

Cahiers **GUT**enberg

☞ L'AVENIR DE T_EX... C'EST WORD[PERFECT] !
UNE ÉBAUCHE : RTF \LaTeX

☞ Daniel TAUPIN

Cahiers GUTenberg, n° 17 (1994), p. 20-31.

http://cahiers.gutenberg.eu.org/fitem?id=CG_1994__17_20_0

© Association GUTenberg, 1994, 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.

L'avenir de T_EX... c'est Word[Perfect]! Une ébauche: RTFLaTeX

Daniel TAUPIN

*Laboratoire de Physique des Solides
associé au CNRS
Bâtiment 510, Centre Universitaire
F - 91405 ORSAY Cédex
taupin@rsovax.lps.u-psud.fr*

Résumé. Quelles que soient ses qualités, T_EX est bien moins utilisé que Word (WordPerfect outre-atlantique). Aussi pensons-nous que la survie à long terme de T_EX — et de ses avatars comme L^AT_EX ou A_MS-T_EX — passe par des logiciels performants de transcription dans les deux sens.

À moins de se contenter de conversions médiocres, de telles transcriptions présentent des difficultés de principe parfois insurmontables ; mais nous présentons néanmoins une première réalisation « potable », en l'occurrence RTFLaTeX.

Abstract. *Although T_EX is admittedly a beautiful software, it is nevertheless much less used by computer typesetters than Word (WordPerfect in North America). Therefore, we think that the future of T_EX — as well as its varieties like L^AT_EX or A_MS-T_EX — implies the existence of powerful transcription softwares in both directions.*

Unless poor conversions were accepted, such transcriptions exhibit severe difficulties in their principle, which may happen to be insuperable ; however, we present a first reasonable attempt, namely RTFLaTeX.

1. La norme, c'est Word...

Quand on centre ses lectures sur les *Cahiers GUTenberg* ou *TUGboat*, on en retire l'impression que T_EX, L^AT_EX et autres dérivés sont des produits usuels... et incontestables ; mais quand on regarde aussi les publications variées qui passent entre nos mains, on constate alors que l'usage de T_EX reste confidentiel, sauf peut-être dans le domaine des mathématiques et de la physique théorique.

1.1. Quelques ordres de grandeurs pseudo-statistiques

Nous donnerons trois exemples :

1. Dans notre laboratoire de physique des solides — laboratoire universitaire associé au CNRS comportant trois cents personnes — nous voyons régulièrement sortir des imprimantes laser chaque année des dizaines de thèses et de *pre-prints* d'articles scientifiques. Tous sont systématiquement composés avec Word (Word 4.0 sur Macintosh en pratique) ; font l'exception deux thèses (théoriques) en cinq ans, comportant beaucoup de maths et composées en \LaTeX , ainsi bien sûr que les travaux de nos collaborateurs immédiats¹.
2. Dans le cadre de notre modeste activité d'enseignement en informatique² nous voyons chaque année passer des dizaines de *rapports de stage*, documents de vingt à cent pages édités dans les entreprises ou laboratoire où nos étudiants vont travailler trois mois d'affilée : en cinq ans, un seul a été fait en \LaTeX à l'INSTM (Saclay).
3. Hormis les *Cahiers GUTenberg*, nous avons depuis un an soumis deux articles à deux revues scientifiques françaises : le Microbulletin du CNRS, et la revue TSI (*Technique et Science Informatiques*) de l'Afcet.
 - La revue TSI écrit explicitement que les articles doivent être envoyés « si possible » sous forme de fichiers Word 4.0, X-Press 3.0 ou MacWrite 5.01 pour Macintosh. Elle accepte cependant les articles composés en \LaTeX mais avec une double restriction [2] :
 - (a) que l'article soit fourni par l'auteur sur papier, prêt à imprimer ;
 - (b) qu'aucune correction de dernière minute ne sera possible :
[...] *authors using TEK or LATEK (sic!) are requested to send their final version. No corrections will be allowed*³.
 - Quant au Microbulletin, la rédaction a recopié tout notre article [1] pour l'incorporer tant bien que mal à Word⁴. Et dans

1. Commentaire en latin : *Asinus asinum fricat!*

2. C4 d'informatique appliquée, Orsay.

3. On remarquera que l'allusion à $T_{E}X$ ou \LaTeX ne figure que dans la version anglaise des *consignes aux auteurs*...

