



L^AT_EX ET LES SCIENCES HUMAINES

Christophe Masutti

© 2010, C. Masutti

[<creativecommons.org/licenses/by-nc-sa/2.0/fr/>](http://creativecommons.org/licenses/by-nc-sa/2.0/fr/)

Ce document est placé sous licence Creative Common

✉ *christophe * masutti * gmail * com*

→ *http://christophe.masutti.name*



Résumé

Ce document a pour objectif de promouvoir l'utilisation de L^AT_EX en sciences humaines. Adressé au néophyte, il s'agira, en quelques pages, de vulgariser certaines notions afin de proposer un ensemble de pratiques inhérentes à l'activité d'écriture.

Table des matières

1. Le temps de l'ignorance	1
1.1 Wysiwyg	3
1.2 La découverte	3
2. Quels sont les besoins ? quelles sont les pratiques ?	4
2.1 L ^A T _E X est adapté à notre travail	4
2.2 La polyvalence des logiciels	5
3. Qu'est-ce que T_EX et L^AT_EX ?	6
3.1 Histoire et objectifs	6
3.2 Paradoxe du retour à un « vieux » système ?	9
3.3 L ^A T _E X en moins de trente secondes	9
4. Créer des documents T_EX	10
4.1 Les distributions	11
4.2 Les éditeurs	12
5. La bibliographie	14
5.1 BibTeX	14
5.2 Les logiciels de gestion bibliographique	15
6. Des présentations	16
7. Conclusion	18

1. Le temps de l'ignorance

Dès le début de mes études de philosophie, j'avais pris l'habitude de dactylographier mes cours et mes travaux personnels. Pour cela, je me rendais au Centre de Ressources Informatiques de ma faculté, où plusieurs ordinateurs Apple, essentiellement ceux de la gamme Mac Classic, étaient en accès libre pour les étudiants. Un jour d'hiver 1996, certains membres du Centre étaient occupés à installer les nouveaux Powermacs fraîchement livrés. Ceux-ci étaient accompagnés d'écrans « portrait » permettant notamment la visualisation d'une page complète lors de la saisie de texte¹. Quelques mois plus tard, j'appris d'un professeur de linguistique la raison pour laquelle les ordinateurs Mac semblaient être fortement prisés : « c'est une question d'habitude, me dit-il. Auparavant, il était beaucoup plus facile d'utiliser des polices pour le grec ancien ou l'hébreu sur ces ordinateurs. Maintenant, les logiciels de traitement de texte de Microsoft permettent de mieux faire encore, mais nous sommes attachés à Apple. »

Des écrans au format « portrait », des polices de caractère à choisir dans des menus déroulants, imprimer ce que l'utilisateur voit sur son écran... Depuis mon premier ordinateur – et on ne rajeuni pas – taper une dissertation à rendre pour le lendemain ne m'avait jamais semblé plus facile, mises à part les longues heures passées à ajuster le texte, vérifier les notes de bas de page, les numéros de partie, l'harmonisation de la bibliographie, etc. Voir d'un seul coup d'oeil le résultat de sa mise en page tout en entrant du contenu me paraissait être un sommet d'innovation pour les usages de l'ordinateur individuel.

Les années passant, je me perfectionnai de plus en plus. J'en venais à apprécier l'utilisation des formats et des styles, les logiciels de gestion de bibliographie et leurs macros, la joie de pouvoir imprimer un texte dont on savait qu'il ne resterait plus qu'une ou deux heures à passer pour les derniers ajustements en écoutant *Iron Maiden*.

1. Aujourd'hui certains écrans plats sont équipés d'un pivot permettant ce type d'affichage, mais après la génération Mac Classic, ce fut une véritable révolution dans les pratiques.

Je n'ai jamais ressenti le besoin d'avoir une quelconque formation pour cela. J'apprenais à cliquer là où il le fallait et quand il le fallait pour obtenir des documents fort acceptables. Certes, je fais partie de cette génération d'élèves qui ont appris le Basic et le Pascal durant les cours d'Éducation Manuelle et Technique au collège. J'ai de même eu la chance d'avoir très tôt un ordinateur à la maison, d'où une certaine habitude des machines. Entre nous, je me suis toujours demandé pour quelle raison il faudrait une formation pour apprendre à se servir d'un logiciel de traitement de texte² alors même que la pratique relève davantage d'un exercice de singe savant capable de cliquer aux bons endroits (il faut tout de même savoir lire le manuel).

Durant ces belles années estudiantines, les logiciels de traitement de texte n'avaient plus de secret pour moi³, et comme l'Internet faisait son apparition dans les chaumières pour un coût raisonnable, j'avais tout le loisir de récupérer des informations lorsque j'avais besoin d'une fonction ou d'une procédure. À l'instar de beaucoup de mes collègues, et pendant presque 5 ans, je me considérais comme un champion du *Wysiwyg*. Et ce sentiment était exacerbé au fur et à mesure de la progression des performances des ordinateurs individuels, suivies de près par la puissance des applications disponibles sur le marché.

2. Dans la suite de ce document, j'appellerai « logiciel de traitement de texte » un logiciel présentant une interface de traitement de texte, par opposition à l'éditeur de texte. Comme nous le verrons L^AT_EX est bien un logiciel de traitement de texte.

3. Et mes chevilles gonflaient, gonflaient...

1.1 *Wysiwyg*

What You See is What you Get : ce que je voyais sur mon écran était la reproduction exacte de ce que je voulais imprimer ou communiquer par voie électronique. Les défauts de cette approche de l'écriture informatisée n'ont d'égal que le fatalisme ambiant au sein de la communauté des chercheurs dont je commençais à faire partie au début de mon doctorat quelques années plus tard.

La première difficulté que je dus affronter consistait à savoir comment communiquer un texte par voie électronique de manière à ce que mon correspondant puisse le voir s'afficher comme je le voulais. Mission impossible. Entre les différents formats de fichiers et les versions d'un même logiciel, mieux vaut passer par la version papier ou PDF. Sauf bien sûr lorsqu'il s'agit de travailler en groupe sur un même document. Et comme je me déclarais « champion du Wysiwyg », c'est souvent à moi qu'incombait la tâche d'aider les collègues dans leurs difficultés quotidiennes, au mépris de mon propre temps de travail. Tout le monde trouvait cela normal, moi y compris.

Ce fatalisme me faisait cependant accepter beaucoup de choses. Manipuler des images dans un texte devenait une tâche que planifiais à l'avance. Réaliser un diaporama⁴ se transformait en une véritable enquête policière pour savoir si l'équipement de la salle de conférence disposait de la même version du logiciel que j'utilisais. Alors que mes activités informatiques en tant qu'étudiant se résumaient à taper du texte et créer des documents dont j'étais seul juge du rendu final, mes activités d'écriture s'étaient multipliées et, par conséquent, les exigences de qualité augmentaient énormément : soumissions d'articles dans des revues ou les actes d'un colloque, écriture de support de cours... Tout cela confirmait mon idée que l'harmonisation de la typographie et la mise en page (surtout la bibliographie!), sont des tâches que personne ne devrait avoir à faire a posteriori. Premièrement parce que c'est stressant et chronophage, et deuxièmement parce qu'il doit bien y avoir des logiciels pour cela, diantre! Mais lesquels?

4. Non, je ne dirai pas « Pohèreponte ».

1.2 *La découverte*

Le sentiment d'avoir réinventé la roue n'est pas franchement agréable. En 2001, alors qu'auprès d'amis mathématiciens je me lamentais sur ma misère quotidienne et imaginai à haute voix un logiciel digne de ce nom, les visages hilares et les explications qui suivirent me firent entrevoir une issue à la fatalité quotidienne. Je dis : « entrevoir », car l'après-midi que je passai alors au laboratoire de mathématiques de la faculté-d'en-face fit sur moi l'effet d'une véritable bombe : tous mes acquis étaient remis en cause. Outre le fait que j'en ressortis avec un Cd-Rom d'une distribution GNU/Linux, j'avais appris l'existence d'un système de composition de document censé combler mes rêves les plus chers : L^AT_EX. Le pire, c'était que ces logiciels étaient connus et utilisés depuis longtemps!

