

Dans la série
LES TUTORIELS LIBRES
présentés par le site FRAMASOFT

Tutorial OpenOffice.org Writer

Yves Combes

Logiciel : OpenOffice.org (Writer)
site officiel : <http://www.openoffice.org>

Niveau : Débutant

Auteur : Yves Combes
site de l'auteur : <http://fynl.free.fr>

date : 09 2002

Licence : licence libre GNU/FDL

FRAMASOFT

« Partir de Windows pour découvrir le libre... »

www.framasoft.net

Tutorial OpenOffice.org Writer

Table des matières

1	Licence de ce document.....	1
2	Présentation rapide.....	1
3	Installation.....	2
4	Démarrage Rapide.....	2
4.1	Avertissement : votre souris a au moins 2 boutons.....	3
4.2	Une Lettre simple.....	3
4.3	La mise en forme simple.....	4
4.4	Pour les caractères.....	5
4.5	Pour les paragraphes.....	5
4.6	Insérer images (ou autres...).....	5
4.7	Les tabulations.....	5
4.8	Les puces et les numéros.....	7
5	Les feuilles de style se ramassent à la pelle... ..	8
5.1	Utilité des styles.....	8
5.1.a	Style de paragraphe.....	8
5.1.b	Autres styles.....	8
5.1.c	Appliquer un style.....	9
6	Un petit exemple avec les styles.....	9
6.1	Indiquer la structure du texte.....	12
6.2	Numérotation des sections.....	13
6.3	Table des matières automatique.....	14
6.4	Modification d'un style.....	15
6.5	Mettre des puces (ou des numéros) avec le style de numérotation.....	17
7	En guise de conclusion.....	18

1 Licence de ce document

L'auteur de ce document est Yves Combe, yvcombe@free.fr.

Ce document est sous licence Gnu F.D.L. (Gnu Free Document Licence ou Licence de Documentation Libre Gnu). Cela signifie que vous êtes libre de le reproduire, le recopier, le réutiliser, le modifier et le distribuer à condition de lui attribuer les mêmes libertés de copie, d'utilisation etc... De plus, il n'est pas nécessaire d'avertir l'auteur que vous réutilisez son oeuvre (même si cela lui fera toujours plaisir de le savoir). En revanche, il vous est demandé de citer l'auteur à l'origine du document.

Cette licence a pour but de protéger le présent ouvrage ainsi que les libertés qui l'accompagnent. Vous ne pourrez donc pas le modifier, le distribuer et dire : « *Vous n'avez pas le droit de redistribuer ce document .»* parce que vous lui aurez supprimé une liberté.

2 Présentation rapide

OpenOffice.org Writer est la partie *traitement de texte* d'**OpenOffice.org**, la suite bureautique libre basée sur StarOffice, celle-ci n'étant pas libre et restant propriété de Sun Inc.

En fait OpenOffice.org 1.0 correspond presque complètement à StarOffice 6.0, et toute documentation pour l'une peut servir à l'autre.

Les différences, liées à des parties non propriétés de Sun Inc., sont la correction orthographique, ...à compléter.

Ce document est un bref manuel pour une utilisation courante. Je n'ai pas l'intention d'aborder les fonctionnalités avancée, ni même toutes les fonctionnalités courantes. Il doit s'y trouver juste le minimum pour taper un mémoire de 30 pages...

3 Installation

Vous trouverez sur <http://lang.openoffice.org/fr/> , un lien vers la dernière version en français, sous forme d'une archive zippée (i.e. compressée au format zip).

Il vous faut un logiciel spécial pour décompresser cette archive. Si vous n'en disposez pas un logiciel appelé 7zip, également libre, se trouve à partir de l'adresse <http://adressede7zip>. Installez-le!

Une fois décompressée (un clic droit i.e. avec le bouton droit de la souris), descendez les répertoires et trouvez le programme **Steup**. Un double-clic dessus va installer OpenOffice.org.

4 Démarrage Rapide

Voici à quoi peut ressembler OpenOffice.org Writer lors de son lancement.

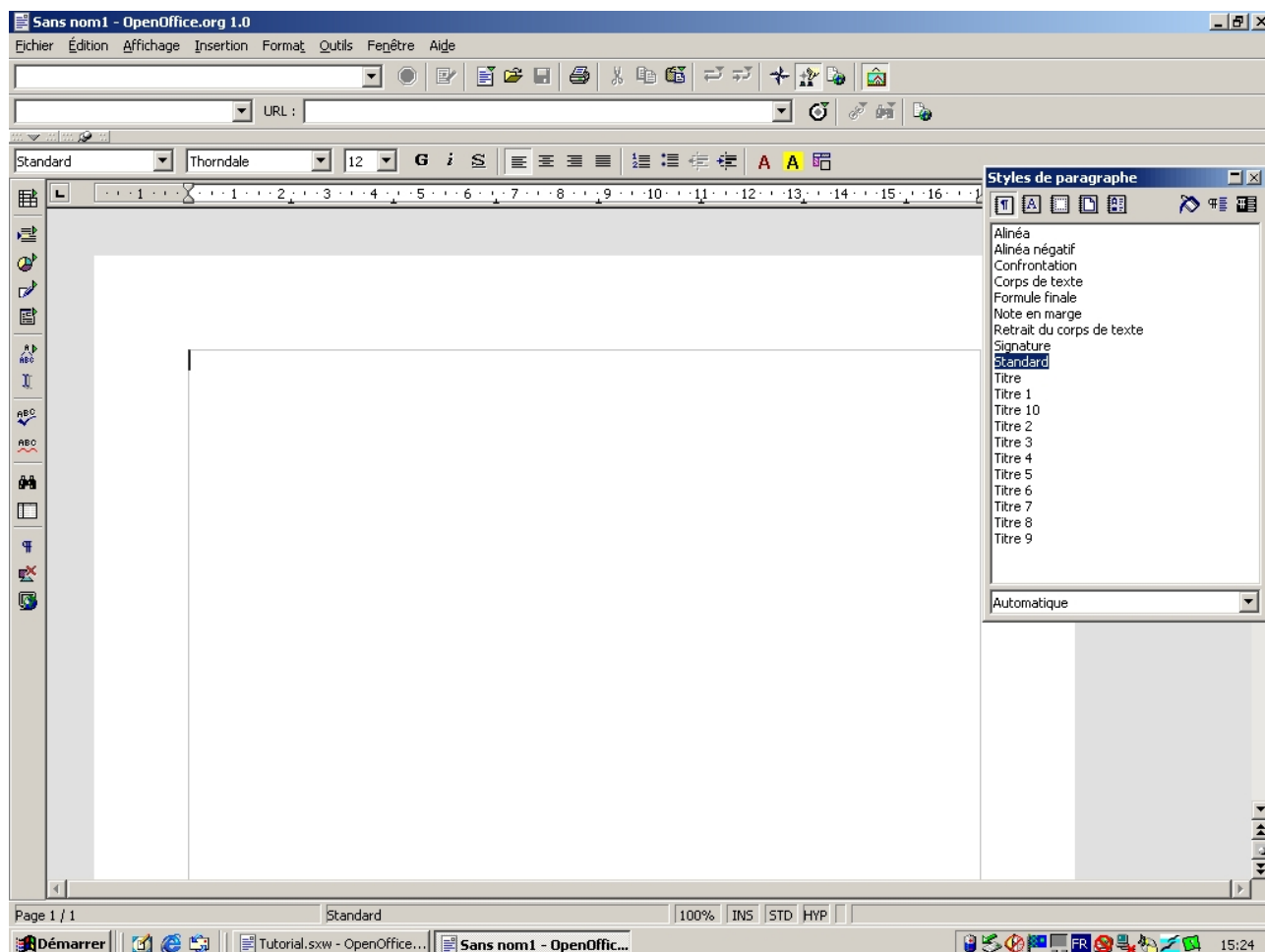


Illustration 1 OpenOffice.org à l'ouverture

Commençons par fermer la petite fenêtre du *styliste* (F11 ou la petite croix en haut à droite). Elle sert tout le temps quand on a compris les principes de base, on va donc commencer par les principes.

