Tagmatica
www.tagmatica.com
Documentation de TagSearch 2.x

Version du 29 décembre 2014

1. Présentation

TagSearch est un outil simple et autonome d'indexation et de recherche de documents développé par Tagmatica (voir www.tagmatica.com). Il fonctionne de manière locale sans utilisation de web-serveur, ni de navigateur. Il est fondé sur l'outil d'indexation Lucene.

TagSearch fonctionne sur Windows, Linux ou Mas-OS. Il est totalement écrit en Java donc portable. La seule chose qui change d'un système à l'autre est le script de lancement qui fait une ligne.

2. Historique des versions

TagSearch 2 fait suite à la version 1 qui ne traitait que les formats de fichiers PDF, HTML, XML et RTF, ainsi que quelques formats professionels spécifiques comme NEWSML pour absorber les flux des nouvelles de l'AFP. La version 2 ajoute la prise en compte des formats Office, c'est-à-dire Word, Excel, Power-Point et Visio. Notons que d'autres formats (plus d'une centaine de formats) sont détectés mais ils sont ignorés volontairement, il s'agit par exemple des formats des images comme JPEG.

Les détails des différentes versions figurent de manière cumulative dans l'onglet "Administration". Cliquer sur "Consulter les notes de livraison".

3. Pré-requis

TagSearch nécessite l'installation de Java en version 1.7 ou 1.8.

Il faut télécharger Java depuis:

http://www.oracle.com/technetwork/java/javase/downloads/index.html

Le JRE (Java Runtime Environment) est suffisant. Si vous avez le JDK (Java Development Kit), le logiciel va fonctionner tout aussi bien, car ce dernier incorpore le JRE. Ajoutons que la version 1.8 est la version la plus à jour. Tagsearch ne fonctionne pas avec Java-1.5 ou Java-1.6 qui sont de très vieilles versions.

Il faut que le système d'exploitation soit un système 64 bits.

4. Installation et lancement du logiciel

Il faut créer un répertoire (ou selon la terminologie Windows, un nouveau dossier). Admettons

qu'il s'appelle "TagSearch".

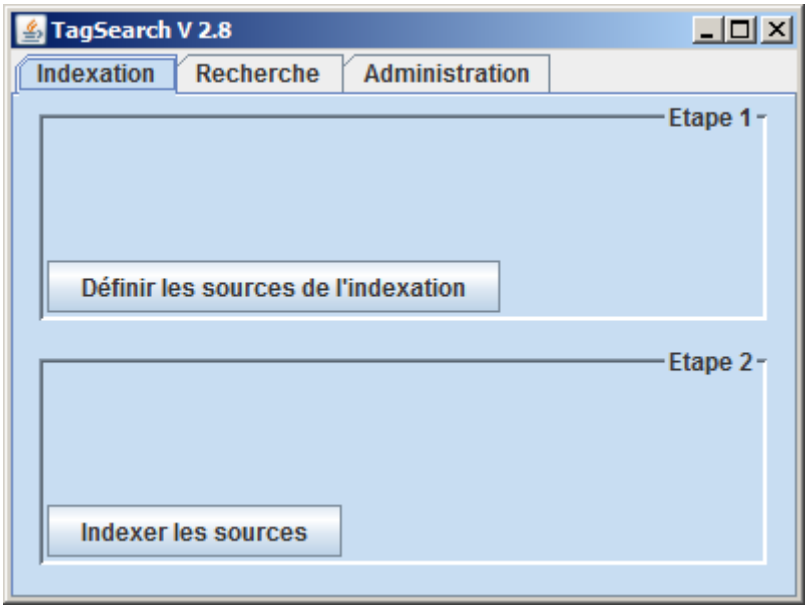
Ensuite, il faut y copier les fichiers du logiciel.

Ouvrir un terminal de commande (un "command.com" selon la terminologie Windows) et vous y positionner avec la commande "cd".

Il faut lancer le script « tagsearch ».