Je remarquai très vite que la possibilité d'écrire des mathématiques avec L^AT_EX ne constitue en fait qu'un seul de ses nombreux avantages. L'envers du décor, c'est qu'il me fallait apprendre à me passer de l'interface des logiciels de traitement de texte à laquelle j'étais habitué. Cela nécessitait certes une phase d'apprentissage et du temps à y consacrer, mais par-dessus tout, il me fallait aussi envisager qu'un logiciel repose sur des principes et pas seulement sur des possibilités techniques. Ce qui avait conduit à la création de L^AT_EX, c'était une véritable réflexion sur ce qu'est réellement un texte et comment un langage informatique pouvait en permettre la production : une pensée, une structure logique, un style et une typographie appropriés au contenu, et certainement pas cliquer à droite / cliquer à gauche, ajuster, sélectionner, glisser, corriger ici et corriger là. Non pas que les logiciels de traitement de texte empêchent l'écriture d'excellents textes, mais ils ne sont qu'un ensemble de procédés techniques prédéfinis et permettant de remplir une page blanche. Par certains côtés, les logiciels de traitement de texte sont des machines à écrire très évoluées et efficaces. L^AT_EX repose essentiellement sur des principes que l'on applique pour formater du contenu en adaptant les outils à ce contenu, et non l'inverse. Dès lors, il agit comme une « béquille mentale », une aide à l'écriture structurée d'un texte en proposant des outils de mise en forme. Comme le dictionnaire est utile à l'orthographe,

L^AT_EX est un typographe qui recèle en lui l'art ancestral (au moins depuis Gutenberg) de présenter du texte afin qu'il soit lisible et ordonné. Le tout est d'apprendre à travailler en équipe avec son petit Gutenberg domestique.

L'utilisation de L^AT_EX suppose un apprentissage, tout comme n'importe quel logiciel. Cet apprentissage peut sembler difficile car il suscite une approche rigoureuse et systématique de l'écriture de texte. L^AT_EX est adapté à la production de toutes sortes de documents scientifiques ou de documents longs en général. Il n'est pas réservé à certaines disciplines.

2. Quels sont les besoins ? quelles sont les pratiques ?

2.1 *L^AT_EX est adapté à notre travail*

Allin Cottrell, professeur à Wake Forest University (Caroline du Nord), a écrit un article très polémique intitulé « Traitements de texte : stupides et inefficaces »⁵. Dans cet article, Cottrell compare deux démarches d'écriture de texte : celle qui utilise les logiciels de traitement de texte, et celle qui repose sur l'utilisation conjointe d'un éditeur de texte et du langage balisé L^AT_EX. Selon Cottrell, l'écriture d'un texte se divise en deux tâches : la composition du texte et la typographie. Ce que proposent les logiciels de traitement de texte vise à effectuer ces deux tâches simultanément, et cela se traduit par :

- la préoccupation de la mise en page au détriment du contenu lui-même ;
- une baisse de la qualité du rendu final ;
- le risque de perdre de vue la structure logique du texte.

5. Allin Cottrell, « Word Processors : Stupid and Inefficient », 1999 (<http://ricardo.ecn.wfu.edu/~cottrell/wp.html>).

Je pense pour ma part que confronter les deux méthodes n'est que peu productif, même si l'auteur en fait ressortir les avantages de L^AT_EX. Tout comme lui, je pense que l'une des raisons de la prévalence des logiciels de traitement de texte dans les pratiques relève d'un succès économique discutable⁶ (alors que L^AT_EX est libre et gratuit) mais je pense aussi qu'à l'instar de logiciels comme LyX⁷, qui proposent de la saisie semi-wysiwig de type L^AT_EX, il aurait été de bon aloi de développer davantage de logiciels libres permettant une approche a priori moins austère. Comme nous le verrons plus loin, il n'en demeure pas moins que les logiciels d'aide à la saisie L^AT_EX sont aujourd'hui fort agréables à l'utilisation et permettent de produire des documents corrects aussi vite qu'avec un logiciel de traitement de texte.

J'ai évoqué plus haut les deux étapes de l'écriture d'un document : la composition et la typographie. Listons un peu quels sont les besoins inhérents à la production de documents en sciences humaines :

- une argumentation obéissant à une structure logique ;
- communiquer des documents à ses collègues et s'assurer qu'ils puissent les lire ;
- savoir créer des documents de plusieurs types : des articles, des livres, des présentations, des supports de cours, des rapports ;
- gérer des références bibliographiques et autres sources documentaires ;

6. On pourra bien sûr m'objecter qu'il existe aujourd'hui des logiciels de traitement de texte libres et souvent gratuits : OpenOffice.org, Abiword, Kword... Il me faut donc corriger le tir : ces logiciels se sont présentés comme une alternative au logiciel dominant et payant MSWord. Choisir MSWord est d'abord un choix économique : accepter de payer les versions successives d'un même logiciel qui n'a pas subi de franches améliorations, au moins dans ses principes, depuis sa version 2.0. Le format de document produit par MSWord s'est alors imposé comme un standard (ce qu'il n'est pas), rendant prisonniers les utilisateurs. Les alternatives libres visent justement à libérer ces utilisateurs. Si L^AT_EX s'est trouvé en situation minoritaire, c'est parce qu'avec MSWord sont nées à l'échelle de la planète les pratiques liées à ce type de logiciel, L^AT_EX étant considéré comme réservé à une population restreinte, celle des universitaires ou des ingénieurs ayant des besoins spécifiques. Ce qui n'est pas faux d'ailleurs : l'utilisation de L^AT_EX n'est pas adaptée à n'importe quel type de document.

7. Voir <http://www.lyx.org/>.

- respecter les règles typographiques⁸ ;
- assurer la présentation esthétique (et claire) du document.

Selon ces besoins, pour optimiser votre production, vous devez utiliser un outil :

- qui automatise ces actions le plus possible ;
- sur lequel vous pouvez compter (qui ne plante pas) ;
- qui applique les bonnes règles typographiques ;
- qui vous permet de faire ce que vous voulez avec votre bibliographie et votre mise en page ;
- qui garanti la lisibilité de votre document pour toujours ;
- qui documente le formatage de votre travail ;
- qui assure à la fois l'esthétique et la cohérence de votre document final.

Cet outil se nomme L^AT_EX. Mais avant d'aller plus loin, j'ai besoin de préciser ma position au sujet des logiciels de traitement de texte.

2.2 La polyvalence des logiciels

Qu'on se le dise : les logiciels de traitement de texte, libres ou non, sont de bons logiciels ! Si vous désirez composer des documents reposant sur un traitement prédéfini comme du publipostage, ou bien encore un prospectus, une lettre d'invitation, ou une lettre simple (si vous réalisez par exemple un squelette incluant des champs et des logos), il est préférable d'utiliser soit un logiciel de traitement de texte soit un logiciel de composition PAO comme Scribus. L^AT_EX peut s'avérer en effet improductif dans certaines situations, à moins de le maîtriser sur le bout des doigts.

Certes, vous pouvez composer vous même un style pour rédiger des lettres, voire même un script vous permettant d'automatiser la composition et la compilation, et vous obtiendrez un très beau document. Néanmoins, L^AT_EX n'est au départ pas destiné à ce type de document. De même, de très beaux posters peuvent être réalisés avec L^AT_EX mais là encore tout dépend de vos besoins : un poster scientifique se fera avec L^AT_EX tandis qu'un poster publicitaire se fera avec un autre logiciel plus approprié. Nous verrons par contre que la réalisation de diaporama est bigrement convainquante avec L^AT_EX.

L'essentiel est de retenir deux choses :

- premièrement, il existe des logiciels adaptés aux documents que vous voulez produire ;
- deuxièmement, utiliser un seul logiciel pour tout faire est un mauvais choix, qu'il s'agisse de L^AT_EX ou d'un logiciel de traitement de texte.

Combien de vos collègues ont toute leur bibliographie écrite au kilomètre dans un document MSWord ? Combien utilisent MSWord pour stocker et communiquer des images ? Combien utilisent MSWord pour créer des tableaux ? Combien utilisent MSword pour prendre de simples notes ? etc.