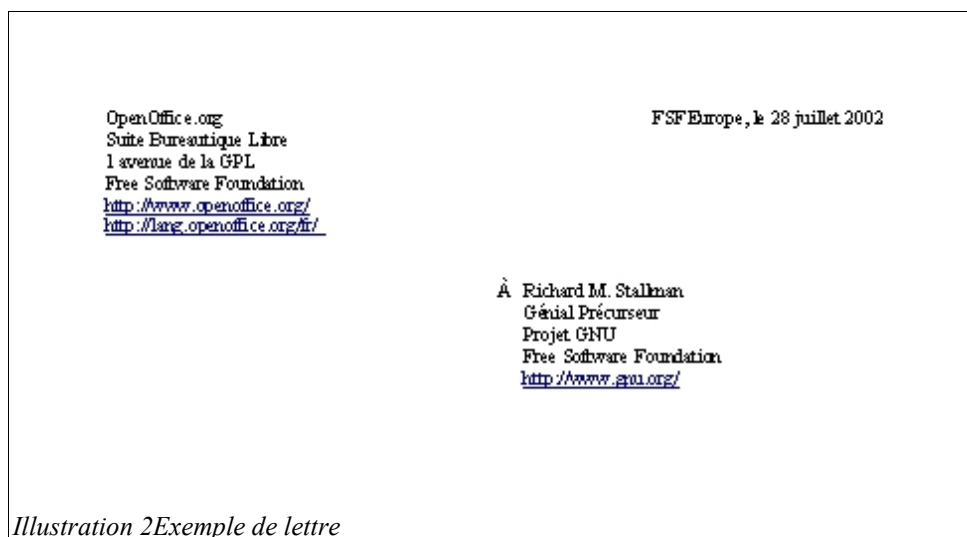
4.1 Avertissement : votre souris a au moins 2 boutons.

En règle générale , les débutants ignorent le bouton droit. Pourtant celui-ci fait en général apparaître un **menu contextuel**, qui contient tout ce qu'on peut faire avec l'objet sur lequel on a cliqué !

Sur **Mac**, en général la souris d'origine n'a qu'un bouton. Appuyez sur CTRL en cliquant sur un objet, vous obtiendrez le même effet.

Si vous cherchez comment faire quelque chose, cliquez sur l'objet concerné avec ce bouton droit. Il y a une très forte chance que la solution soit dans ce menu contextuel.

4.2 Une Lettre simple



La mise en page d'une lettre peut paraître un exemple sans intérêt. Toutefois la mise en page est suffisamment riche pour aborder la plupart des principes d'utilisation du traitement de texte.

Il fut une époque où la seule façon de réaliser ce genre de travail était d'utiliser les *tabulations*. Comme leur nom l'indique cela permet une mise en colonne propre comme dans un tableau.

Voici une méthode générale simple & efficace pour remplacer les tabulations: mettez un tableau sans bordure. Je vous laisse chercher l'icône , vous pouvez toujours passer par les menus : Insertion □ Tableau.

Faites un tableau de 2 lignes et 3 colonnes comme suit, **sans bordure**. Voici le résultat avec une bordure pour être parlant:

OpenOffice.org Suite Bureautique Libre 1 avenue de la GPL Free Software Foundation http://www.openoffice.org/ http://lang.openoffice.org/fr/	FSF Europe, le 17 août 2002
	À M. Richard M. Stallman Génial Précurseur Projet GNU Free Software Foundation http://www.gnu.org/

La seule difficulté consiste à redimensionner les colonnes (avec la souris et faisant *glisser* le séparateur) et à mettre la date à droite (Format□Paragraphe, ou les boutons). Un clic droit sur la Barre d'objets, ou un clic sur le bouton à droite avec le triangle, permet de remettre les icônes de mise en forme du texte.

Dernière indication, la date est obtenue par l'insertion d'un *champ* (Insertion□Champ) qui automatise sa saisie. Un double clic sur ce champ permet de choisir sa mise en forme : 17/08/02 ou samedi 17 août 2002 par exemple.

Finissez donc cette lettre et cherchons à la visualiser avant de l'imprimer. Ne cherchez pas l'icône, elle n'y est pas; passez par les menus (Fichier□Aperçu avant impression).

Si cette icône vous manque vous pouvez la remettre avec un clic droit sur la barre de boutons, puis Personnaliser, et dans la fenêtre cliquez sur affichage. Le bouton est en forme de microscope, cliquez dessus et faites-le glisser jusqu'à l'endroit où il doit être.

4.3 La mise en forme simple

C'est la mise en forme avec les boutons que vous avez sous les yeux.



Illustration 3 Des boutons, des boutons, toujours des boutons...

Quelqu'un qui débiterait avec un tel traitement de texte (ce n'est sûrement pas votre cas! Si?) aurait peut-être un peu de mal avec la notion essentielle suivante: certaines fonctions agissent sur le paragraphe courant entier, d'autre sur le texte sélectionné ou en cours de frappe, i.e. les caractères.

Ensuite les barres d'outils, contenant les boutons, sont aujourd'hui tellement riches qu'il n'est plus vraiment possible de connaître par cœur les fonctions de tous ceux qui sont affichés. Rappelons-nous simplement que laisser la souris quelques instant sur un bouton fait apparaître un texte qui indique sa fonction.

4.4 Pour les caractères

Ce sont les fonctions de mises en formes suivantes :

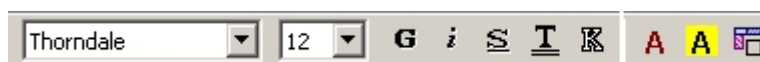


Illustration 4

gras, *italique*, souligné, contours, souligné double, mais aussi choix de la police, taille, couleur & couleur de fond :

Vous avez l'impression qu'il vous manque certains boutons ? C'est possible, puisque j'en ai rajouté avant de prendre la photo. Et vous croyez que toutes les fonctions pour le formatage des caractères sont là ? Perdu ! Il faut quand même prendre la peine d'aller voir dans le menu **Format** > **Caractères** pour se rendre compte de l'étendue des possibilités.

4.5 Pour les paragraphes

Bien sûr il y a les sempiternels alignements gauches, centres, droits, justifiés (cf. Illustration 5). Mais dans cette catégorie on trouve aussi les lettrines, tabulations, espaces avant & après, interlignes, numérotation.

En fait le paragraphe est l'entité typographique de base. On joue sur la différenciation des caractères à l'intérieur d'un paragraphe seulement lorsque l'on veut **mettre en valeur** un mot ou une phrase (citation courte par exemple).



Illustration 5

4.6 Insérer images (ou autres...)

Comme toujours, il y a le menu ou les boutons. Une fois votre image choisie et insérée, **et tant que** les poignées vertes sont toujours présentes autour de l'image, cliquez *avec le bouton droit*:

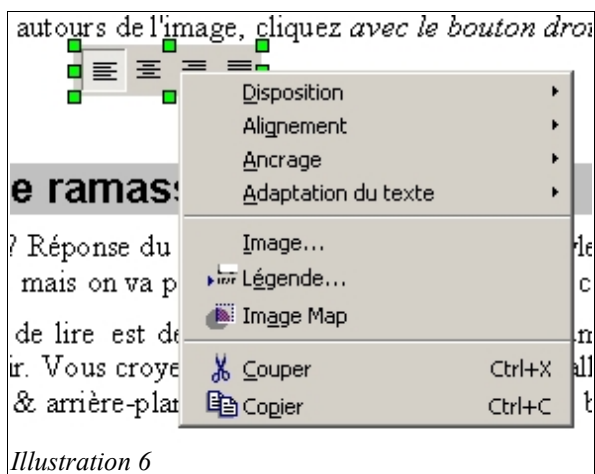


Illustration 6

Vous obtenez un **menu contextuel**, qui a la bonne idée d'afficher ce qu'on peut faire comme travail au niveau de l'alignement, du placement, etc... Si vous voulez le texte autour de l'image, c'est par là que ça se passe. Si vous voulez une légende aussi.

