geq$  et  $\leq$ ) ; nous reparlerons de ce problème de (re)définition de commandes dans une rubrique ultérieure.
- Au sujet des intervalles, il y aurait plusieurs choses à dire. Nous y reviendrons. Notons pour l'instant qu'avec un intervalle fermé, il n'y a généralement pas de problème.