

SCI6005 Information numérique et informatique documentaire (H2025)

Anton Boudreau Ninkov, EBSI, UdeM

H2025 20 mars 2025

*Cours 8 - Structuration de l'information dans un traitement de texte
SCI6005*

Table des matières

I - Cours 8 - Structuration de l'information dans un traitement de texte	3
1. + Au programme aujourd'hui	3
2. + Alignement pédagogique	3
3. Introduction	5
4. Modélisation	6
4.1. Modélisation d'un type de document : Conceptualisation (1ère étape)	7
4.2. Modélisation d'un type de document : Représentation de la conceptualisation (2ième étape)...	7
4.3. Modélisation d'un type de document : Validation du modèle (3ième étape)	8
5. Structuration dans un traitement de texte	9
5.1. Couche sémantique ajoutée aux fichiers : Métadonnées.....	9
5.2. Couche sémantique ajoutée au contenu	12
5.3. Couche sémantique présente dans le contenu : Expressions régulières.....	20
6. Ressources en lien avec le cours	23
Glossaire	24
Index	25
Crédits des ressources	26

Cours 8 - Structuration de l'information dans un traitement de texte



1. + Au programme aujourd'hui

- **Modélisation** d'un type de document
- **Métadonnées** : couche sémantique ajoutée aux fichiers
- Fonctionnalités de **structuration dans un traitement de texte** (styles, légendes et renvois, tables et index, modèles) : couche sémantique ajoutée au contenu des documents
- **Expressions régulières** : exploitation de la couche sémantique présente dans les contenus

Expressions régulières



Remarque

Les expressions régulières sont insérées dans ces notes comme elles relèvent du traitement de texte. Cependant, nous les aborderons plutôt la semaine prochaine, comme vous les exploitez uniquement dans le cadre du *TP sur la structuration dans un tableau*.

2. + Alignement pédagogique



Objectifs visés, matériel du cours et évaluation : Examen final

Objectif général : Concevoir et planter des systèmes et services d'information numérique		
Objectifs spécifiques	Compétences à développer	Matériel associé
Exécuter des manipulations de base sur des documents, données et fichiers numériques	Appliquer l'utilisation des expressions régulières pour des cas simples	Section <i>Structuration dans un traitement de texte > Couche sémantique présente dans le contenu : Expressions régulières</i> <i>TP Structuration dans un tableau</i>
Appliquer une approche systématique pour la mise sur pied de systèmes et de services d'information numérique	Expliquer les étapes du processus de modélisation	Section <i>Modélisation</i>
	Modéliser un type de document	<i>TP Structuration dans un traitement de texte</i>

	<p>Nommer les avantages de la structuration de l'information dans un traitement de texte</p>	<p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée aux fichiers : Métadonnées > Avantages</i></p> <p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée contenu > Styles > Avantages</i></p> <p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée contenu > Légendes et renvois > Avantages</i></p> <p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée contenu > Tables des matières, autres tables ou index > Avantages</i></p> <p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée contenu > Modèles > Avantages</i></p>
Exploiter la structuration de l'information numérique dans différents contextes	<p>Distinguer les métadonnées descriptives et techniques</p>	<p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée aux fichiers : Métadonnées</i></p>
	<p>Manipuler des styles (création, modification, application)</p>	<p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée au contenu > Styles</i></p> <p><i>TP Structuration dans un traitement de texte</i></p>
	<p>Expliquer l'utilité des légendes et des renvois</p>	<p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée au contenu > Légendes et renvois</i></p>
	<p>Insérer des tables et des index</p>	<p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée au contenu > Tables des matières et autres tables ou index</i></p> <p><i>TP Structuration dans un traitement de texte</i></p>
	<p>Créer un modèle de documents</p>	<p>Section <i>Structuration dans un traitement de texte > Couche sémantique ajoutée au contenu > Modèles</i></p> <p><i>TP Structuration dans un traitement de texte</i></p>

Lien entre les objectifs, les compétences à développer et le matériel du cours 8



Objectifs visés, matière du cours et activités associées

Objectif général : Concevoir et implanter des systèmes et services d'information numérique		
Objectifs spécifiques	Matière	Activités
Appliquer une approche systématique pour la mise sur pied de systèmes et de services d'information numérique	Modélisation d'un type de document	<i>TP Structuration dans un traitement de texte</i>
Exploiter la structuration de l'information numérique dans différents contextes	Styles	<i>TP Structuration dans un traitement de texte</i>
	Tables et index	<i>TP Structuration dans un traitement de texte</i>
	Modèle de document	<i>TP Structuration dans un traitement de texte</i>
Exécuter des manipulations de base sur des documents, données et fichiers numériques	Expressions régulières	<i>TP Structuration dans un tableur</i>

Lien entre les objectifs, la matière du cours 8 et les activités associées

3. Introduction

Professionnels et professionnelles de l'information croiseront plusieurs occasions d'utiliser leur **expertise en structuration et organisation de l'information** au profit d'une **bonne gestion des documents** dans leurs milieux. Une des manières d'y contribuer est d'intervenir dès la *création* des documents afin de proposer des **modèles de documents** qui permettent d'exploiter au mieux l'information qui s'y trouve. Nous nous attarderons cette semaine à la création de documents dans un

traitement de texte pour, dans un premier temps, formaliser la démarche de *modélisation* d'un type de document afin, dans un deuxième temps, d'examiner les différentes possibilités d'*ajout de couches sémantiques* aux fichiers et à leurs contenus et d'*exploitation* de ces couches sémantiques.

4. Modélisation

« La modélisation peut se définir de la manière suivante. C'est un processus qui passe par toutes les phases de la démarche scientifique, avec, à partir de l'observation, une conceptualisation du phénomène, une traduction dans un langage (un modèle) et une confrontation du modèle avec de nouvelles observations. Cela peut apparaître trivial, mais il faut rappeler que l'ensemble des phases est influencé par l'expérience et la science de l'observateur. »

Source : Schème Modelling & Environment, 2013, <https://www.scheme-rd.fr/notre-methodologie-la-modélisation>.

La définition de modélisation ci-dessus résume bien l'essence même de ce processus qui permet de représenter une certaine réalité à l'aide d'un modèle. Cette démarche, qui se doit d'être **systématique**, peut prendre différentes formes selon le type de réalité que l'on veut modéliser et l'environnement de modélisation utilisé, mais les grandes étapes demeurent sensiblement les mêmes :

- **Conceptualisation** d'un phénomène à partir d'observations
- **Représentation** de la conceptualisation à l'aide d'un certain langage
- **Validation** du modèle à partir de nouvelles observations

La modélisation peut se faire dans différents contextes et de différentes manières, se voulant parfois descriptive et d'autre fois prédictive, par exemple :

- Modélisation mathématique de certains phénomènes
- Modélisation 3D d'une structure en architecture et ingénierie
- Modélisation des processus en entreprise
- Modélisation de systèmes d'information en informatique

Dans le cadre du cours, nous avons déjà croisé différentes modélisations, que ce soit les modèles des écologies informationnelles, de l'informatique sociale, de l'appropriation des technologies, du processus d'implantation de solutions documentaires par exemple.

Comme professionnelles et professionnels de l'information, différentes occasions se présenteront pour modéliser des types de documents, des comportements informationnels, des processus d'affaires, etc. Il est ainsi utile de se familiariser avec ce processus, ce que nous ferons dans le cadre du cours pour les prochaines semaines :

- Nous nous intéresserons à la modélisation de différents **types de documents** afin d'en faire ressortir la *structure logique*.
- Nous verrons par la suite comment, selon le type de documents et de structures logiques, il est possible d'**exploiter** au mieux les possibilités de **structuration** de l'information dans **différents types de logiciels d'application** (traitement de texte, tableur, système de gestion de bases de données).

