

The ESIEEcv*package

Benjamin BAYART
Institut Gaspard MONGE
Université de Marne-la-Vallée
bayartb@edgard.fdn.fr

Pierre LE MAGUET
Valeo Borg
Karlsruhe

Imprimé le 13 avril 2009
Dernière mise à jour le 1997/12/29

Résumé

Ce document décrit l'utilisation du package `ESIEEcv` dont le but est de permettre la mise en page de curriculum-vitæ de manière simple et efficace, tels que les entreprises françaises les attendent.

Il a été écrit par moi (Benjamin BAYART) sur une idée originale (pour ce qui est de la mise en page et de la syntaxe finale) de Pierre LE MAGUET.

English :

This document describe the use of the package `ESIEEcv` to setup curriculum vitæ efficiently and easily as french employers will expect it.

The current documentation is in French.

This package was written by my (Benjamin Bayart) from an original idea by Pierre Le Maguet (regarding design and final syntax).

Table des matières

1	« The name of the game »	2
2	Organisation du document	2
3	Syntaxe exacte	2
4	Paramétrage du système	3
5	Le code utile à tous	4
6	Le code le moins utile	4
7	Code de gestion des rubriques	7

*Ce document décrit la version v2.0a du package. La documentation est datée de 1997/12/14.

1 « The name of the game »

Ce package s'appelle ESIEEcv simplement parce que Pierre et moi étions étudiants à l'ESIEE¹ lorsqu'il a été écrit pour la première fois.

2 Organisation du document

Un cv, pour ce package, est divisé en plusieurs rubriques, classiquement 3 : « Formation initiale », « Expériences », et « Divers ».

En plus des rubriques, on trouvera généralement au début du cv un texte libre présentant la personne.

Dans chaque rubrique, on trouve des sous-rubriques permettant de décrire chaque information du cv. Par exemple une sous-rubrique par diplôme dans la rubrique « formation » et une par expérience dans la rubrique suivante.

Dans la troisième rubrique, les sous-rubriques sont généralement beaucoup plus simples, pour ce qu'elles contiennent moins d'information ; typiquement pour chaque langue on donne juste son niveau, alors que pour un stage ou un ancien emploi on donne plus d'informations.

Les sous-rubriques les plus complexes contiendront des appels aux commandes `\Titre`, `\Date`, `\Duree`, `\Descr`, et `\Apport`. Toutes ces commandes sont optionnelles (si un champ est omis, la mise en page est corrigée en conséquence), et la commande `\Apport` peut être utilisée plusieurs fois pour souligner les points forts d'une expérience ou d'une formation.

Les sous-rubriques simples comportent deux champs : `\Descr` et `\Competence`. En général, on place dans la compétence, par exemple, la langue concernée, et dans la description le niveau que l'on a.

3 Syntaxe exacte

Un exemple de sous-rubrique simple :

```
\begin{sousrubrique}
\Competence{Espagnol}
\Descr{Niveau scolaire}
\end{sousrubrique}
```

Un exemple de sous-rubrique complexe :

```
\begin{sousrubrique}
\Titre{Ingénieur ESIEE (\'Ecole Sup\'erieurs d'Ing\'enieurs
en \'Electronique et \'Electrotechnique)}
\Date{1991-1996}
\Duree{5 ans}
\Descr{Spécialisation en informatique.}
\Apport{Programmation syst\`eme}
```

1. École Supérieure d'Ingénieurs en Électronique et Électrotechnique.

```

\Apport{Conception et programmation objet}
\Apport{Théorie des langages}
\Apport{Langages interprétés}
\Apport{Programmation logique}
\Apport{Programmation des interfaces graphiques}
\Apport{SGBD}
\end{sousrubrique}

```

Attention, si on ne souhaite donner que le lieu et pas le titre, il est souhaitable de faire le contraire, quitte à devoir forcer le choix de la fonte dans la donnée. En effet, le changement de ligne qui doit avoir lieu entre le titre et la description n'aura effectivement lieu que si et seulement s'il y a un titre *et* une description.

Une rubrique contiendra exclusivement une ou des sous-rubrique, et on devra lui passer un titre, par exemple :

```

\begin{rubrique}{Formation initiale}
\begin{sousrubrique}
\Titre{Baccalauréat...}
\end{sousrubrique}
\end{rubrique}

```

4 Paramétrage du système

Un certain nombre de choix typographiques peuvent être modifiés de manière simple, en redéfinissant des commandes.

La première chose que l'on peut modifier est la structure même des apports. Par défaut, avant chaque apport est placé le symbole \triangleright suivi d'un espace ; et après chaque apport un point et un changement de ligne. Avant l'ensemble des apports est placé un changement de ligne, et après l'ensemble des apports rien.

