

# *Cahiers* **GUT** *enberg*

☞ FAIRE UN GROS OUVRAGE PORTABLE EN

L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

☞ Daniel TAUPIN

*Cahiers GUTenberg*, n° 21 (1995), p. 96-106.

<[http://cahiers.gutenberg.eu.org/fitem?id=CG\\_1995\\_\\_21\\_96\\_0](http://cahiers.gutenberg.eu.org/fitem?id=CG_1995__21_96_0)>

© Association GUTenberg, 1995, tous droits réservés.

L'accès aux articles des *Cahiers GUTenberg*

(<http://cahiers.gutenberg.eu.org/>),

implique l'accord avec les conditions générales

d'utilisation (<http://cahiers.gutenberg.eu.org/legal.html>).

Toute utilisation commerciale ou impression systématique

est constitutive d'une infraction pénale. Toute copie ou impression

de ce fichier doit contenir la présente mention de copyright.

# Faire un ouvrage portable en $\text{\LaTeX} 2_{\epsilon}$

---

Daniel TAUPIN

*Laboratoire de Physique des Solides*  
*F-91405 Orsay cedex*  
`TAUPIN@rsouvax.lps.u-psud.fr`

**Abstract.** *Writing a big book in  $\text{\LaTeX} 2_{\epsilon}$  with many figures raises several problems of “ $\text{\TeX}$  capacity” and also some portability problems between the various kinds of printers and drivers. We solved that challenge by inserting figures in PCX code, and using several well known and more confidential MS-DOS tools which enable enlarging and shrinking these figures, although minimising distortions and information losses.*

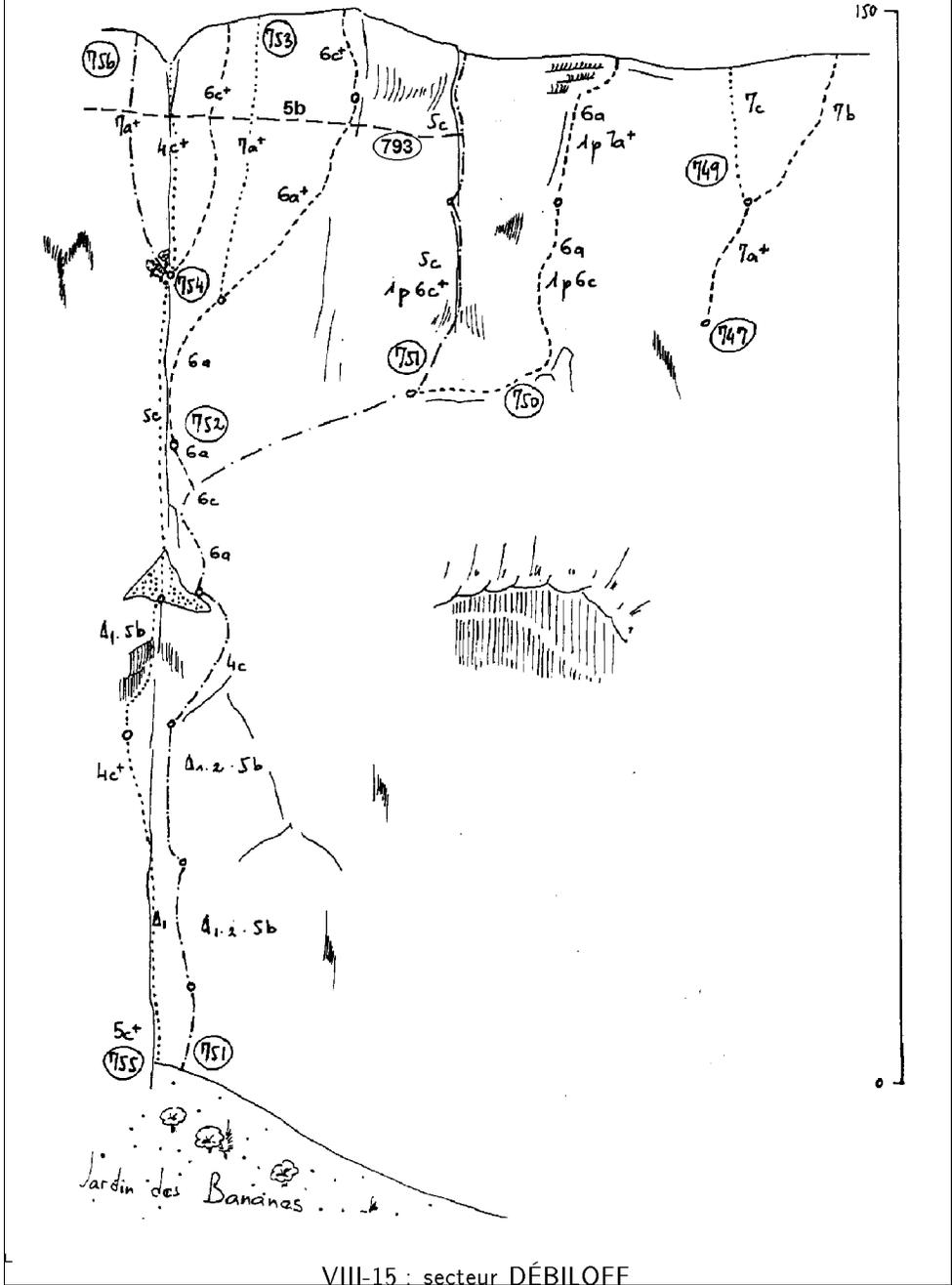
**Résumé.** Écrire un gros ouvrage en  $\text{\LaTeX} 2_{\epsilon}$  avec beaucoup de figures pose des problèmes de «  $\text{\TeX}$  capacity » et de portabilité entre les divers types d'imprimantes et de pilotes. Nous avons résolu le problème en incorporant les figures en PCX, et en utilisant plusieurs utilitaires MS-DOS plus ou moins connus permettant d'agrandir ou de réduire ces figures tout en minimisant les distorsions et les pertes d'information.