Les différentes étapes de cette démarche de modélisation d'un type de documents sont présentées ci-dessous.

4.1. Modélisation d'un type de document : Conceptualisation (1ère étape)

La première étape consiste à *observer* des **exemples représentatifs** du **type de documents** que l'on veut modéliser pour s'assurer de bien en comprendre le **contenu** et la **structure**. On part ainsi habituellement d'un *certain corpus* (c'est-à-dire d'un ensemble de documents) auquel s'ajouteront bien entendu les *connaissances de ce type de documents* que possède la personne qui en fait la modélisation.

La conceptualisation est ainsi un exercice d'**observation** et d'**analyse** :

- Il faut réussir à *identifier les composantes communes* à chacun des documents du corpus, celles qui sont *obligatoires* et celles qui sont *facultatives*.
- Il faut comprendre comme les composantes sont *relées entre elles*.
- Il faut identifier la *nature des contenus* qu'on y retrouve, la *manière de les écrire* s'il y a lieu (par exemple, si on retrouve une date, y a-t-il un format privilégié?).

Conceptualisation d'un courriel

?

Exemple

L'**observation de plusieurs courriels** permet de mettre au jour les caractéristiques suivantes :

- Un *courriel* est composé d'un *expéditeur*, d'un ou plusieurs *destinataires*, d'un *objet* et d'un *message*.
- Les *destinataires* sont composés d'une ou plusieurs *adresses courriel* tandis que, pour l'*expéditeur*, il n'y en a qu'*une seule*. Une adresse courriel a la forme *abc@def.ghi*.
- L'*objet* d'un courriel est une *simple phrase*.
- Le *message* d'un courriel peut être formé d'un ou plusieurs *paragraphes* de texte.

Note : C'est cette conceptualisation qui a permis de présenter lors d'un cours précédent la représentation XML d'un courriel!

4.2. Modélisation d'un type de document : Représentation de la conceptualisation (2ième étape)

Une fois que l'on a bien compris le contenu et la structure du type de documents que l'on veut modéliser, il faut **représenter la conceptualisation obtenue** afin de l'expliciter. Différents types de représentation sont possibles, entre autres en fonction du type de documents. Ce peut être un *schéma*, un *tableau*, une *carte conceptuelle*, etc. Pour certains types de modélisation, il existe des représentations bien établies comme, par exemple :

- pour la modélisation de systèmes informatiques avec UML (*Unified Modeling Language*¹),
- pour la modélisation de bases de données avec le modèle Entité-Relation (ou Entité-Association²).

Lorsqu'il n'y a pas de méthode bien établie, il s'agit d'en choisir une qui correspond bien au type de modélisation à représenter.

La représentation de la conceptualisation sous une forme ou une autre peut amener à **déceler des erreurs de logique dans la conceptualisation**. Il s'agit alors de faire des **allers-retours** entre *conceptualisation* et *représentation* jusqu'à la création d'une représentation qui semble *stabilisée* (c'est-à-dire où l'on ne retrouve pas d'erreur).

¹<https://www.uml.org/>

²https://fr.wikipedia.org/wiki/Mod%C3%A8le_entit%C3%A9-association

Représentation de la conceptualisation d'un courriel

?

Exemple

Dans le cadre d'un type de documents textuel que l'on peut produire dans un **traitement de texte**, on pourrait en représenter la structure à l'aide d'une **liste à puces hiérarchisée** de la manière suivante :

- courriel
 - expéditeur [forme abc@abc.com]
 - destinataire
 - mel + [forme abc@abc.com]
 - objet
 - message
 - para +

Plus précisément, cette représentation est basée sur une syntaxe et une sémantique très simples retenues pour les besoins du *TP Structuration dans un traitement de texte* :

1. La **hiérarchie** des puces permet de représenter les *relations d'inclusion* entre des éléments et leurs sous-éléments, comme c'est le cas ici de *mel* qui est un sous-élément de destinataire.
2. Il est intéressant de pouvoir y indiquer aussi les éléments qui peuvent se **répéter**. C'est le cas, par exemple, de l'élément *para*. On pourrait retrouver plusieurs paragraphes à l'intérieur du message. C'est ce que le **+** signifie.
3. De plus, il peut être utile de consigner des informations sur les **règles d'écriture** particulières de certains contenus comme, par exemple, la forme d'une adresse courriel. C'est ce qui est indiqué entre **crochets carrés** dans l'exemple [...].
4. Finalement, on pourrait utiliser l'**italique** pour indiquer les éléments qui sont **facultatifs**. Dans cet exemple, on ne retrouve aucun italique comme tous les éléments sont obligatoires.

4.3. Modélisation d'un type de document : Validation du modèle (3ième étape)

Il est finalement conseillé de **valider le modèle** ainsi défini à partir de **nouveaux exemples** de documents du type de documents représenté. Cette étape pourrait amener à **ajuster** le modèle (tant sa conceptualisation que sa représentation), comme les nouveaux exemples pourraient avoir fait ressortir des éléments de contenu et de structure absents du corpus initial.

Une autre manière de valider une modélisation est de la **présenter à des personnes externes** familières avec le type de document. Elles pourraient ainsi confirmer ou infirmer la qualité de la modélisation obtenue.

Validation de la conceptualisation d'un courriel

?

Exemple

Imaginez, par exemple, que vous ayez modélisé la structure d'un courriel avec uniquement des exemples possédant un destinataire unique. En ce cas, vous n'auriez pas défini l'élément *mel* comme répétable (+). La validation avec de nouveaux courriels, dont certains avec plusieurs destinataires, vous amènerait à revoir cette décision de modélisation.

Une personne externe pourrait aussi proposer des ajouts à votre modélisation sur la base de son expérience comme :

- La **présence potentielle** de pièces jointes
- Une **composante temporelle** (date d'envoi)

La représentation de cette nouvelle modélisation serait ainsi la suivante :

- courriel
 - expéditeur [forme abc@abc.com]
 - destinataire
 - mel + [forme abc@abc.com]
 - date [forme aaaa-mm-jj]
 - objet
 - message
 - para +
 - pièce jointe
 - fichier +

5. Structuration dans un traitement de texte

Les traitements de texte offrent plusieurs moyens d'ajouter une **couche sémantique** aux contenus pour mieux les *structurer* et les *organiser*, mieux les *repérer* et mieux les *gérer*. Cette section des notes de cours présentera *trois types de couches sémantiques* que l'on peut exploiter :

- Les couches sémantiques que l'on peut **ajouter aux fichiers** comme tel, soit les *métadonnées* ^{p.24}.
- Les couches sémantiques que l'on peut **ajouter au contenu** des documents par le biais de différentes fonctionnalités.
- Les couches sémantiques **incluses explicitement dans le contenu** que l'on peut exploiter.

5.1. Couche sémantique ajoutée aux fichiers : Métadonnées

Avantages

Les métadonnées associées à un fichier représentent une couche sémantique importante pour assurer une bonne gestion des fichiers. Des métadonnées pertinentes faciliteront entre autres l'*organisation* et le *repérage* des fichiers.

On retrouve deux types de métadonnées : (1) des **métadonnées techniques** et (2) des **métadonnées descriptives**. Ces métadonnées peuvent être visualisées et modifiées de différentes façons.