5. Démarrage rapide

Le présent chapitre ne couvre pas toutes les fonctionnalités possibles mais permet de démarrer l'utilisation du logiciel par un exemple simple et typique qui constitue un canevas. Le reste des fonctionnalités est en fait un ensemble d'options autour de ce canevas et il vaut mieux bien comprendre le canevas avant d'expérimenter les options.

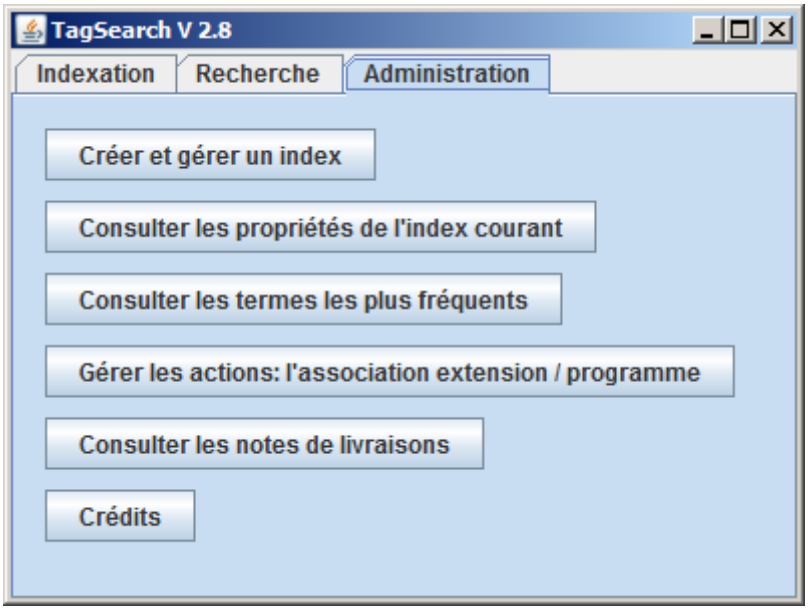


Il s'agit de réaliser les quatre opérations suivantes, dans cet ordre:

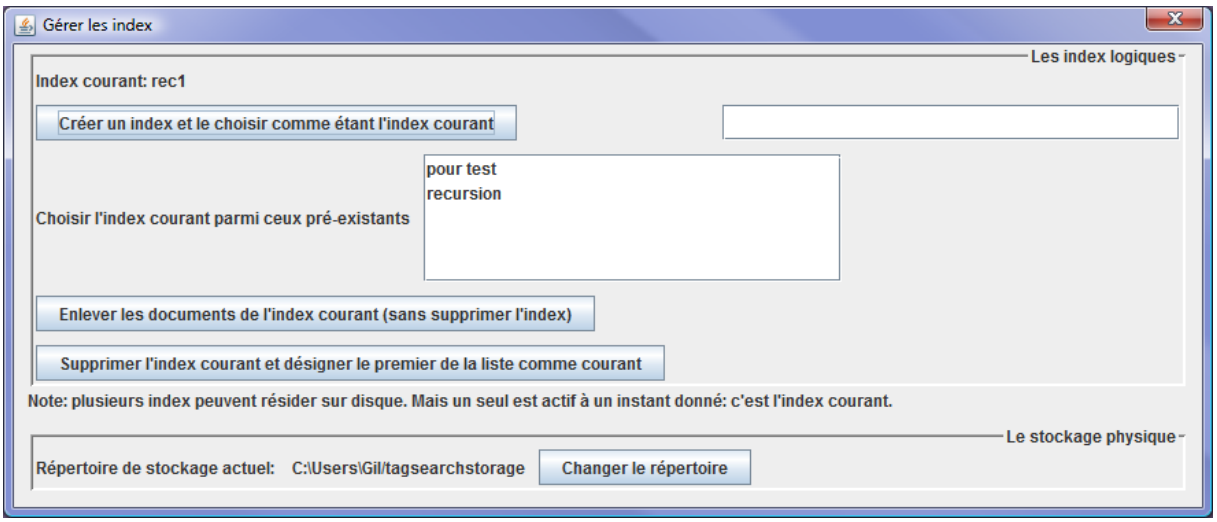
1. créer un index,
2. associer un type de fichier à une action,
3. indexer des documents,
4. rechercher ces documents.