Je me répète : à toutes ces tâches correspondent des logiciels différents. Lorsque ces derniers sont libres, ils sont inter-opérables et facilitent la communication des données. L^AT_EX en fait partie, et il est temps dé-

8. Je pense aux pauvres thésards cherchant désespérément un document de référence, n'en trouvant point, et obéissant à leur directeur de thèse lui-même ignorant en la matière. Si aucune formation sérieuse n'existe au sujet de la typographie dans un cursus étudiant, c'est tout simplement parce que les logiciels de traitement de texte ne se prêtent pas à l'exercice d'une bonne typographie. Ce n'est pas en apprenant à des étudiants à cliquer aux bons endroits qu'on en fera de bons rédacteurs. Voir à ce propos un excellent guide de typographie à la française : <http://jacques-andre.fr/faqtypo/lessons.pdf>

sormais de passer aux explications sérieuses.

La soi-disante polyvalence des logiciels courants de bureautique induit en erreur la plupart d'entre nous. Ces logiciels disposent d'une batterie d'outils de plus en plus imposante, complexe et diversifiée. Or, nous en utilisons à peine le dixième puisque si ces logiciels sont polyvalents, nous ne réalisons qu'un nombre réduit de tâches régulières. Mieux vaut donc choisir des outils spécifiques à nos besoins selon les tâches que nous désirons accomplir. L^AT_EX n'échappe pas à la règle et ne remplace donc pas tous les logiciels.

3. Qu'est-ce que T_EX et L^AT_EX ?

3.1 Histoire et objectifs

T_EX (prononcez [tek]) est un système de composition typographique inventé par Donald Knuth, de l'université de Stanford, à la fin des années 1970. Le nom de ce système est dérivé du grec τεχ, un diminutif de τέχνη (technique, savoir-faire), d'où la prononciation. D. Knuth, en effet, avait créé ce système afin de produire des documents techniques structurés et de qualité. De tels documents ne doivent pas seulement être compréhensibles, ils doivent être lisibles, c'est à dire par exemple que la manière d'écrire telle ou telle formule mathématique doit être renseignée. Ces documents doivent aussi être affichés de la même manière sur n'importe quelle machine sans dépendre d'un quelconque changement de version. Pour cela, il faut bien séparer la composition du rendu final.

Imaginez que, comme le professeur Tournesol, vous reveniez d'expédition à bord de l'Aurore, et que vous étudiez un morceau de l'aérolithe tombée dans l'océan⁹. Pour avoir la primeur de la découverte scientifique sur les concurrents, vous décidez de dicter votre article par radio, à savoir la démonstration de cette nouvelle sensationnelle et la composition chimique de l'aérolithe. Si vous avez pleine confiance en votre correspondant, il va néanmoins vous falloir décrire précisément les éléments

basiques de la structure de votre article (titres et sous-titres, la manière de mettre en page vos calculs, le contenu d'un tableau, etc.). Pour le reste, vous faites confiance à la revue qui calibrera votre article selon les standards en cours : titre, auteur, abstract, plan et corps de l'article, bibliographie, etc.

Et bien, c'est exactement ce qu'a réalisé D. Knuth : un système qui permette d'**interpréter** vos commandes (commandes de titre, commande de sous-titre, par exemple) et de produire un document final formaté selon vos souhaits et d'après des fichiers de styles prédéfinis (ou définis par vos soins). Pour cela il faut *baliser* (voir Section 3.3) votre texte, c'est à dire placer des commandes qui formatent le texte exactement comme vous le souhaitez. Cela revient à doubler le texte d'un ensemble de commandes qui permettent de structurer le contenu. Mais cela revient aussi à automatiser le traitement du texte lui-même : les commandes ne sont là que pour les besoins de la structuration. Selon le style choisi et la disposition générale des éléments (marges, colonnes, taille des caractères, espacements, sauts de page, etc.), c'est le rendu L^AT_EX, après compilation, qui déterminera à quoi ressemblera votre entête de chapitre ou votre bibliographie. Tel est l'avantage de séparer la forme et le contenu.

Cependant, du point de vue historique, c'est un petit peu plus compliqué que cela. D. Knuth a créé deux « compilateurs » qui interprètent le contenu au format texte que vous avez entré et dans lequel vous avez utilisé des commandes. Le premier compilateur est T_EX qui, effectivement, interprète les commandes. Le second est *Metafont* qui, pour faire court, dessine littéralement les lettres (D. Knuth est un grand mathématicien : il a utilisé les courbes de Bézier pour dessiner le contours des lettres et la manière de calculer tout ceci!).

L'apport évident de cet élément *Metafont*, c'est que le document final, destiné à être imprimé où même seulement lisible à l'écran, aura une qualité typographique digne des plus grands imprimeurs. Sont calculés et calibrés les éléments comme : hauteur et largeur des lettres, la distance entre les lettres, les espaces accompagnant les ponctuations, les espaces séparant les lignes et les paragraphes, etc. Quand à T_EX, il s'amuse à

9. Voir Hergé, *L'étoile mystérieuse...*

interpréter la structure de votre composition mais traite aussi le placement d'une figure dans le texte, l'hyphénation, etc. En somme c'est l'esthétique de votre document qui est assurée, *en plus* des commandes que vous avez entrées.

Afin de faciliter l'entrée des commandes qui créeront la structure typographique des pages que vous voulez produire, un autre chercheur de très haut niveau, Leslie Lamport¹⁰ a imaginé un système de commandes permettant d'interagir avec T_EX à un plus haut niveau¹¹. Il nomma ce système La-T_EX (L-a pour les deux premières lettres de son nom). Il s'agit d'un ensemble de macro-commandes toutes faites pour des mises en pages calibrées à l'avance. Ainsi, vous pouvez choisir d'écrire un livre ou un article scientifique en spécifiant simplement le type de document que vous voulez produire au bout du compte. Pour le reste, vous pouvez soit vous contenter des commandes les plus simples soit ajouter des séquences plus complexes selon vos besoins.

Vous voulez un exemple concret ? C'est normal. Voici l'exemple très simple du double point (vous lirez dans le second encart le texte tel qu'il apparaît une fois compilé) :

————— Le double point —————

Il existe deux règles différentes en anglais et en français: les uns utilisent une espace insécable entre le mot et le double point qui le suit; les autres n'utilisent pas d'espace avant le double point.

Ainsi, dans cet exemple, notez que je ne me soucie pas de la règle française: je laisse le compilateur gérer cela tout seul. Pour cela j'ai spécifié au début de ma composition la langue dans laquelle je désire produire mon document.

10. Voir http://en.wikipedia.org/wiki/Leslie_Lamport.

11. Voir http://www.ctan.org/what_is_tex.html.

Il existe deux règles différentes en anglais et en français : les uns utilisent une espace insécable entre le mot et le double point qui le suit ; les autres n'utilisent pas d'espace avant le double point. Ainsi, dans cet exemple, notez que je ne me soucie pas de la règle française : je laisse le compilateur gérer cela tout seul. Pour cela j'ai spécifié au début de ma composition la langue dans laquelle je désire produire mon document.

Et c'est pareil pour les règles typographiques concernant les guillemets, les numéros de notes de bas page¹², les alinéas, et tous les petits détails qui vous gâchent un après-midi.

En guise de second exemple, nous pouvons voir que l'insertion de figures pourrait paraître a priori austère, mais à bien y prendre garde, cela demande un surcroît de rigueur :

————— Insertion de figure —————

```

1 \begin{figure}[!h]
2 \centering
3 \includegraphics[scale=1]{lettreT.png}
4 \caption{Ceci est la lettre T}
5 \label{lettreT}
6 \end{figure}

```

Sans entrer dans les détails, voici une petite interprétation du code : je déclare un environnement « figure » (lignes 1 et 7), que je centre (ligne 2), et je fais appel à l'image que je souhaite afficher, à l'échelle 1/1 (ligne 3). La légende (« caption », ligne 4) est renseignée ainsi que le label (ligne 5), qui me servira pour faire appel à la figure dans le corps de ma composition sans me soucier de son numéro, qui d'ailleurs peut changer si je décide de rajouter une figure dans les pages précédentes.