4. C'était un article sur $T_{E}X$, un comble !

un courrier personnel, la rédaction précisait qu'elle n'avait jamais investi dans T_EX parce qu'aucun des auteurs n'avait jamais proposé de fournir des articles en (L^A)T_EX.

En revanche, le Journal de Physique accepte maintenant volontiers les articles écrits en L^AT_EX... bien que le *style* propre à ce journal soit incompatible avec NFSS!

1.2. L'opinion des gens

Évidemment, comme pour toute personne qui se passionne pour ses outils de travail, la tentation du prosélytisme ne nous a pas épargné et il nous arrive souvent de discuter de PAO avec des collègues de travail ou du monde associatif. Au mieux, on considère les « T_EXeurs » comme de gentils originaux au même titre qu'un collectionneur exclusif de timbres du Zimbabwe ou un qu'un linguiste féru de la langue inuit; au pire on nous considère comme des *déviants*, surtout si le document doit être travaillé par plusieurs co-auteurs.

Dans ces conditions, discuter des qualités respectives de T_EX et de Word est aussi stupide que comparer les systèmes UNIX et VMS ou bien les réseaux de protocoles SNA et TCP/IP. Dans le cas d'UNIX et de TCP/IP, on ne discute pas des qualités de ces produits: ce sont des normes⁵; or on ne discute pas les normes, on s'y plie. Pour la PAO usuelle, la situation est la même: la norme s'appelle Word (de Microsoft et avec toute la puissance commerciale de Microsoft) en France, et WordPerfect aux États-Unis.

Cela dit, le fait que — au moins en France — Word soit la norme *de facto* biaise inévitablement les jugements sur (L^A)T_EX. En pratique on peut distinguer :

- ce que T_EX fait aussi bien que Word : c'est un bon point pour T_EX ;
- ce que T_EX a du mal à faire alors que Word peut, et qui *ipso facto* est « axiomatiquement » une tare intolérable de T_EX :
 - incorporer des figures en restant *bien au chaud* dans le monde Word/Microsoft⁶,

5. Au point que la société Tandem — qui fabrique des ordinateurs à haute tolérance aux pannes — a dû renoncer à certaines caractéristiques hautement fiables pour se rallier à la norme UNIX.

6. T_EX peut incorporer des figures, mais on doit se soucier du protocole de l'imprimante, PostScript, PCL ou autre; quant à **BM2FONT**, il peut d'une part saturer le nombre acceptable de

- avoir des polices exotiques (par exemple à la fois *sans sérif*, *gras* et *petites capitales*),
 - ajuster l'aspect d'une formule ou d'un tableau par essais et erreurs, ce qui nécessite une compilation/visualisation en une fraction de seconde... ou un système *wysiwyg* ;
- ce que T_EX fait bien mieux que Word, par exemple les *maths*, les coupures de mots multilingues, les tables des matières (en L^AT_EX) et les index. Nous autres *T_EXeurs* plus ou moins émérites en faisons des qualités suprêmes, mais le *Wordiste moyen* les considère comme négligeables, d'une part parce qu'il ne s'en sert pas⁷, d'autre part parce qu'il utilise Word pour privilégier la vitesse par rapport à la qualité⁸ et à la facilité de maintenance.