Remarque

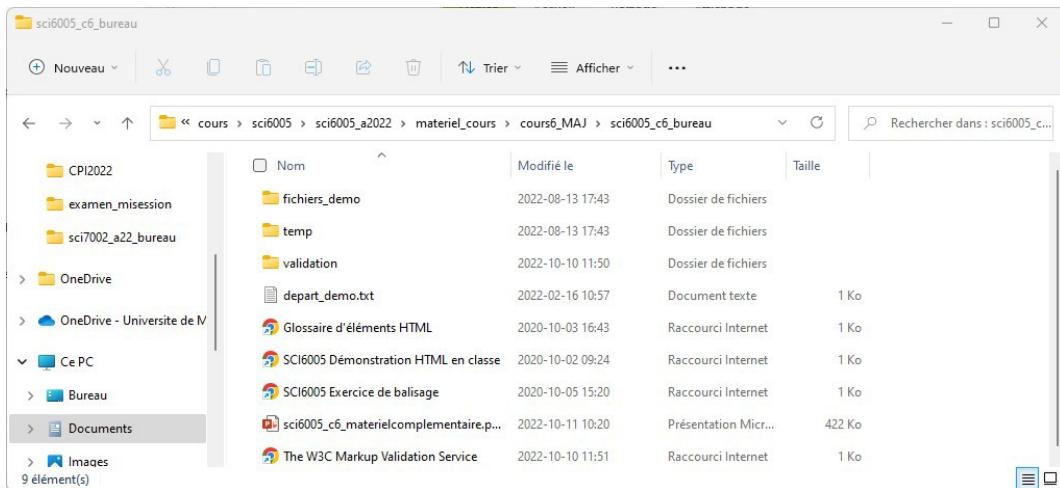
Il est à noter que ce ne sont pas que les fichiers de traitement de texte qui peuvent bénéficier de cette couche sémantique, mais bien tous les fichiers!

a) Métadonnées techniques

Les **métadonnées techniques** informent sur les *caractéristiques des fichiers* comme tels, par exemple :

- le *nom* du fichier
- la *taille* du fichier
- la *date de création/modification* du fichier

Ce sont des métadonnées de base qui s'affichent habituellement dans le gestionnaire de fichiers comme illustré ci-dessous.

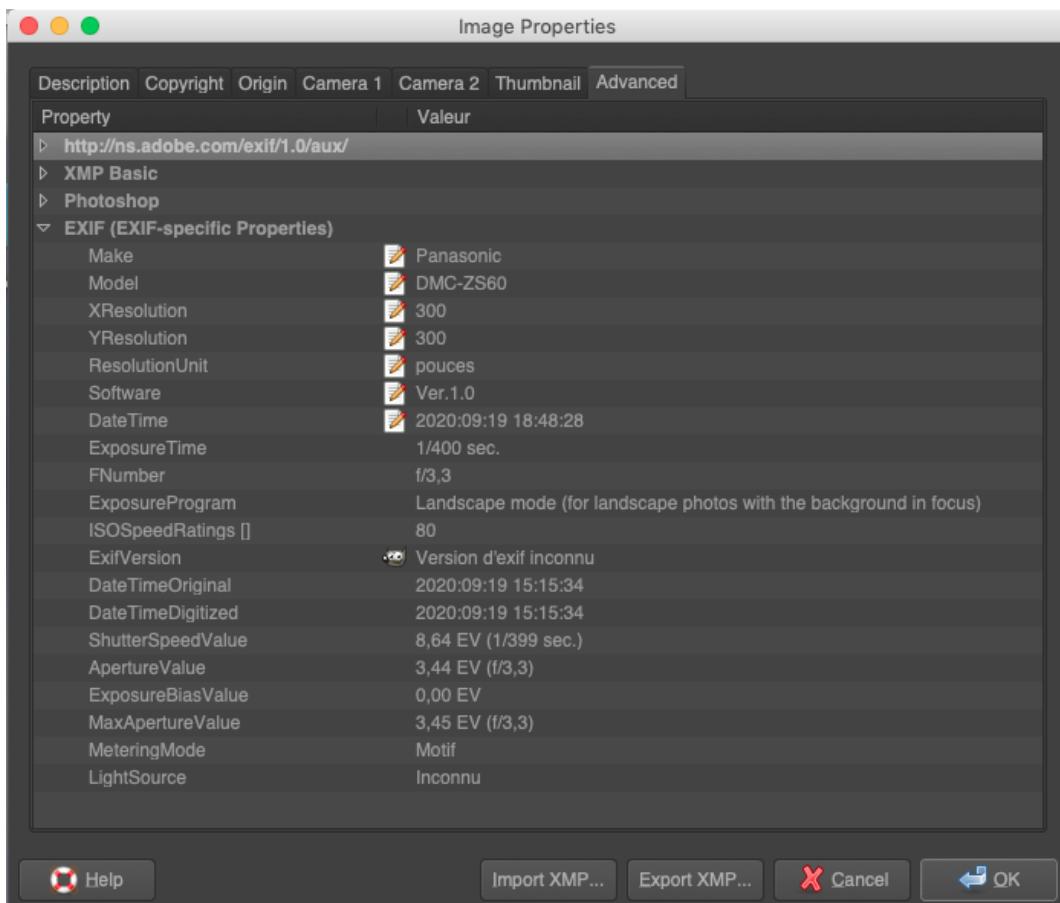


Métadonnées techniques de base affichées dans le gestionnaire de fichiers (Windows)

D'autres **métadonnées techniques plus spécialisées** s'ajoutent à ces métadonnées de base en fonction du *type de fichier* comme, par exemple :

- la *résolution* d'une image
- la *longueur* pour une vidéo
- le *débit binaire* pour un fichier audio

La copie d'écran ci-dessous montre des métadonnées associées à une photographie et affichées dans le logiciel d'édition photo libre *GIMP*. On y remarque bien la spécificité des métadonnées par rapport au type de fichier (photographie).