12. ← Comme vous le voyez dans ce document, ce sont les « frenchfootnotes » qui sont utilisées, c'est à dire le numéro de la même taille que le texte, situé sur la même ligne et suivi d'une espace. Pour que ce style soit appliqué, il suffit de déclarer son utilisation au début du document.



FIGURE 1 – Ceci est la lettre T

Utilisation de label

Pour faire référence à la Figure `\ref{lettreT}`, je fais appel à son nom, ce qui m'évite de numéroter manuellement les figures que j'emploie.

Pour faire référence à la Figure 1, je fais appel à son nom, ce qui m'évite de numéroter manuellement les figures que j'emploie.

Vous noterez par ailleurs que L^AT_EX calcule seul la bonne disposition de la figure en respectant les règles de mises en page. Je n'ai pas à me soucier de l'emplacement, de la distance avec le texte et de la légende.

Je ne saurais terminer cette section sans mentionner le travail remarquable de Bernard Gaulle sur le projet *french*. Pour cela il faut bien comprendre comment se structure le *préambule* d'un document *.tex*. Dans cette partie se déclarent les « grandes » commandes que l'on souhaite utiliser et qui seront interprétées par le compilateur comme des commandes qui formatent l'ensemble du document. Les « petites » commandes elles, se déclarent au fil du texte, par exemple pour signaler un mot en italique. Parmi les « grandes » commandes il y a celles qui visent à déclarer l'utilisation de *paquets* spécifiques, c'est à dire des ensembles de commandes tellement complexes qu'elles définissent à elles seules un style de document, ou une manière spécifique de formater le texte. Ainsi le paquet *babel*, vise à faire respecter les règles typographiques communes à l'alphabet latin (l'un des avantages de L^AT_EX est de permettre l'utili-

sation d'un très grand nombre de langues et pas forcément latines). En complément de *babel*, Bernard Gaulle a créé un paquet nommé *french* destiné à la langue française et chargé de rendre un texte conforme aux règles typographiques françaises en général. Grâce au travail de Bernard Gaulle¹³, il est possible, par exemple, de permettre à L^AT_EX d'harmoniser les guillemets, les notes de bas de pages, la bibliographie, les index mais aussi écrire des lettres « à la française » (*french* permet par exemple de placer correctement les adresses de l'expéditeur et du destinataire) et bien d'autres choses encore. C'est un bel exemple de l'étendue des capacités de L^AT_EX : parce que c'est un logiciel libre, il est possible de l'améliorer sans cesse et d'y adjoindre une infinité de fonctions supplémentaires correspondant aux besoins des utilisateurs. Et si cela ne suffit pas, moyennant un peu de pratique, vous pouvez vous même créer vos propres commandes.

13. Bernard Gaulle, entre autre président fondateur de l'association GUTemberg (Groupe francophone des utilisateurs de T_EX, L^AT_EX et logiciels compagnons – gutenber.org) est décédé en 2007. Le projet *french* poursuit actuellement sa course sous le nom *efrench* (<http://projects.tuxfamily.org/group.pl?name=efrench>). Voir à ce propos l'article de Laurent Bloch (<http://www.laurentbloch.org/spip.php?article166>).

3.2 Paradoxe du retour à un « vieux » système ?

Comme je l'ai déjà dit, L^AT_EX n'est pas recommandé pour tous les types de documents. Il vaut mieux employer les logiciels appropriés à la production voulue. Ainsi, il en va de la différence entre la PAO et le traitement de texte ou entre le logiciel de graphisme et le schéma réalisé avec les outils de « dessin ».

Dès lors, L^AT_EX serait-il fait pour les nostalgiques des caractères en plombs ? Depuis l'époque de Gutenberg, du chemin a été parcouru, en particulier le passage des machines à écrire aux imprimantes électroniques, tout en répondant aux règles de qualité typographique de plus en plus exigeantes. Comme pour le dessin des caractères, la composition de texte a elle aussi progressé ! Nous ne donnons plus nos manuscrits à l'imprimeur-typographe, mais nous pouvons profiter des fonctions typographiques intégrées à notre logiciel de composition.

L'utilisation de L^AT_EX paraît bien plus « manuelle » que celle d'un logiciel de traitement de texte *Wysiwyg*. Et pourtant, moyennant l'apprentissage de quelques notions très simples, celle d'un *langage balisé*, il est possible d'obtenir un résultat que le commun des utilisateurs de logiciels de traitement de texte ne pourra jamais obtenir, ou alors avec beaucoup de difficultés. Au contraire, comme nous le verrons plus loin, l'utilisateur de L^AT_EX devra simplement se servir d'un éditeur de texte (voir Section 4.2) et lancer quelques commandes, autrement dit, utiliser une interface réduite, des ressources système minimalistes (inutile d'avoir une machine ultra-performante, une « vieille bécane » suffit), et tout le potentiel de son propre cerveau.

Théoriquement, tout est possible à l'aide d'un simple éditeur de texte. Tout dépend du niveau de difficulté que vous êtes prêt à affronter pour réaliser une tâche. Par exemple, la fonction de suivi de modifications intégrée à la majorité des logiciels de traitement de texte est chose bien utile, mobilisant des codes de couleurs, et permettant de travailler à plusieurs sur un même document. Mais vous pouvez aussi comparer deux fichiers texte avec *diff*¹⁴, y compris en utilisant la coloration syntaxique.

14. Définition sur Wikipédia : <http://fr.wikipedia.org/wiki/Diff>.

C'est une question de préférence et non d'efficacité, mais ce n'est surtout pas une question de rapidité. De la même manière, ce n'est pas parce qu'on utilise un logiciel qui n'a pas de boutons qui clignotent, que le résultat produit sera moins bon...

3.3 L^AT_EX en moins de trente secondes

Considérons un instant l'exemple suivant :

Facile!

```
\textbf{Bonjour} \\
--- Quelle est la différence entre un
\textit{hot dog} et un \textit{sandwich}?
```

Bonjour

— Quelle est la différence entre un *hot dog* et un *sandwich* ?

Je vous vois venir : « Comment ? taper des commandes et des signes bizarres pour obtenir de simples effets comme le gras, l'italique et un saut de ligne ? C'est scandaleusement contraignant ! ». Et vous ne croyez pas si bien dire. Il est si facile de sélectionner le texte et cliquer sur « italique » pour obtenir un texte en italique. . . quoique quitter le clavier, saisir la souris, sélectionner le texte, et cliquer sur « italique » demande un certain temps¹⁵.

Cela dit, en moins de trente secondes, vous venez de comprendre ce qu'est un langage balisé ainsi que sa syntaxe. C'est un langage qui formate le texte en fonction des commandes qu'on lui donne. Ainsi l'italique et le gras sont obtenus en *balisant* le texte par une déclaration de commande, une accolade de début et une accolade de fin.

15. Oui, les raccourcis clavier au cours de la frappe permettent d'aller plus vite.

Un texte ainsi composé se « compile », c'est à dire qu'il faut lancer une commande qui fera en sorte d'instruire votre document pour le formater et produire ensuite, par exemple, un document lisible aux formats Postscript, DVI, PDF, HTML, etc.

Vous avez aussi compris (il s'en passe des choses en trente secondes!) que pour composer un tel texte, vous n'avez pas besoin d'un logiciel très complexe. Un simple éditeur de texte peut suffire. Et comme votre document L^AT_EX (qui portera le plus souvent un nom de fichier avec une extension .tex) est écrit en plein texte, il est par définition lisible à partir n'importe quel logiciel censé afficher du texte. En d'autres termes, vous n'avez pas à vous soucier de savoir si le correspondant à qui vous envoyez le texte dispose bien du même logiciel que vous.