On objectera que tous ces propos sont bien pessimistes, mais nous rappellerons Jean de LA FONTAINE⁹ :

« *Selon que vous serez puissant ou misérable
Les jugements de cour vous rendent blanc ou noir* »

Est-ce une raison pour déclarer que T_EX est moribond ? Nous ne le croyons pas et, surtout, nous ne le souhaitons pas compte tenu de ses immenses possibilités, notamment la possibilité de déclarer de nouvelles *macros*. Cela dit, compte tenu de caractère ultra-minoritaire de T_EX¹⁰, il ne peut survivre à long terme qu'en symbiose avec le modèle dominant, c'est-à-dire Word[Perfect]. Aussi, et en priorité sur d'autres incontestables améliorations comme L^AT_EX3, l'avenir de (L^A)T_EX est donc avant tout :

1. de constituer un post-processeur pour peaufiner des documents préparés initialement avec Word[Perfect] ;
2. de constituer un pré-processeur pour réaliser des documents impeccables, ensuite incorporés dans des ouvrages plus globaux édités avec Word[Perfect].

polices dans le cas de gros ouvrages, et d'autre part il ne tourne pas à notre connaissance sur Macintosh. Or le Macintosh est aussi une *norme de fait* pour la PAO ordinaire.

7. Il ne s'en sert pas parce qu'il ne peut pas s'en servir.

8. Les *T_EXeurs* sont en général des perfectionnistes tandis que les *Wordistes* sont le plus souvent des adeptes du *quick and dirty* !

9. *Les animaux malades de la peste*.

10. Sauf dans quelques domaines comme les maths ou la physique théorique.

2. Les niveaux de symbiose Word \leftrightarrow (\LaTeX) \TeX

Il est évident que les exigences de qualité de conversion Word \leftrightarrow (\LaTeX) \TeX seront différentes selon qu'il s'agit d'une conversion occasionnelle, une fois pour toutes pour un document, ou d'un processus itératif de mise-à-jour. En fait la conversion peut se placer à cinq niveaux :

1. Le *niveau de base* : transcription des simples caractères ASCII imprimables. Ceci est fait directement par Word quand on lui demande d'inclure un document qu'il ne reconnaît pas comme étant de son format. Dans l'autre sens, n'importe quel programmeur débutant peut écrire un processeur qui filtre les caractères imprimables dans n'importe quel fichier (binaire ou texte) et en fait un fichier à *peu près* acceptable par (\LaTeX) \TeX moyennant un petit travail à l'éditeur.
2. Le *niveau de base avec lettres accentuées* : malheureusement, Word[Perfect] ne code pas les lettres accentuées sous la forme \TeX à 7 bits (par exemple $\backslash'e$ ou $\backslashoe\{}$) ni dans les codes à 8 bits comme le code 850 des PC ou la norme Iso-Latin 1. Il est clair que ceci est indispensable pour les langues européennes, sous peine de devoir revoir chaque mot ou presque. Mais dans les deux sens, la réalisation d'un tel processeur est chose facile.
3. La *transcription des changements de police et des notes de bas de page* : ceci n'est pratiquement faisable qu'en utilisant ou en créant des fichiers sous la forme RTF¹¹ qui est, sauf erreur de notre part, un format inventé par Microsoft (Word) et également reconnu par WordPerfect. RTF a l'avantage de n'utiliser que des caractères ASCII à 7 bits (codes 32–127) et d'offrir une (coupable) ressemblance avec la syntaxe de \TeX . Son autre avantage est de comporter en (presque) clair les indications de changements de police et de ne pas reporter les notes de bas de page en fin de document.
4. La *transcription des « formules »* de Word : bien que plus rudimentaire que le mode *mathématique* de \TeX , Word sait fabriquer des fractions, des exposants, des sommations ou des intégrales [3]. Leur transcription de/vers (\LaTeX) \TeX est nécessaire, même si elle est parfois approximative.

11. *Rich Text Format*.

5. La *transcription réversible* $Word \Leftrightarrow (\mathbb{A})T_{E}X$: elle est *indispensable* pour un document qui doit être régulièrement mis-à-jour par des équipes donc certains membres sont allergiques à $(\mathbb{A})T_{E}X$. Cette transcription de *Word[Perfect]* vers $(\mathbb{A})T_{E}X$ doit alors respecter *tous* les détails inscrits avec *Word*, y compris sans doute les en-têtes et surtout les figures. Et dans l'autre sens, elle exige une conversion rigoureuse, notamment des espacements et des longueurs de lignes.