Il faut commencer par créer un index.

Aller dans l'onglet "Administration": la fenêtre suivante apparaît:

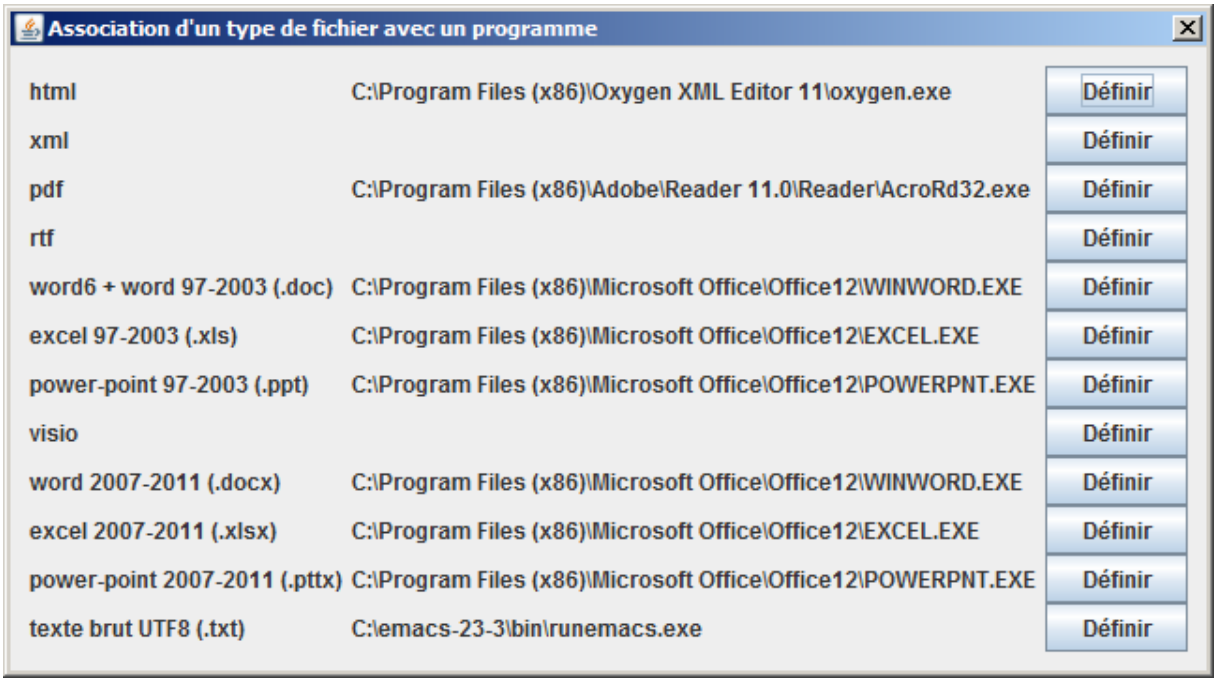


Choisir "Créer et gérer un index", ce qui donne:



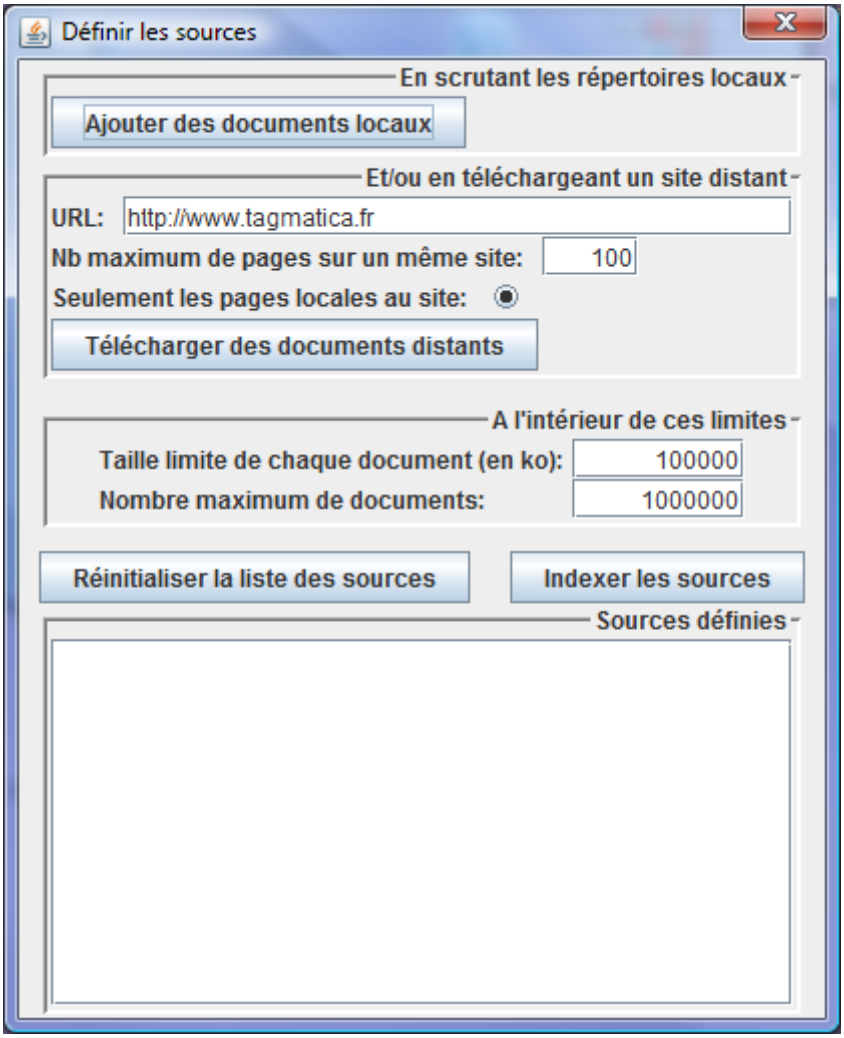
Taper un nom d'index dans le premier champ textuel de saisie. Cliquer sur "Créer un index et le choisir comme étant l'index courant". Fermer la fenêtre avec la croix en haut à droite.

Dans l'onglet "Administration", cliquer sur "Gérer les actions: l'association extension / programme" pour associer un programme à un type de fichier:



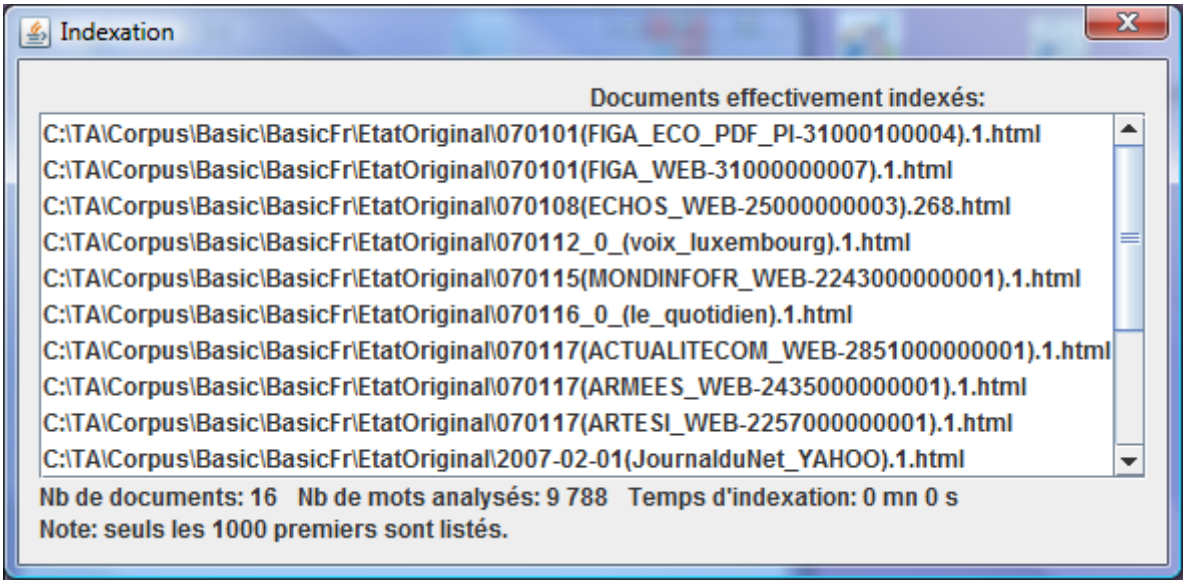
Définir quelques associations puis fermer la fenêtre.

