

Traitements de Texte : Stupides et Inefficaces

par Allin COTTRELL
(cottrell@wfu.edu)

Traduction d'Arnaud ZUFFEREY (arnaud.zufferey@epfl.ch)
Révisé par Xavier PERSÉCUERS (xavier.perseguers@epfl.ch)

Table des matières

1. - L'affirmation	2
2. - Documents imprimés	2
a) Composition et typographie	2
b) Les pièges du WYSIWYG	4
c) Documenter la structure	4
d) Éditeurs de texte	5
e) Les vertus de l'ASCII	6
f) Le programme de composition typographique	6
g) En pratique	7
3. - Diffusion numérique	8
a) Documents simples	8
b) Documents complexes	9
4. - Précisions	9
5. - Grrr, grrr	10
6. - Références	11

1. - L'affirmation

Un programme de traitement de texte est un outil stupide et particulièrement inefficace pour préparer du texte destiné à la communication avec d'autres. C'est l'affirmation que je défendrai ci-dessous. Cela vous semblera probablement bizarre à première vue. Si je suis contre les programmes de traitement de texte, qu'est ce que je recommande donc à la place ? Que nous écrivions à la main, ou que nous utilisions une machine à écrire ? Non ! Bien qu'il y ait des choses à dire en faveur de ces modes de préparation de texte, je pars du principe que la plupart des lecteurs de cet essai feront la majeure partie de leurs écrits à l'aide d'un ordinateur, comme moi. Mon affirmation est qu'il y a de meilleurs moyens pour rédiger un texte à l'aide d'un ordinateur que les traitements de texte.

La formulation de ma thèse est volontairement provocatrice, mais laissez-moi préciser ceci : quand je dis que *les programmes de traitement de texte* sont stupides je ne dis pas que *vous*, si vous êtes un utilisateur de ces programmes, êtes stupide. Je critique une technologie qui est assidûment défendue par les principaux fournisseurs de logiciels, et qui est devenue un standard *de facto*. À moins que vous ne soyez trouvé au bon moment au bon endroit, vous ignorez probablement l'existence d'alternatives. Ces alternatives ne sont pas favorisées par les principaux fournisseurs, pour une simple raison : comme nous le verrons plus loin, elles sont disponibles gratuitement.

Partons du produit final. Les textes destinés à communiquer des idées et des informations à d'autres personnes sont distribués de deux manières principales :

1. en tant que document imprimé ;
2. par des moyens numériques : courrier électronique, pages Web, documents conçus pour être lus à l'écran.

La frontière entre ces deux catégories n'est pas toujours bien nette. Par exemple, un document qui est prévu pour être imprimé peut être distribué sous forme numérique, en supposant que le destinataire peut l'imprimer. Mais examinons ces deux catégories à tour de rôle.

2. - Documents imprimés

Vous voulez dactylographier un document avec votre ordinateur, et vous voulez obtenir un bon résultat sur votre imprimante. Naturellement, vous ne voulez pas que ceci se produise en temps réel (le texte s'imprimant au fur et à mesure que vous saisissez le texte). Vous voulez dactylographier le document d'abord puis le sauvegarder sur un disque. Vous voulez pouvoir ouvrir le document plus tard, le modifier, et l'imprimer seulement lorsque vous serez satisfait du résultat. Un traitement de texte tel que Microsoft Word est certainement la manière « naturelle » de faire tout ceci ? C'est un moyen, en effet, mais pas le meilleur. Pourquoi ?

a) *Composition et typographie*

En fait la préparation d'un document imprimable en utilisant un logiciel de traitement de texte vous force à combiner deux tâches qui sont *conceptuellement* distinctes et qui devraient être *pratiquement* séparées, afin de s'assurer que votre temps soit utilisé au mieux et que la communication finale soit la plus efficace. Ces deux tâches sont :

1. La composition du texte elle-même. Par ceci je veux dire le choix réel des mots pour exprimer ses idées, et structurer le texte de façon logique. Cette structure peut prendre diverses formes selon la nature du document : division du texte en paragraphes, sections ou chapitres, choix du texte qui apparaîtra dans le texte ou dans les notes de bas de page, mise en exergue de certaines parties de texte, citations, ...
2. La typographie du document. Elle comprend le choix des polices de caractères, et la manière dont les éléments de structure doivent être représentés. Est-ce que les titres doivent être en gras ou en petites majuscules ? Doivent-ils être alignés à droite ou centrés ? Le texte doit-il être justifié ou non ? Les notes doivent-elles apparaître en bas de page ou à la fin ? Est-ce que le texte doit apparaître en deux colonnes ou en une seule ? Etc.