Métadonnées techniques pour une photographie dans le logiciel GIMP

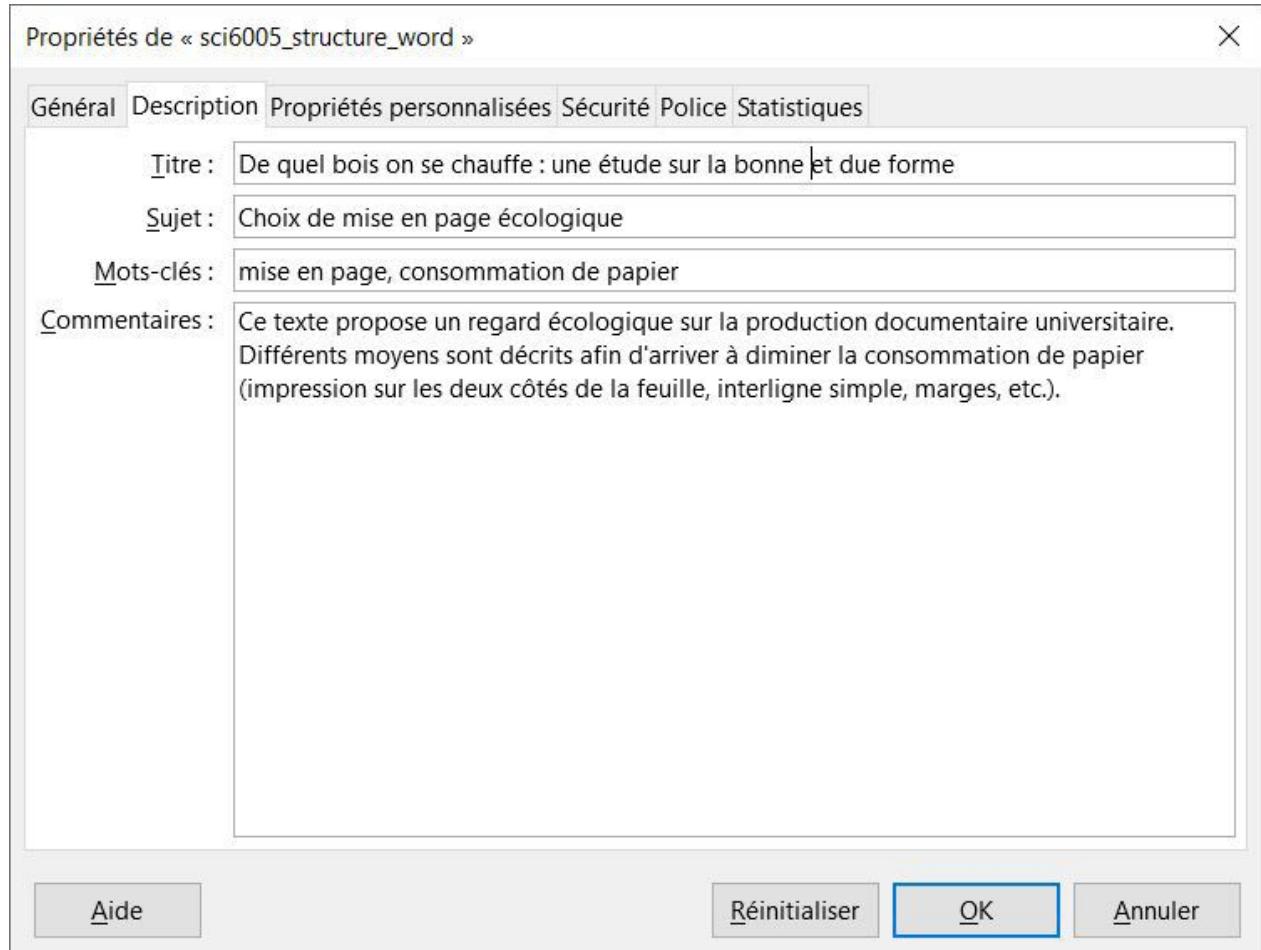
Les métadonnées techniques sont des métadonnées qui sont habituellement générées automatiquement lors de la création des fichiers.

b) Métadonnées descriptives

Les **métadonnées descriptives** visent à *décrire le contenu ou le contexte de création* comme, par exemple :

- les *auteurs* du document
- le *sujet* du document
- des *mots-clés* décrivant le document

La copie d'écran ci-dessous montre les propriétés que l'on peut indiquer, dans *Writer (LibreOffice)*, pour décrire le contenu d'un fichier.

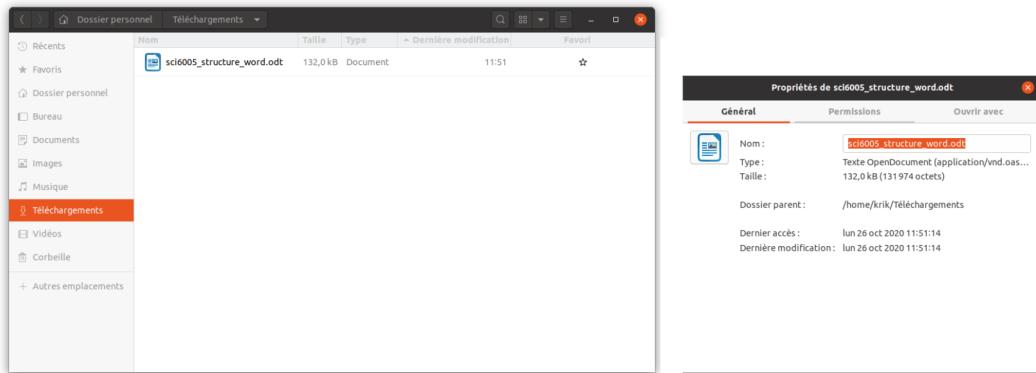


Métadonnées descriptives affichées dans *Writer (LibreOffice)*

Ces métadonnées, pour la plupart, sont *créées manuellement* comme elles demandent de créer des éléments descriptifs. La création peut dans certains cas être semi-automatique lorsque les métadonnées sont dérivées de métadonnées associées à une identité prédéfinie par exemple. C'est le cas si on se branche à son compte sur *Office 365*; la métadonnée *Auteur* est automatiquement dérivée.

c) Visualisation et modification des métadonnées

Comme les exemples présentés ci-dessus l'illustrent, on peut visualiser les métadonnées de différentes manières. Le *système d'exploitation* permet d'en visualiser plusieurs, dont les métadonnées techniques de base, que ce soit à l'aide du **gestionnaire de fichiers** ou de l'affichage des **propriétés** d'un fichier :



Métadonnées sous Linux dans le gestionnaire de fichiers ainsi que les propriétés du fichier

Certaines métadonnées plus étroitement liées au type de fichier sont uniquement visibles à partir du **logiciel de création ou d'édition** utilisé, comme illustré dans la copie d'écran du logiciel *GIMP* présentée précédemment.

Certaines métadonnées peuvent être *modifiées*, tout particulièrement les métadonnées descriptives. Cette modification, bien qu'elle puisse se faire pour certaines d'entre elles dans la fenêtre des propriétés générée par le système d'exploitation, se fait souvent à même le *logiciel d'application*. D'autres métadonnées ne peuvent être modifiées manuellement, entre autres des métadonnées techniques comme la taille d'un fichier ou les caractéristiques techniques d'une photographie.

5.2. Couche sémantique ajoutée au contenu

Les traitements de texte offrent plusieurs moyens d'**ajouter au contenu des documents une couche sémantique** pour permettre une *meilleure structuration* et une *meilleure exploitation* des contenus. Nous nous attarderons tout particulièrement sur les fonctionnalités suivantes :

- *Styles*
- *Légendes et renvois*
- *Table des matières et autres tables*
- *Modèles*

Le *TP Structuration dans un traitement de texte* permettra d'explorer plusieurs de ces fonctionnalités. Bien que la mise en pratique dans ce TP se fera dans un traitement de texte en particulier, soit *Writer* de la suite bureautique *LibreOffice*, il est à noter que ces fonctionnalités se retrouvent en tout ou en partie présentes dans tous les traitements de texte. Vous retrouverez ainsi, dans les notes de cours, des exemples provenant soit de *Writer* ou de *Word*.

a) Styles

Avantages

Les **styles** dans un traitement de texte permettent d'identifier la **nature de certains contenus** pour y associer une certaine **mise en page**. Un parallèle très direct peut être fait avec les éléments HTML dans une page Web qui permettent d'identifier la nature des contenus et les règles CSS qui sont associées aux éléments HTML pour définir une certaine mise en page de ces contenus. La différence dans le traitement de texte est que la distinction "*nature des contenus*" et "*mise en page*" est moins explicite comme l'application des styles fait les deux simultanément (identifie la nature du contenu en même temps que fait la mise en page associée).

Des *avantages* très similaires à ceux évoqués pour la structuration apportée par les éléments HTML et leur mise en page en CSS justifient l'utilisation des styles dans un traitement de texte. L'utilisation des styles permet de/d' :

- **Faciliter la mise à jour** du visuel d'un document
- **Uniformiser la mise en page** des documents
- **Augmenter les possibilités de réutilisation** des contenus

Types de styles

Les traitements de texte offrent un ensemble de **styles prédéfinis** que l'on peut appliquer dans un document, comme illustré ci-dessous par quelques exemples.