Depuis la fenêtre principale, choisir l'onglet "Indexation". Cliquer sur "Définir les sources de l'indexation". La fenêtre suivante apparaît:



Cliquer sur "Ajouter des documents locaux" pour choisir d'indexer des documents qui sont mémorisés sur votre disque.

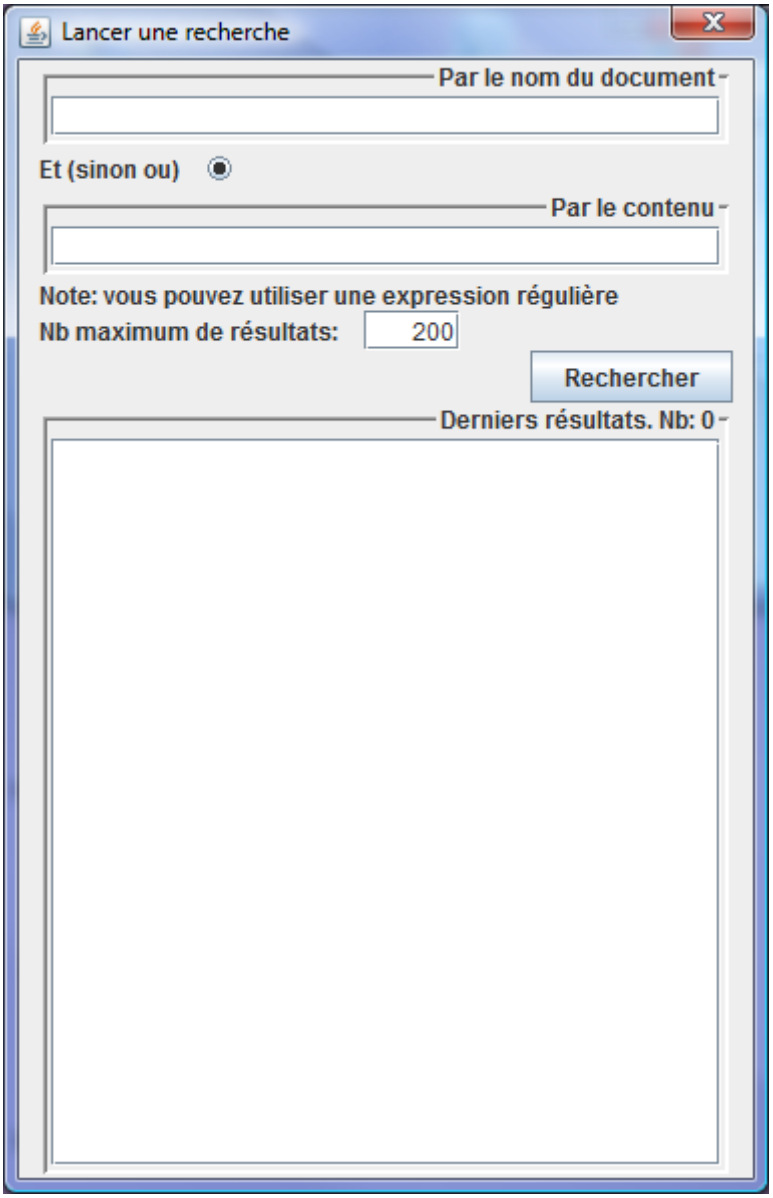
Une fois la sélection faite, cliquer sur "Indexer les sources". Selon le nombre de documents, l'indexation va prendre plus ou moins de temps. A la fin de l'indexation, une fenêtre comme celle-ci va être présentée:



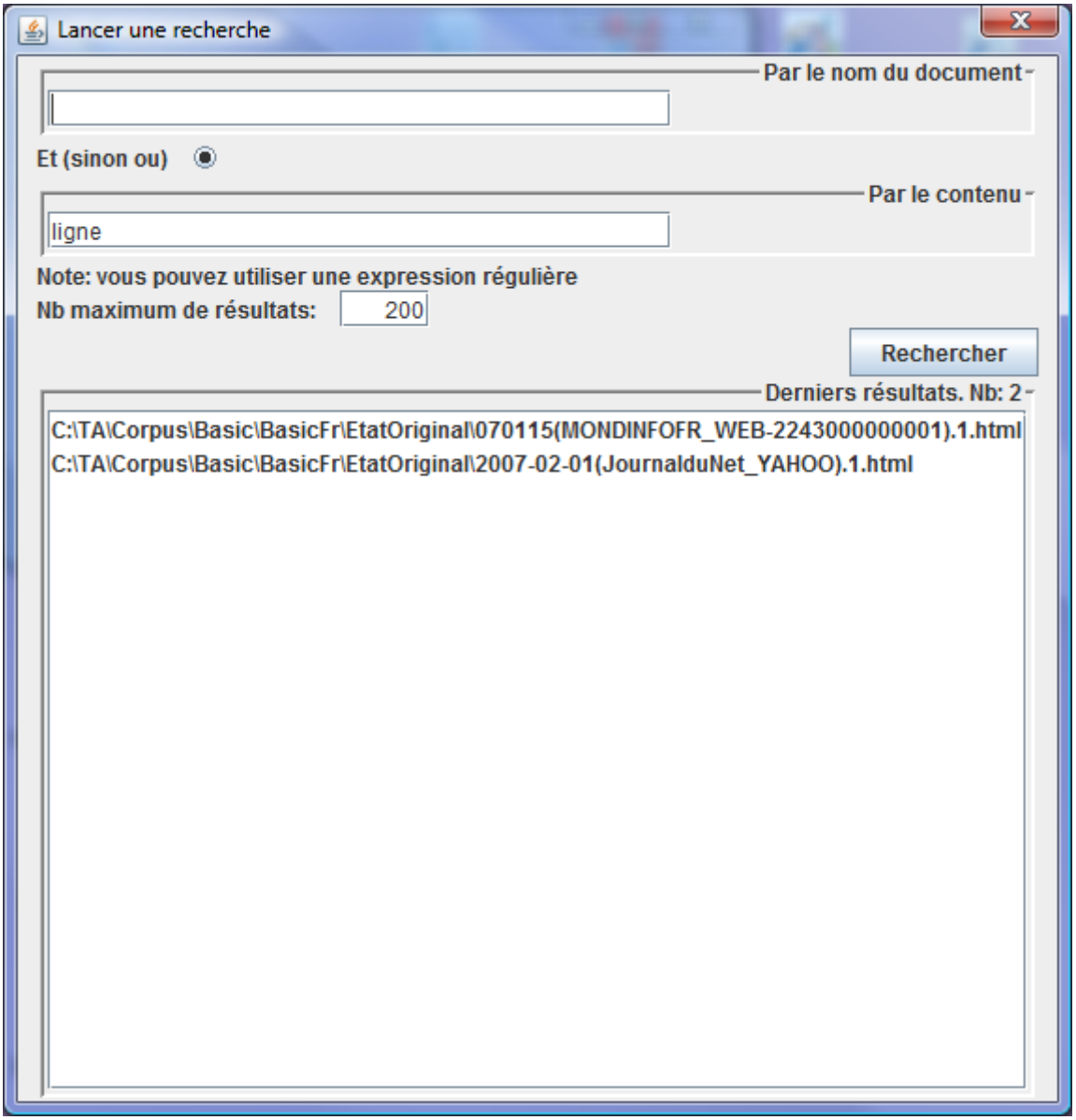
Vous pouvez double cliquer sur un nom du document pour l'ouvrir dans la mesure où l'action associée à son type a été définie dans l'administration.

