

# Sur les codes à barre EAN

Laurent Guillopé

*Institut Fourier, Laboratoire de Mathématiques, URA 188 du CNRS, Université de Grenoble I,  
B. P. 74, F-38402 Saint-Martin d'Hères Cedex.  
Email : guil@frgren81.bitnet*

**Résumé** On décrit la structure des codes à barre EAN (un code EAN13 orne souvent la dernière de couverture d'un livre) et un ensemble de macros  $\TeX$  utilisé pour les réaliser. On termine par quelques remarques sur la réalisation effective.

**Abstract** We describe the structure of the barcodes of the EAN family (a EAN13 code is often printed nowadays on the last cover-page of a book) and a set of  $\TeX$ -macros used to realize them. We end with some remarks on the effective realisation.

(...) les lettres O, A, M, R et L surmontent des lignes aux destins divers : la ligne O reste droite, les lignes A et M se rapprochent puis s'éloignent, les lignes R et L, longtemps parallèles, finissent par se rejoindre.

G. Perec, *Still life/Style leaf, Le fou parle*, n° 18, septembre 1981.

Afin de renforcer l'informatisation de la gestion du prêt, la bibliothèque de l'Institut Fourier (Laboratoire de Mathématiques, URA 188 du CNRS, Université de Grenoble I) a décidé d'équiper tous les ouvrages de son fonds (21 000 volumes) d'un code à barre. Pour profiter au maximum du catalogage informatique existant, une édition locale de ces codes a paru la meilleure solution. On décrit ici la structure des codes à barre de type EAN13 et les quelques macros mises en place pour leur réalisation.

## 1. Anatomie

Les codes de la famille EAN<sup>1</sup> (on se limitera à la description du code EAN13, un EAN8 existe aussi) sont des codes

<sup>1</sup> Association internationale rassemblant des organisations de codification. Le GENCOD (Groupement d'études de normalisation et codification, 13 bd Le-fèvre 75015 Paris) y représente la France.



Figure 1: Le code à barre 9 782020 108997

numériques. Le code EAN13 (utilisé dans la distribution, et notamment sur la dernière de couverture des livres) représente une suite de 13 chiffres  $(c_i)_{0 \leq i \leq 12}$ , dont seuls 12 sont indépendants : est imposée la congruence linéaire

$$\sum_{i=0}^6 c_{2i} + \sum_{i=1}^6 3 * c_{2i-1} \equiv 0 \quad [10],$$

vérifiée par tout instrument-lecteur au moment de la lecture (cette condition de cohérence limite les éventuelles erreurs de lecture, rares au demeurant).

Un code à barre imprimé est un symbole de deux parties à six caractères ( $d = (d_i)_{1 \leq i \leq 6}$  et  $g = (g_i)_{1 \leq i \leq 6}$ ), jointes par un séparateur central et encadrées par un indicatif de début et de fin. Les barres de ces trois séparateurs sont légèrement plus longues que celles constituant les caractères numériques ; on conviendra que ces dernières sont sans profondeur et en en donnant une aux caractères-séparateurs, l'humain distinguera sans ambiguïté la partie gauche  $g$  de la partie droite  $d$  du symbole.

En fait, il y a trois familles ( $A, B, C$ ) de code-caractères, ceux de la famille  $A$  ( $B, C$ ) étant impairs (*resp.* pairs). Chacun des douze caractères du symbole

0	AAAAAA
1	AABABB
2	AABBAB
3	AABBBA
4	ABAABB
5	ABBAAB
6	ABBBAA
7	ABABAB
8	ABABBA
9	ABBABA

Table 1:  $c_0$  et son  $g$ -mot des familles associé

0	bbbnnbn
1	bbnnbbn
2	bbnbbnn
3	bnnnnbn
4	bnbbbnn
5	bnnbbbn
6	bnbnbbb
7	bnnnbn
8	bnnbnn
9	bbbnnbn

Table 2: Le  $\mathcal{A}$ -mot des couleurs d'un chiffre

(codant un chiffre :  $g_i \sim c_i$ ,  $d_i \sim g_{6+i}$ ) appartient à l'une d'entre elles ; le mot-famille à douze lettres correspondant est déterminé par le chiffre  $c_0$ . Les caractères de la partie droite sont tous de la famille  $C$ , le tableau 1 indique comment le chiffre  $c_0$  détermine le mot des familles de  $g$ . Cette partie gauche contient au moins un caractère pair et un caractère impair, ceci induit une orientation pour le code et permet ainsi à l'instrument-lecteur de le reconstituer indépendamment de la manière dont il est présenté.

