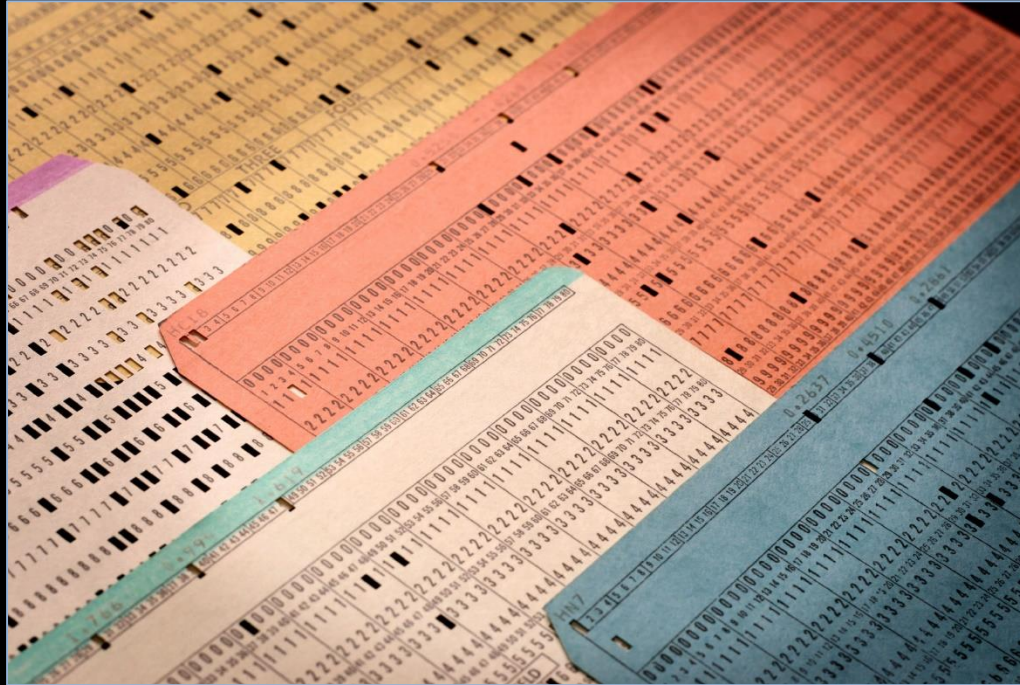


Initiation à la fouille de textes, et à la lemmatisation via TXM



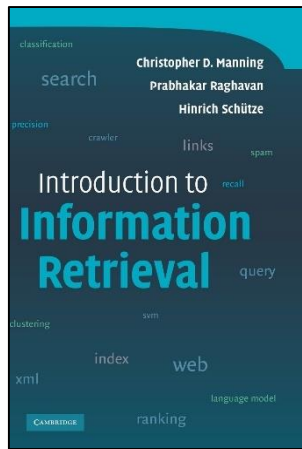
Lot de cartes perforées (c. 1960)

Objectifs de l'initiation

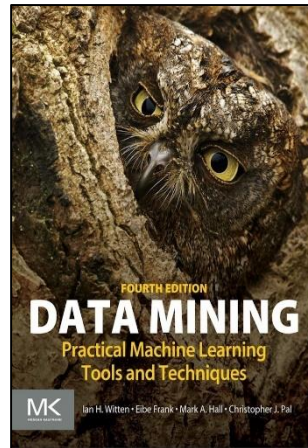
- Comprendre les **enjeux** de la fouille de données en histoire.
- Donner une **idée des méthodes** (beaucoup à inventer).
 - Présenter quelques **logiciels** / procédures.
 - **Lemmatiser** un petit corpus via TXM.
- Présentation rapide de **méthodes plus avancées**.



(2004)



(2008)



(2016)

- A. Guerneau, *Statistique pour historiens*, 2004 (en ligne).
- C. Manning ; P. Raghavan ; H. Schütze, *Introduction to Information Retrieval*, 2008.
- I. Witten ; E. Frank ; M. Hall, *Data Mining*, 2016 (4^e édition).

PLAN

I. Introduction à la fouille de textes

- I.1. Contexte général
- I.2. Qu'est-ce que le *Text Mining* ?
- I.3. Construire son corpus

II. Lemmatiser un corpus avec TXM

II. Explorer les CBMA sous TXM

- II.1. Concordances simples / complexes (CQL)
- II.2. Analyse des cooccurents
- II.3. Sous-corpus et outils de visualisation