L'auteur d'un texte devrait, du moins dans un premier temps, se concentrer uniquement sur le premier ensemble de tâches. C'est le rôle de l'auteur. L'illustre Adam SMITH¹ a montré les grands avantages qui découlent de la division du travail. La composition et la structuration logique du texte est la contribution spécifique de l'auteur à la production d'un texte imprimé. La typographie est l'affaire du typographe. Cette division du travail a naturellement été accomplie dans la production traditionnelle des livres et des articles avant l'arrivée de l'informatique. L'auteur écrivait et indiquait à l'éditeur la structure logique du texte au moyen de diverses annotations. Le typographe convertissait le texte de l'auteur en document imprimé, en représentant la conception logique de l'auteur par une typographie adéquate. Imaginez par exemple Jane AUSTEN² se demander dans quelle police mettre les titres de chapitres d'*Orgueil et préjugé* pour voir comment la notion est ridicule. Jane AUSTEN était un grand auteur ; elle n'était pas un typographe.

Vous pouvez penser que c'est hors de propos. Les textes de Jane AUSTEN étaient excellents et les typographes professionnels étaient donc intéressés à les éditer. Vous et moi ne sommes pas aussi chanceux ; si nous voulons un article imprimé nous devons le faire nous-mêmes (et, en outre, nous le voulons réalisé beaucoup plus rapidement que par l'intermédiaire de l'édition traditionnelle). Ce n'est qu'en partie vrai. Nous devons le faire nous-mêmes (sur nos propres ordinateurs), mais nous avons beaucoup d'outils à notre disposition. En particulier nous avons à notre disposition un programme de composition de qualité professionnelle. *Ce programme (ou ensemble de programmes) fera en effet pour nous, gratuitement et en quelques secondes ou fractions de seconde, le travail que les typographes traditionnels ont fait pour William SHAKESPEARE, Jane AUSTEN, Sir Walter SCOTT et tous les autres.* Nous devons juste fournir au programme un texte convenablement annoté, comme le font les auteurs traditionnels.

Je suggère donc qu'il devrait y avoir deux moments distincts dans la production d'un texte imprimé à l'aide d'un ordinateur. En premier lieu on dactylographie son texte et le structure au moyen d'annotations simples. Cette tâche est accomplie au moyen d'un *éditeur de texte*, un programme qu'il ne faut pas confondre avec un logiciel de traitement de texte (j'expliquerai la différence plus loin.) Dans la deuxième étape on « passe la main » à un programme de composition typographique qui compose le texte et retourne dans un temps très court le résultat final.

1. Adam SMITH (1723–1790), économiste, est le père du capitalisme.

2. Jane AUSTEN (1775–1817)

b) *Les pièges du WYSIWYG*

Ces deux tâches sont intimement liées dans les traitements de texte WYSIWYG³. Vous dactylographiez votre texte, et *pendant que vous saisissez le texte* il apparaît sur l'écran de l'ordinateur, dans une représentation typographique concrète qui correspond étroitement à ce que vous verrez quand vous enverrez le document à l'impression (bien que pour différentes raisons cela ne se passe pas toujours ainsi). En effet, le texte est composé à la volée pendant que vous le tapez.

À première vue ceci peut sembler être un grand avantage ; nous verrons lors d'une inspection plus minutieuse que c'est une malédiction, et ce pour trois raisons :

1. l'auteur est distrait du travail de composition par des choix typographiques pour lesquels il peut ne pas avoir de connaissances (« tripoter les marges et les polices » quand il devrait se concentrer sur le contenu) ;
2. l'algorithme de composition utilisé par le programme de traitement de texte WYSIWYG sacrifie la qualité au profit de la vitesse exigée par l'arrangement et le réajustement en temps réel des caractères entrés par l'utilisateur. Le résultat final est d'une qualité très inférieure à celle d'un vrai programme de composition. ;
3. l'utilisateur d'un logiciel de traitement de texte est soumis à une grande tentation de perdre de vue la structure logique du texte et de s'occuper de problèmes typographiques superficiels.