Il n'y a qu'à essayer les différentes possibilités. Si vous envisager de faire de votre texte une **page web**, alors après avoir mis éventuellement une légende, choisissez un **ancrage comme caractère**.

4.7 Les tabulations

J'ai indiqué dans l'exemple de la lettre qu'on pouvait remplacer les tabulations par un tableau sans bordure. Cela peut toutefois être plus rapide d'utiliser des tabulations, et celles-ci permettent d'utiliser des **points de conduite** pour remplir l'espace vide. Ce sont les deux raisons pour lesquels je vais expliquer comment elles marchent:

Jean	165	165,80	hier
Paul	134	67,09	demain
Pierre	34	4567,89	hier

Illustration 7

On peut le juger illisible, voici quelques points de conduite pour faciliter la lecture:

Jean	_____ 165 165,80	hier
Paul	_____ 134 67,09	demain
Pierre	_____ 34 4567,89	hier

Illustration 8

Ce tableau utilise une tabulation à gauche, une centrée décimale (i.e. sur la virgule), et une à droite.

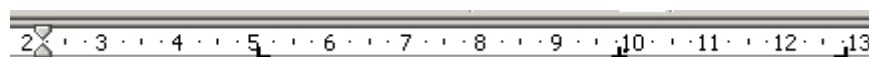


Illustration 9

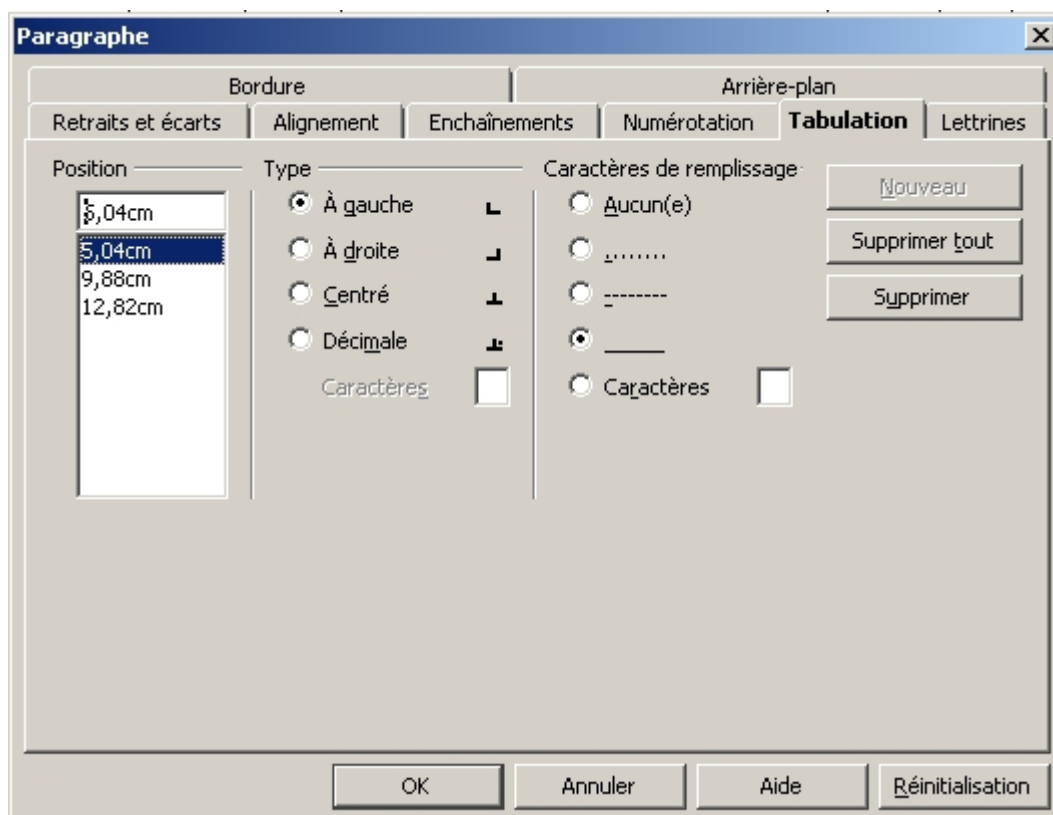


Illustration 10

On pose les tabulations à la souris, un clic sur le bouton à gauche de la règle permet de choisir son

type, un clic sur la règle la pose (on peut la faire glisser). On peut aussi utiliser **Format**▢**Paragraphe**▢**Tabulations** pour définir les tabulations précisément, et en profiter pour mettre des points de remplissage (points de conduite: Illustration 10).

4.8 Les puces et les numéros

Ici je ne parle pas de la numérotation des section ou des chapitres, mais de faire des listes numérotées ou avec des puces.

Exemple:

- | | |
|------------------------|-------------------------|
| - Premier élément | 1. Premier élément |
| - Deuxième élément | a. Deuxième élément |
| • Premier sous-élément | b. Premier sous-élément |
| • Deuxième sou-élément | 2. Deuxième sou-élément |
| - Troisième élément | 3. Troisième élément |

Pour cela c'est les boutons suivant qui servent.



Pour les « sous-élément » appuyez sur **TAB** au début de la ligne.

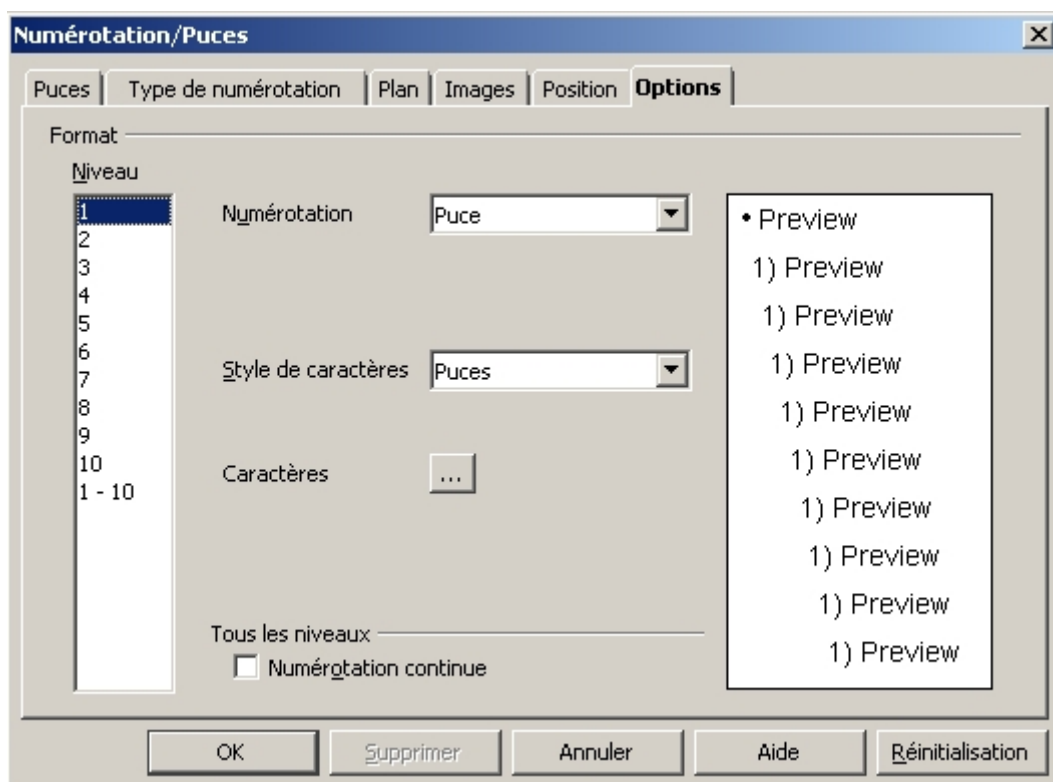


Illustration 11

Mais pour modifier les puces comme je l'ai fait, pour modifier les retraits, le type de numérotation, il faut passer par **Format**▢**Numérotation/puces**.