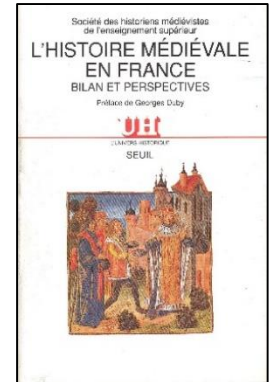
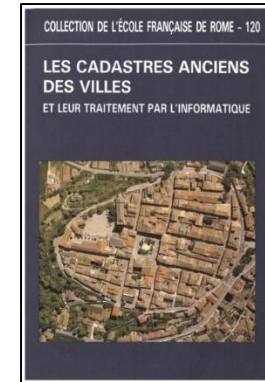
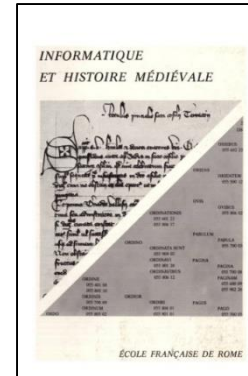
Perspectives de recherche (CWB / R)

I.1. Contexte général

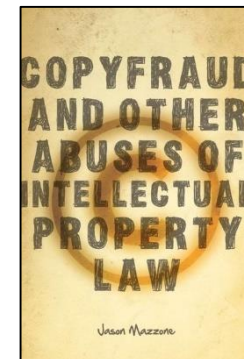
- Importance considérable prise par l'**Intelligence artificielle** et la **fouille de données** (Text mining) depuis le début du XXI^e siècle.
- Plus globalement : place de l'**électronique** et de l'**algorithmique** après la Seconde guerre mondiale (A. Turing ; C. Schanon).

- Certains **médiévistes** ont vu tôt ces changements :

(J.-P. Genet ; A. Guerreau ; Lucie Fossier ;
E. Carpentier ; P. Tombeur ;
équipe du *Médiéviste et l'ordinateur* ; etc.).



- Important : il n'y a pas de **droit d'auteur** sur les textes médiévaux
(cf. Convention de Berne ; Code de la propriété intellectuelle).
Pas de création = pas de droit d'auteur !
Voir Jason Mazzone, *Copyfraud* [...].



(1977)

(1989)

(1989)

(2006)

I.2. Qu'est-ce que le Text Mining ?

- **Spectre très large** : de la **recherche simple** d'une forme ou d'une chaîne textuelle à la **modélisation sémantique** :
 - Recherche d'information dans un corpus (*Information Retrieval*) ;
 - Analyse des cooccurrences, modélisation et représentation graphique ;
 - Apprentissages complexes (*Learning ; Topic Model ; etc.*).
- Implique des **outils** et des **formats** bien différents.
 - Il est donc important de **définir ses objectifs** : pas besoin de cadre complexe si les attentes sont simples.
- Série de **blocages** : techniques mais aujourd'hui **épistémologiques**.
 - Question fréquente : **que faire avec « mon corpus »** ?

I.3. Construire son corpus 1 : les textes

Trois situations sont envisageables pour obtenir un répertoire fichiers TXT (UTF-8).

➤ Extraction-compilation à partir de **sites préexistants** :

- Nécessite de récupérer les fichiers.
- Si corpus limité (auteur, typologie particulière) : la récupération à la main possible.
 - Sinon, création d'un script avec la commande « wget », par ex. :
wget "https://epistolae.cml.columbia.edu/letter/1.html" -O episto_0001.html

➤ Une **édition papier existe**, mais pas d'édition en mode texte :

- Utilisation d'un logiciel **OCR** (Tesseract ; Gamera ; Abbyy, etc.).
- Importance centrale de la **netteté des caractères** = qualité des **scans**.
 - Les **erreurs mécaniques** des OCR nécessitent des corrections : développement de scripts (à écrire) ou corrections à la main.
- On peut réduire les erreurs en **entraînant les logiciels** aux caractères employés.

➤ S'il n'existe pas d'édition... alors il faut **transcrire** !
(non-opposition qualitatif / quantitatif).

I.3. Construire son corpus 2 : formaliser

Partant d'un répertoire de **textes bruts (UTF-8)**, on doit obtenir un répertoire de **fichiers lemmatisés**, associés à des **métadonnées**.

Nécessite un minimum de **programmation** (Python / Perl).

Utilisation des **paramètres Omnia** pour **Treetagger** (glossaria.eu).

Omnibus Maticensis ecclesie filiis presentibus
scilicet atque futuris notum fieri dignum duximus,
quia cum preesset dominus Berardus venerabilis
presul sacrosancte Dei ecclesie in honore Sancti
Vincentii martiris dedicate, venit in capitulum
canonicorum suorum, et petiit ab eis quod erat ei de
rebus illorum necessarium, curtilum videlicet quem
tenebat Umbertus de Insula juxta muros civitatis ad
orientalem portam, et salicem de rivulo manante a
fonte Leireitana ad hoc ut viridarium ibi plantaret et
edificaret. Canonici vero, petitioni illius consentientes,
concesserunt ei tantum in vita sua tali pacto de jam ut
bene edificaret illud, [...]