Bien que quasiment impossible, une telle réversibilité est néanmoins impérative si l'on veut¹² qu'un même document puisse être alternativement travaillé en $(\mathbb{A})T_{E}X$ et en *Word[Perfect]*.

Le logiciel $RTFLaTeX$ ¹³ essaye donc de transcrire un fichier RTF ¹⁴ en $LaTeX$; le résultat $LaTeX$ est cependant loin d'être parfait, d'une part à cause des imperfections de la version actuelle (1.64), d'autre part à cause de difficultés de principe inhérentes à *Word* et à RTF ; aussi est-il nécessaire, une fois la transcription faite, d'*éditer* le fichier $LaTeX$ pour le rendre plus convenable, notamment en mode mathématique.

3. $RTFLaTeX$: les difficultés

3.1. Une documentation RTF déficiente

Les fichiers RTF sont aisément éditables pourvu que l'éditeur — nous utilisons *Micro-Emacs* — puisse gérer des enregistrements d'au moins 256 caractères. Sa structure interne ressemble beaucoup à $T_{E}X$, avec un parenthésage des polices entre accolades ($\{...\}$) et des mots clé commençant par des « \backslash ». Mais la documentation est incomplète quoique non nulle : il n'y a certes aucune référence bibliographique à la définition de RTF dans les manuels *Word*¹⁵ et notre seule documentation est un fichier anonyme $RTF.TXT$ trouvé sur le serveur *ftp* de Stuttgart.

Ce fichier $RTF.TXT$ décrit sommairement la plupart des commandes RTF , mais quand il dit par exemple que \backslashpard a pour effet *reset to de-*

12. Et des messages électroniques nous sont arrivés dans ce sens à la suite de l'annonce de $RTFLaTeX$.

13. *NDLR* : Un autre logiciel nommé $RTF2TeX$ existe et travaille aussi dans ce domaine.

14. À partir de *Word*, une simple option de sauvegarde permet de stocker le fichier sous la forme RTF plutôt que strictement *Word*. Et selon notre expérience, cette sauvegarde semble bien réversible.

15. Procédé fréquent chez les marchands de logiciels qui ne veulent pas faciliter le travail à des concurrents.

fault para properties, il ne dit pas quelles sont ces propriétés par défaut, ni quelles sont celles qui sont affectées. Il nous a donc fallu deviner les spécifications de RTF en regardant ce qu'elles produisaient en imprimant les documents via Word.

3.2. Une imbrication à deux niveaux indépendants

Le parenthésage des polices (type et taille) est certes fait comme en $\text{T}_{\text{E}}\text{X}$ avec des accolades, mais quand on entre dans les « formules » — c'est à dire l'embryon de composition mathématique de Word — alors le parenthésage est réalisé non plus par des accolades mais par des parenthèses rondes « (...) ». En fait, alors que $\text{T}_{\text{E}}\text{X}$ utilise les accolades à la fois pour parenthéser les groupes qui délimitent la *portée* de paramètres locaux — dont les polices actives — et pour parenthéser les arguments des *macros*, RTF utilise pour ses macros (figées) qui réalisent les formules un parenthésage indépendant de celui des polices. On peut donc avoir en RTF un *groupage*, par exemple pour spécifier une police grasse, commençant à l'extérieur d'une racine carrée, et se terminant au milieu des arguments de cette racine, un peu comme si on écrivait :

$$\$ a+d+c \{ \backslash \mathbf{f} +d \sqrt{2.0+j} \} +k \$$$

pour obtenir :

$$a + d + c + \mathbf{d} \sqrt{2.0 + j + k}$$

avec ce petit détail que — contrairement à Word — ni $\text{T}_{\text{E}}\text{X}$ ni $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ne savent mettre le symbole $\backslash\text{sqrt}$ en gras¹⁶. Mais l'important ici est surtout que la traduction des formules RTF en $\text{T}_{\text{E}}\text{X}$ se heurtent à un principe fondamental de $\text{T}_{\text{E}}\text{X}$: l'unicité du parenthésage *à la fois* pour les polices et pour les arguments de macros¹⁷.