- Pour modifier les puces, cliquez sur l'onglet *puce*, sélectionnez une des puces, puis sur option vous choisirez niveau par niveau le caractère qui vous convient (avec le bouton marqué *Caractères*).
- Pour modifier les numéros, cliquez sur l'onglet *Type de numérotation*, sélectionnez une des

numérotation, puis sur option vous choisirez niveau par niveau le type qui vous convient (avec le menu marqué *Numérotation*).

- En choisissant ensuite l'onglet *Position* vous pourrez mettre un retrait personnalisé pour chaque niveau. Personnellement avec le niveau *1-10* (i.e. pour tous) je mets un retrait de 0,5 cm.

5 Les feuilles de style se ramassent à la pelle...

Qu'est-ce donc qu'une feuille de style ? Réponse du matheux: c'est l'ensemble des styles regroupés dans un même fichier. Ouais... Mais on va pas aller loin si on n'a pas idée de ce qu'est un style.

Le document que vous êtes en train de lire est découpé en sections, qui sont numérotées, en gros caractères, et qui sont sur fond gris clair. Vous croyez que j'ai fait ça à la main? En allant à chaque fois dans les menus *modifier police, taille & arrière-plan* ? Vous me flattez, mais je suis bien plus feignant que ça... Cela dit personne ne vous interdit de le faire, mais en cas d'ajout, de modification, il faut veiller à tout modifier. Si vous voulez faire une table des matières il faudra aussi la créer et la modifier à la main. Bon courage!

5.1 Utilité des styles

5.1.a Style de paragraphe

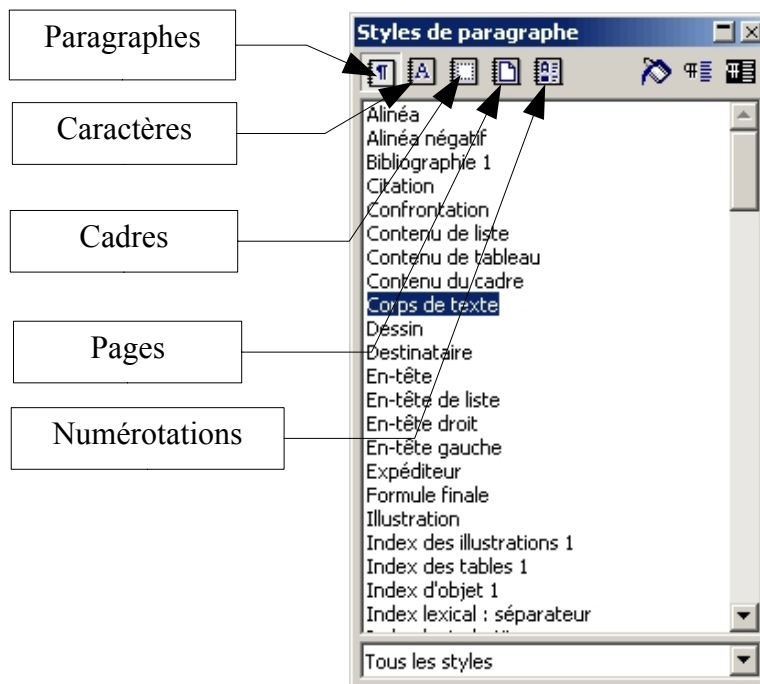
Les titres de sections de ce documents sont faites avec le style **Titre 1**. Pourquoi celui-là ? Parce qu'il est prédéfini comme le titre de niveau 1 dans la numérotation des chapitres. De même **Titre 2** est prédéfini comme titre de niveau 2, et ainsi de suite... Ainsi lorsque l'on demande la numérotation des chapitres, et la table des matières automatique, les sections, sous-sections, sous-sous-sections, ..., sont directement connues avec leur hiérarchie. C'est très pratique.

Pour appliquer un style, le mieux est d'utiliser la fenêtre du **styliste** : elle est là pour ça ! Appuyez sur la touche F11 pour la faire apparaître (vous avez noté que cette même touche permet de la faire disparaître si elle vous gêne ? Pratique, non ?).

5.1.b Autres styles

En plus des styles de paragraphes on peut définir des styles de caractères, pour mettre en gras souligné Times New Roman 13 pt petites capitales d'un seul clic, par exemple, ou même des styles de page, pour avoir des pages non numérotées avec la table des matières, par exemple.

5.1.c Appliquer un style



Dessin 1

Cliquez sur le bouton **Paragraphes** du styliste, puis sélectionnez **Titre 1**. Votre paragraphe en cours devient plus gros! C'est la modification la plus visible, ça marche. On verra comment mettre un fond gris automatique et la numérotation plus tard.

6 Un petit exemple avec les styles

Voici un texte à taper et à structurer :

Logiciels libres, une introduction

Qu'est ce qu'un logiciel ?

Pour comprendre le concept de logiciel libre, vous devez d'abord comprendre ce qu'est un logiciel (ou encore programme ou application).

Du point de vue de l'utilisateur, un logiciel est une application qui répond à l'un de ses besoins (traitement de textes, programme de dessin, jeu, ...). C'est une suite de petites instructions invisibles pour l'utilisateur, qui forment un tout cohérent.

Ces logiciels ont besoin d'un système d'exploitation pour fonctionner. Le système d'exploitation permet d'accéder aux ressources de la machine (lecteur de disquette, écran, clavier, ...). C'est aussi le système d'exploitation qui se charge d'exécuter le logiciel.

Les instructions d'un logiciel sont écrites dans un langage que l'ordinateur peut comprendre, le langage machine (ou langage binaire). En revanche, il est très difficile (voir impossible) à un humain de comprendre ce langage (c'est de plus interdit dans de nombreux pays). La manière la plus courante de créer un logiciel est de l'écrire dans un langage informatique compréhensible par des humains, et ensuite de le traduire vers le langage binaire. Cette traduction est effectuée par un logiciel appelé compilateur.

Le logiciel dans sa forme compréhensible est appelé code source, et dans sa version en langage machine binaire.

Qu'est-ce qu'un logiciel libre ?

Imaginez que vous vous trouviez dans un restaurant et que vous mangiez un excellent plat. Peut-être aurez vous l'envie de le cuisiner chez vous pour vos amis ?

C'est impossible, car vous n'avez pas la recette du plat. Vous pouvez toujours le manger dans le restaurant, mais même si vous connaissez le goût, vous ne savez comment le reproduire.

En informatique, c'est la même chose avec un logiciel. Le code source est la recette, le binaire est le plat cuisiné. La plupart des logiciels sont distribués sans leur code source, et il est interdit d'essayer de comprendre leur fonctionnement. Il est interdit de les partager avec vos amis, et il est interdit d'essayer de les modifier pour les adapter à vos besoins.

Un logiciel libre vous garantit plusieurs libertés :

la liberté de copier le logiciel pour vous ou vos amis

la liberté de comprendre son fonctionnement si vous le désirez

la liberté de le modifier et de distribuer vos modifications

La FSF et le projet GNU

Richard Stallman, considéré par tous comme le père des logiciels libres a commencé à penser au logiciel libre lorsqu'il travaillait au laboratoire d'intelligence artificielle au MIT. Le laboratoire possédait une imprimante qui tombait souvent en panne, mais comme les chercheurs avaient le code source du pilote^{note}{Le pilote est le logiciel qui sert à accéder et à utiliser l'imprimante.} de l'imprimante, ils avaient modifié le programme pour que l'imprimante leur envoie un signal à chaque panne.