Les deux premiers points ci-dessus devraient être explicites. Laissez-moi développer le dernier point (son importance dépend du type de document considéré.)

c) *Documenter la structure*

Prenons par exemple un titre de section. En ce qui concerne la structure logique du document, tout ce qui importe c'est qu'un bout de texte soit « marqué » d'une certaine manière comme titre de section. On pourrait par exemple dactylographier `\section{Texte du titre}`. La représentation typographique correspondante dans le document imprimé est une question distincte. Cependant, quand vous utilisez un logiciel de traitement de texte ce que vous voyez est (seulement !) ce que vous obtenez. Vous êtes forcé de décider d'un aspect typographique spécifique pour votre titre au moment où vous le créez.

Supposons vous décidiez que vous voudriez vos titres en caractères gras, et légèrement plus grands que le reste du texte. Comment allez-vous réaliser cet aspect ? Il y a plus d'une façon de réaliser ceci, mais pour la plupart des personnes la manière la plus évidente et la plus intuitive (dans le contexte d'un logiciel WYSIWYG) est de dactylographier le texte du titre, le sélectionner, cliquer sur l'icône « caractère gras », dérouler la liste de tailles pour la police et de choisir une plus grande taille. Le titre est maintenant « gras » et grand.

Génial ! Mais qu'est-ce qui indique que c'est un titre ? Il n'y a rien dans votre document qui identifie logiquement ce petit bout de texte comme titre de section. Supposons que vous décidiez un peu plus tard qu'en fait vous préféreriez avoir les titres en petites majuscules, ou numérotés avec des chiffres romains, ou centré ou que sais-je. Ce que vous voudriez pouvoir faire est d'ordonner à votre logiciel de changer tous les titres de section selon vos nouvelles préférences. Mais si vous avez

3. "What You See Is What You Get", ce que vous voyez à l'écran est conforme à ce que vous verrez sur papier.

appliqué le formatage comme décrit ci-dessus, vous devrez passer en revue votre document entier et changer chaque titre manuellement.

Bien sûr *il y a* une manière d'indiquer la structure de morceaux de texte dans les logiciels de traitement de texte comme par exemple Microsoft Word. Vous *pouvez*, si vous faites attention, réaliser des effets tels que changer l'aspect de tous les titres de section avec une commande⁴. Mais peu d'utilisateurs de Word exploitent ceci uniformément, et cela n'est pas étonnant : l'approche WYSIWYG n'encourage pas le souci de la structure. Vous pouvez facilement – trop facilement – simuler la structure par des commandes de formatage de bas niveau. Par contre, en saisissant le texte avec un éditeur de texte, la nécessité d'indiquer la structure est immédiatement évidente.

d) Éditeurs de texte

Bien ! il est probablement temps d'expliquer ce qu'est un éditeur de texte, et en quoi il diffère d'une unité de traitement de texte. Un éditeur de texte moderne ressemble un peu à un programme de traitement de texte. Il comprend les fonctionnalités standards comme des menus déroulants, des icônes clicables pour l'ouverture ou la sauvegarde de fichiers, les fonctions de recherche ou remplacement de texte, correction orthographique, etc. Mais un tel programme n'a pas de fonctionnalité de composition typographique. Le texte que vous dactylographiez apparaît sur l'écran dans une représentation visuelle claire, mais sans la prétention de représenter l'aspect final du document imprimé.