Styles	Writer (LibreOffice)	Word (Microsoft Office)
Titres et sous-titres	Titre principal, sous-titre, Titre 1 ... Titre 10 (<i>styles de paragraphe</i>)	Titre, sous-titre, Titre 1 ... Titre 9 (<i>styles liés - paragraphe et caractère</i>)
Paragraphe de texte	Corps de texte (<i>styles de paragraphe</i>)	Corps de texte (<i>style lié - paragraphe et caractère</i>)
Liste à puces	Puce -, Puce •, ... (<i>styles de puce</i>)	Liste à puces (2 ... 5) (<i>styles de paragraphe</i>)
Liste numérotée	Numérotation 123, Numérotation abc, ... (<i>styles de numérotation</i>)	Liste à numéros (2 ... 5) (<i>styles de paragraphe</i>)
Accentuation	Accentuation, Accentuation forte (<i>styles de caractère</i>)	Accentuation, Accentuation légère, Accentuation intense (<i>styles de caractère</i>)

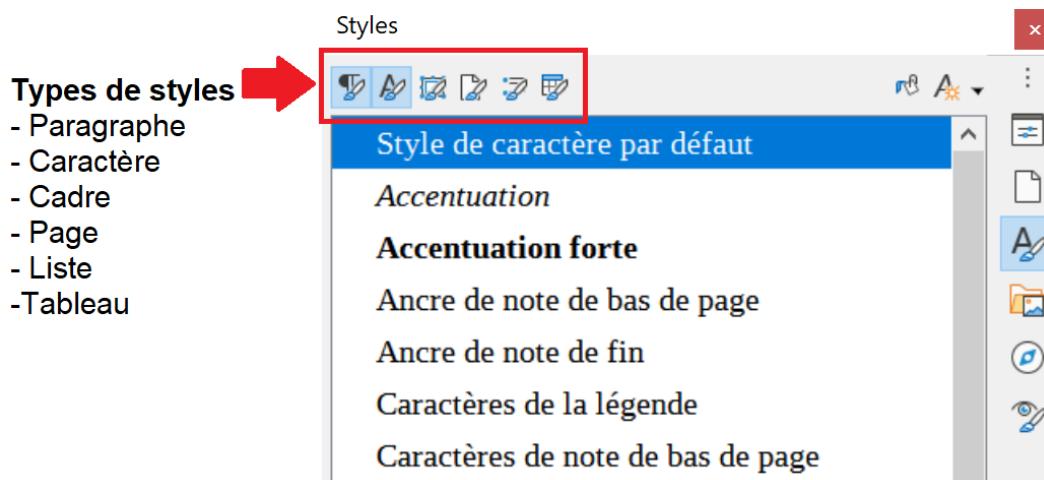
Exemples de styles par défaut dans un traitement de texte

Comme le tableau le montre, les styles peuvent appartenir à *différentes catégories* :

- Certains styles vont s'appliquer aux **paragraphes** en entier auquel cas, il ne sera pas possible d'appliquer ces styles uniquement à un mot dans un paragraphe.
- D'autres styles peuvent modifier la mise en page à un niveau plus fin soit aux **caractères**.

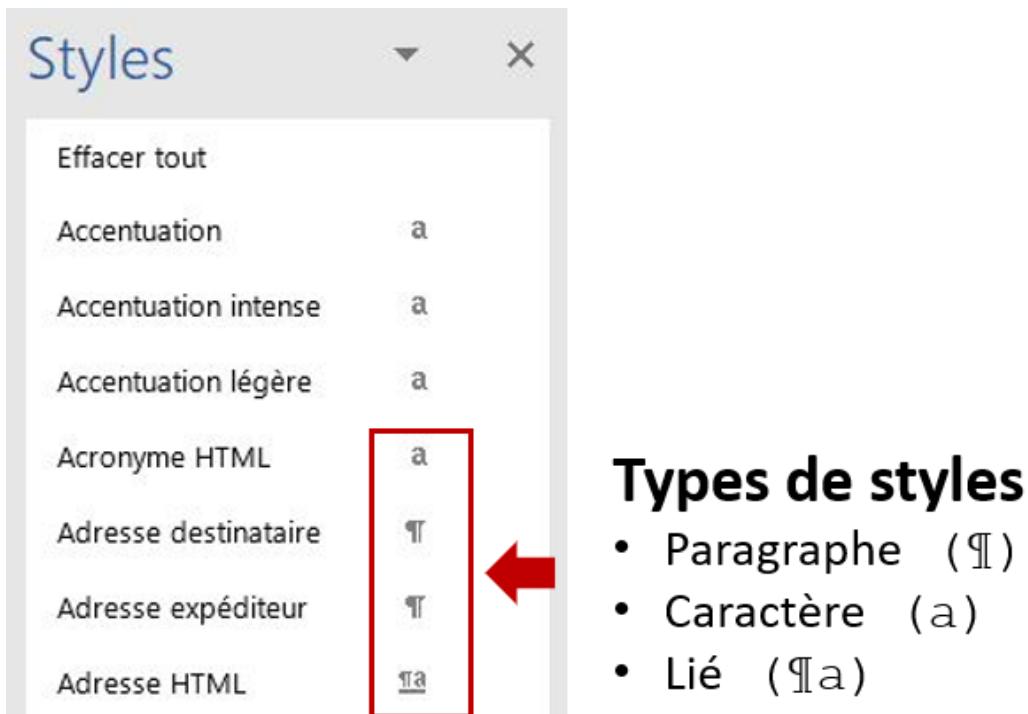
Bien que les choix faits pour les styles de base peuvent différer selon le traitement de texte utilisé, par exemple dans *Word* il est possible d'appliquer le style *Titre 1* à un seul mot dans une phrase ce que *Writer* ne permettra pas, cette notion de styles qui peuvent s'appliquer soit à des paragraphes, soit à des caractères, demeure présente. Il est ainsi utile de s'assurer de bien en comprendre la différence pour bien les appliquer.

L'interface de la zone de styles dans *Writer* permet de facilement voir le type de styles comme les **styles sont regroupés par type dans des onglets distincts** :



Onglets des styles dans *Writer* (LibreOffice)

Dans *Word*, dans la fenêtre pour les styles, c'est l'icône à la droite du nom d'un style qui nous informe de son type :



Fenêtre des styles dans *Word* (Microsoft Office)

Opérations possibles sur les styles

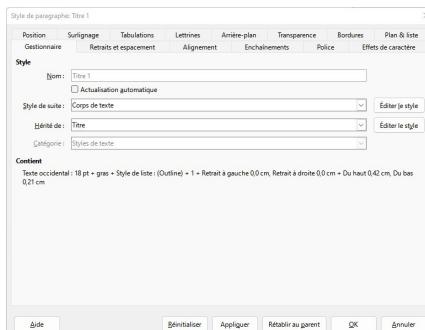
Plusieurs **opérations** sont possibles avec les styles :

- On peut les **appliquer à du contenu** en sélectionnant le contenu et en cliquant sur le style désiré.
- Il est possible de **modifier le visuel d'un style existant**. Les manières d'y parvenir peuvent varier selon le traitement de texte utilisé, mais habituellement deux manières principales sont proposées :
 1. par le truchement d'une *fenêtre d'édition* des caractéristiques du style
 2. en *appliquant le visuel que vous avez défini* dans votre document à un style existant pour le modifier.

- On peut ajouter un **nouveau style**. Cela permet ainsi de personnaliser les styles associés à un document.
- Il est possible d'**importer des styles** à partir d'un autre document.