3.3. L'indécidabilité des polices à utiliser dans les formules de Word

$\text{T}_{\text{E}}\text{X}$ (et Donald E. KNUTH[4]) font le choix que, dès qu'on passe en mode mathématique, les lettres deviennent des caractères en *italique maigre* sauf spécification contraire à préciser explicitement. Au contraire, dans ses

16. Donc une déficience de $\text{T}_{\text{E}}\text{X}$ vis-à-vis de Word!

17. Donc une autre déficience de $\text{T}_{\text{E}}\text{X}$ vis-à-vis de Word!

formules, Word conserve la *forme* et la *graisse* du texte environnant. Alors, quand on traduit un texte Word qui se lit :

La valeur de l'exponentielle est e^m

doit-on laisser le texte sous la forme `e{\hbox{\bf m}}` ou l'*interpréter* comme

La valeur de l'exponentielle est e^m

ce qui se code bien plus simplement `em`? Et quand on trouve N₂, s'agit-il de l'élément n° 2 de l'ensemble N, qui devrait alors se typographier N₂, ou bien s'agit-il de la molécule d'azote qui s'écrit effectivement N₂?

Il est clair que ceci est *indécidable* au même titre que la typographie de « a » dans « la variable a a pour représentation la lettre a », et que quel que soit le choix fait par RTFLaTeX celui-ci sera attaquable, soit pour ne pas respecter la *lettre* du texte Word, soit pour le pas respecter l'*esprit* dans lequel il a été composé. Nous entrons donc dans un domaine passionnel, où les uns reprocheront à Word de permettre sans barrage un codage hideux des maths et où les autres reprocheront à T_EX — ou à RTFLaTeX — un abus de pouvoir inadmissible.

3.4. Le respect des polices de Word

Word offre au moins neuf formes de polices: *Chicago*, *Geneva*, *Courier Helvetica*, *Monaco*, *New York*, *Times*, *Venice* et *Zapf Dingbats*¹⁸. Il est assez évident que l'on peut interpréter *Times* par `\rm`, *Helvetica* par `\sf` (sans sérif) et *Courier* par `\tt` car, même si la forme diffère légèrement, l'aspect est similaire. Mais il n'existe pas dans les polices standard de T_EX (série cmxxx ou série dcxxx) d'équivalent à *Chicago*.

Qui plus est, Word propose pour chacune de ces polices la transformation en *gras*, *italique*, *souligné* (cf. 3.5), *relief*¹⁹, *ombré*, *petites capitales*, *majuscules*. Et ces transformations sont combinables, ce qui permet d'avoir du *sans sérif gras penché en petites capitales*. Heureusement, peu de gens s'amuse à des combinaisons aussi farfelues, mais l'impossibilité de le faire

¹⁸. D'après le manuel de référence pour Word 4, mais notre installation en offre une quinzaine d'autres.

¹⁹. On ne dessine que la frontière entre le noir initial et le blanc des œils ou l'extérieur.

avec `TeX` est un argument qu'on retiendra si on veut le condamner! Sans parler des caractères en « relief » (*outline* en anglais) qui ne figurent ni dans les polices `cmxxx` ni dans les `dcxxx` des distributions standard. En cherchant bien, on en trouve quand même sur les serveurs `ftp` dédiés à `TeX`, et il est facile de les étendre, par exemple :

Police `cmossb10` (`ocmssbx10`) grandie à 1728
Police `cmobx12` (`ocmbx12`) grandie à 1200

3.5. Le soulignage

Dans le cadre des formats de caractères, Word propose le soulignage, lequel peut être maigre ou gras, et inclure ou non les espaces. Il est clair que le soulignage est considéré comme *typographiquement méprisable* en ce sens qu'il n'est qu'un procédé d'indication de l'*italique* ou du *gras* dans les manuscrits (éventuellement dactylographiés) ou les corrections d'épreuves. Cela n'empêche pas maints typographes amateurs de recourir au soulignage pour les titres, les mots voire les paragraphes à mettre en valeur, fut-ce de manière redondante s'il y a aussi changement de police.