M.C. Ragut (éd.), *Cartulaire de Saint-Vincent de
Mâcon*, n° 3 (1096-1124).

```
<?xml version="1.0" encoding="UTF-8"?>
<texte>
<s>
<w pos="QLF" lemma="omnis">omnibus</w>
<w pos="NAM" lemma="-">maticensis</w>
<w pos="SUB" lemma="ecclesia">ecclesie</w>
<w pos="SUB" lemma="filius">filiis</w>
<w pos="QLF" lemma="presens">presentibus</w>
<w pos="ADV" lemma="scilicet">scilicet</w>
<w pos="CON" lemma="atque">atque</w>
<w pos="QLF" lemma="futurus">futuris</w>
<w pos="VBE" lemma="nosco">notum</w>
<w pos="VBE" lemma="fio">fier</w>
<w pos="QLF" lemma="dignus">dignum</w>
<w pos="VBE" lemma="duco">duximus</w>
<w pos="PON" lemma=",">,</w>
<w pos="CON" lemma="quia">quia</w>
<w pos="CON" lemma="cum2">cum</w>
<w pos="VBE" lemma="presum">preesset</w>
<w pos="SUB" lemma="dominus">dominus</w>
<w pos="NAM" lemma="-">berardus</w>
<w pos="QLF" lemma="uenerabilis">uenerabilis</w>
<w pos="SUB" lemma="presul">presul</w>
<w pos="QLF" lemma="sacrosanctus">sacrosancte</w>
<w pos="SUB" lemma="deus">dei</w>
<w pos="SUB" lemma="ecclesia">ecclesie</w>
<w pos="PRE" lemma="in">in</w>
<w pos="SUB" lemma="honor">honore</w>
<w pos="QLF" lemma="sanctus1">sancti</w>
<w pos="NAM" lemma="-">uinentii</w>
<w pos="SUB" lemma="martur">martiris</w>
<w pos="VBE" lemma="dedico1">dedicate</w>
```

I.3. Construire son corpus 3 : les logiciels

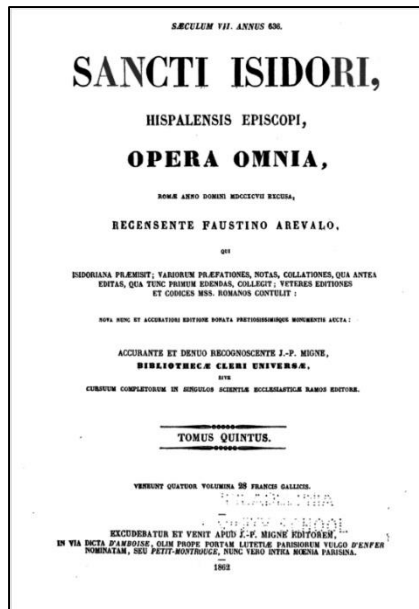
Il n'existe **pas de logiciel « ultime »**.

Voir A. Guerreau, « Textes anciens en série. Outils informatiques d'organisation et de manipulation de bases de données textuelles », *BUCEMA*, Collection CBMA, 2012, <https://journals.openedition.org/cem/12177>

- **Enquêtes simples** : Geany (éditeur de texte), AntConc.
 - **Niveau intermédiaire** : Philologic.
 - **Enquêtes avancées** : TXM, CWB.
- **Philologic** fonctionne bien pour alterner qualitatif et quantitatif. Reste difficile à installer (cf. TGIR HumaNuM).
- À ce jour, seuls **TXM** et **CWB** gèrent la lemmatisation de façon satisfaisante.
 - **TXM** s'installe facilement (Linux, Windows, OS X) et constitue un excellent choix pour débiter en fouille de textes.