Caractéristiques des styles

L'observation de la **fenêtre d'édition des styles** de *Writer* (*LibreOffice*) permet d'identifier plusieurs caractéristiques des styles.

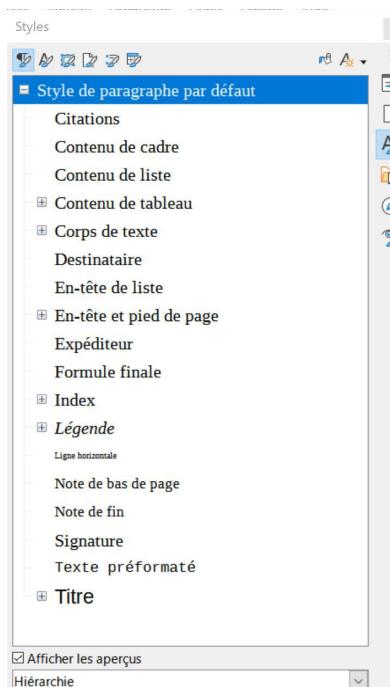


Fenêtre d'édition des styles dans *Writer* (*LibreOffice*)

Ainsi un style aura, en sus d'un *nom significatif*:

- Une **catégorie** (type de style) comme *styles de paragraphe*
- Un ensemble de **caractéristiques de mise en page** (alignement, police, etc.)
- Dans certains cas, un **style "parent"** sur lequel il se base (il *hérite* de ses caractéristiques). Cette notion d'héritage est importante à comprendre. En effet, si vous modifiez les caractéristiques de certains styles, d'autres styles pourraient être affectés.
- Dans certains cas (pour les styles de paragraphe), l'identification du **style de ce qui le suivra**. Cela peut augmenter l'efficacité de la saisie. Par exemple, pour un style de titre, lorsque l'on fait un retour de ligne après sa saisie, le traitement de texte se mettra automatiquement en style *Corps de texte*.

On remarque, dans *Writer*, le bouton *Éditer le style* à la droite du style de suite et du style parent qui permet un accès facile en édition à ces styles. De plus, dans la fenêtre où s'affiche la liste des styles, les liens de "parenté" sont visibles par la hiérarchisation des styles (mode *Hiérarchie* illustré ci-dessous). On peut ainsi facilement identifier les styles de base qui sont à la source de plusieurs styles.



Fenêtre d'affichage de la liste des styles dans *Writer* (*LibreOffice*)

b) Légendes et renvois

Avantages

L'ajout de légendes pour identifier certains types d'objet dans un document (comme des images ou des tableaux) ainsi que de renvois dynamiques à ces légendes représente des avantages sur deux plans :

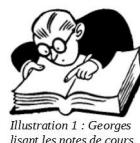
- **Sur le plan de la rédaction** : L'ajout de légendes et de renvois systématise la *bonne pratique*, d'une part, de toujours intituler un objet inséré dans un texte et, d'autre part, de prendre soin de faire un renvoi dans le texte aux objets insérés.
- **Sur le plan de l'efficacité** : Leur *caractère dynamique* (par exemple, les légendes sont numérotées automatiquement et le renvoi à la page d'une légende se met à jour automatiquement si celle-ci change de place) représente un gain net en efficacité. Il n'est en effet plus nécessaire, si on déplace une figure, de mettre à jour manuellement sa numérotation ou le numéro de sa page dans un renvoi.

i) Légendes

Il est possible d'associer à *différents types d'objet* (des images, des dessins, des tableaux par exemple) une **légende** permettant :

- D'identifier le **type d'objet** (par exemple, illustration, figure, tableau)
- De **numéroter automatiquement** ce type d'objet
- De **décrire** l'objet

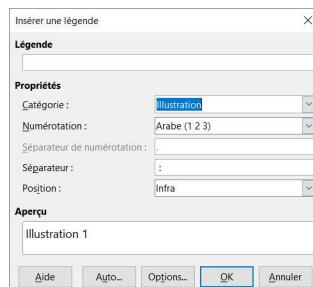
Dans l'exemple ci-dessous, la légende "*Illustration 1 : Georges lisant les notes de cours*" a été ajoutée à l'image pour préciser qu'il s'agit d'une illustration. Comme il s'agit de la première illustration dans le document, le numéro 1 lui a été associé.



Légende associée à une image dans Writer (LibreOffice)

Ces légendes représentent de **bonnes pratiques** sur le plan de la **rédaction**, comme il est toujours souhaitable de clairement identifier les objets insérés dans un document en précisant le type (illustration, figure, tableau, ...) ainsi que la nature du contenu (titre). Il s'agit aussi d'une bonne pratique sur le plan de l'**efficacité** du fait d'une **numérotation automatique**; dès qu'un document est volumineux et comporte plusieurs tableaux et figures, le gain en efficacité est rapidement visible comme vous n'avez pas vous-mêmes à la numérotter! De plus, comme nous le verrons dans une section subséquente, c'est le fait d'insérer des légendes qui permet d'*ajouter automatiquement une liste des illustrations* dans un document.

Lorsque l'on procède à l'ajout d'une légende, les traitements de texte permettent habituellement de choisir parmi des **catégories existantes** (par exemple, Figure, Tableau) et même d'en **créer de nouvelles**. Comme illustré ci-dessous, dans *Writer*, il est aussi possible de choisir le **type de numérotation** désiré ainsi que de **positionner** la légende par rapport à l'objet auquel elle est associée.



Fenêtre d'ajout d'une légende dans Writer (LibreOffice)

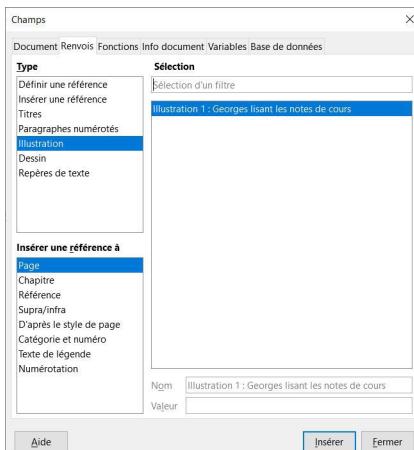
ii) Renvois

Un avantage supplémentaire des **légendes** vient du fait qu'en marquant ainsi un objet dans un document, il est possible d'en exploiter la **position** dans un **renvoi**. Les légendes sont en quelque sorte des signets! On peut les exploiter en insérant un renvoi dynamique dans le texte comme, par exemple " (voir l'illustration 1 en page 2)".

Les renvois sont ainsi de bonnes pratiques de **réécriture**. Un objet inséré dans un document doit habituellement être explicitement rattaché au texte en lui faisant référence par exemple par son numéro (Illustration 1) et/ou sa page (p. 2). L'insertion de **renvois dynamiques** rejoint ainsi ces bonnes pratiques, mais aussi en augmentant l'**efficacité** comme le numéro de l'objet et sa page sont dynamiques (c'est-à-dire que si on déplace une illustration et que son numéro et sa page changent, le renvoi sera mis à jour dynamiquement).

Lors de l'**insertion d'un renvoi**, il faut, d'une part, **choisir l'élément que l'on veut pointer**. Comme on peut le remarquer dans la copie d'écran ci-dessous, non seulement il est possible de sélectionner des *légendes* (illustration et dessin), mais aussi d'*autres types d'éléments* dans un document (titres et paragraphes numérotés par exemple). L'idée à retenir est que du fait d'avoir "marqué" de différentes manières des éléments d'un document, par exemple avec une légende ou avec un style de titre, ces éléments deviennent "repérables" et peuvent être exploités entre autres pour faire des renvois.