Enfin, après avoir digéré ce que je viens d'affirmer, vous allez comprendre que le fait de structurer son texte à l'aide de commandes vous permet de vous concentrer sur le fond : paradoxalement, en entrant les commandes, vous structurez votre texte, ses parties, ses notes de bas de page, les exemples et les définitions qu'il comporte, etc. En somme, vous construisez la « grammaire » de votre texte : les redondances, les répétitions, les références inutiles, l'équilibrage et la hiérarchie des parties, chapitres, sections et sous-sections, tout cela vous apparaît sans que vous ayez à modifier l'affichage en temps réel. Si, de surcroît, vous définissez dans le préambule¹⁶ le style et la classe de votre article (sa mise en page finale), le calibrage se fait automatiquement, et vous savez déjà sous quelle forme apparaîtra votre document, sans vous en soucier au cours de sa composition. De quoi peut-on rêver de mieux pour écrire en toute tranquillité ?

Avant de poursuivre, il me faut faire ici une mise au point. Mon objectif n'est pas de vous apprendre à vous servir de L^AT_EX. Je ne cherche qu'à vous indiquer des pistes qui vous permettront d'aller plus loin dans cet apprentissage. Vous trouverez à la fin de ce texte une liste de sites

16. Je rappelle que le préambule est la partie située au début du fichier où l'utilisateur détermine les commandes de mise en page générales.

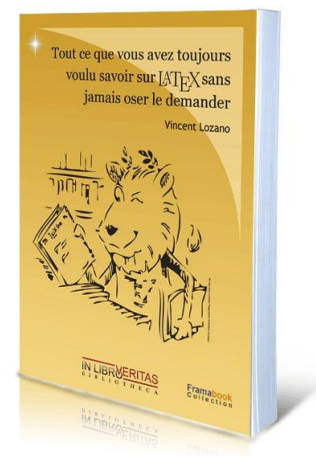


FIGURE 2 – *Tout ce que vous avez toujours voulu savoir...* (V. Lozano)

internet et une bibliographie qui vous seront d'une grande aide. En attendant, je me permets de signaler l'ouvrage de Vincent Lozano [4], particulièrement accessible et adapté.

Il est temps maintenant d'aborder quelques questions d'ordre pratique.

4. Créer des documents T_EX

Avant tout, il convient ici de préciser que les logiciels que je vais citer ne sont pas gratuits : ils sont *libres*. Pour savoir ce qu'est un logiciel libre, je vous invite à vous renseigner par vous même. De nombreux sites internet sont dédiés au logiciel libre¹⁷. Je fais cette distinction ici pour que le lecteur comprenne bien que même si T_EX et les logiciels associées sont disponibles gratuitement, c'est toute une communauté d'utilisateurs qui les maintient, et avec qui il est important d'échanger, non seulement afin d'apprendre à se servir de ces logiciels (des forums, des listes de discussion et des revues sont disponibles), mais aussi pour aider les utilisateurs

17. Commencez par ici : http://fr.wikipedia.org/wiki/Logiciel_libre.

moins avancés, et contribuer même indirectement à l'amélioration de ces logiciels¹⁸. En somme, si vous utilisez un logiciel libre, le prix à payer est en quelque sorte la participation à la communauté, dans la mesure de vos possibilités.

4.1 Les distributions

Comme nous l'avons vu précédemment, L^AT_EX étant en quelque sorte une sur-couche de T_EX, il y a plusieurs versions de L^AT_EX disponibles. Ces versions sont nommées des *distributions*, un peu comme à la manière du système GNU/Linux.

Je me contenterai de citer deux distributions qui satisferont pleinement les utilisateurs. Il en existe d'autres, notamment ConTeXt¹⁹, un outil très puissant pour l'édition de documents scientifiques, et T_EX_E²⁰, une distribution qui n'est plus maintenue à ce jour. Les deux distributions suivantes sont en fait les plus courantes :

Texlive

Il s'agit de la distribution la plus en vogue²¹. Elle est accessible pour les trois systèmes d'exploitation les plus connus : GNU/Linux, MacOS (MacTeX est basée sur Texlive²²), et MSWindows. TeXLive utilise tout le matériel des archives CTAN* et constitue de ce fait une distribution très complète²³. Le téléchargement et l'installation de TeXLive (DVD ou installateur en ligne) sont décrits sur le site officiel et le manuel de l'utilisateur. Certains systèmes GNU/Linux, notamment Ubuntu, utilisent TeXLive comme la distribution par défaut à installer directement via les dépôts, comme n'importe quel autre logiciel.

18. Tout le monde ne sait pas programmer, mais tout le monde peut reporter des bugs, faire part d'un besoin, participer aux traductions, aider les autres utilisateurs, faire un don à telle association d'utilisateurs, etc.

19. Voir ici <http://www.pragma-ade.nl/> et là : http://wiki.contextgarden.net/Main_Page.

20. Voir <http://www.tug.org/tetex/>.

21. Voir <http://www.tug.org/texlive/>.

22. Il s'agit de l'ajout à TeXLive d'un installateur spécifique à MacOS, voir <http://www.tug.org/mactex/2009/>.

23. Voir le manuel de l'utilisateur de TeXLive en français sur la page <http://tug.org/texlive/doc/texlive-fr/texlive-fr.html>.

MiKTeX

Pour Windows, MikTeX²⁴ est une distribution efficace, facile à l'installation et à la prise en main. Cette distribution inclut de même un système de mise à jour automatique. Elle est aussi capable de proposer le téléchargement et l'installation à la volée d'un paquet manquant lorsque l'utilisateur tente de compiler un document (fonctionnalité pratique mais peu pédagogique). MikTeX fonctionne en binôme avec TeXnicCenter, un environnement d'édition de documents T_EX.

***CTAN**

The Comprehensive T_EX Archive Network (CTAN) constitue le site de référence pour tout le matériel concernant T_EX. Il s'agit du stockage des dépôts des différents logiciels, styles, et autres paquets mis à jour par la communauté des utilisateurs. Si vous utilisez une distribution comme TeXLive, le besoin d'aller télécharger et d'installer « manuellement » un paquet spécifique sera très rare. Les mises à jour de cette distribution sont basées sur la disponibilité des dernières versions de ces paquets.

24. Voir <http://www.miktex.org/>.

Je ne parlerai pas des spécificités techniques des distributions pour la simple et bonne raison qu'il n'est pas nécessaire d'entrer dans de tels détails pour commencer avec L^AT_EX. Retenons simplement que quelle que soit la distribution que vous utilisez, vous pouvez utiliser n'importe quel éditeur de texte pour réaliser vos documents. Certains proposent une panoplie de commandes permettant de lancer la compilation des documents sans trop avoir à se soucier du rendu lorsqu'on crée un simple document, ni même d'avoir une connaissance précise de ce qui se passe lorsqu'on compile un document `.tex`, par exemple pour en faire une sortie en PDF. Ainsi, si vous utilisez la ligne de commande, la compilation d'un document pour produire un PDF pourra s'obtenir en entrant `pdflatex nomdufichier.tex`. Et si vous utilisez un éditeur spécialisé pour la composition de documents `.tex`, un clic sur l'icône correspondante suffira.

4.2 Les éditeurs

Les éditeurs L^AT_EX sont des assistants à la saisie de code T_EX et au lancement des commandes. Un document peut parfaitement être rédigé à l'aide d'un simple éditeur de texte classique et les commandes qui serviront à le compiler peuvent être lancées à partir d'une console. C'est d'ailleurs la manière que préfèrent ceux qui abhorrent ces éditeurs spécialisés. Pour ma part, je pense que les éditeurs L^AT_EX permettent une prise en main rapide et facilitent la compréhension des commandes à condition que l'utilisateur tente de comprendre en même temps ce qu'il fait. Une chose est d'aider en un clic à mettre un mot en italique, une autre est de bien structurer son contenu, le compiler correctement et éventuellement chercher les erreurs.