## 1. Exposé du problème

### 1.1. La taille de l'ouvrage

Il s'agit d'un *topo-guide* d'escalade consacré aux *Gorges du Verdon* ; cela dit, l'important n'est pas le sujet traité — sauf pour quelques amateurs éclairés — mais le volume du document :

- 352 pages au format A5,
- trois index alphabétiques,
- texte introductif en cinq langues (français, anglais, allemand, italien, espagnol),
- une partie des indications sous forme de pictogrammes (cf. page 98),
- et surtout : 160 figures incluses dans le texte (cf. 97).



**VIII-15 DÉBILOFF**accès : pp. 94, 116 ;  9-16

- 756 – LE MANÈGE ENCHANTÉ** 7a+, 35 m, ↓<sup>752</sup> ;  ;  (1983).
- 755 – SIDARDENDAILLE** A2 & 5c+, TD, 150 m, ↓<sup>792</sup> ;  (20) .
- 754 – DÉBILOFF PROFFONDICUM** 6c+, 30 m, ↓<sup>752</sup> ;    ;  (1991).
- 753 – SINFONIE DEL VENTO** 7a+, 30 m, ↓<sup>752</sup> ;   ;  (1991).
- 752 – TURLUT** 6c+(6c), TD+, 80 m, ↓ ;   ;  (1986).
- 751 – QUI S'Y FROTTE S'Y PIQUE** A2 & 6c+, TD+, 160 m, 4 h 30, ↓<sup>752</sup> ;  ;   (1983) ;  .
- 750 – AU BORD DES ABÎMES** 7a+, TD+, 80 m, ↓<sup>752</sup> ;  ;   ; .
- 749 – LES JUPONS CAPITEUX** 7c, ↓ ;  (1988).
- 747 – BOULEVARD DES STARS** 7b(7a), ED-, 60 m, ↓ ;   ;  (1989).

# Achtung auf Blitzschlaggefahr!

Meiden Sie den Felsrand bei Gewitter:  
dort schlägt der Blitz mit Vorliebe ein!

# Attenti al fulmine !

In caso di temporale, andate via dall'orlo delle  
falesie : questi sono i traguardi più favoriti dal ful-  
mine.