II. Lemmatiser un corpus avec TXM

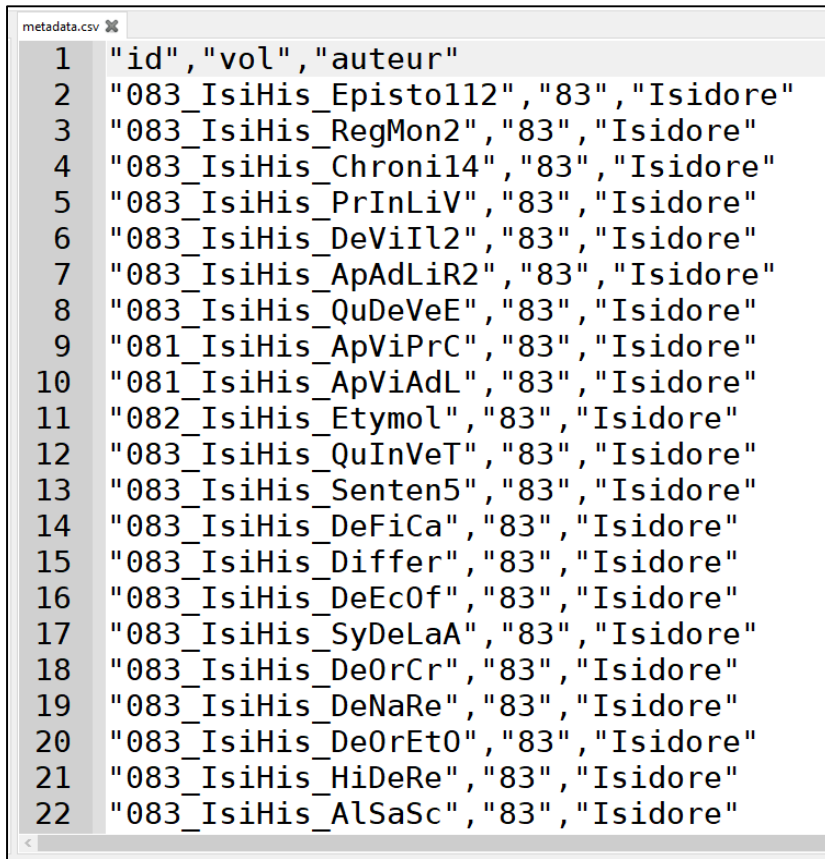
- Choix d'un **petit corpus** issu de la *Patrologie latine*
- Textes d'**Isidore de Séville** [†636] et de **Raban Maur** [†856]
(uniquement les textes attribués avec certitudes par la PL...).



PL, t. 83 (1862)

- **76 textes** au total (environ 22 Mo).
- Tomes 83, 103, 107-112 de la PL.
- En particulier le ***De Universo*** de RM et les ***Étymologies*** d'Isidore.
- Fichiers TXT « bruts » disponibles et facilement modifiables (OMNIA-Zürich)

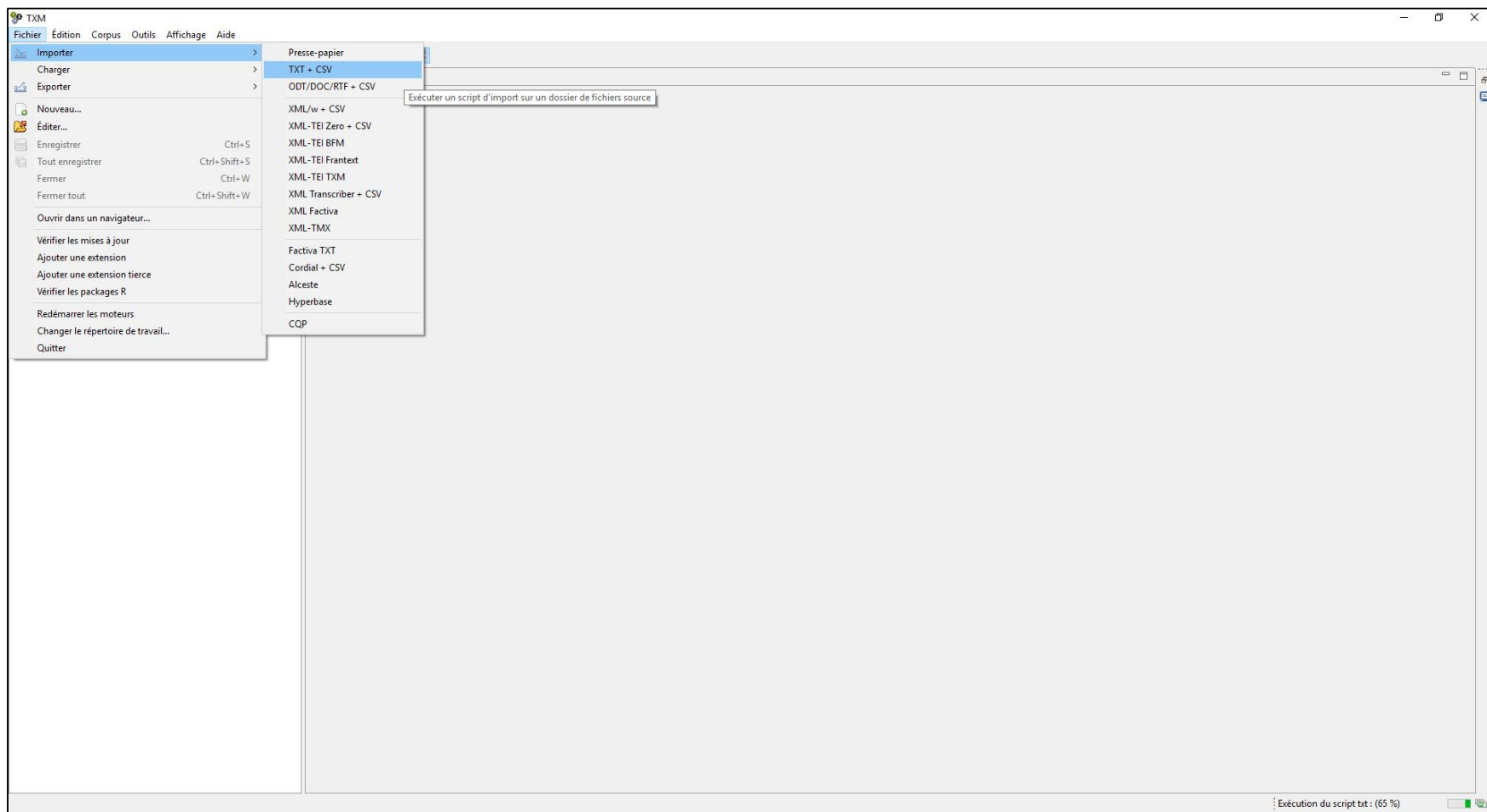
- Les corpus TXT pour TXM peuvent être accompagnées de **métadonnées (fichier .CSV)**.



```
1 "id","vol","auteur"
2 "083_IsiHis_Episto112","83","Isidore"
3 "083_IsiHis_RegMon2","83","Isidore"
4 "083_IsiHis_Chroni14","83","Isidore"
5 "083_IsiHis_PrInLiV","83","Isidore"
6 "083_IsiHis_DeViIl2","83","Isidore"
7 "083_IsiHis_ApAdLiR2","83","Isidore"
8 "083_IsiHis_QuDeVeE","83","Isidore"
9 "081_IsiHis_ApViPrC","83","Isidore"
10 "081_IsiHis_ApViAdL","83","Isidore"
11 "082_IsiHis_Etymol","83","Isidore"
12 "083_IsiHis_QuInVeT","83","Isidore"
13 "083_IsiHis_Senten5","83","Isidore"
14 "083_IsiHis_DeFiCa","83","Isidore"
15 "083_IsiHis_Differ","83","Isidore"
16 "083_IsiHis_DeEcOf","83","Isidore"
17 "083_IsiHis_SyDeLaA","83","Isidore"
18 "083_IsiHis_DeOrCr","83","Isidore"
19 "083_IsiHis_DeNaRe","83","Isidore"
20 "083_IsiHis_DeOrEt0","83","Isidore"
21 "083_IsiHis_HiDeRe","83","Isidore"
22 "083_IsiHis_AlSaSc","83","Isidore"
```