Un éditeur de texte est habituellement fourni avec le système d'exploitation de votre machine. Il ne faut pas le confondre avec un logiciel de traitement de texte. Comme nous l'avons vu précédemment, un fichier T_EX est d'abord un fichier texte comportant des instructions qui seront interprétées. Un fichier texte est une suite de caractères imprimables, d'espaces et de retours à la ligne. Le plus souvent, lorsque vous sauve-

gardez un fichier texte, il le sera au format ASCII (American Standard Code for Information Interchange), la norme de codage la plus largement compatible²⁵. Si bien que la grande vertu des fichiers texte est qu'ils seront toujours lisibles et que vos correspondants pourront toujours les lire, indépendamment de l'éditeur qu'ils utilisent. A contrario, votre fichier `zaza.tghb` que vous avez créé avec le logiciel *Machinealaver 9.02* provenant de la firme Grobug qui a elle-même breveté le format `.tghb`, risque de n'être lisible que par le logiciel *Machinealaver*, éventuellement avec des versions précises de ce logiciel, que votre correspondant ne possède pas forcément.

Les éditeurs, eux, sont très nombreux, offrant diverses fonctionnalités des plus basiques aux plus complexes : le copier/coller, chercher/remplacer, des macro-commandes, des interfaces spécifiques, etc. L'inventivité des créateurs d'éditeurs de texte est sans doute la plus illustrative du monde du logiciel libre. En effet, l'édition de fichiers texte est l'activité principale du programmeur informatique, il est donc tout à fait normal que les programmeurs affectionnent et améliorent tout particulièrement leurs outils principaux. L'histoire du célèbre Emacs²⁶, créé par Richard Stallman, montre combien les éditeurs peuvent être optimisés et améliorés pour faciliter la programmation. Emacs est plus qu'un éditeur : les commandes, les raccourcis, les fonctionnalités dont il s'est enrichi au fil du

25. Ce n'est pas tout à fait exact. L'ASCII définit 128 caractères (de 0 à 127) mais il y a de grandes chances que votre texte soit sauvegardé dans l'un des formats qui étendent l'ASCII, et qui prennent en compte, par exemple, les caractères accentués. Vous avez sans doute croisé les normes ISO 8859-1 (Latin-1) et UTF-8. Généralement les éditeurs de texte proposent de choisir l'encodage dans lequel vous souhaitez sauvegarder votre fichier texte. Ils proposent aussi de définir l'encodage avec lequel vous désirez lire un fichier. Vous avez certainement déjà fait face à ce genre de difficultés, par exemple lorsque votre correspondant vous envoie un courriel encodé d'une certaine manière et que vous peinez à le lire parce que votre logiciel de courriel affiche le message en utilisant un autre encodage. La difficulté est surmontée aisément en sélectionnant le bon encodage dans les options de votre logiciel. Il en va de même pour les pages web (sous Firefox, il vous suffit d'utiliser le menu Affichage/encodage des caractères), même si une bonne page web définit elle-même l'encodage utilisé pour que votre navigateur utilise le bon.

26. Voir <http://www.gnu.org/software/emacs/>.

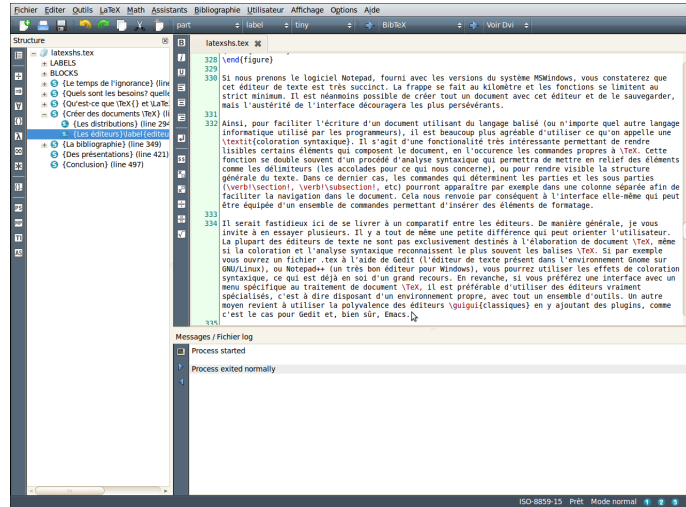


FIGURE 3 – Texmaker, un éditeur L^AT_EX avec coloration syntaxique et de multiples outils.

temps ont fait de lui un environnement auto-suffisant, que l'utilisateur peut configurer lui-même à volonté, jusqu'à pouvoir y lire son courriel.

Si nous prenons le logiciel Notepad, fourni avec les versions du système MSWindows, vous constaterez que cet éditeur de texte est très succinct. La frappe se fait au kilomètre et les fonctions se limitent au strict minimum. Il est néanmoins possible de créer tout un document avec cet éditeur et de le sauvegarder, mais l'austérité de l'interface découragera les plus persévérants.

Ainsi, pour faciliter l'écriture d'un document utilisant du langage balisé (ou n'importe quel autre langage informatique utilisé par les programmeurs), il est beaucoup plus agréable d'utiliser ce qu'on appelle une *coloration syntaxique*. Il s'agit d'une fonctionnalité très intéressante permettant de rendre lisibles certains éléments qui composent le document, en l'occurrence les commandes propres à T_EX. Cette fonction se double souvent d'un procédé d'analyse syntaxique qui permettra de mettre en

relief des éléments comme les délimiteurs (les accolades pour ce qui nous concerne), ou pour rendre visible la structure générale du texte. Dans ce dernier cas, les commandes qui déterminent les parties et les sous parties (`\section`, `\subsection`, etc) pourront apparaître par exemple dans une colonne séparée afin de faciliter la navigation dans le document. Cela nous renvoie par conséquent à l'interface elle-même qui peut être équipée d'un ensemble de commandes permettant d'insérer des éléments de formatage.

Il serait fastidieux ici de se livrer à un comparatif entre les éditeurs. De manière générale, je vous invite à en essayer plusieurs. Il y a tout de même une petite différence qui peut orienter l'utilisateur. La plupart des éditeurs de texte ne sont pas exclusivement destinés à l'élaboration de document T_EX, même si la coloration et l'analyse syntaxique reconnaissent le plus souvent les balises T_EX. Si par exemple vous ouvrez un fichier `.tex` à l'aide de Gedit (l'éditeur de texte présent dans l'environnement Gnome sur GNU/Linux), ou Notepad++ (un très bon éditeur pour Windows), vous pourrez utiliser les effets de coloration syntaxique, ce qui est déjà en soi d'un grand recours. En revanche, si vous préférez une interface avec un menu spécifique au traitement de document T_EX, il est préférable d'utiliser des éditeurs vraiment spécialisés, c'est à dire disposant d'un environnement propre, avec tout un ensemble d'outils. Un autre moyen revient à utiliser la polyvalence des éditeurs « classiques » en y ajoutant des plugins, comme c'est le cas pour Gedit et, bien sûr, Emacs.

Voici une liste non exhaustive des éditeurs spécialisés (et libres) :

- TeXnicCenter – <http://www.texniccenter.org/>
- Texworks – <http://tug.org/texworks/>
- Texmaker – http://www.xmlmath.net/texmaker/index_fr.html
- Kile – <http://kile.sourceforge.net/>
- Gummi²⁷ – <http://gummi.midnightcoding.org/>

27. On notera que la particularité de Gummi est d'offrir une mise à jour en temps réel du rendu DVI du document T_EX produit par l'utilisateur. C'est

- TeXShop – <http://www.uoregon.edu/~koch/texshop/>
- Latexila – <http://latexila.sourceforge.net/>
- Emacs + AucTeX – <http://www.gnu.org/software/auctex/>

5. La bibliographie

La gestion de la bibliographie n'est pas seulement un élément indispensable à l'exercice de création de documents structurés. Une part de ce travail ne s'obtient qu'avec l'aide de logiciels spécifiques. Là encore, c'est un changement de pratique qui doit s'opérer, du moins pour ceux qui ne gèrent pas (encore) leurs données bibliographiques.

5.1 BibTeX

BibTeX est à la fois un logiciel de gestion de références bibliographiques et un format de fichier (.bib) conçu par Leslie Lamport (voir plus haut) et Oren Patashnik en 1985. En tant que logiciel, BibTeX fonctionne avec L^AT_EX : à la fin du document ou dans un fichier .bib séparé, on entre les références bibliographiques puis on procède à la compilation. Cela aura pour effet, selon le style choisi, de produire une bibliographie qui apparaîtra à l'endroit souhaité.