Quand vous sauvez votre document, il est sauvé sous forme *de texte pur*, qui dans le contexte des USA signifie habituellement dans le format ASCII (American Standard Code for Information Interchange)

L'ASCII comporte 128 caractères (parfois désigné sous le nom de « jeu de caractères 7 bits » puisqu'il exige 7 éléments binaires pour son codage : $2^7 = 128$). Ce jeu de caractères inclut les numéros 0 à 9, l'alphabet romain en majuscule et minuscule, les signes de ponctuation standards, et un certain nombre de caractères spéciaux. L'ASCII est le plus petit dénominateur commun de la communication textuelle en forme numérique. Un message ASCII sera « compréhensible » par n'importe quel ordinateur dans le monde. Si vous envoyez un tel message, vous pouvez être sûr que le destinataire verra avec précision ce que vous avez dactylographié.

En revanche, quand vous sauvez un fichier d'un logiciel de traitement de texte, le fichier contient divers caractères de contrôle, non compris dans la norme ASCII. Ces caractères représentent le formatage que vous avez appliqué (par exemple des caractères en gras ou en italique) plus de diverses sortes de commandes internes spécifiques au traitement de texte. Ces commandes ne sont pas universellement « compréhensibles ». Pour les interpréter, vous avez besoin d'une copie du logiciel avec lequel le document a été créé (ou un filtre de conversion approprié). Si vous ouvrez un fichier créé avec un traitement de texte dans un éditeur de texte vous verrez (en plus du texte) beaucoup de drôles de symboles : c'est le code binaire de formatage.

Puisqu'un éditeur de texte n'insère aucun code binaire de formatage, si vous voulez représenter des formatages tels que l'italique, vous devez faire ceci par l'intermédiaire de commandes, ou *balises*. C'est-à-dire que vous saisissez une annotation (en n'employant que des caractères ASCII) qui indiquera au logiciel chargé de la composition typographique de mettre le texte indiqué en italique. \LaTeX est

4. Ceci passe par l'utilisation de styles prédéfinis.

un exemple de programme de composition typographique (plus de détails un peu plus loin). Pour lui indiquer un texte en italique, vous dactylographieriez `\textit{texte que vous voulez en italique}`. En fait, si vous employez un éditeur de texte qui est conçu pour coopérer avec \LaTeX vous ne devriez pas dactylographier ceci vous-même. Vous saisissez seulement un certain raccourci, ou une entrée d'un menu, ou cliquez sur une icône, et l'annotation appropriée est insérée pour vous ; les mécanismes pour dactylographier un document ASCII approprié au traitement par \LaTeX ne diffèrent pas beaucoup de l'utilisation d'un traitement de texte moderne.

e) *Les vertus de l'ASCII*

L'approche de saisir votre texte en ASCII pur en utilisant un éditeur de texte, puis en le composant avec un programme séparé, a plusieurs avantages « fortuits » :

1. Portabilité : comme mentionné ci-dessus, *quiconque*, en utilisant n'importe quelle plateforme informatique, pourra lire votre texte annoté, même s'ils n'ont pas les moyens de regarder ou imprimer la version composée. En revanche votre fichier de traitement de texte Snazz 9.0 peut être complètement incompréhensible pour un destinataire qui n'a pas la même marque et version du traitement de texte, à moins qu'il ne soit un expert en informatique et soit capable d'extraire le texte réel à partir du « fouillis » binaire. Et ceci s'applique aussi à *vous* dans le temps. Vous pourriez très bien avoir des problèmes à lire votre fichier Snazz 8.0 avec Snazz 9.0 ou inversement, mais vous n'aurez jamais d'ennuis pour lire votre vieux fichier ASCII.
2. Compacité : un fichier ASCII représente vos *idées*, et pas des commandes spécifiques à un traitement de texte. Pour de petits documents en particulier, les fichiers de traitements de texte peuvent être pas moins de 10 fois plus volumineux qu'un fichier ASCII correspondant contenant la même information.
3. Sécurité : l'approche « structure puis composition » vous garantit que vous n'aurez jamais de problème de corruption de vos documents (à moins que votre disque dur ne tombe accidentellement en panne, ou autres calamités dans le style). Le texte sera toujours lisible, même si le programme de composition échoue pour quelque raison. Si vous employez régulièrement un programme de traitement de texte, et que vous n'avez *jamais* eu de problème de fichier corrompu, vous êtes très chanceux.