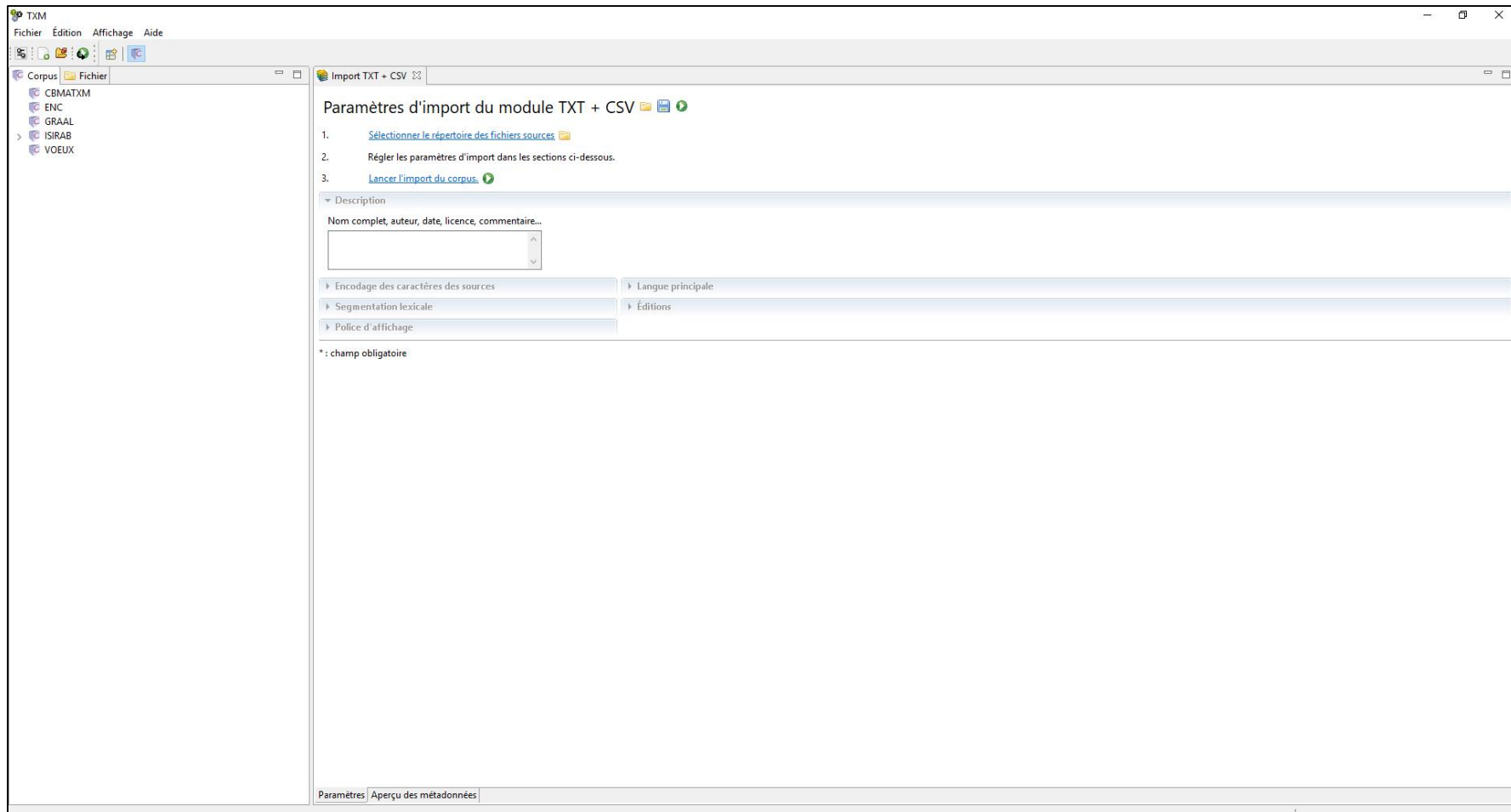
Extrait du fichier **metadata.csv**

- La structure retenue pour l'exemple est très simple :
 - Colonne « **id** » = nom du fichier (sans extension)
 - Colonne « **vol** » = volume de la PL dans lequel se trouve le texte
 - Colonne « **auteur** » = auteur (?) du texte.
 - Ces métadonnées peuvent être multipliées (ex. CBMA), du moment que l'on conserve le bon format.
 - On utilise des **guillemets simples** comme séparateurs de champs.
 - On utilise une **virgule** pour séparer les colonnes.

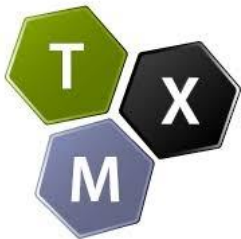


TXM peut lemmatiser les fichiers TXT « bruts » accompagnés de métadonnées, à l'aide d'une version du TreeTagger intégrée au logiciel.

... le processus est souvent très long !