Un beau jour, le laboratoire achète une nouvelle imprimante de marque Xérox, plus fiable. En revanche, le pilote de l'imprimante n'est pas fourni à la livraison. Richard Stallman entend parler plus tard d'un laboratoire qui possède les sources de ce pilote. S'y rendant, on lui répond que le laboratoire s'était engagé à ne pas diffuser les sources du pilote. Ressentant ce refus comme une agression, Richard Stallman prend alors conscience du danger de la logique propriétaire.

Richard Stallman décide alors de fonder la Free Software Foundation. Conscient qu'il est impossible d'utiliser un ordinateur sans utiliser de logiciels propriétaires, il initialise le premier projet de la fondation, le projet GNU (GNU est un jeu de mots récursif signifiant GNU's Not Unix). Ce projet vise à concevoir un système d'exploitation complet et entièrement libre. Ce système sera compatible avec UNIX, mais sera différent. Aujourd'hui ce système existe, et s'appelle GNU/Linux.

Pour valider ce système, une base légale est nécessaire. Cette base légale est la GNU GPL, pour GNU General Public Licence. La GNU GPL est la licence des logiciels libres par excellence. Elle détermine des conditions de distribution qui garantissent les libertés de l'utilisateur. Un programme protégé par la GPL est libre, mais la GPL impose aussi que tout travail dérivé de ce logiciel reste libre.

GNU/Linux, une réalité

Le projet GNU a commencé de zéro, et ses premiers développements avaient pour but de créer un environnement de développement utilisable. Les premiers efforts se sont donc très logiquement portés sur l'élaboration d'un compilateur (gcc), d'un éditeur de textes (Emacs), ainsi que d'un débogueur (gdb). Un noyau (Hurd) était aussi prévu, mais son élaboration prend plus de temps que prévu.

En 1991, Linus Torvalds commence à écrire son noyau, qui connaîtra plus tard le succès que l'on sait. Le projet GNU l'adopte en attendant que Hurd soit opérationnel, et le noyau Linux devient alors composant du système GNU/Linux.

Le système GNU/Linux est réputé pour sa fiabilité et sa robustesse. Ceci est dû en partie à la liberté des logiciels qui le composent, l'accès aux sources permettant de corriger très facilement et rapidement une erreur de programmation.

Le système GNU/Linux est fortement POSIX (POSIX est un ensemble de normes définissant un système UNIX idéal), ce qui le rend très similaire à la plupart des UNIX propriétaires existants. Le noyau Linux est multi-tâches, multi-utilisateurs, et intègre la plupart des technologies les plus récentes (SMP, clustering, RAID, ...).

GNU/Linux est aujourd'hui utilisé aussi bien par les entreprises que par les utilisateurs finaux.

De nombreuses applications sont disponibles pour l'utilisateur non informaticien. Par exemple, l'environnement graphique Gnome permet d'utiliser son ordinateur sans utiliser la ligne de commande. Gimp est un logiciel de traitement d'images très puissant. Gnumeric est un tableur intégré au projet Gnome. Ce ne sont que des exemples, de nombreuses applications peuvent rendre d'immenses services à l'utilisateur. La sécurité et la fiabilité de ce système sont un argument supplémentaire pour l'utiliser (les virus n'existent pas). Plusieurs distributions existent, et l'installation du système GNU/Linux est désormais facile.

Pour l'entreprise, les champs d'application sont vastes. La robustesse du système couplée à l'excellence des applications orientées réseau (Apache par exemple) en fait un système de choix pour les serveurs (le couple GNU/Linux et Apache est le plus utilisé au monde en matière de serveurs web). Les outils de développement (en particuliers les outils GNU : Emacs, gcc, cvs, ...) font de GNU/Linux une plate-forme de développement puissante et agréable à utiliser. En dehors de ces considérations techniques, l'utilisation de logiciels libres affranchit de plus l'entreprise de la main-mise d'un éditeur.

Pourquoi le logiciel libre est t'il meilleur ?

Pour le technicien

Par définition, les sources d'un logiciel libre sont publiques et librement utilisables. Pour le technicien, c'est d'un intérêt inestimable. D'abord, il peut se permettre d'utiliser un code source codé par un autre programmeur sans crainte de poursuites. Il n'a pas besoin de "réinventer la roue".

De plus, les erreurs d'un logiciel libre seront plus rapidement corrigées. L'ouverture des sources permet aux informaticiens (ou tout simplement aux utilisateurs ayant quelques connaissances) de corriger eux-mêmes les erreurs qu'ils rencontrent dans les programmes qu'ils utilisent. Les corrections seront intégrées dans la version suivante du logiciel.

Pour les étudiants en informatique ou même les professionnels, l'accès aux sources permet un apprentissage des plus profonds. Pour beaucoup de professionnels, "la meilleure documentation qui puisse exister, ce sont les sources elles-mêmes".

Pour l'utilisateur

L'utilisateur ne bénéficie pas directement de l'accès aux sources. La plupart des automobilistes ne connaissent pas le fonctionnement d'un moteur, c'est la même chose en informatique. Maintenant, imaginons un monde où 10% de la population fait de la mécanique pendant ses moments perdus, et passe son temps à améliorer son véhicule. Si vous achetez une voiture d'occasion, vous êtes sûr que le moteur a été amélioré par son propriétaire précédent. On gagne en fiabilité, en performance et en qualité.

En informatique, on a plus ou moins le même concept, mais là vous avez une voiture neuve qui profite de toutes les technologies développées dans des petits garages indépendants par des artisans habiles, et dont les meilleures sont rassemblées dans votre véhicule. Vous êtes sûr de la qualité tout comme de la sécurité. Dans le cas d'un vice de conception, vous êtes certain que quelqu'un aura détecté l'erreur avant vous et l'aura corrigée.

Pour les entreprises

Comme pour les utilisateurs, les entreprises bénéficient de la qualité du logiciel. Mais en plus, le modèle de développement des logiciels libres est un avantage à lui tout seul. Il permet de corriger très rapidement une erreur de programmation dont les effets sont critiques pour l'entreprise, et aucune logique marchande ne conditionne les sorties des différentes versions des logiciels libres.

Si une entreprise décide de jouer le jeu et de développer des logiciels libres, elle bénéficie du support de toute la communauté du logiciel libre, tout comme elle garantit la pérennité de son programme.

Brevets sur les logiciels

D'après la directive européenne du 14 mai 1991 concernant la protection juridique des programmes d'ordinateurs, ceux-ci sont protégés par le droit d'auteur, en tant qu'oeuvre littéraire au sens de la convention de Berne pour la protection des oeuvres littéraires et artistiques.

La directive précise que seule l'expression d'un programme d'ordinateur est protégée et que les idées et les principes qui sont à la base de la logique, des algorithmes et des langages de programmation ne sont pas protégés par le droit d'auteur. De même, les idées et principes qui sont à la base de ses interfaces ne sont pas protégés par le droit d'auteur.

L'unique critère pour déterminer si un programme peut bénéficier d'une protection est son originalité, en ce sens que ce programme est la création intellectuelle propre à son auteur.

Un projet de directive européenne propose de remettre en cause la non brevetabilité des programmes d'ordinateurs garantie par la directive européenne de 1991 et la Convention de Munich.

Effets dangereux des brevets sur les logiciels

Quelques effets pervers des brevets sur les logiciels :

Pour l'innovation

Le brevet permet de restreindre l'utilisation d'une innovation pendant vingt ans.