Cela dit, le rôle d'un système de PAO est d'inciter à une bonne typographie, il n'est pas de son ressort d'*interdire* ce qui est contraire aux usages. Donc, bon ou mauvais, la possibilité du soulignage est un *droit* du typographe amateur, droit que `TeX` lui dénie sauf acrobaties de `TeXperts`. En effet, si `TeX` permet facilement de souligner un mot avec `\underline{...}`, le soulignage systématique d'un paragraphe dont le nombre et les coupures de lignes sont inconnus à l'avance n'est pas prévu²⁰. Ou alors il faut disposer de *variantes* soulignées de toutes les polices, droites, penchées, maigres, grasses, avec ou sans sérif, c'est-à-dire doubler le nombre des polices... rajouter une nouvelle classe — en plus de *family*, *series* et *shape* — dans NFSS[5].

Aussi, dans l'état actuel des choses, `RTFLaTeX` ne sait souligner que des mots ou de petites suites de caractères, avec la restriction que les portions soulignées deviennent *insécables*.

²⁰ NDLR: Ceci est vrai tant que l'on n'utilise pas de macro-commandes supplémentaires comme celles définies dans l'option de style `ulem.sty`. L'auteur pense d'ailleurs s'en inspirer pour les prochaines versions de `RTFLaTeX`.

3.6. Les figures incorporées par Word

On les retrouve effectivement, mais en hexadécimal dans le texte RTF ; il n'est pas difficile pour RTFLaTeX de les extraire pour les mettre dans un fichier. La question est alors : qu'est-ce qu'on en fait ? Un examen des fichiers produits permet d'y reconnaître des caractères du texte imprimé dans les figures, mais aucune information ne permet de trouver la clé pour décoder ces figures qui apparaissent donc comme un rectangle vide dans le produit de RTFLaTeX.

3.7. Les équations réalisées par l'« éditeur d'équations »

Word 5.0 pour Macintosh possède un *éditeur d'équations* qui produit un texte mathématique d'une présentation très proche de celle de $T_{E}X$, donc fort acceptable en comparaison avec ce que produisent les « formules » de Word quand elles sont maniées sans soin particulier. L'ennui est que ces équations sont alors incorporées dans le fichier RTF comme du code PostScript, lui-même emballé dans un code numérique dont la structure n'est pas intelligible. Faute de documentation, RTFLaTeX ne peut donc convertir ces équations en $T_{E}X$ ²¹ et elles restent des boîtes vides, même si l'imprimante comprend le PostScript.

3.8. L'absence de structure des documents Word

En pratique, un document RTF transporte une pauvre quantité d'information par rapport à $T_{E}X$ et surtout $L_{A}T_{E}X$. En effet, à part les numéros des notes de bas de page et les numéros de page, RTF ne contient aucun registre *variable* comme par exemple la numérotation des chapitres ou des sections. Il est aussi pauvre qu'un document réalisé en « plain $T_{E}X$ » sans astuce particulière.

Ceci signifie que les numéros de sections, de chapitres ou d'énumérations y sont *figés*. Aussi, pour pouvoir remanier le document $L_{A}T_{E}X$ — par exemple pour y ajouter ou supprimer des chapitres — il faudra réviser le texte et y introduire des commandes $L_{A}T_{E}X$ telles que `\chapter` ou `\section`.

21. Si quelqu'un sait convertir en $T_{E}X$ des équations codées en PostScript, l'auteur recevrait volontiers un message d'information !

4. Comment installer RTFLaTeX

Actuellement²², le traducteur est programmé en TurboPascal très élémentaire. Le texte source, le programme exécutable pour PC sous DOS, l'option de style L^AT_EX attendant (`rtflatex.sty`), quelques fichiers annexes spécifiant les ordres RTF à ignorer et les sources METAFONT des polices en mode *outline* sont accessibles par ftp anonyme à `rsovax.lps.u-psud.fr`, dans le *directory* [`anonymous.rtf2tex`]²³.