---

## 1.2. Les contraintes

### 1.2.1. Les imprimantes destinataires

Elles sont au nombre de trois, et bien sûr le DVI doit être unique, sous peine de risque d'erreurs catastrophiques, toujours en dernière minute selon la *loi de Murphy* :

1. une imprimante HP Laserjet 4 $\ell$  (300 dpi, langage PCL) pour les épreuves de travail usuelles,
2. une imprimante HP Laserjet 4*si* (300/600 dpi, langages PCL et PostScript) pour les épreuves finales,
3. la photocomposeuse de l'imprimeur, auquel nous fournissons l'intégralité de l'ouvrage — sauf quelques photos additionnelles — sous forme d'une douzaine de fichiers PostScript 600 dpi, *zippés* sur des disquettes MS-DOS.

### 1.2.2. Les ordinateurs utilisés

Pour le travail ordinaire, il s'agit simplement de deux 486 DX33, avec **emTeX** (**tex386**), **Version 3.1415 [3c-beta12]** et les pilotes associés (version 1.5h).

Pour les épreuves finales, il s'agit d'un HP 9000 (HP\_UX) avec METAFONT 2.71 et **dvips** 5.516, configurable pour obtenir, soit du PostScript 300 dpi, soit du 600 dpi. Nous avons évidemment utilisé cette deuxième option.

## 2. Les problèmes mineurs

La figure en page 98 montre que le style est différent du standard de L<sup>A</sup>T<sub>E</sub>X, mais ceci ne mérite pas qu'on s'y attarde : il s'agit essentiellement de problèmes de mise en page dont le plus délicat est de dupliquer la figure de la page paire, si la page impaire déborde de la limite du `\textheight`. De même, un topo-guide d'escalade performant comporte trois index :

1. un index général, pour les rubriques pratiques, documentaires, touristiques, sécuritaires, etc.,
2. un index des itinéraires d'escalade (933 en l'occurrence) classés par ordre alphabétique ; l'ordre alphabétique des lettres accentuées et des `\OE` pose quelques problèmes à **makeindex** (version 2.12), que nous avons résolu à l'aide d'un programme Fortran faisant une conversion préalable des clés de classement lues par **makeindx 2.12**. On aurait pu rechercher la version 3.0, mais il y avait le risque de découvrir *in extremis* que certains cas particuliers n'étaient peut-être pas résolus...

3. un autre index des itinéraires d'escalade, classés cette fois-ci par ordre de difficulté croissante et, à difficulté égale, par ordre alphabétique ; il s'agissait du même problème que ci-dessus, sauf quelques difficultés dues aux symboles spéciaux comme "." et "□&□", résolues par le petit programme de réorganisation des clés de tri.

### 3. Le problème des figures

#### 3.1. Le choix du format de représentation

À partir du moment où nous avons annoncé que la photocomposeuse finale lisait le PostScript, il paraissait évident que nous *devions* avoir des figures codées en *encapsulated PostScript*. Pourtant cette évidence était contredite par deux contraintes :

1. les *scanners* dont nous disposons produisent des fichiers aux formats BMP, TIFF et PCX. Ils ne produisent pas d'*encapsulated PostScript*.
2. les pilotes d'Eberhardt MATTES tournant sur 386/486 savent inclure des figures sous les formes :

```
\specialfem: graph < fichier >.pcx}
```

```
\specialfem: graph < fichier >.msp}
```

mais ils ne savent pas inclure de l'*encapsulated PostScript*.

En revanche, que ce soit sur 386/486 ou sur UNIX, **dvips** 5.516 comprend — ô miracle ! — exactement la même commande **\special** que les pilotes d'Eberhardt MATTES.

D'où la conclusion que, quel que soit le langage de l'imprimante, PCL ou PostScript, le format le plus portable pour les images est PCX ou MSP, puisqu'il permet à la fois de générer du PCL pour les Laserjets bon marché, qu'il est produit par nos *scanners* sur 486 (logiciel Scanjet sous Windows), et qu'il est manipulable par PaintBrush (sous Windows) pour des éventuelles retouches. Nous avons donc choisi PCX.