Le lecteur intéressé pourra trouver davantage d'informations sur la page de Sam STEINGOLD intitulée “[No Proprietary Binary Data Formats](#)”.

f) *Le programme de composition typographique*

Il est temps que je vous donne un peu plus de détails sur le logiciel de composition typographique utilisé dans la stratégie que je préconise. Je n'entrerai pas dans les détails techniques ici, mais j'essaierai d'en dire assez pour vous donner une certaine idée de ce dont je parle.

Le programme de composition de base que j'ai à l'esprit s'appelle \TeX , et a été écrit par Donald KNUTH de l'université de Stanford. \TeX est disponible gratuitement pour la plupart des plateformes informatiques⁵. Knuth a commencé à développer \TeX en 1977. En 1990 il a annoncé qu'il cessait le développement, non par manque

5. \LaTeX est disponible au téléchargement sur de nombreux sites Internet. Vous pouvez aussi commander un CD-ROM complet pour quelques euros seulement.

d'intérêt, mais plutôt parce qu'à ce moment le programme avait atteint le stade de la perfection. Il permet de composer n'importe quel type de contenu, du simple texte aux formules mathématiques les plus complexes.

J'ai fait référence plus haut à \LaTeX . \TeX est le moteur de composition de base, et \LaTeX est un grand ensemble de macros, initialement développées par Leslie LAMPORT dans les années 80 et aujourd'hui maintenu par un groupe international d'experts. Ces macros facilitent beaucoup la vie à l'utilisateur moyen du système. \LaTeX est toujours en cours de développement, car de nouvelles possibilités et paquets sont ajoutés au moteur de base. Les divers modules pour \TeX sont également en cours de développement, comme par exemple un système qui vous permet de faire des documents PDF⁶ directement à partir de vos fichiers source ASCII (lorsque je dis « en cours de développement » je veux juste dire qu'ils sont améliorés continuellement. Les programmes sont déjà très complets et stables).

Comme mentionné ci-dessus, vous indiquez la structure et le formatage désirés de votre document à \LaTeX sous forme d'annotations. Il y a beaucoup de livres (et sites web) qui détaillent ces commandes, et je n'entrerai pas dans les détails ici. Les balises de base sont simples à retenir, et les éditeurs de texte compatibles avec \LaTeX rendent la tâche encore plus facile.

Un aspect très attrayant de \LaTeX est la capacité de changer l'apparence finale de votre texte rigoureusement et *uniformément* à l'aide de quelques commandes. L'aspect final est contrôlé par :

1. le type de document que vous choisissez (par exemple rapport, lettre, article, livre);
2. les modules optionnels que vous décidez de charger.

Vous pouvez, par exemple, complètement changer la famille de police (uniformément à travers le texte, les titres de section, les notes de bas de page et autre) et/ou les tailles des polices employées, en changeant juste un ou deux paramètres dans le « préambule » de votre fichier source ASCII. De même, vous pouvez répartir le texte sur une ou deux colonnes, ou passer du mode portrait au mode paysage. Il est possible d'accomplir quelque chose de semblable en utilisant un traitement de texte, mais la procédure est généralement moins commode, et le risque est plus grand d'introduire des inconsistances dans le formatage.

Vous pouvez composer des documents aussi complexes que vous le souhaitez avec \LaTeX . Vous pouvez choisir l'approche « par défaut » et préciser uniquement le type de document. Cette approche produit généralement de bons résultats, avec une qualité bien supérieure à celle obtenue avec un traitement de texte. (Naturellement la numérotation des titres, des notes de bas de page ou des références croisées sont générées automatiquement.) Vous pouvez aussi choisir une approche plus « interventionniste » et charger divers modules optionnels (ou en écrire vous-mêmes), pour contrôler divers aspects de formatage. Vous pouvez ainsi obtenir de superbes documents individuels.

g) *En pratique*

Laissez-moi vous donner juste une brève idée de la façon dont tout ceci fonctionne. Si vous avez un système avec \TeX installé : Vous saisissez votre texte dans un éditeur compatible \TeX . Vous pouvez dactylographier les annotations directement ou les faire insérer par l'intermédiaire des menus ou des boutons. Quand vous

6. Portable Document Format d'Adobe

atteignez un point où vous voudriez jeter un coup d'oeil à la version composée vous faites un choix d'une option de menu ou cliquez sur un bouton dans l'éditeur pour lancer le moteur de composition. Une autre option de menu ouvrira un aperçu dans lequel vous verrez le document final, tel qu'il apparaîtra à l'impression. L'aperçu est généralement un vrai WYSIWYG, il montrera une version très précise du résultat imprimé. Vous pouvez zoomer, passer à la page suivante, etc. Vous pouvez ensuite choisir d'imprimer le résultat ou retourner en mode édition.