Les utilisateurs d'ordinateurs non-DOS devront recompiler le programme, mais ceci ne devrait pas poser de problèmes puisque RTFLaTeX ne fait appel à aucune fonction spécifique de DOS sauf pour retrouver la ligne de commande.

5. Utilisation de RTFLaTeX

Pour avoir le mode d'emploi, il suffit de lancer la commande `rtflatex` sans argument, ce qui donne :

```
Usage: RTFLATEX [options] <input-file> [options] <output-file>
Options:
  -b           : set explicit \baselineskip at par ends
  -d<nnn>     : debug level: 1=default, 0=quiet, >1=verbose
  -f<templ>   : file template to put figures (need a * )
  -l<style>   : latex style (default is "report")
  -m           : put debugging marks in output TeX
  -n           : assume NFSS in further [La]TeX-ing
  -o<option>  : latex style option (cumulative)
  -p<value>   : std font size (10, 11, 12)
  -r<file>    : LOG output file (default is none)
  -s           : use slanted instead of italic
  -t           : omit LaTeX \documentstyle etc.
  -x           : inhibit output simplification
  -z<file>    : file containing list of ignored keywords
```

6. Conclusions

RTFLaTeX est une tentative de permettre une transcription relativement fidèle de Word[Perfect] (en pratique RTF) vers L^AT_EX. Aux erreurs mineures

22. Version 1.64 au 4 mars 1994.

23. Pour ceux qui n'ont pas l'habitude de VMS, il suffit de faire `cd rtf2tex` après la connexion par ftp.

près — et moyennant des améliorations futures de la gestion des polices, par exemple en introduisant les polices *en relief* — il satisfait à peu près aux exigences du point 4 de la section 2.

On peut penser que RTFLaTeX convient quand on veut transcrire définitivement en (L^A)T_EX un document créé sous Word[Perfect]; en revanche nous confessons qu'il est notoirement insuffisant pour faire des conversions systématiques en L^AT_EX d'un document dont le *fichier maître* serait maintenu en Word[Perfect].

Quant à la conversion inverse, elle nous paraît théoriquement plus facile car il est en principe plus facile de passer d'une structure riche comme L^AT_EX à une structure pauvre comme RTF ou Word. Mais dans ce sens, les exigences sont bien plus draconiennes en ce sens que le texte imprimé par Word doit coïncider avec le texte L^AT_EX au dixième de point près, ce qui implique un format L^AT_EX utilisant uniquement les polices de Word... qui ne sont pas du domaine public! Notons cependant que ce problème serait contournable en ayant des polices possédant les mêmes TFM que celles utilisées par Word[Perfect], ce qui obligerait simplement à avoir une variante du *format* L^AT_EX. Alors, une conversion fidèle²⁴ pourrait être obtenue en créant un *pilote* qui pourrait s'appeler *dvirtf* et qui aurait bien des ressemblances avec *dvips*...

En résumé, les exigences que nous déduisons de nos observations conduisent à des gageures pratiquement impossibles. Mais, citant une fois de plus Jean de LA FONTAINE, nous dirons que *l'impossibilité n'est pas une excuse valable* car « la raison du plus fort est toujours la meilleure! »

Références bibliographiques

- [1] Daniel TAUPIN, « T_EX: grandeur(s) et servitude(s) », *Microbulletin* **49**, 47–54, 1993.
- [2] *Consignes aux auteurs / Instructions to authors*, TSI **12** - N° **1**, 113–120, 1993.
- [3] Microsoft Word, Manuel de référence, *Microsoft Corporation*, Version 4 pour Apple Macintosh, 168–177, 1989.
- [4] Donald E. KNUTH, The T_EXbook, *Addison-Wesley Publishing Company*, 1984.
- [5] Frank MITTELBACH, Rainer SCHÖPF, « A New Font Selection Scheme for T_EX Macro Packages », TUGboat10#2, 222–238, 1989.

24. Quoique difficilement modifiable sous Word.