Vous pouvez fermer la fenêtre.

Ensuite, choisir l'onglet "Recherche". Cliquer sur "Lancer une recherche", ce qui provoquera l'affichage suivant:



Dans le deuxième champ, saisir la valeur "ligne" par exemple pour chercher les documents qui comporte le mot "ligne" et cliquer sur le bouton "Rechercher". Le résultat aura cette forme:



Vous pouvez double cliquer sur un nom du document pour l'ouvrir dans la mesure où l'action a été définie dans l'administration.

6. syntaxe des expressions régulières

La syntaxe est identique pour les différents champs de recherche, que cela soit pour le nom du document ou son contenu. La syntaxe est celle de Lucene qui est d'un usage très fréquent dans le domaine de la recherche et indexation des documents en entreprise. La documentation complète figure sur le site de Lucene[1]. La recherche est très rapide. Ces expressions sont très puissantes sans qu'il soit nécessaire d'en connaître toutes les subtilités pour en faire un usage simple. Pour un utilisateur expert, les mécanismes flous et le favoritisme (voir plus loin) permettent de calculer une pondération qui a pour effet de présenter les résultats dans un

[1] http://lucene.apache.org/core/4_10_2/queryparser/org/apache/lucene/queryparser/classic/package-summary.html#Boolean_operators

certain ordre (i.e. ranking en anglais). En voici une description par des exemples:

expression	effet
java	contient le terme "java" strictement.
java*	contient les termes qui commencent par "java".
ja*va	contient les termes qui commencent par : "ja" suivi d'un nombre indéterminé de caractères et ensuite "va".
ja?va	contient les termes qui commencent par : "ja" suivi d'un caractère et ensuite "va".
java~	contient les termes qui sont proches de "java" comme "lava". Dans le jargon Lucene, c'est la recherche floue.
java~0.8	contient les termes qui sont proches de "java" avec une similarité de 0.8. La proximité doit être comprise entre 0 et 1. C'est une recherche floue.
java junit	contient "java" ou "junit". En d'autres termes, l'absence d'opérateur entre les termes est la disjonction logique (i.e le OU).
+java +junit	contient à la fois "java" et "junit". En d'autres termes, pour exprimer une conjonction logique (i.e. le ET), il faut l'indiquer explicitement avec le signe +.
+java -unit	contient "java" mais pas "unit".
+metho (agile extreme)	contient "metho" et doit contenir aussi "agile" et/ou "extreme". Voir le tableau suivant pour la combinaison des opérateurs.
"junit in action"	la totalité du contenu doit être "junit in action". Ce n'est pas très utile pour rechercher un contenu puisqu'il faudrait que la totalité du contenu soit égal à la chaîne de recherche, en revanche, cela permet de retrouver un document dont on connaît le nom exact.
"junit action"~5	contient 'junit" et "action" à cinq positions l'un de l'autre. Dans le jargon Lucene, c'est la recherche de proximité. Cela permet donc aussi de rechercher les suites ordonnées de termes comme "junit action"~0 dans lesquelles "junit" est suivi immédiatement de "action".
{Aida TO Carmen}	permet d'exprimer des intervalles de valeurs.
jakarta^4 apache	permet de favoriser le terme "jakarta" par rapport à "apache" dans la présentation des résultats.