Un peu plus tard dans le processus, vous voulez voir le résultat mis à jour. Il suffit d'appuyer à nouveau sur le bouton du programme de composition. Si l'aperçu est resté ouvert, le résultat est mis à jour. Quand vous avez terminé avec un article, vous pouvez supprimer la version composée et garder uniquement le fichier source ASCII pour économiser de l'espace disque. La version composée peut facilement être recréée lorsque vous avez besoin d'elle.

3. - Diffusion numérique

La section précédente était orientée vers la production de documents destinés à l'impression. D'autres facteurs sont à prendre en compte lorsque vous créez un document avec l'intention de le distribuer de façon numérique (email, pages web, etc).

Commençons par l'email. Typiquement si les gens souhaitent envoyer un court message ils dactylographient ce message directement dans un programme d'email, comme Pine, Netscape ou Eudora. Le message sera envoyé en texte pur ou en HTML⁷. Mais que faire si voulez envoyer un plus long morceau de texte que vous avez déjà rédigé indépendamment de votre programme de client d'email ?

Pour cela il est de plus en plus courant « d'attacher » un document issu d'un traitement de texte. Comment fonctionne cette stratégie alternative dans ce cas ?

Bien, nous devons distinguer deux situations : le texte en question est-il relativement court et peu compliqué (une note, une lettre, compte rendu d'une réunion, une liste d'ordre du jour, un programme pour une réunion) ou est-il plus complexe (un article scientifique contenant peut-être beaucoup de formules mathématiques, un rapport avec des illustrations, un manuscrit de livre) ? L'approche « document ASCII plus composition typographique » conduit à différentes suggestions dans les deux cas.

a) *Documents simples*

Pour des documents simples, nous devons nous poser la question : avons-nous vraiment besoin de la composition, de polices spéciales et tout le reste ? N'est-il pas plus approprié, du point de vue d'une communication économique et efficace, d'envoyer uniquement le texte pur en ASCII, avec le formatage minimal permis par ce dernier ? Ceci préserve la bande passante des communications (rappelez-vous qu'un fichier de traitement de texte peut être *beaucoup* plus volumineux qu'un fichier ASCII contenant le même texte) et garantit que personne ne sera exclu de la discussion parce qu'il s'avère qu'il ne possède pas Snazz 9.0. Vous pouvez attacher un fichier ASCII, créé dans un éditeur de texte, de la même manière que vous attacheriez un fichier de traitement de texte, ou vous pouvez simplement le coller dans le corps de l'email (puisque ce n'est rien d'autre que du texte pur). Puisque les fichiers source de

7. Hyper Text Markup Language, langage utilisé pour créer les pages web, composé principalement d'ASCII.

T_EX ne sont rien de plus que du texte ASCII – et puisque nous parlons de documents simples qui ne contiennent que peu d’annotations facilement compréhensibles – ils peuvent être traités de la même manière.

b) *Documents complexes*

Des documents plus longs et plus complexes peuvent être bien plus lisibles sous leur forme composée. Il est bien plus difficile d’exprimer des formules mathématiques en ASCII et de toute façon les diagrammes et les images sont exclus d’office. Alors que dire de T_EX dans ce contexte ? J’ai affirmé que les fichiers de traitement de texte peuvent être problématiques, parce que votre correspondant pourrait ne pas avoir Snazz 9.0 comme vous. Mais cela n’exclut-il pas les deux méthodes ? Même si vous êtes suffisamment motivé pour essayer T_EX, combien de vos correspondants disposent de T_EX sur leur machine ? C’est une bonne question. Si vous voulez que votre correspondant puisse lire une version composée de votre document, et qu’il ne dispose pas de T_EX, vous avez plusieurs possibilités :