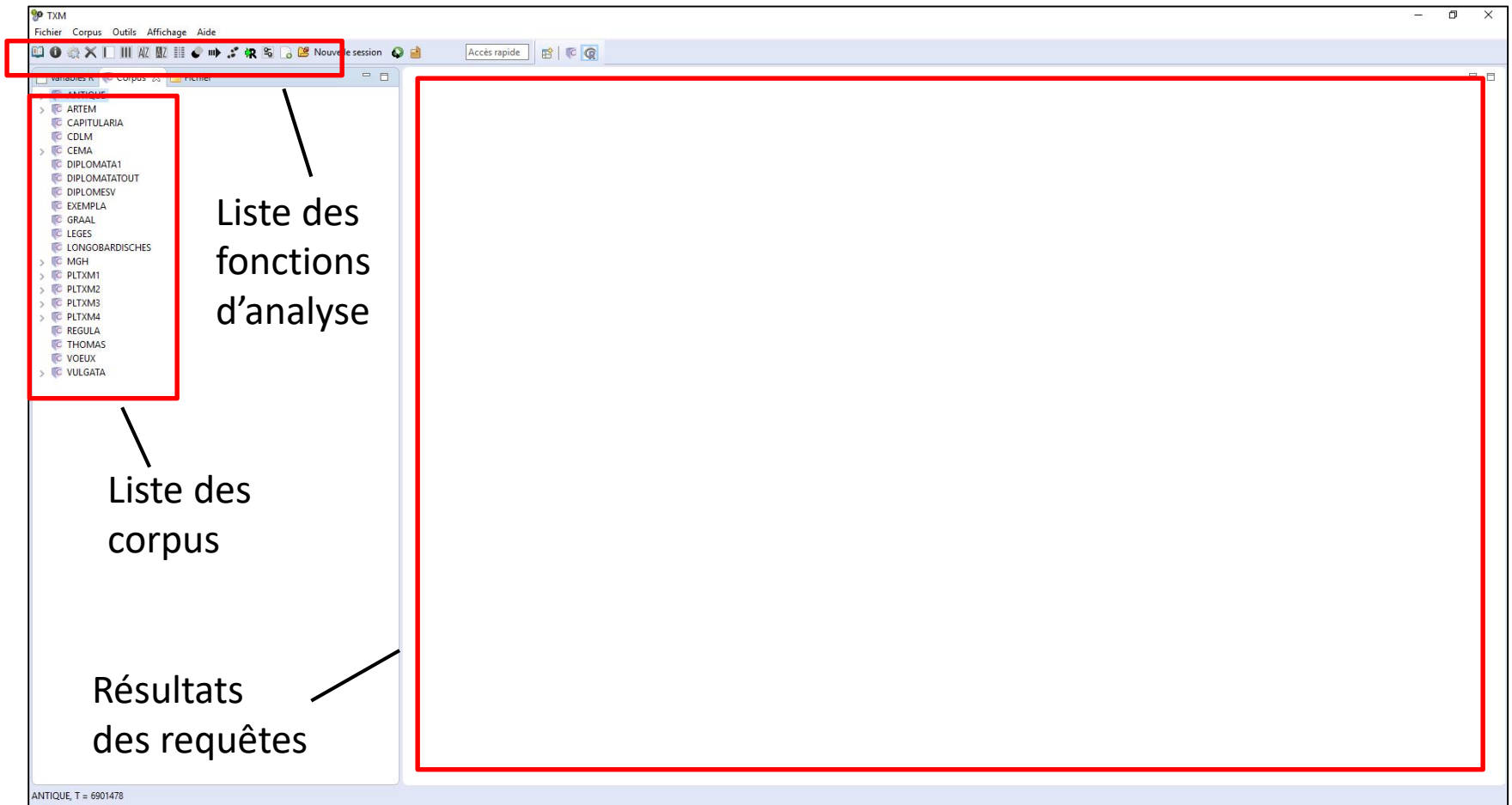
III. Explorer les CBMA sous TXM



- 1^{er} corpus diplomatique lemmatisé en accès ouvert :
<http://www.cbma-project.eu>
- Dirigé par Eliana Magnani (LaMOP).
Soutenu par l'ANR (Espachar, Charcis),
la Région Bourgogne, la MSH de Dijon, le **LaMOP** et **Cosme**.
- Herbergé par le TGIR Huma-Num (2010).
- 27 094 documents, essentiellement latins
et répartis du VI^e au XIV^e siècle,
plus de 6 millions de mots.
- Base disponible sous Philologic et TXM.

III. Explorer les CBMA sous TXM

III.1. Concordances simples / complexes

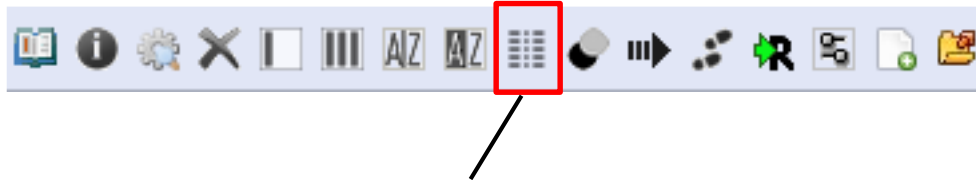


Vue de l'interface sous TXM.




Propriétés du corpus

- Donne des **statistiques** sur le corpus :
En particulier : nombre de mots (6 175 838).
- **Propriétés des unités lexicales** :
Dont une liste des POS (ici en particulier : SUB, QLF, VBE).
- **Propriétés des structures (= chartes)** :
Donne la liste des métadonnées du corpus. Pour les CBMA :
analysecbma ; analyseed ; auteurcode ; auteurlieu ;
auteurnom ; beneficiairecode ; beneficiairelieu ; beneficiaireprenom ;
date ; dateed ; diocese ; edition ; editionref ; genre ;
id ; ncbma ; tradition ; texte
- Le champ “**date**” correspond à une date à 4 chiffres
(utilisation du terminus a quo ; si aucune date = 9999).



Requête lexicale (concordance)

- TXM fonctionne en **langage CQL** (= Corpus Query Language).
- “Code” développé pour le **logiciel CQP** (= Corpus Query Processor),
à l’Université de Stuttgart : utilisé par **TXM** et **CWB**.
 - Il s’agit d’un **langage formel**,
permettant des requêtes flexibles (+ éléments regex).
http://cwb.sourceforge.net/files/CQP_Tutorial/
- Deux méthodes : l’assistant  ; entrer sa requête au clavier.
L’assistant est parfois utile si on oublie la syntaxe.
Le plus efficace est souvent d’entrer sa requête, afin d’être certain du résultat.