BibTeX consiste à traiter des champs qui correspondent aux différents éléments constitutifs de la bibliographie : le type de document (article, monographie, thèse, etc.), l'auteur, la date, le titre, etc. Le format .bib est un format de fichier contenant ces éléments ordonnés de la manière suivante :

— Dans un .bib —

```
@BOOK{Dupont1988,
  author = {Jean Dupont},
  title = {Les sycophantes dans le système
          juridique athénien},
  publisher = {PUF},
  address = {Paris},
  year = {1988},
}
```

L'utilisation de BibTeX permet une chose essentielle : pouvoir générer les références bibliographiques d'un document en les faisant correspondre à style précis.

Si, dans le corps du texte, j'entre ceci :

— Exemple —

```
On peut voir chez J. Dupont\cite{Dupont1988} à quel point
les jurys populaires athéniens appréciaient
le travail des délateurs professionnels.

\bibliographystyle{lestyle}
```

presque du Wysiwyg... Mais cette compilation à tout va est davantage une source de difficultés pour l'utilisateur non averti. Mieux vaut dans ce cas se rapprocher de TeXMacS (<http://www.texmacs.org/tmdoc/index.fr.html>) ou de LyX (<http://www.lyx.org>).

```
\bibliography{fichier_biblio}
```

J'obtiens quelque chose comme :

On peut voir chez J. Dupont[1] à quel point les jurys populaires athéniens appréciaient le travail des délateurs professionnels.

Références

[1] Jean DUPONT (1988), *Les sycophantes dans le système juridique athénien*, Paris : PUF.

Évidemment, les références dans le corps du texte comme la liste bibliographique et ses entrées peuvent être affinés et configurés à loisir, selon les besoins. Il peut même arriver que BibTeX ne suffise pas à lui seul. Dans ce cas, d'autres outils permettent d'élaborer un ensemble de manière plus fouillée comme par exemple le paquet BibLatex²⁸, spécialement dédié aux sciences humaines.

5.2 Les logiciels de gestion bibliographique

Pour générer des fichiers .bib, on utilise des logiciels de gestion bibliographique. Ces derniers sont très évolués et permettent bien d'autres choses que la simple sortie BibTeX. C'est avant tout une question de pratique. J'ai souvent été surpris par le nombre de personnes de mon entourage professionnel qui ignorent l'existence de ce type de logiciel. Et pourtant... que de temps gagné à ne pas avoir à rechercher fastidieusement une référence perdue dans un document rédigé!

Un autre sujet de mécontentement de ma part concerne les revues. Je trouve particulièrement effarant de voir des revues professionnelles indiquer aux auteurs que c'est à eux d'arranger les références bibliographiques de leurs articles. Qui plus est, chaque revue demande aux

²⁸. Voir <http://linuxetleschoses.tuxfamily.org/Articles/biblatex.php>.

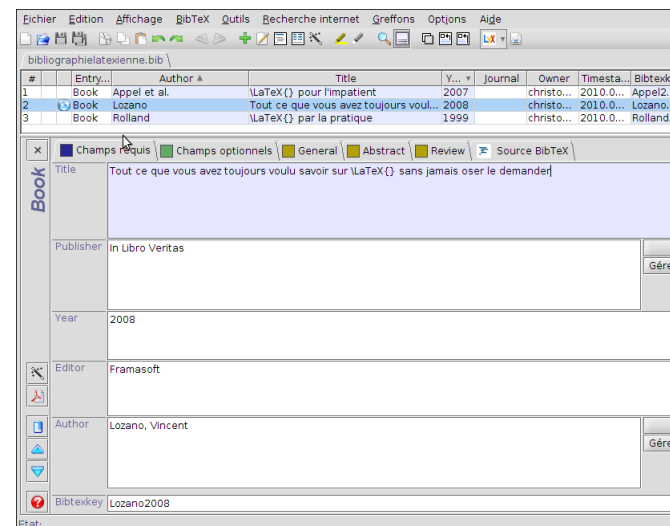


FIGURE 4 – JabRef, un logiciel de gestion bibliographique efficace.

auteurs un arrangement qui, bien souvent, ne correspond à aucun standard en la matière²⁹. Il suffirait de demander un fichier BibTeX aux auteurs pour résoudre ce genre de difficulté, synonyme d'une perte de temps considérable. En effet, même sans utiliser L^AT_EX, le format BibTeX est parfaitement adapté à l'exercice : certains logiciels de gestion bibliographique possèdent un jeu de macros destinées à créer un « tunnel » entre eux et les logiciels utilisés pour créer les documents, qu'il s'agisse d'éditeurs L^AT_EX ou d'autres logiciels de traitement de texte comme OpenOffice.org. Par conséquent, même les revues qui n'utilisent pas L^AT_EX pourraient très bien se servir de ce format parfaitement compatible.

²⁹. Oui, on parle bien de standard. La norme internationale ISO 690 définit les principes de citation de documents. Voir http://fr.wikipedia.org/wiki/ISO_690. On peut aussi se demander pourquoi les revues n'utilisent pas des styles semblables, un peu comme si chaque comité éditorial trouvait un plaisir sadique à inventer de nouveaux styles.

Enfin, l'usage de tels logiciels est adapté à la phase de recherche bibliographique. La plupart d'entre eux permettent en effet de récupérer des fiches bibliographiques au format BibTeX par exemple sur les catalogues en ligne de type SuDoc. Dans la pratique, non seulement il est très sain d'utiliser BibTeX, mais l'efficacité se trouve multipliée par l'usage de L^AT_EX en garantissant la production de documents dont les références bibliographiques sont harmonieuses et parfaitement traitées.

Vous trouverez une liste assez complète de logiciels de gestion bibliographique libres à l'adresse <http://linuxetleschoses.tuxfamily.org/Logiciels/biblioref.php>.

6. Des présentations

L^AT_EX ne permet pas seulement d'élaborer de bons articles ou de bons livres. J'ai cité précédemment la possibilité d'écrire des lettres (grâce au paquet *french*), même si je persiste à penser que L^AT_EX n'est pas forcément le meilleur outil pour cela. Là où L^AT_EX permet aussi d'exceller, c'est dans l'élaboration des diaporamas que l'on projette par exemple à l'occasion de cours ou de conférences et autres séminaires.

L'outil sur lequel je souhaite m'attarder se nomme *Beamer*. Il s'agit d'une classe de document, au même titre que « book » ou « article ». Cela signifie simplement que, dans le préambule, au lieu de mentionner :

```

----- Classe article -----
\documentclass[10pt,a4paper]{article}

```

ou bien :

```

----- Classe book -----
\documentclass[10pt,a4paper]{book}

```

il suffit de mentionner :

```

----- Classe Beamer -----
\documentclass[t,10pt]{beamer}

```

Puis, tout comme un article et un livre ne s'organisent pas de la même manière, au lieu des chapitres et des sections, on divise le document en autant de diapositives que l'on souhaite.

Là où l'élaboration d'un diaporama avec L^AT_EX prend tout son sens, c'est dans la rigueur de l'élaboration du diaporama avec une mise en page automatique : donner un titre à chaque diapositive, faire un sommaire (qui deviendra un menu), hiérarchiser les idées et utiliser les styles appropriés à l'exposition des éléments d'une diapositive... Tout cela entraîne l'utilisateur à respecter les aspects les plus importants d'un diaporama :

- être lisible par l'assistance ;
- ne pas surcharger les diapositives et aller à l'essentiel ;
- proposer une esthétique soignée, cohérente, et agréable à l'oeil.