Graphiquement, un code-caractère est la donnée de sept traits adjacents (de largeur `\unit`) bicoloriés (avec une couleur claire et une foncée, en général blanc et noir) telle qu'apparaissent quatre barres ; un code-caractère est dit pair si le nombre de traits foncés l'est. Cette donnée est exprimable par un mot en les lettres `b` et `n`, représentant les couleurs claire et foncée *resp.* Le tableau 2 liste ces mots pour la famille  $\mathcal{A}$ . Les autres familles s'obtiennent par symétries : la famille  $C$  à partir de  $\mathcal{A}$  par symétrie de couleurs et la famille  $B$  à partir de  $C$  par symétrie spatiale par rapport au caractère central du mot. Par *ex.*, 0 est associé aux mots `bbbnnbn`, `bnbbnbn` et `nbnbbnn` dans les familles  $A$ ,  $B$ ,  $C$  (*resp.*). Les

séparateurs `\d`, `\c`, `\f` sont tracés suivant les mots `nbn`, `bnbnb`, `nbn` *resp.*

La norme indique une taille nominale de 0.33 mm, 22.85 mm, 24.50 mm pour `\unit`, `\haut`, `\prof` *resp.*, avec une latitude permise de 80% à 300%.

## 2. Genèse

Soit le code à barre 9 782020 108997 (fig. 1), qu'on écrira en  $\text{\TeX}$

```
\dA7\B8\B2\A0\B2\A0\c\c1\c0\c8\c9\c9\c7\c
```

où `\d` `\c` `\f` désignent les séparateurs de début, de milieu et de fin *resp.* et `\A7` le chiffre 7 dans la famille  $\mathcal{A}$  (...).

Le caractère `\A7` se développe en

```
\b\n\n\n\b\n\n.
```

On peut définir alors simplement `\n` par un `\vrule width\unit height\haut` et `\b` par `\kern\unit`.

Pour limiter la taille des fichiers `dvi` générés, il est préférable (quitte à demander à  $\text{\TeX}$  de faire davantage d'arithmétique, rallongeant ainsi le temps de compilation) de procéder autrement. Deux compteurs `\Cb`, `\Cn` sont introduits. La macro `\b` incrémente d'une unité le registre `\Cb`, après avoir vidé le registre `\Cn` en un

```
\vrule width\Cn\unit height\haut
```

si le registre `\Cb` était nul. La macro `\n` opère de manière analogue et les séparateurs `\d`, `\c` et `\f` sont définis dans le même esprit (avec une profondeur `\prof` non nulle pour les barres noires).

### 3. Remarques

Il est très important de choisir la largeur des traits élémentaires comme un multiple entier de la dimension unitaire du dispositif d'impression. Avec une résolution de 300 points par pouce, on définira simplement `\unit` par

```
\unit=4in\divide\unit by 300
```

Une `\unit` incommensurable à cette dimension unitaire donne éventuellement des codes à barre très fantaisistes. Ces apparences d'erreurs<sup>2</sup> sont dues aux arrondis effectués par le pilote d'impression lors des conversions en *points* (d'imprimante) des dimensions horizontales représentant la largeur des barres des codes.

Notons par ailleurs les difficultés techniques rencontrées lors d'impressions à grande échelle. Le choix du papier-étiquette est très important : il ne doit pas être trop glacé (l'encre n'adhère pas, même avec du papier convenable il est conseillé de recouvrir le code à barre par un scotch de bonne qualité, ie ne jaunissant pas avec l'âge) et ne doit pas déposer la moindre trace de colle sur le rouleau de cuisson de l'imprimante (sinon au bout d'une dizaine de feuilles, l'imprimante s'arrête avec le message `paper jam`). Par expérience, il ne semble pas toujours facile de trouver le papier adéquat (ou un fournisseur qui ne change pas inopinément son type de papier...).