Or, dans le cas des programmes d'ordinateurs, les brevets résultent le plus souvent de la découverte ou de la simple application d'une propriété mathématique ou d'un algorithme, qui fait partie du patrimoine de l'humanité. La plupart du temps, les découvertes effectuées par un programmeur ne sont que l'adaptation d'idées déjà existantes à un problème bien défini.

Aux Etats-Unis, plus de 50 000 brevets concernant l'informatique sont déposés. Dans un tel contexte, il est impossible à un programmeur d'accomplir son métier sans transgresser un brevet à son insu. Les brevets informatiques couvrent des domaines divers : les éléments les plus simples de l'interface d'un programme, des algorithmes qu'un étudiant de première année dans une école informatique pourrait inventer à nouveau, et même l'utilisation des couleurs. Il faudrait mettre un ou deux avocats derrière chaque programmeur, et, dans la plupart des cas, il serait impossible d'écrire un programme fonctionnel, car les algorithmes les plus efficaces ou évidents sont protégés par des brevets.

De grands groupes détiennent la majorité des brevets et s'en servent pour bloquer leurs concurrents, le brevet étant ici une arme offensive et non plus défensive. Seules les très grandes compagnies de logiciels auraient le droit à la création. Ce serait la fin du droit de chacun d'innover et de créer.

Pour l'informatique libre

L'informatique libre est forcément la plus touchée par la brevetabilité des programmes d'ordinateurs. Un logiciel libre utilise par nature des techniques et des inventions découvertes par autrui, tout comme il autorise l'utilisation future des techniques développées par son programmeur.

Un programmeur de logiciel libre ne peut pas non plus se permettre d'embaucher des avocats pour se protéger dans le cas probable où il enfreindrait un brevet par inadvertance.

De plus, l'informatique libre posant comme principe le partage et l'utilisation mutuelle des connaissances, pour le bien de la communauté, ne peut accepter la brevetabilité des programmes d'ordinateurs.

Les idées appartiennent à l'humanité.

Pour en savoir plus

Si vous voulez en savoir plus, n'hésitez pas à faire un tour sur les sites suivants :

<http://www.gnu.org>, le site web du projet GNU et de la FSF. C'est le site web de référence lorsqu'on parle de logiciel libre. Il contient la philosophie GNU qui explique pourquoi il faut du logiciel libre et de nombreuses autres choses comme la GPL.

<http://www.april.org>, le site de l'association APRIL (Association pour la Promotion et la Recherche en Informatique Libre). Le web contient des informations sur l'association, sur leurs travaux en cours, sur les manifestations organisées par l'association ainsi que sur la philosophie de l'informatique libre.

<http://www.debian.org>, le site de la distribution de GNU/Linux la plus libre.

<http://www.linux.org>, le site de départ pour toute information ou recherche de logiciels concernant le système GNU/Linux.

<http://www.linux-france.org>, site d'informations (avec de nombreuses documentations) en français autour de GNU/Linux.

<http://www.linux-center.fr>, site d'actualités en français autour de GNU/Linux.

<http://lpf.ai.mit.edu>, le site de la League For Programing Freedom, qui s'intéresse au problème des brevets sur les logiciels.

Généré le 29/08/1999 par Benjamin Drieu

Sauf mention contraire indiquée plus haut, le présent document est soumis aux conditions d'exploitation suivantes :

Copyright 1999 APRIL Ce document peut être reproduit par n'importe quel moyen que ce soit, pourvu qu'aucune modification ne soit effectuée et que cette notice soit préservée.

Ce texte est un peu long, mais il est intéressant. Sa structure apparaît nettement à la lecture. Vous le trouverez sur le [site de l'APRIL](#), l' Association Pour la Recherche en Informatique Libre.

Le mieux est de partir d'un fichier ne contenant que ce texte. Ouvrez un document texte vide et recopiez-le dedans¹.

6.1 Indiquer la structure du texte.

Ensuite il faut indiquer la structure; voici le tableau avec les titres et le style à leur appliquer :

Logiciels libres, une introduction	Titre principal
Qu'est ce qu'un logiciel ?	Titre 1
Qu'est-ce qu'un logiciel libre ?	Titre 1
La FSF et le projet GNU	Titre 1
GNU/Linux, une réalité	Titre 1
Pourquoi le logiciel libre est t'il meilleur ?	Titre 1
Pour le technicien	Titre 2
Pour l'utilisateur	Titre 2
Pour les entreprises	Titre 2
Brevets sur les logiciels	Titre 1
Effets dangereux des brevets sur les logiciels	Titre 2
Pour l'innovation	Titre 3
Pour l'informatique libre	Titre 3
Pour en savoir plus	Titre 1

¹ Utilisez le copier-coller si vous utilisez une version électronique de ce document!

Logiciels libres, une introduction

Qu'est ce qu'un logiciel ?

Pour comprendre le concept de logiciel libre, vous devez d'abord comprendre qu'est-ce qu'un logiciel (ou encore programme ou application).
Du point de vue de l'utilisateur, un logiciel est une application qui répond à l'un de ses besoins (traitement de texte, jeu, programme de dessin, jeu...). C'est un ensemble de petites instructions inscrites pour l'utilisateur, qui s'exécutent un jour ou l'autre.
Ces logiciels ont besoin d'un système d'exploitation pour fonctionner. Le système d'exploitation permet d'accéder aux ressources de la machine (lecteur de disque, écran, clavier...). C'est au système d'exploitation qu'est chargé d'exécuter le logiciel.
Les instructions d'un logiciel sont écrites dans un langage que l'ordinateur ne comprend pas, le langage machine (ou langage binaire). En revanche, il est très difficile (voire impossible) à un humain de comprendre un langage (c'est le plus utilisé dans de nombreux pays). La manière la plus courante de coder un logiciel est de l'écrire dans un langage informatique compréhensible par des humains, et ensuite de le traduire vers le langage binaire. Cette traduction s'effectue par un logiciel appelé compilateur.
Le logiciel dans sa forme compréhensible est appelé code source, et dans sa version langage machine binaire.

Qu'est-ce qu'un logiciel libre ?

Imaginez que vous vous trouviez dans une restaurant que vous mangiez un excellent plat. Par votre amour pour l'artisan de la cuisine, vous le partagez avec vos amis ?
C'est impossible, car vous n'avez pas la recette du plat. Vous pouvez toujours le manger dans le restaurant, mais même si vous connaissez le goût, vous ne savez comment le reproduire.
En informatique, c'est la même chose avec un logiciel. Le code source est la recette, le binaire est le plat cuisiné. La plupart des logiciels sont distribués sans leur code source, et il est très difficile d'essayer de comprendre le fonctionnement. Il est impossible de les partager avec vos amis, et il est impossible d'essayer de les modifier pour les adapter à vos besoins.
Un logiciel libre vous garantit plusieurs libertés :
la liberté de copier le logiciel pour vous ou vos amis
la liberté de comprendre son fonctionnement et de le débiter
la liberté de le modifier et de publier vos modifications

La FSF et le projet GNU

Richard Stallman considère par exemple comme le père des logiciels libres a commencé à penser au logiciel libre lorsqu'il travaillait au laboratoire d'intelligence artificielle au MIT. Le laboratoire possédait une imprimante qui tombait souvent en panne, mais comme les chercheurs avaient le code source du pilote (Le pilote est le logiciel qui sert à contrôler et à utiliser l'imprimante.) de l'imprimante, ils avaient modifié le programme pour que l'imprimante leur envoie un signal à chaque panne.
Un jour, le laboratoire achète une nouvelle imprimante de marque Xerox plus fiable. En revanche, le pilote de l'imprimante n'est pas fourni à la maison. Richard Stallman ne peut plus parler à l'imprimante qui possède le code source du pilote. Symplicité on lui répond que le laboratoire s'est engagé à ne pas diffuser le code source du pilote. En conséquence, il ne peut pas le modifier, Richard Stallman prend alors conscience du danger de la logique propriétaire.