1. Convertir le fichier source de T_EX en HTML. Il y a à cet effet de bons programmes de conversion. Le HTML et le code T_EX ont une forte ressemblance du fait qu’ils impliquent tous les deux un marquage au moyen de balises de texte, et donc la conversion entre les deux formats peut être accomplie avec un haut degré de fiabilité.⁸ Votre correspondant peut ensuite lire votre texte en utilisant un navigateur Internet.
2. Votre correspondant a-t-il accès à une imprimante PostScript ? Ceci est souvent le cas dans un environnement académique ou professionnel. Dans ce cas vous pourriez envoyer une version entièrement composée de votre document sous forme de fichier PostScript, ce dernier pouvant être imprimé directement. Ces fichiers peuvent être lus à l’écran en utilisant GhostView, un programme de visualisation disponible gratuitement sur Internet.
3. Votre correspondant a-t-il le programme gratuit Adobe Acrobat Reader qui permet de lire les fichiers PDF ? Si oui, vous pouvez envoyer une version PDF de votre document composé.

En discutant des moyens de transmission de document par email, nous avons déjà abordé le problème de la préparation de texte pour des pages web. Vous avez la possibilité d’écrire directement du HTML. Si vous ne voulez pas le faire, vous pouvez écrire du HTML indirectement en utilisant un éditeur avec interface graphique comme Mozilla Composer par exemple. Bien sûr vous pouvez aussi produire du HTML avec MS Word (cependant le code produit est horrible, illisible, plein de balises inutiles qui rendent le texte fastidieux à modifier à posteriori avec un autre éditeur). Si vous travaillez avec T_EX il est facile de convertir vos documents en HTML (le code généré est propre et conforme aux standards).

4. - Précisions

J’ai essayé de vous montrer la supériorité de l’approche « ASCII plus composition » par rapport aux logiciels de traitement de texte. J’admets cependant qu’il y a *quelques* types de documents pour lesquels un traitement de texte WYSIWYG

8. L’encodage binaire utilisé par les traitements de texte sont bien différents, et donc la conversion entre T_EX et les formats de traitements de texte n’est pas facile. De plus comme T_EX est un langage de plus haut niveau, il est en principe impossible de convertir un document T_EX en Word par exemple, sans perte d’informations.

est en fait un outil naturel. Je pense en particulier à des documents courts, avec un formatage important : papillons, posters, lettres d'invitation et autres documents similaires. Vous pourriez les réaliser avec T_EX, mais cela ne serait pas très efficace. Les classes de document standard de L^AT_EX (rapport, article, ...) ne sont pas faites pour ça. L^AT_EX choisit automatiquement les polices de caractère appropriées pour la composition de documents formels, mais il n'est pas conçu pour gérer un mélange de polices fantaisistes dont vous pourriez avoir besoin dans une production occasionnelle. La structure logique du document n'entre pas en ligne de compte ici : vous vous intéressez dans ce cas uniquement au formatage « brut ». Vous voulez savoir, par exemple, si vous mettez *cette* ligne dans une police de taille 36 ; si oui ou non la dernière ligne de la page va être déplacée sur la page suivante. Dans ce cas, le WYSIWYG est votre homme.⁹

Si la majeure partie de votre travail de traitement de texte est de ce type, vous avez probablement cessé de lire il y a longtemps. Si la majeure partie de votre travail de préparation de textes implique la production des documents relativement formels, cette section n'affecte pas les fondements de ma démonstration.

5. - Grrr, grrr

Vous avez certainement remarqué que je suis un peu fanatique sur ce sujet. Oui je le suis ! Le problème est qu'il ne s'agit pas que d'un débat académique sur les différentes façon d'écrire du texte. C'est un combat dans lequel les éditeurs de logiciels investissent de gros moyens. Pour être francs, nous considérons une situation dans laquelle MS Word est devenu pour la majorité *la* norme pour la préparation de documents à l'aide d'un ordinateur. Cependant Word est une norme qui a peu en sa faveur d'autre que c'est une norme (ou du moins aspire à l'être).