Par ailleurs, les aplats rendus par les imprimantes à laser *première génération*

---

<sup>2</sup>C'est Y. Roy qui m'expliqua ce phénomène resté très longtemps mystérieux.

(du type LaserJet ou LaserWriter) n'étaient pas très profonds et rendaient parfois un contraste *couleur foncée/couleur claire* entre le noir et le blanc des codes à barre insuffisamment net (ainsi certains codes à barre d'une planche n'étaient pas lisibles), il semblerait que ce problème soit maintenant résolu avec la technologie récente (mise en œuvre par *ex.* dans les LaserJet II ou LaserWriter NTX)

Pour générer l'expression d'un code à barre en une séquence d'ordres `TeX`, on a utilisé un *préprocesseur* dont l'élément central est la fonction `trans_code()` donnée en annexe. Nous laisserons à d'autres le soin de l'implanter en une macro `TeX` qu'on appellerait suivant `\eantreize9782020108997`.

Un usage fréquent de codes à barre (les journaux d'annonce de programmes de télévision utilisent des codes à barre ainsi que l'administration des *Mathematical Reviews*<sup>3</sup>, ces codes à barre ne sont pas du type EAN certes) réclame-t-il l'existence de polices ? Si certains constructeurs, tels Xerox ou HP, en ont à leur catalogue, la création de fichiers de polices (`pk`, `pxl...`) paraît exagérément lourde. L'introduction de *polices virtuelles*, dont Knuth vient de définir<sup>4</sup> le format, est bien plus appropriée (du moins pour une application avec une production importante de codes à barre).

---

<sup>3</sup>`TeX` at Mathematical Reviews, d.c. Lattner et w.b. Wolf, *Tugboat*, 10, 1989, 639-654.

<sup>4</sup>`TeX`hax V90 #11, #12 et #13.

## 4. Annexe

### 4.1. Quelques macros

Nous laissons au lecteur le soin de définir les caractères des familles  $\mathcal{A}$ ,  $\mathcal{B}$ ,  $\mathcal{C}$  en termes des macros élémentaires `\b` et `\n`.

```
\newdimen\unit
\unit=1in
\multiply\unit by 4
\divide\unit by 300
\newcount\Cn
\newcount\Cb
\def\Barre{\vrule height35pt width\unit depth 6pt}
\def\barre#1{\vrule height35pt width#1\unit depth 0pt}
\def\n{\ifnum\Cn>0\advance\Cn by 1\else\kern\Cb\unit\Cb0\Cn1\fi\relax}
\def\b{\ifnum\Cb>0\advance\Cb by 1\else\barre\Cn\Cb1\Cn0\fi\relax}

\def\d{\Barre\kern\unit\Barre\Cn0\Cb0\relax}      %separateur de debut
\def\c{\ifnum\Cn>0\barre\Cn\kern\unit\else\advance\Cb by1\kern\Cb\unit%
\fi\Barre\kern\unit\Barre\Cn0\Cb1\relax}      %separateur central
\def\f{\ifnum\Cn>0\barre\Cn\else\kern\Cb\unit\fi\Barre%
\kern\unit\Barre\relax}      %separateur de fin
```

### 4.2. Un générateur de séquence- $\text{\TeX}$ de codes à barre

```
char * codage[]={"AAAAAA","AABABB","AABBAB","AABBBA","ABAABB",
                "ABBAAB","ABBBAA","ABABAB","ABABBA","ABBABA"};

trans_code(int *code)
{
    int i;

    printf("\\d");
    for(i=1;i<7;i++)printf("\\c%d",codage[code[0]][i-1],code[i]);
    printf("\\c");
    for(i=7;i<13;i++)printf("\\C%d",code[i]);
    printf("\\f");
}
```