Illustration 12

Le texte commence à prendre forme. Il devrait ressembler à ceci (cf Illustration 12). Toutefois le résultat, s'il est plus lisible, ne correspond peut-être pas encore à ce que l'on espère. On peut améliorer la chose.

6.2 Numérotation des sections.

Commençons par numéroter les sections de types Titre 1 & Titre 2. Cela se passe dans le menu *Outils* □ *Numérotation des chapitres*. Vous obtiendrez une fenêtre comme celle de l'illustration 13.

Dans la colonne **Niveau**, on indique le niveau à numéroter.

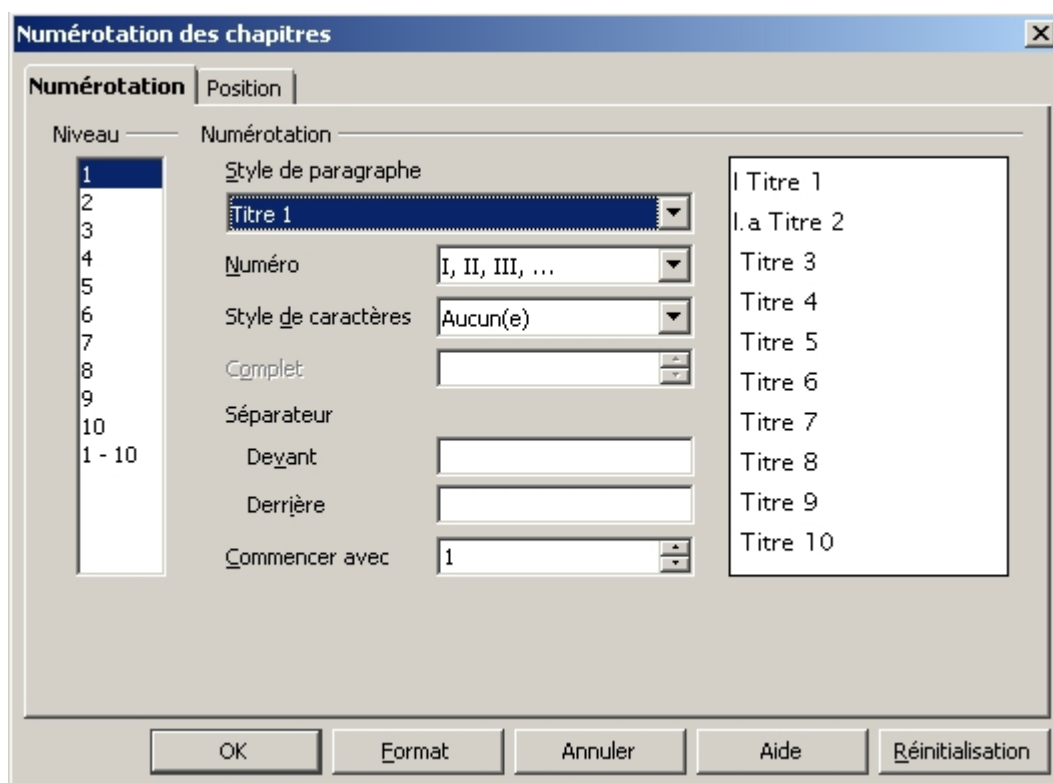


Illustration 13

Ici pour le **Niveau 1** on laisse le style **Titre 1**, que l'on a appliqué pour cela. On choisit le style de numéros voulus (ici I, II, III, ...). On ajoute un espace comme séparateur arrière par soucis d'esthétisme (vous pouvez mettre un « - » par exemple).

Pour le **Niveau 2** on laisse le style **Titre 2**, on choisit la numérotation a, b, c, ... Les modifications sont indiquées au fur & à mesure dans le cadre de droite. On passera le *champ Complet* à 2 pour obtenir la numérotation des titres de sous-section du type I.a plutôt que a seulement.

6.3 Table des matières automatique

Allez je vous laisse chercher². Demandez donc à l'aide d'OpenOffice.org comment insérer une table des matières, et insérez-en une après le titre principal. Mettez **Evaluation jusqu'au niveau** à 3 puis 2 puis 1 et remettez-le à 3 après avoir compris sa signification.

Voici à quoi ressemble maintenant notre texte (Illustration 14).

² Pas trop non plus: [insertion->index->index](#)

Logiciels libres, une introduction

Table des matières

I Qu'est ce qu'un logiciel ?	1
II Qu'est ce qu'un logiciel libre ?	1
III La FFF et le projet GNU	2
IV GNU/Linux, une réalité	2
V Pourquoi le logiciel libre est-il meilleur ?	3
VA Pour le technicien	3
VB Pour l'utilisateur	3
VC Pour les autres personnes	3
VI	
VII Pourquoi les logiciels	3
VIIA Effet d'annonce des boutons sur les logiciels	4
Pour l'information	4
Pour l'information libre	4
VIIIPour en savoir plus	4

I Qu'est ce qu'un logiciel ?

Pour comprendre le concept de logiciel libre, vous devez d'abord comprendre ce qu'est un logiciel (ou encore programme ou application).

Du point de vue de l'utilisateur, un logiciel est une application qui répond à l'un de ses besoins (traitement de texte, programme de dessin, jeu, ...). C'est une suite de petites instructions invisibles pour l'utilisateur, qui forment un programme.

Ces logiciels ont besoin d'un système d'exploitation pour fonctionner. Le système d'exploitation permet d'accéder aux ressources de la machine (lecteur de disque, écran, clavier, ...). C'est au système d'exploitation qu'est chargé de faire vivre le logiciel.

Les instructions d'un logiciel sont écrites dans un langage que l'ordinateur peut comprendre, le langage machine (ou langage binaire). En revanche, il est très difficile (voire impossible) à un humain de comprendre un langage (c'est le plus intimidant dans de nombreux pays). Le langage le plus courant de ces logiciels est écrit dans un langage informatique compréhensible par des humains, et est écrit dans le langage machine. Cette traduction est effectuée par un logiciel appelé compilateur.

Le logiciel dans sa forme compréhensible est appelé le code source, et dans sa version langage machine binaire.

II Qu'est ce qu'un logiciel libre ?

Imaginez que vous êtes invité dans un restaurant que vous mangiez un excellent plat. Par votre amie vous l'avez fait de la cuisine chez vous pour vos amis ?

C'est impossible, car vous n'avez pas la recette du plat. Vous pouvez toujours le manger dans le restaurant mais même si vous connaissez le goût, vous ne savez comment le reproduire.

En informatique, c'est la même chose avec un logiciel. Le code source est la recette, le binaire est le plat cuisiné. La plupart des logiciels sont distribués sans leur code source, et il est interdit d'essayer de comprendre le fonctionnement. Il est interdit de les partager avec vos amis, et il est interdit d'essayer de les modifier pour les adapter à vos besoins.

Un logiciel libre vous garantit plusieurs libertés :

- la liberté de copier le logiciel pour vous ou vos amis
- la liberté de comprendre son fonctionnement et de le modifier

Illustration 14

6.4 Modification d'un style

Le résultat commence à devenir correct, du moins pour la structure. Toutefois, pour comprendre mieux comment agir sur nos styles, nous allons mettre les titres de section, i.e. les **Titre 1** sur un fond gris clair.

Je vous ai dit comment faire apparaître un *menu contextuel* ? Mais si, au début. Bon allez je répète : c'est avec le bouton droit de la souris. CTRL enfoncé en cliquant si vous êtes sur Mac.

Allez dans le styliste, sélectionnez le **Titre 1**, et cliquez avec le bouton droit (Illustration 15). Choisissez **Modifier**.