Une fois l'élément choisi, **plusieurs types d'information peuvent être inclus** dans la mention de renvoi comme la *page* ou la *catégorie* et le *numéro*.



Insertion d'un renvoi dans Writer (LibreOffice)

c) Table des matières et autres tables ou index

Avantages

Certaines des **couches sémantiques ajoutées au contenu** des documents (par exemple, en ajoutant des légendes ou en utilisant des styles de titre), en offrant une certaine prise sur ces contenus, permettent la **création** et la **mise à jour automatique** de différents types de **tables** ou **index** (tables des matières, tables des illustrations, etc.). Ces tables et index ajoutent à la lisibilité des documents en offrant différents **outils de navigation** dans leur contenu.

i) Table des matières

L'utilisation des **styles de titres et de sous-titres** permet l'insertion dans un document d'une **table des matières**. Cette table des matières est **dynamique**, c'est-à-dire qu'il est possible de la mettre à jour pour prendre en considération les modifications de la structure d'un document. Cette mise à jour peut souvent se faire à partir du *menu contextuel* de la table des matières.

Les traitements de texte offrent un certain contrôle sur la table des matières générée, notamment le choix du *nombre de niveaux* à inclure. La mise en page peut varier sur certains points, mais les tables des matières se présenteront sous *forme hiérarchique*, comme illustré ci-dessous.

Table des matières	
De quel bois on se chauffe : une étude sur la bonne et due forme	a
Table des matières	a
Table des figures.....	a
Résumé	1
Introduction : le sacrifice des arbres.....	1
Développement.....	1
Le sacrifice des bras	1
Le sacrifice des yeux	2
Conclusion : sacrifices plutôt le gaspillage!	2
Sources consultées.....	2

Table des matières dans Word (Microsoft Office)

ii) Table des illustrations

C'est l'insertion de **légendes** qui permettra d'ajouter une **table (index) des figures ou des tableaux** dans un document. La liste présentera le numéro de la figure ou du tableau ainsi que son titre et, finalement, la page où l'élément apparaît. Cette liste est **dynamique** et pourra être mise à jour, souvent à partir de son menu contextuel, pour tenir compte des modifications du document.

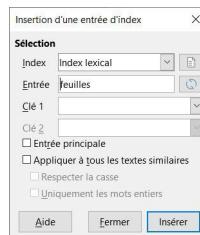
Table des figures	
Figure 1: Pile des travaux.....	2
Figure 2: Fatigue oculaire selon la distance (en km) parcourue par l'œil.....	2

Index des figures dans Writer (LibreOffice)

iii) Index

Il est aussi possible de **marquer certains mots ou expressions dans un texte** pour pouvoir ensuite produire automatiquement un **index**. Ce processus est ainsi fait en deux temps :

- Il faut premièrement *identifier ce que l'on veut mettre dans l'index*. Ce faisant, la fenêtre d'ajout d'une entrée offre des possibilités qui peuvent varier selon le traitement de texte. Dans *Word*, par exemple, il est possible de faire des renvois. Dans *Writer*, on peut préciser s'il s'agit ou non d'une entrée principale, hiérarchiser les entrées et demander d'indexer toutes les expressions similaires.



Définition d'une entrée d'index lexical dans Writer (LibreOffice)

- Dans un deuxième temps, on peut *insérer dans le document un index* qui identifiera les entrées d'index ajoutées dans le texte pour générer automatiquement l'index. Cet index pourra être mis à jour au fur et à mesure de la rédaction du document.

Index lexical	
consommation de papier.....	1
regard écologique.....	1
interligne.....	1, 3
marges.....	1, 3
Recto Verso.....	1

Index lexical produit dans Writer (LibreOffice)

d) Modèles

Avantages

Un **modèle** dans un traitement de texte est en quelque sorte une « *coquille* » prête à l'emploi présentant la **structure de base d'un type de document** et les **règles de présentation des éléments** qui le composent. Les **avantages** à utiliser des modèles pour les principaux types de documents utilisés dans une organisation sont nombreux :

- D'une part, les modèles vont permettre de s'assurer d'une **uniformité dans les contenus présentés** ainsi que dans leur **structuration**. La mémoire organisationnelle qui y sera ainsi consignée sera ainsi plus constante d'une personne à l'autre. Leur consultation en sera ainsi facilitée.

- D'autre part, il sera possible d'assurer une **uniformité aussi dans le visuel retenu**.
- Finalement, les modèles facilitent la **prise en main** des tâches d'édition par des personnes nouvellement arrivées.

Composantes d'un modèle

Prenons, par exemple, le modèle de CV ci-dessous proposé par Word (Microsoft Office) :



Modèle de CV dans Word (Microsoft Office)

On y remarque :

- Des **titres de section** pour *structurer* le document
- Des **exemples de contenus** pour *guider* l'utilisateur lors de l'édition du document
- Un **visuel prédefini**

On retrouve dans les traitements de texte certains **modèles de base**. Par exemple, dans *Word (Microsoft Office)*, lorsque l'on demande de créer un nouveau document (*Fichier > Nouveau*), différents modèles apparaissent (*Document vierge, Lettre de motivation, Rapport, etc.*) ainsi qu'il est possible de chercher de *nouveaux modèles en ligne*. Du côté de *Writer (LibreOffice)*, c'est par l'option *Modèles de Fichier > Nouveau* que l'on peut accéder à divers modèles (*CV, Résumé, Carte de visite avec logo, etc.*).

En sus des modèles existants, il est possible de **créer des modèles de document adaptés** aux besoins d'une organisation par exemple. Les professionnelles et professionnels de l'information sont bien placés pour créer ces modèles en raison de leur expertise sur le plan de la *structuration de l'information* et de leur *rôle conseil dans la mise en place de bonnes pratiques de création de documents*.

Comme vous l'explorerez dans le cadre du *TP Structuration dans un traitement de texte*, le développement d'un modèle de document dans un traitement de texte implique deux étapes principales :

1. la **modélisation** du type de document
2. l'**implantation** de la modélisation dans un modèle de document dans un **traitement de texte**.

Cette deuxième étape amène à exploiter toutes les **fonctionnalités** d'un traitement de texte nécessaires et utiles pour bien exploiter la structure du document vues précédemment (*styles, légendes, table des matières, etc.*).

5.3. Couche sémantique présente dans le contenu : Expressions régulières

On ne peut parler de structuration de l'information dans un traitement de texte sans mentionner, à tout le moins rapidement, une autre fonctionnalité souvent méconnue, soit la possibilité de faire des recherches et des remplacements en exploitant des **expressions régulières** (appelées aussi *expressions rationnelles*). Les expressions régulières permettent de chercher des *chaînes de caractères possédant certaines caractéristiques structurelles*.

Il ne s'agit donc pas ici d'ajouter une couche sémantique à des contenus, mais plutôt d'**exploiter certaines régularités dans un texte**, qui sont habituellement des "*marqueurs sémantiques*", pour faire des recherches. Commençons par un exemple simple, soit la liste des noms des membres de l'équipe d'*ABC courtage informationnel* ci-dessous.