C'est un peu comme QWERTY. Connaissez-vous cette histoire ? Pourquoi est-ce que la disposition des touches sur les machines à écrire (et sur les ordinateurs par extension) a les lettres QWERTYUIOP sur la rangée supérieure ? Ce n'était pas la disposition initiale. Il a été conçu dans le but de *ralentir les dactylos*. Le problème était que l'expertise des premières dactylos a rapidement dépassé les possibilités des machines à écrire mécaniques : une dactylo rapide pouvait bloquer les touches, en tapant plus rapidement qu'elles ne pouvaient revenir à leur position initiale après avoir percuté le ruban encreur. La disposition QWERTY répartit les touches de telle manière qu'on ne puisse plus aller aussi vite. C'est clair que cette disposition est absurde pour les claviers d'ordinateurs, mais c'est trop tard : QWERTY est un standard, et toutes les tentatives de nouvelle disposition plus rationnelles des touches, par exemple le DVORAK, ont échoué.

De même, j'affirme que MS Word n'a aucun *droit* d'être une norme pour la préparation de documents, puisqu'elle est clairement moins efficace (pour la plupart des objectifs) et que des solutions de rechange sont disponibles. J'espère qu'il n'est pas trop tard dans ce cas-ci, et qu'il est encore possible de dire Non à Word. En fait, dans un certain sens, Word est pire que QWERTY : ce n'est pas une norme mais plutôt un tracteur qui traîne derrière lui le monopole de Windows (une autre question que je n'aborderai pas ici).

Le « standard » de Microsoft pour la représentation binaire du formatage des documents est quelque chose qui est sujet aux caprices de la Microsoft Corporation.

9. Note du traducteur : dans ce cas j'utilise le traitement de texte OpenOffice.org, qui est similaire à Word, mais libre et gratuit et qui permet de produire des documents PDF.

Et tant qu'ils ne sont pas mis sous pression par des concurrents commerciaux, ils n'ont aucun intérêt à établir une norme à long terme. Au contraire, ils ont un vif intérêt à vous forcer à mettre à jour votre version de Word à intervalles réguliers. Oh votre Word N.0 ne peut pas lire le fichier Word N+1 que votre collègue vient de vous envoyer ? Bien, vous feriez mieux de mettre votre logiciel à jour, n'est-ce pas ? Même si il n'y a pas de fonctionnalités dans la version N+1 qui vous sont utiles.¹⁰

6. - Références

Si vous m'avez lu jusqu'ici, vous pouvez être intéressés par de bons éditeurs de texte, par le système de composition de \TeX et ainsi de suite.

Le meilleur endroit pour trouver de l'information au sujet de \TeX est la [page des utilisateurs de \$\text{\TeX}\$](#) .¹¹ Vous y trouverez tous les liens dont vous avez besoin. Un autre site important est le [CTAN](#) depuis lequel vous pourrez télécharger \TeX pour la plupart des plateformes (inclu le moteur de composition, de nombreuses macros, un logiciel de visualisation et des logiciels pour convertir vos documents en fichiers imprimables).

Les distributions gratuites de \TeX n'incluent généralement pas d'éditeur de texte ce qui est pourtant nécessaire (à moins que vous ne disposiez déjà d'un éditeur de texte qui vous convient). Il existe de nombreux éditeurs, mais mon favori pour travailler avec des fichiers \TeX est Emacs avec les modules AUCTex et RefTeX. Ces derniers rendent Emacs très convivial pour taper du code \TeX : ils colorent les balises de texte vous permettant de détecter d'éventuelles erreurs immédiatement, et offrent des commandes liées à \TeX dans des menus pratiques.

Si vous êtes intéressés il y a une [capture d'écran](#) sur mon site web qui montre Emacs avec un document \TeX ouvert (40 Ko).

10. En ce qui me concerne, j'ai utilisé Word pendant plusieurs années avant de passer à \TeX . Je n'ai noté aucune évolution utile depuis Word 2.0 en 1990 en ce qui concerne la composition de documents.

11. Note du traducteur : un bon site pour les francophones est « [L \$\text{\TeX}\$ à portée de clic](#) ».