Illustration 15

Comme c'est un style de paragraphe qu'on modifie, la fenêtre qui s'ouvre (Illustration 16) est celle qui s'applique au paragraphe. Choisissez **Arrière-plan** et mettez une couleur gris assez clair (20 % par exemple). Validez et admirez le résultat : **tous** les titres de section ont été modifiés en même temps.

Vous pouvez utiliser la même méthode pour justifier le texte. Tous les paragraphes hormis les titres doivent être en *corps de texte*. S'il n'y sont pas cliquez sur ce style dans le styliste puis sur le bouton en forme de seau en train de verser. Et d'un clic sur chaque paragraphe ce style est appliqué! Ensuite il n'y a plus qu'à modifier le style *corps de texte* et à lui ajouter la justification. Facile, non?

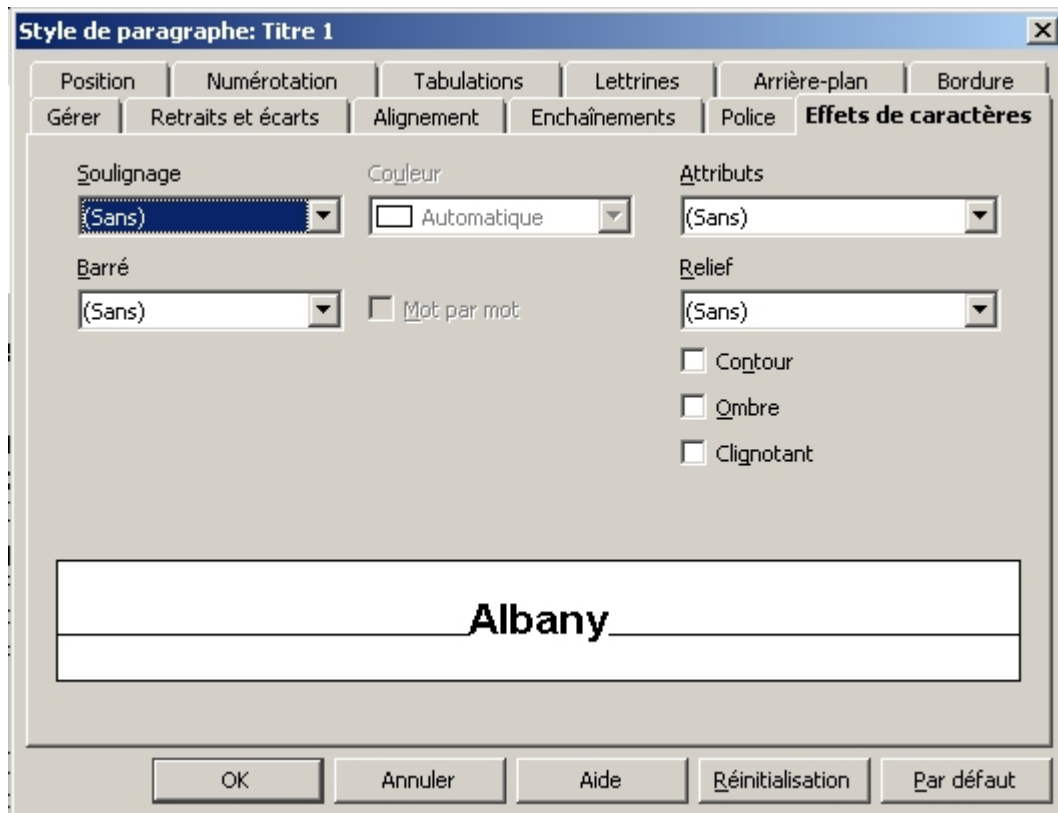


Illustration 16

6.5 Mettre des puces (ou des numéros) avec le style de numérotation

Il manque encore des puces aux lignes qui commencent par « la liberté » et à celles qui commencent par « http ». Dans le styliste le 5^e bouton indique (en restant dessus avec la souris: c'est une *bulle d'aide*) qu'il indique les *styles de numérotation*. On y trouve les styles de puces.

Choisissons **Puce 1** et appliquons le là où les puces manquent. Puis modifions ce style. Dans l'onglet *Options* on choisit le caractère voulu pour chaque niveau, et *Position* on place un retrait relatif de 0,5 cm pour le niveau 1-10 (c'est-à-dire tous).

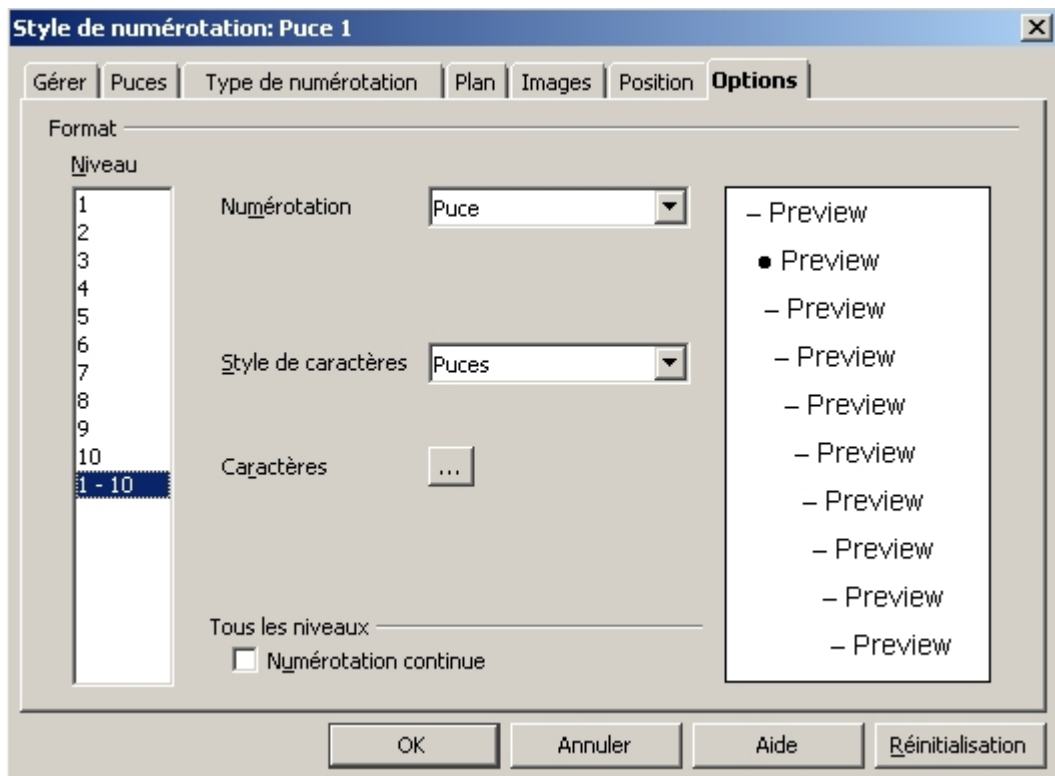


Illustration 17

Le même genre de modification permet de faire des listes numérotées sans se poser plus de problèmes.

Voilà j'en ai fini avec ce document. Il est plus sympa comme ça, non?

7 En guise de conclusion

Je pense avoir parlé de ce qui est le plus utilisé. Ceci dit on peut faire beaucoup plus avec OpenOffice.org.

On peut créer des bibliographies, des index, insérer des objets de plein de types différents, faire un gros document avec des fichiers séparés, mettre des liens hypertextes, créer des pages Web, etc...

N'hésitez pas à demander de l'aide à l'aide intégrée! Ça ne casse rien!

Ah oui, un dernier conseil: si vous faites un document long, passez par *Outils* \square *Options* et dans *Chargement/Enregistrement, Général*, mettez un enregistrement automatique toutes les cinq minutes. C'est désagréable de perdre deux heures de boulot en cas de plantage système !

Amusez-vous bien!