Courtiers et courtières en information d'*ABC courtage informationnel*

Barnett, Tyqueria

Bonner, Ceyonne

Carson, Cheakina

Carter, Zeva

Drapeau, Alucia

Dubé, Victorin

Erika, Nagaoka

Etsuko, Kamoto

Frazier, Kashayla

Gaudet, Sénard

Hatsuka, Tano

Horton, Yahoska

Johnson, Darelle

Jones, Samantha

Juneau, Arthurien

Kiko, Nagashima

Laberge, Tommy

Lacoste, Edward

Laprise, Ophina

Loiselle, Laura

Love, Addrian

Mccoy, Jerilene

Meilleur, Aldélina

Nolin, Bersilinie

Ome, Naya

Pitre, Jacob

Rainville, Kevin

Rhéaume, Sophia

Ryu, Wakamatsu
 Santerre, Edmilda
 Savard, Kayla
 Sénécal, Roméric
 Suzu, Naru
 Sylvain, Patrick
 Tansho, Kawabata
 Toshiko, Hora
 Trudeau, Daphnee
 Vallée, Elvani
 Wakako, Nakagome
 Walton, Cynda

On remarque, dans cette liste, certains **éléments structurels explicites**, soit :

1. la *virgule* qui sépare le nom de famille du prénom
2. le *retour de ligne* à la fin de chaque nom.

Ces éléments structurels peuvent être exploités dans une recherche exploitant les expressions régulières afin d'inverser l'ordre *nom-prénom* dans la liste (par exemple, pour avoir *Cynda Walton* au lieu de *Walton, Cynda*). Lorsque l'on maîtrise les expressions régulières, ce type de transformation se fait en un seul *rechercher-replacer*!

Il est à noter que les expressions régulières ne se retrouvent pas que dans les traitements de texte. Plusieurs environnements qui permettent le traitement du texte les incorporent. Ceci dit, l'implantation des expressions régulières peut varier d'un environnement à l'autre, certains environnements les implantant en totalité et d'autres uniquement partiellement. De plus la syntaxe peut varier. Les principales expressions régulières que l'on retrouve sont les suivantes :

Fonction	Exemple	Writer (LibreOffice)		Word (Microsoft Office)	
		Syntaxe générique	Syntaxe pour l'exemple	Syntaxe générique	Syntaxe pour l'exemple

Trouver n'importe quel caractère alphanumérique (alphanumérique = lettres, chiffres, caractères spéciaux)	Pour trouver <i>crème</i> et <i>crime</i>	Le point . représente n'importe quel caractère alphanumérique	cr.me	Le point d'interrogation ? représente n'importe quel caractère alphanumérique	cr?me
Trouver un caractère dans une plage de caractères	Pour trouver les lettres entre <i>a</i> et <i>h</i>	[premier caractère-dernier caractère]	[a-h]	[premier caractère-dernier caractère]	[a-h]
Trouver un certain nombre d'instances d'un caractère alphanumérique	Pour trouver une suite d'une ou plusieurs lettres <i>e</i> (par exemple, trouver : <i>e, ee, eee, ...</i>)	Le plus + pour au moins un caractère L'astérisque * pour 0 ou plusieurs caractères	e+ (au moins un e) e* (aucun e ou plusieurs)	L'arobase @ pour au moins un caractère L'astérisque * pour 0 ou plusieurs caractères	e@ (au moins un e) e* (aucun e ou plusieurs)
Trouver un mot à la fin d'un paragraphe	Pour trouver le mot <i>information</i> à la fin d'un paragraphe	Le signe de dollar \$ permet de retrouver un mot en fin de paragraphe	information\$	Ne s'applique pas car on peut retrouver la fin d'un paragraphe sans passer par les expressions régulières en cherchant ^p.	n.a.
Mettre en mémoire la chaîne trouvée pour pouvoir la réutiliser dans le <i>Remplacer</i>	Par exemple, mettre en mémoire la lettre trouvée dans une plage de caractères	La mise entre parenthèse permet de mettre en mémoire la chaîne retrouvée. Dans le <i>remplacer</i> , le signe de dollar \$ suivi de l'ordre de mise en mémoire (on pourrait en effet mettre plus d'une chaîne de caractères en mémoire) permet	([a-h]) Pour réutiliser dans le <i>remplacer</i> : \$1	La mise entre parenthèse permet de mettre en mémoire la chaîne retrouvée. Dans le <i>remplacer</i> , la barre oblique inversée \ suivie de l'ordre de mise en mémoire (on pourrait en effet mettre plus d'une chaîne de caractères en mémoire)	([a-h]) Pour réutiliser dans le <i>remplacer</i> : \\1

		d'insérer dans le texte ce qui a été mis en mémoire		permet d'insérer dans le texte ce qui a été mis en mémoire	
Caractère d'échappement	<p>Pour chercher le caractère [sans qu'il soit interprété comme le caractère d'une expression régulière</p> <p><i>Note</i> : à utiliser pour tous les caractères exploités dans la syntaxe des expressions régulières (par exemple, pour [,], (), * , +)</p>	<p>La barre oblique inversée \</p>	\ [<p>La barre oblique inversée \</p>	\ [

Principales expressions régulières

Les différents éléments présentés ci-dessus peuvent être combinés dans une même expression de recherche pour représenter la forme du contenu recherché. Reprenons notre exemple de la liste des noms. Chacune des lignes possède une forme récurrente : *nom, prénom*[retour de ligne] qu'il est possible de rechercher ainsi dans *Writer* : (.+) , (.+) \$

- (.+) , permet de mettre en mémoire (en raison des parenthèses) la chaîne de caractères (en raison du point qui représente un caractère et du + qui précise qu'il y en a potentiellement un certain nombre) jusqu'à la virgule suivie de l'espace.
- (.+) \$ permet de mettre en mémoire la chaîne de caractères après l'espace jusqu'à la fin du paragraphe (en raison du signe de dollar).

L'inversion se fera dans la boîte de remplacement ainsi : \$2 \$1 qui indique en effet à *Writer* de retourner premièrement le deuxième élément (soit le prénom) mis en mémoire, d'ajouter ensuite une espace typographique et finalement d'ajouter le deuxième élément mis en mémoire (soit le nom).

L'exemple utilisé demeure un exemple très simple d'utilisation des expressions régulières. Ces dernières sont très puissantes pour le traitement de données. Dans le cadre du cours, vous les explorerez un peu dans le *TP Structuration dans un tableur*. Ce qui a été présenté dans les notes de cours suffiront pour faire le travail. Éventuellement, si vous avez à faire des traitements de données plus avancés, il est conseillé de les approfondir. Ce sont des outils fort utiles!

6. Ressources en lien avec le cours

Matériel de cours

- *Notes de cours* [cf.]

Note : Du matériel complémentaire est précisé dans le protocole du *TP Structuration dans un traitement de texte* en lien avec les manipulations dans *Writer* (*LibreOffice*).



Glossaire

Métadonnées

"Une métadonnée est, à proprement parler, une donnée à propos de données (le préfixe grec meta indiquant l'autoréférence). En documentation, une métadonnée est une donnée servant à définir ou décrire d'autres données (par exemple un document). La valeur sémantique d'une métadonnée est généralement décrite et définie dans un schéma de métadonnées. À titre d'exemple, une date associée à un document est une métadonnée servant à déterminer la date de publication, de dépôt légal, de copyright, de création, de dernière modification, etc. Les métadonnées peuvent être de nature descriptive, structurelle ou administrative." (Arsenault, Clément, Terminologie en sciences de l'information - EBSI / Métadonnées (Metadata)¹)

¹ <https://clip.ebsi.umontreal.ca/terminologie/sci/fr/index.php?tema=101&/metadonnee>

Index



Légendes.....	16
Métadonnées descriptives.....	9
Métadonnées techniques	9
Modèles	18
Renvois	16
Ressources en lien avec le cours	23
Styles	12



Crédits des ressources

p. 3

<http://creativecommons.org/licenses/publicdomain/4.0/fr/>, johnny_automatic

p. 5

<http://creativecommons.org/licenses/publicdomain/4.0/fr/>, maoriveros