#### 3.2. Les difficultés avec PCX

##### 3.2.1. Problèmes de retouches

Les dessins originaux étaient faits au crayon ou au feutre, avec de possibles grattages et des traits parasites dus aux bords du papier. Quel que soit le soin

---

apporté au réglage du *scanner*, il était toujours nécessaire de retoucher les fichiers PCX.

Pour ce faire, le logiciel PaintBrush de MicroSoft, inclus dans le progiciel Windows fait l'affaire. Mais son défaut (“that’s not a bug, that’s a feature!”) est que les indications de taille d’image et de résolution sont « caviardées » après chaque manipulation<sup>1</sup>, et ceci entraîne des rejets avec d’autres logiciels que nous exposerons plus loin.

Fort heureusement, les pilotes d’Eberhardt MATTES ont le mérite de fonctionner de manière *basique* : ils ignorent toutes les indications sujettes à caution et ne considèrent que les pixels de l’image PCX, qu’ils traitent avec la résolution spécifiée pour l’imprimante, par exemple 300 ou 600 dpi.

En plus, depuis la version 1.4r, les pilotes d’Eberhardt MATTES offrent une option qui produit non plus du PCL ni des *bitmaps* destinés aux imprimantes simplistes comme les BJ10\*, mais du PCX. La commande sur 386/486 est assez lourde, mais elle marche parfaitement :

```
dvidrv dvidot pcx @lj.cnf ;entrée; ;sortie; /l0pt /t0pt /w210mm
/h290mm ;options;
```

où `lj.cnf` est une configuration d’imprimante — ici Laserjet 300 dpi — compatible avec la résolution du DVI, où `entrée;` est le nom du fichier DVI dont on veut faire un PCX, où `sortie;` est le nom du fichier PCX voulu comme résultat, et où `options;` sont des options complémentaires décrites dans `dvidrv.doc` de la distribution d’Eberhardt MATTES.

Parmi les options de ces pilotes, l’une d’elle est particulièrement utile, à savoir `/om+`. Elle a pour effet d’enlever de l’image résultante toutes les marges supérieure, inférieure, gauche et droite qui seraient remplies de blancs. De ce fait, pour « nettoyer » un fichier `monimage.pcx` qui vient, soit du *scanner*, soit d’être retouché par PaintBrush, il suffit d’écrire un petit programme `TeX`, par exemple `pcxclean.tex` :

```
\shipout\vbox{\null\special{em: graph monimage.pcx}}\bye
```

de le `TeX`er et d’exécuter ensuite :

```
dvidrv dvidot pcx @lj.cnf pcxclean monimage.pc3 /l0pt /t0pt
/om+ /w210mm /h290mm
```

où `monimage.pc3` est la version « propre » de `monimage.pcx`, ici en 300 dpi. Il faudra évidemment la renommer pour les usages ultérieurs.

---

1. Sans doute parce que PaintBrush suppose une résolution de 72 dpi et non de 300.

*REMARQUE*: contrairement à ce qu'on peut craindre, ces manipulations sont inoffensives quant à la qualité de l'image ; en effet l'information dans les fichiers PCX (et la plupart des autres formats graphiques) est stockée pixel par pixel : il n'y a pas de perte d'information... tant qu'on reste dans le domaine du **noir et blanc**.

### 3.2.2. Problèmes de dimensions de figures

À partir du moment où on agrandit les figures, et quand en plus on coupe les marges blanches ou devenues blanches, l'insertion des figures dans le fichier  $\LaTeX$  amène à des tâtonnements pénibles.

Nous citons donc un petit logiciel dû à Henk HUYSMAN (Delft University), `pcx2piq` qui lit les informations contenues dans le fichier PCX et produit le morceau de fichier  $\LaTeX$  à insérer pour inclure la figure PCX. Voici un exemple de résultat :

```
\unitlength=1mm
\begin{picture}(119.21,78.74)
\put(0,78.74){\special{em:graph voutn.pcx}}
\end{picture}
```

Notons que `pcx2piq` ne tient compte que des dimensions comptées en pixels dans le fichier, et les multiplie par la résolution de l'imprimante.