Ces quatre choix sont symbolisés par les commandes `\PreApport`, `\PostApport`, `\PrePreApports` et `\PostPostApports`.

Pour ma part, j'aime utiliser le symbole `\rightslice` du package `stmaryrd` et ne pas changer de ligne entre deux apports. Pour cela, je redéfini les commandes `\PreApport` et `\PostApport` comme suit :

```

\renewcommand{\PreApport}{\ensuremath{\rightslice} }
\renewcommand{\PostApport}{. }

```

On peut aussi, en redéfinissant les commandes `\FonteApport`, `\TailleApport`, `\FonteLieu` et `\FonteTitre` changer certains détails de la mise en forme.

Enfin, la colonne de gauche (celle qui contient les dates dans le document final) est de largeur fixe. Cette largeur (trois centimètres par défaut) est défini par la longueur `\largeurcolonne` que l'on peut aisément redéfinir, par exemple en faisant :

```

\setlength{largeurcolonne}{2.5cm}

```

5 Le code utile à tous

Tout d'abord les bêtises habituelles permettant d'identifier le package :

```
1 <*package>
2 \ProvidesPackage{ESIEEcv}[1997/12/29 v2.0a Style ESIEEcv]
3
4 \RequirePackage{tabularx}
```

On précise que l'on ne veut pas de numéro de page, ni d'indentation de paragraphes :

```
5 \pagestyle{empty}
6 \parindent=0pt
```

Enfin la définition des huit commandes utilisées comme paramètres par le reste du package :

```
\PreApport
\PostApport 7 \newcommand{\PreApport}{\ensuremath{\triangleright} }
\PrePreApports 8 \newcommand{\PostApport}{.\newline}
\PostPostApports 9 \newcommand{\PrePreApports}{\newline}
\FonteApport 10 \newcommand{\PostPostApports}{}
\TailleApport 11 \newcommand{\FonteApport}{\itshape}
\FonteLieu 12 \newcommand{\TailleApport}{\small}
\FonteTitre 13 \newcommand{\FonteLieu}{\scshape}
14 \newcommand{\FonteTitre}{\scshape}
```

6 Le code le moins utile

6.1 Définitions des longueurs et des variables de mémorisation

```
15 \newlength{\Tampon}
16 \newlength{\Offset}
17 \newlength{\largeurcolonne}
18 \setlength{\largeurcolonne}{3cm}
19 \newlength{\Space}
20 \setlength{\Space}{3mm}
21 \def\@@date{}
22 \def\@@titre{}
23 \def\@@duree{}
24 \def\@@lieu{}
25 \def\@@descr{}
26 \def\@@apports{}
27 \def\@@competence{}
```

6.2 Le système de gestion de liste

Pour toute explication, voire annexe D du T_EXbook.

`\CV@AppendItem`

```
28 \toksdef\ta=0 \toksdef\tb=2
29 \def\CV@AppendItem#1#2{%
30 \ta={#1}%
31 \tb=\expandafter{#2}%
32 \global\edef#2{\the\tb\the\ta}}
```

6.3 Initialisation du système

`\CV@ajout@apport` Ajoute un texte donné à la liste des apports en cours en utilisant `\CV@AppendItem`.

```
33 \newcommand{\CV@ajout@apport}[1]
34 {\CV@AppendItem{\PreApport#1\PostApport}{\@@apports}}
```

`\CV@init` Définit la commande `\@cde` dont le nom est passé en paramètre. L'intérêt est simplement d'améliorer la compréhension lors de la lecture du code :

```
35 \newcommand{\CV@init}[1]
36 {\expandafter\global\expandafter\def\csname @@#1\endcsname{}}
```

6.4 Code de gestion des sous-rubriques

sousrubrique

```
37 \newenvironment{sousrubrique}
```

On commence par initialiser toutes les variables :

```
38 {\CV@init{apports}\CV@init{date}\CV@init{titre}\CV@init{duree}%
39 \CV@init{lieu}\CV@init{descr}\CV@init{competence}\CV@init{compcomment}}%
```

Enfin on définit (localement donc) les six commandes permettant la mémorisation des données de la sous-rubrique en cours.

```
40 \newcommand{\Date}[1]{\global\def\@date{##1}}
41 \newcommand{\Duree}[1]{\global\def\@duree{##1}}
42 \newcommand{\Titre}[1]{\global\def\@titre{##1}}
43 \newcommand{\Lieu}[1]{\global\def\@lieu{##1}}
44 \newcommand{\Descr}[1]{\global\def\@descr{##1}}
45 \newcommand{\Competence}[1]{\global\def\@competence{##1}}
46 \let\Apport\CV@ajout@apport
47 }
```

Toutes les informations collectées vont maintenant être mises en forme. Pour cela, on les balise avec les commandes \TeX appropriées, puis on les mémorise dans une liste (`\CV@delayed`). Le contenu de la liste sera « éjecté » une fois l'environnement terminé. En effet, ces données peuvent générer plusieurs lignes dans le tableau qui formera le cv. Ce qui nous force à ce qu'elles ne soient pas incluses dans un groupe. Or l'environnement est un groupe.