Requêtes lexicales simples (formes)

- Pour obtenir la liste des occurrences d'une forme
> [word="aqua"]
- Tris sur les résultats : **contextes** (gauche / droite).
(exemple *Sicut aqua extinguat...*).
 - Tris par **dates** facile, grâce à l'encodage CBMA.
- Possibilité de cliquer sur l'occurrence pour avoir accès au texte complet + métadonnées.
 - Chercher une forme tronquée :
> [word="aqu.*"]
 - Chercher plusieurs formes simultanément :
> [word="aqua|aquam|aquis"]

Requêtes lexicales simples (lemmes)

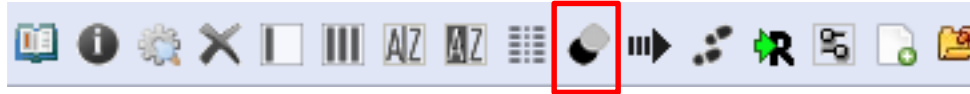
- L'approche par les formes n'est pas toujours la plus efficace.
On peut donc interroger les lemmes :
 - > [lemma="molendinum"]
- On peut aussi employer les formes tronquées :
 - > [lemma="molendi.*"]Mais encore les combinaisons :
 - > [lemma="molendinum | farinarium"]
- Il est aussi possible d'ajouter une marge d'incertitude en utilisant ".?" (ex. : > [word="mol.?endi.*"]).
- Au-delà des lemmes, on peut aussi utiliser les autres POS pour faire des enquêtes avancées :
 - > [lemma="campus"][pos="QLF"]

Requêtes lexicales complexes : expressions

- Il est bien entendu possible de combiner ces éléments :
 - > `[word="sicut"][lemma="aqua"]`
- On peut aussi signaler à TXM une distance potentielle entre les éléments de la requête :
 - > `[lemma="terra"][]{0,5}[word="com+un.*"]`
- Il est de même possible de demander deux requêtes complexes simultanément :
 - > `[lemma="aqua"][]{0,5}[lemma="silua"] | [lemma="silua"][]{0,5}[lemma="aqua"]`

http://txm.ish-lyon.cnrs.fr/bfm/files/QuickRef_CQL_BFM.pdf
http://txm.ish-lyon.cnrs.fr/bfm/files/Tutoriel_TXM_BFM_V1.pdf

III.2. Analyse des cooccurents



- La syntaxe des requêtes est toujours la même (CQL).
- Il est important de choisir “lemma” pour les résultats.
! TXM supporte assez difficilement les gros corpus
(ou les corpus avec de nombreux fichiers) !
 - D’autres paramètres peuvent être modifiés,
en particulier la distance lexicale autour du pivot
(= problème fondamental de sémantique historique).
- Il est bien entendu possible de calculer les cooccurents
pour une expression.

- Exemple autour du lemme **pater** :
> [lemma="pater"]

- L'analyse fait apparaître **5 colonnes** :

le cooccurrent ; la fréquence du cooccurrent ; la cofréquence (= cooccurrence) ;
un indice ; distance moyenne.

- **L'indice** (de spécificité) est calculé selon un algorithme donné par Pierre Lafon (« Sur la variabilité de la fréquence des formes dans un corpus », *Mots*, 1, 1980, p. 127-165).

- Le plus souvent, il est bon d'observer les résultats en **triant les indices** et les **cofréquences**. Cela offre deux visions complémentaires.
- Enfin, on peut **cliquer sur un cooccurrent** pour obtenir une concordance associant ce dernier et le pivot.

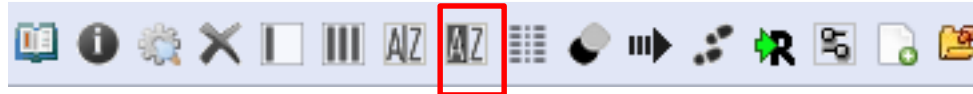
III.3. Sous-corpus et outils de visualisation

- Un des atouts de TXM est sa **flexibilité** en matière de **création de sous-corpus** (= partitions).



- Possibilité particulièrement intéressante avec les **CBMA**.
 - Ces sous-corpus sont utiles pour **comparer des résultats par périodes, par types d'auteurs (PA, EPS), etc.**
 - Une fois créés, on peut en effet y chercher des cooccurents pour un même pivot, et ainsi ébaucher une **sémantique diachronique**.
- > Ex. avec les dates ; autre exemple avec les évêques.

- Un autre atout de TXM réside dans sa capacité à générer des listes lexicales à partir d'une requête.



- Par exemple avec > [lemma="aqua"]
 - La liste obtenue permet de repérer les formes les plus importantes (« formules de pertinence »), de repérer les cas rares, mais aussi de voir certaines erreurs du lemmatiseur :
 - > [lemma="bannus"]
- Cette approche « purement fréquentielle » est très utile pour **distinguer les termes fréquents et ceux plus rares.**

- TXM met aussi à disposition d'autres outils, dont l'**analyse de progression**.



- Elle permet de contrôler l'**évolution d'une requête** dans les CBMA.
 - Par exemple : *fluvius*, *rivus*, *aqua*.
- Malheureusement, l'outil n'est **pas très lisible**.
- Cela reste une possibilité pour **comparer la dynamique de plusieurs termes** dans le corpus.