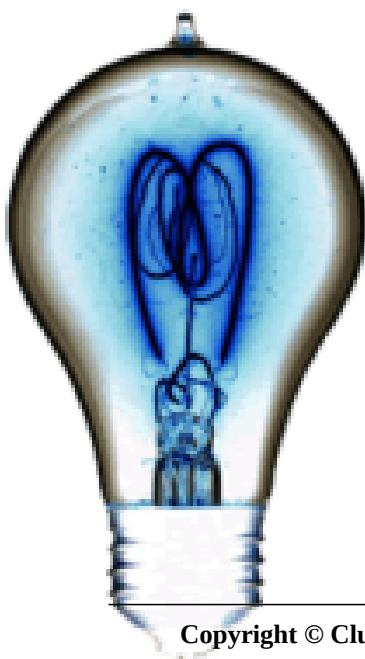


<https://clx.asso.fr/spip/?Imprimer-un-grand-document-sur-des>



Imprimer un grand document sur des feuilles A4

- Documentations - Education - Transition vers le Libre... -



Date de mise en ligne : mardi 30 janvier 2007

Copyright © Club Linux Nord-Pas de Calais - Tous droits réservés

Le driver de mon imprimante ne le faisant pas, j'ai alors cherché sur Internet le moyen de découper mon document et là ... pas beaucoup de solutions, voire pas du tout !

Ainsi, ce document fait un peu plus en hauteur qu'une page A4 en orientation portrait (pas de chance). On va donc le réduire un peu, en enlevant de notre page A4 1cm en bas et en haut, il nous reste 27.7 cm, ce qui nous fait une réduction d'environ 17%. Le coefficient multiplicateur à appliquer est donc 0.83

Notre largeur sera alors d'environ 159.9 cm, presque 1m 60.

Là encore, en enlevant 1 cm de chaque côté de notre page A4, il nous faudra 9 pages.

Le principe

Nous allons faire un document LaTeX constitué de 9 pages, chacune d'elles contenant le bon bout de notre fichier EPS. Pour cela nous utiliserons le package *graphicx* donnant accès à la commande « `\includegraphics` ». Cette commande peut afficher, grâce à l'option *bb*, une partie d'une image EPS en indiquant la zone à afficher par le principe de la « Bounding Box ».

Je dois donc former 9 « Bounding Box » consécutives, les voilà :

`bb=-2700 -36 -2095 908`

`bb=-2095 -36 -1490 908`

`bb=-1490 -36 -885 908`

`bb=-885 -36 -280 908`

`bb=-280 -36 325 908`

`bb=325 -36 930 908`

`bb=930 -36 1535 908`

`bb=1535 -36 2140 908`

`bb=2140 -36 2745 908`

On remarque que la division par 9 ne tombe pas juste, il faut donc réajuster de 4 points postscript notre largeur, la dernière sera donc réajustée à :

`bb=2140 -36 2749 908`

Le fichier LaTeX

Il n'y a donc plus qu'à constituer notre fichier LaTeX :

```
\documentclass{article} \usepackage[a4paper,text={19cm,27.7cm},dvips]{geometry} \usepackage[dvips]{graphicx}
\usepackage{pstricks} \pagestyle{empty} \begin{document} \includegraphics[bb=-2700 -36 -2095
908,clip,scale=.83]%% {monfichier.eps} \newpage \includegraphics[bb=-2095 -36 -1490 908,clip,scale=.83]%%
{monfichier.eps} \newpage \includegraphics[bb=-1490 -36 -885 908,clip,scale=.83]%% {monfichier.eps} \newpage
```

```
\includegraphics[bb=-885 -36 -280 908,clip,scale=.83]%% {monfichier.eps} \newpage \includegraphics[bb=-280
-36 325 908,clip,scale=.83]%% {monfichier.eps} \newpage \includegraphics[bb=325 -36 930
908,clip,scale=.83]%% {monfichier.eps} \newpage \includegraphics[bb=930 -36 1535 908,clip,scale=.83]%%
{monfichier.eps} \newpage \includegraphics[bb=1535 -36 2140 908,clip,scale=.83]%% {monfichier.eps} \newpage
\includegraphics[bb=2140 -36 2749 908,clip,scale=.83]%% {monfichier.eps} \end{document}
```

Cerise sur le gâteau

C'est bien mais pour découper c'est pas terriblement efficace sans pointillés (ou plutôt traits-tillés (!)), alors on fait une commande de pointillés :

```
\newcommand{\separation}{%% \begin{pspicture}(0,0)%%
\psline[linestyle=dashed,linewidth=.05mm](0pt,-1.5cm)(0pt,28.5cm)%% \end{pspicture}}
```

Et notre code LaTeX devient :

```
\documentclass{article} \usepackage[a4paper,text={19cm,27.7cm},dvips]{geometry} \usepackage[dvips]{graphicx}
\usepackage{pstricks} \pagestyle{empty} \newcommand{\separation}{%% \begin{pspicture}(0,0)%%
\psline[linestyle=dashed,linewidth=.05mm](0pt,-1.5cm)(0pt,28.5cm)%% \end{pspicture}} \begin{document}
\includegraphics[bb=-2700 -36 -2095 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=-2095 -36 -1490 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=-1490 -36 -885 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=-885 -36 -280 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=-280 -36 325 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=325 -36 930 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=930 -36 1535 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=1535 -36 2140 908,clip,scale=.83]%% {monfichier.eps}\separation \newpage \separation%%
\includegraphics[bb=2140 -36 2749 908,clip,scale=.83]%% {monfichier.eps}\separation \end{document}
```

Il n'y a plus qu'à compiler notre *monfichier.tex* :

```
$ latex monfichier
```

puis de produire un fichier PostScript :

```
$ dvips monfichier.dvi -o
```

Et nous obtenons notre joli *monfichier.ps*

Le gâteau est un peu pourri

Génial direz-vous...mais non, en fait pas terriblement efficace, en effet le fichier que je viens de produire, ici à la maison :-), pèse finalement environ 2,4 Mo alors que mon image de départ pèse à peine 251 Ko ! Hé bien oui, c'est bien loin d'être un découpage, l'image est tout simplement copié 9 fois dans le fichier produit finalement.

Automatiser

Lire avec une expression régulière la ligne de la « Bounding Box » est une évidence, faire une petite conversion d'unités et une petite division aussi.

Donc le processus décrit dans cette page peut relativement facilement être scripté ; par contre, son intérêt est limité par la taille des fichiers obtenus : si mon arbre augmente en hauteur, j'aurais au moins 2 pages A4 en hauteur et au total 18 feuilles d'où un fichier d'environ 5 Mo... le jeu n'en vaut assurément pas la chandelle !

Post-scriptum :

Certains parleront de la commande poster. Chez moi, elle n'a tout simplement pas fonctionné malgré des heures d'efforts et des passages par *pstops*.

C'est bien dommage car dans l'absolu, c'est une commande qui simplifie largement les choses !

Si vous avez une solution qui est plus efficace notamment au niveau de la taille finale, n'hésitez pas à me contacter : elle m'intéresse vivement !