Commençons pas initialiser cette liste.

```
48 {%
49 \global\def\CV@delayed{}
```

En premier, la date. Elle est mise en gras dans la première colonne, suivie d'un changement de ligne s'il y a une durée. En effet, s'il y a une date et pas de durée, il faut supprimer le changement de ligne de la première colonne, sans quoi la case correspondante du tableau serait contrainte à une hauteur d'au moins deux ligne, ce qui n'est pas acceptable.

```
50 \ifx\@@date\empty\else
51 \CV@AppendItem{\textbf{\@@date}}%
52 \ifx\@@duree\empty\else
53 \newline
54 \fi
55 }\CV@delayed}%
56 \fi
```

La durée est ajoutée, si nécessaire, entre parenthèse, qu'il y ait eu une date ou pas. On est encore dans la première colonne.

```
57 \ifx\@@duree\empty\else
58 \CV@AppendItem{(\@@duree)}\CV@delayed}%
59 \fi
```

S'il y a une compétence de précisée, elle est ajoutée dans la première colonne. On notera que le package n'impose pas que si on travaille avec une sous-rubrique complexe alors il n'y a pas de compétence. De même il n'impose pas que si on est dans une sous-rubrique simple il doit ne pas y avoir de compétence. C'est un choix de design : toujours faire simple. De plus, cela permet de remplacer la date par n'importe quoi d'autre d'intéressant en ne mettant ni date ni durée mais simplement une compétence, et ce même dans une sous-rubrique complexe.

```
60 \ifx\@@competence\empty\else
61 \CV@AppendItem{\@@competence}\CV@delayed}
62 \fi
```

Maintenant, la première colonne est complète. On peut donc passer à la seconde. Pour ce, on ajoute un &.

```
63 \CV@AppendItem{&}\CV@delayed}
```

Le lieu et le titre sont mis en premier dans la seconde colonne, chacun dans la bonne fonte, et en les faisant suivre d'un point. Un espace est ajouté entre les deux si et seulement si les deux sont présents.

```
64 \ifx\@@lieu\empty\else
65 \CV@AppendItem{\FontELieu{\@@lieu}.}\CV@delayed}
66 \fi
67 \ifx\@@lieu\empty\else\ifx\@@titre\empty\else
68 \CV@AppendItem{ }\CV@delayed}
69 \fi\fi
70 \ifx\@@titre\empty\else
71 \CV@AppendItem{\FontETitre{\@@titre}.}\CV@delayed}
72 \fi
```

S'il y a un titre *et* une description, on insère un saut de ligne dans la seconde colonne (celle de droite). Ensuite, on ajoute, si elle existe, la description.

```
73 \ifx\@@titre\empty\else\ifx\@@descr\empty\else
```

```

74 \CV@AppendItem{\newline}{\CV@delayed}
75 \fi\fi
76 \ifx\@@descr\empty\else
77 \CV@AppendItem{\@@descr}{\CV@delayed}%
78 \fi

```

Ne reste plus comme donnée à ajouter que les apports dans la bonne fonte et dans la bonne taille. On ajoute bien entendu le `\PrePreApports` et le `\PostPostApports`.

```

79 \ifx\@@apports\empty\else
80 \CV@AppendItem{\TailleApport
81               \FonteApport
82               {\PrePreApports\@@apports\PostPostApports}}}%
83               {\CV@delayed}%
84 \fi

```

Maintenant que toutes les données ont été convenablement balisées dans la liste, ne reste plus qu'à ajouter le changement de ligne dans le tableau. Ensuite, on « éjectera » le contenu de la liste une fois le groupe correspondant à l'environnement terminé.

```

85 \CV@AppendItem{\}\{\CV@delayed}
86 \aftergroup\CV@delayed
87 }

```

7 Code de gestion des rubriques

`rubrique` Cet environnement est en fait extrêmement simple. Il met son titre en gros et en gras, puis commence un tableau dont la largeur sera exactement celle de la page.

```

88 \newenvironment{rubrique}[1]{%%
89 \bigskip
90 \textbf{\large #1\phantom{Pj}}
91 \medskip
92 \par
93 \tabularx{\linewidth}{p{\largeurcolonne}X}}
94 {\endtabularx}

95 </package>

```