### 3.2.3. PCX 300 dpi et PCX 600 dpi

À la fois les pilotes de Eberhardt MATTES et `dvips` de Tom ROKICKI lisent « bêtement » les fichiers PCX, ce qui tous calculs faits est un compliment car on sait que la difficulté en informatique n'est pas d'obtenir d'un logiciel qu'il fasse ce qu'on désire, mais de l'empêcher de faire ce qu'on ne veut pas. Il en résulte quand même que si l'on a un PCX de résolution 300 dpi et qu'on le communique à `dvihplj/dvidrv` ou à `dvips` configurés en 600 dpi, le pilote de prendra pour argent comptant, et s'abstiendra de dilater.

De ce fait, il est nécessaire d'avoir des PCX distincts avec *le même nom de fichier*, selon qu'on imprime en 300, en 360, en 240 ou en 600 dpi. Le problème ne se produit pas pour les polices, car dans ce cas le pilote sait qu'il doit multiplier la taille de la police par 300 si on travaille en 300 dpi, et par 600 si l'on est configuré en 600 dpi. Et il appelle METAFONT s'il ne dispose pas de la police voulue dans l'agrandissement voulu.

Dans le cas qui nous concerne, c'est à dire la dualité 300/600 dpi, une solution simple nous a encore été fournie par Henk HUYSMAN :

```
pcx2x monimage.pcx monimage.pc6
```

---

transforme `monimage.pcx` en un fichier PCX 2×2 fois plus grand appelé `minimage.pc6`, qu'il faudra évidemment renommer pour le donner à consommer au pilote PCL ou `dvips`.

La seule difficulté est alors une simple difficulté de gestion : savoir si le fichier `monimage.pcx` est la version 300 ou 600 dpi. D'où l'utilité des sauvegardes<sup>2</sup>...

## 4. La manipulation des fichiers PCX sous MS-DOS/Windows

### 4.1. Corrections et retouches

Le plus commode que nous ayons trouvé est PaintBrush, inclus dans Windows et de ce fait accessible à tous, ou presque.

### 4.2. Dilatation ou rétrécissement de fichiers PCX

#### 4.2.1. `bm2font`

Parmi les diverses fonctions offertes par `bm2font` (dû à Friedhelm SOWA) figure la mise à l'échelle des figures, en spécifiant la dimension finale, soit verticalement, soit horizontalement, soit les deux.

Notons que `bm2font` peut lire du PCX, du MSP, du GIF, du TIFF, et des *bitmaps* bruts. Ce qu'il produit n'est pas un fichier graphique, mais des polices de caractères, c'est-à-dire :

- un ou plusieurs fichiers `*.pk`,
- un ou plusieurs fichiers `*.tfm`,
- un fichier T<sub>E</sub>X dont l'insertion par `\input` insère la figure où il faut... sauf erreur de l'utilisateur.

Initialement nous avons songé à utiliser `bm2font` pour inclure des figures dans un autre ouvrage du même style, mais `bm2font` a un grave inconvénient : il produit une ou deux polices de caractères différentes pour chaque figure. Or le nombre de polices est limité par T<sub>E</sub>X à 255, alors que L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> en réserve environ une centaine. Aussi, au delà d'une centaine de figures, on retombe sur les sempiternels « **TeX capacity exceeded** ».

---

2. Pour notre part, nous avons stocké les PCX 300 sur nos PC, et les PCX sur l'ordinateur où tourne le `dvips` configuré en 600 dpi

On peut contourner cette limite en utilisant les facilités des pilotes d'Eberhardt MATTES, en écrivant un tout petit programme  $\text{\TeX}$ , fabriquant simplement l'image souhaitée et en demandant au pilote de fabriquer un PCX à partir du DVI élémentaire.

#### 4.2.2. `dtpic`

`dtpic` est aussi l'œuvre de Friedhelm SOWA mais, jusqu'à affirmation contraire de l'auteur, ce n'est pas un *freeware*. Toutefois, il nous en a fait cadeau à titre personnel et nous l'avons grandement apprécié.