Note: on ne peut pas utiliser * ou ? comme premier caractère d'une expression.

La combinaison de deux expressions autorise deux syntaxes:

syntaxe verbeuse	syntaxe courte
a AND b	+a +b
a OR b	a b
a AND NOT b	+a -b

Les caractères séparateurs entre les mots sont l'espace, la tabulation, les ponctuations, les caractères de contrôle et le caractère CR et le caractère LF. La spécification précise est celle de l'Unicode sur: http://unicode.org/reports/tr29/

7. Evolutions possibles

Point#1

Gérer des noms d'utilisateurs avec des droits et mémoriser les différentes actions comme le lancement de l'indexation et la consultation de tel fichier.

Point#2

Eventuellement, il serait possible de détecter automatiquement de quelle génération est un fichier Office (97-2003 ou 2007). Cela aurait l'avantage de simplifier la déclaration des actions mais aurait l'inconvénient que l'on ne pourrait pas affecter un programme particulier avec telle génération. A voir.

Point#3

Une fois qu'on a lancé une indexation, avoir un dispositif automatique qui rafraichit l'index en fonction des changements sur le disque sans qu'il soit nécessaire de relancer l'indexation comme actuellement.

Point#4

Eventuellement, traiter le format EML (message Outlook) pour en extraire le contenu.

Point#5

S'assurer que le logiciel n'est pas lancé deux fois sur la même machine et l'interdire explicitement, ceci pour éviter les confusions car les fenêtres vont se mélanger.

Point#6

Pouvoir consulter quand on indexe. Le système sous-jacent le permet (i.e. Lucene[2]) mais il faut le prévoir dans l'interface graphique.

Point#7

Ajouter dans les résultats, le contexte d'apparition des mots recherchés sous forme d'un KWIC (key word in context, voir wikipedia anglais) à la façon de Google.

Point#8

En plus du nom de fichier et du contenu, avoir un critère de sélection de recherche portant sur le type de fichier.

Point#9

Idem mais sur les sources pour n'indexer qu'un ensemble restreint de types de fichiers.

Point#10

Il est possible d'ajouter des fonctions linguistiques pour cibler de manière sémantique la recherche d'informations:

a) Reconnaître la langue parmi 153 langues et appliquer sur les langues européennes un anti-dictionnaire des mots ne sont pas signifiants langue par langue. Cela permettrait de pouvoir interroger sur la langue, ce qui n'est pas possible actuellement. Aujourd'hui, c'est l'anti-dictionnaire de l'anglais de l'option par défaut de Lucene qui est mis en place.

b) Effectuer une reconnaissance des entités nommées pour les noms de personnes, d'organisation, de lieux, de dates et de marques/produits en 18 langues. Ce module reconnaît des formes spécifiques (comme un prénom suivi d'un nom commençant par une majuscule ou bien une date) mais peut catégoriser plus finement des noms propres qui figurent dans une base de connaissances de grande taille. Cette dernière est issue de Wikipedia pour récolter 800 000 noms propres catégorisés selon une ontologie (voir publication[3]). On pourrait ainsi rechercher quels sont les documents qui mentionnent des noms d'organisation dont le nom comporte "Surrey" ou bien encore quels sont les documents qui mentionnent des footballeurs.

c) Il serait possible alors d'afficher les résultats avec des "rich snippets" (voir wikipedia anglais).

---

[2] Lucene permet d'ouvrir un index en lecture quand celui est ouvert en écriture.

[3] http://tagmatica.fr/publications/FrancopouloACLISOWorkshopWithinInternationalConferenceOnComputationalSemantics2011.pdf