J'entends déjà les critiques. Oui, tout ceci peut s'obtenir avec l'outil de présentation d'OpenOffice.org, pour ne pas citer un autre logiciel fort connu. Certes, en utilisant les formats et les styles, tout comme dans un logiciel de traitement de texte, il est fort probable qu'un diaporama puisse obéir à ces règles et se projeter de manière tout à fait honorable. Pourtant, toute personne ayant assisté à au moins un séminaire avec plusieurs intervenants sera d'accord avec moi : les personnes utilisant les formats et les styles sont très rares, les diapositives sont habituellement surchargées et illisibles (ce qui brouille le message de l'intervenant), l'esthétique est souvent plus que douteuse, les effets de transition tombent souvent à l'eau parce que la version du logiciel présent sur la machine est presque toujours différent de celle que l'intervenant a utilisé pour créer son diaporama, etc. En fait, pour utiliser les outils de formats et styles, prédéfinir le graphisme, et structurer le diaporama, il faut réfléchir et avoir une idée très précise de ce que l'on veut produire. Si bien que, nos cerveaux étant ce qu'ils sont – c'est-à-dire limités –, et que les logiciels de présentation permettent facilement de s'échapper de la rigueur des styles pour transformer les diapositives en autant de murs blancs sur lesquels on « tague » n'importe comment, le résultat manque à la fois de cohérence et d'esthétisme.

Paradoxalement, parce que L^AT_EX apporte un cadre très rigoureux, il est plus facile d'élaborer un bon diaporama qu'avec un logiciel permet-

tant une très (trop) grande liberté de mouvement. Bien entendu, il est possible de faire exactement ce que l'on veut avec L^AT_EX, mais les outils proposés par *Beamer* ont été spécifiquement pensés pour une communication d'ordre scientifique. Les logiciels de présentation, comme celui d'OpenOffice.org, sont pensés de manière à correspondre à une plus large palette de besoins, de la présentation publicitaire automatiquement projetée sur le mur d'un magasin, à la présentation de cours en passant par le dessin. Or, je ne le dirai jamais assez : un logiciel destiné à une tâche spécifique permettra d'accomplir cette dernière de manière plus efficace qu'un logiciel incluant cette tâche mais destiné à en remplir plusieurs. Avec la classe *Beamer*, vous réaliserez des diaporamas destinés à des communications scientifiques, et pas autre chose.

Pour terminer cette partie sur l'utilisation de *Beamer*, je vous invite à regarder la figure 5 qui montre un exemple de production de diapositive. *Beamer* possède une batterie d'outils préconfigurés qui assurent une présentation très esthétique. Il suffit de placer bonnes commandes d'*environnement* aux endroits souhaités (comme par exemple l'environnement « *definition* »).

Pour obtenir cette diapositive, il a donc fallu entrer le texte suivant (vous noterez que la signification des commandes est ici assez intuitive pour ne pas avoir à les commenter) :

```

----- Utilisation de Beamer -----
\begin{frame}{Partie I}
  \frametitle{Critique du jugement}
  \framesubtitle{Les bons conseils d'E. Kant}
  \begin{definition}
    Le défaut de jugement est proprement ce que l'on
    nomme stupidité,
    et c'est là un \underline{vice} auquel il n'y a
    \underline{pas de remède}.
    (E. Kant, \textit{CRP})
  \end{definition}

```

```

\begin{block}{Quelques nuances}
  \begin{itemize}
    \item On peut être instruit et manquer
    de jugement.
    \item Les hommes stupides et instruits
    font un mauvais usage de leur
    science\ldots
    \item \ldots qui s'étale comme
    de la confiture,
    \item Oh My God! que de poncifs !
  \end{itemize}
\end{block}

\end{frame}

```

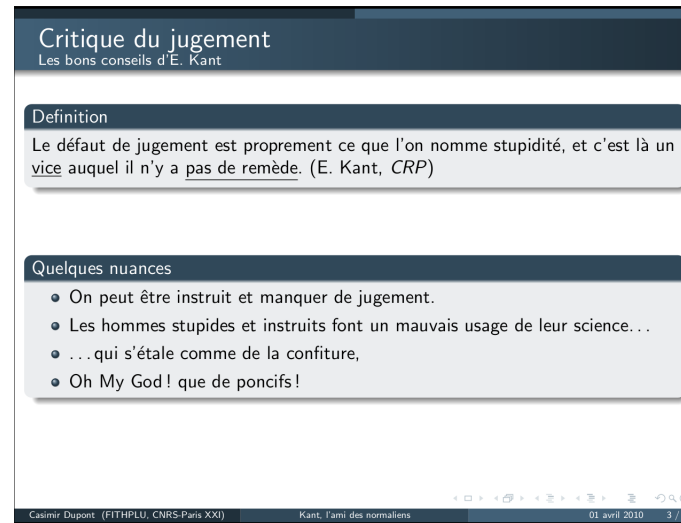


FIGURE 5 – Une diapositive obtenue avec *Beamer*.

7. Conclusion

Il y aurait tant à dire sur L^AT_EX ! Je ne suis pourtant qu'un simple utilisateur que l'on pourrait qualifier de débutant. J'aurais pu, par exemple, mentionner les outils de graphisme comme PSTricks³⁰, ou toute la batterie d'outils destinés à créer des documents pédagogiques et des manuels, ou encore m'étendre plus largement sur les paquets que j'utilise moi-même dans le cadre de mon travail. Mais l'objectif de ce document n'était pas de créer un énième manuel d'utilisation de L^AT_EX, d'autant plus que je ne maîtrise pas la moitié du quart du dixième des paquets j'ai mis dans le préambule de ce document, par exemple.

Je préfère donc vous confier ci-dessous une liste de sites sur lesquels vous trouverez de l'information pertinente. S'ensuivra une bibliographie indicative destinée aux débutants.

Liste de sites internet

Tug.org Le groupe d'utilisateur de T_EX fondé en 1980. Il s'agit du « canal historique » des premiers pionniers. L'association édite deux revues (*TuGboat* et *The PracTeX Journal*), encadre la distribution TeXLive, et reste très active dans le domaine, en organisant des conférences et en soutenant différents projets.
<http://tug.org/>.

GUTenberg Le Groupe francophone des utilisateurs de T_EX, L^AT_EX et logiciels compagnons est une association loi 1901 fondée en 1988. Elle publie régulièrement une Lettre et édite la revue *Les Cahiers GUTenberg*.
<http://www.gutenberg.eu.org>.

30. Voir <http://tug.org/PSTricks/>. Je conseille l'utilisation de LaTeX-Draw pour générer facilement du code PSTricks, même si une fois encore les puristes vont me tomber dessus : <http://latexdraw.sourceforge.net/fr/index.html>.

Tuteurs ENS La page L^AT_EX des tuteurs de l'ENS. Idéal pour commencer à se familiariser avec les bases de L^AT_EX.

<http://www.tuteurs.ens.fr/logiciels/latex/>.

Présentation rapide... Une présentation rapide de L^AT_EX2e, qui peut-être qualifiée de « gros » aide-mémoire portant sur les bases de L^AT_EX.

http://people.via.ecp.fr/~flo/2000/presentation_LaTeX/.

Latex-project.org Le site officiel du projet L^AT_EX.

<http://www.latex-project.org/>.

Références

- [1] MARC BAUDOIN, *Apprends L^AT_EX !*
Un manuel écrit pour les élèves de l'école nationale supérieure de techniques avancées (ENSTA ParisTech) mais qui se prête parfaitement à un premier apprentissage de L^AT_EX. Le manuel est disponible à l'adresse http://www.babafou.eu.org/Apprends_LaTeX/Apprends_LaTeX.pdf.
- [2] CÉLINE CHEVALIER & AL., *L^AT_EX pour l' impatient. Offrez une présentation irréprochable à vos documents*. Paris : H et K Editions. 2007.
- [3] LESLIE LAMPORT, *L^AT_EX, A Documentation Preparation System*. Reading (Mass) : Addison Wesley (Pearson-education). 1994.
- [4] VINCENT LOZANO, *Tout ce que vous avez toujours voulu savoir sur L^AT_EX sans jamais oser le demander*. Paris : In Libro Veritas – Framasoft, Collection Framabook. 2008.
Il s'agit du livre que je conseille à quiconque serait intéressé par un manuel pédagogique et non rébarbatif pour commencer avec L^AT_EX. En plus, c'est un livre libre que l'on peut se procurer ici : <http://www.framabook.org/latex.html>.
- [5] FRANK MITTELBACH, MICHEL GOOSSENS & AL., *L^AT_EX Companion (seconde édition)*. Paris : Pearson Education France. 2005.