En fait `dtpic` est une variante de `bm2font` et la différence est qu'il produit directement, soit un fichier PCX, soit un fichier TIFF, à partir de *bitmaps*, de GIF, de TIFF ou de PCX. Pour le reste il possède les mêmes caractéristiques que `bm2font`, notamment l'agrandissement/rétrécissement carré ou rectangulaire.

C'est ce que nous avons couramment utilisé pour les mises à l'échelle de nos figures.

#### 4.2.3. *Graphics Workshop*

Il s'agit là d'un outil *shareware* à 40 \$ que l'on trouve dans les « SimTel ». Pour la France, il est accessible par `ftp` anonyme à :

`ftp.loria.fr` : *directory* pub/pc/simtel20/graphics/grfwk20e.zip

`ftp.jussieu.fr` : *directory* pub/pc/SimTel/msdos/graphics/grfwk20e.zip

et on doit pouvoir le trouver ailleurs en demandant “`quote site index grfwk`”.

Ce programme offre beaucoup de possibilités de conversions, agrandissements, rétrécissements, rotations de multiples de 90°, etc. Contrairement à `bm2font` et `dtpic`, il fonctionne en mode interactif, ce qui est plus agréable pour des transformations isolées, mais gênant pour l'inclure dans des procédures systématiques.

En ce qui concerne les rotations, notons avec regret que les pilotes d'Eberhardt MATTES savent faire tourner les textes... mais pas les figures incluses.

### 4.3. Pour passer du gris au noir-et-blanc

Il s'agit évidemment de l'opération usuellement appelée *tramage*. Friedhelm SOWA a considérablement étudié ce problème<sup>3</sup> et expose comment paramétrer le tramage des photos avec des niveaux de gris dans la documentation de `bm2font`. Précisons que `dtpic` produit les mêmes résultats, avec les mêmes qualités et les mêmes défauts.

---

3. Graphics Workshop semble aussi savoir le faire, mais nous ne l'avons pas testé.

---

Après quelques essais nous avons tenté d'introduire des photos tramées dans l'un de nos topo-guides d'escalade. Il s'agissait en fait de diapositives couleur correctement contrastées, ayant fait l'objet d'un tirage papier ensuite soumis au *scanner* en mode « gray scale » alors que les dessins précédents étaient numérisés en mode « black and white ». Quelques essais avec `dtpic/bm2font` nous ont permis d'obtenir un contraste agréable sur notre imprimante Laserjet 4*l* (300 dpi), avec hélas une trame grossière due à la faible résolution de cette imprimante. Nous donnons en page 106 un exemple.

Nos déboires ont commencé lorsque le même fichier a dû être imprimé sur une Laserjet 4*si*, de résolution 600 dpi : le zone noires devenaient pâles et la photo prenait un aspect « délavé ». Le résultat est devenu encore pire quand le fichier PostScript envoyé à l'imprimeur a produit les premières épreuves.

En fait, quand on lit la documentation de `bm2font`, on remarque que Friedhelm SOWA propose plusieurs paramètres pour rendre les zones sombres un peu moins sombres, sauf si elles sont 100 % noires. En fait les corrections qu'il propose sont fortement liées à la nature de l'imprimante, c'est-à-dire au fait que les pixels élémentaires sont exactement à la résolution, ou qu'ils sont trop maigres, ou un peu baveux.

Notre conclusion a finalement été que les paramètres optimaux du tramage d'une photo à niveaux de gris dépendaient crucialement des caractéristiques de l'imprimante, non seulement des caractéristiques de construction, mais aussi de l'état des réglages le jour du tirage, y compris la qualité du *toner*. De ce fait, il faut réviser le fichier tramé chaque jour et pour chaque imprimante, au prix de nombreux essais.

Si l'imprimante définitive est dans le bureau d'à côté, la partie reste jouable, mais que faire avec la photocomposeuse 2000 dpi d'un imprimeur situé à des centaines de kilomètres? On peut court-circuiter le professionnel pour la typographie ordinaire et les dessins, pas pour les photos de qualité...

