

## **Un ensemble de ressources informatisées et intégrées pour l'étude du français : FRANTEXT, TLFi, Dictionnaires de l'Académie et logiciel Stella, présentation et apprentissage de leurs exploitations<sup>1</sup>**

Pascale Bernard, Jacques Dendien, Josette Lecomte, Jean-Marie Pierrel  
contact@inalf.fr

ATILF-CNRS (Analyse et Traitement Informatique de la Langue Française)  
44 Avenue de la Libération BP30687 F-54063 Nancy-Cedex

### **Résumé**

Nous proposons de présenter quelques-unes des ressources linguistiques informatisées que le laboratoire ATILF propose sur la toile et leurs diversités d'exploitation potentielle.

Ces importantes ressources sur la langue française regroupent un ensemble de divers dictionnaires et lexiques, et de bases de données dont les plus importants sont le TLFi (Trésor de la Langue Française informatisé) et Frantext (plus de 3500 textes, dont la plupart catégorisés). Elles exploitent, pour la plupart, les fonctionnalités du logiciel Stella, qui correspond à un véritable moteur de recherche dédié aux bases textuelles s'appuyant sur une nouvelle théorie des objets textuels. Tous les spécialistes de traitement automatique de la langue ainsi que tous les linguistes, syntacticiens aussi bien que sémanticiens, stylisticiens et autres peuvent exploiter avec bonheur les possibilités offertes par Stella sur le TLFi et autres ressources offertes par l'ATILF.

Ces recherches peuvent s'articuler autour des axes suivants : études en vue de repérer des cooccurrences et collocations, extraction de sous-lexiques, études morphologiques, études de syntaxe locale, études de sémantique, études de stylistique, etc.

Nous proposons de démystifier le maniement des requêtes sur le TLFi, FRANTEXT et nos autres ressources à l'aide du logiciel Stella, et d'expliquer et de montrer comment interroger au mieux ces ressources et utiliser l'hyper-navigation mise en place entre ces ressources pour en tirer les meilleurs bénéfices.

### **Mots Clés**

Ressources linguistiques, corpus, dictionnaires, lexiques, Frantext, TLFi, Stella

---

<sup>1</sup> Ce support de tutorial est inspiré (pour FRANTEXT) de celui proposé par l'URFIST de Bordeaux, site <http://www.montesquieu.u-bordeaux.fr/urfist/niveau2/frantext/frantext0.htm>

## **1. Présentation des ressources textuelles**

### **1.1. La base Frantext**

La base Frantext (1992) peut être définie comme l'association d'une part d'un vaste corpus de textes littéraires français, et d'autre part d'un logiciel offrant une interface Web avec des possibilités d'interrogation de consultation et d'hyper-navigation. Historiquement, le but premier de ce corpus textuel était de permettre la constitution d'une base d'exemples destinée aux rédacteurs des articles du TLF.

La base Frantext contient 3417 textes appartenant aux domaines des sciences, des arts, de la littérature, des techniques, qui couvrent 5 siècles de littérature (Du 16<sup>e</sup> au 20<sup>e</sup>). Elle est accessible sur Internet, moyennant un abonnement ou des accords spéciaux. En cette année 2002, il existe une sous-base constituée de 30 textes de Victor Hugo, qui est en accès libre sur le Web. De même, une sous-base constituée des textes de l'Agrégation. Deux versions de Frantext sont proposées :

- L'intégralité de la base (3417 textes, environ 209 millions d'occurrences, environ un millier d'auteurs). Les œuvres se répartissent pour 80% d'œuvres littéraires et 20% d'œuvres scientifiques ou techniques.

- Une sous-partie constituée de 1940 textes, soit environ 127 millions d'occurrences, qui ont fait l'objet d'un codage grammatical selon les Parties du Discours. Aux fonctionnalités du Frantext intégral, ont été ajoutées des possibilités de requêtes portant sur les codes grammaticaux.

Pour plus de renseignements se connecter au site : <http://www.inalf.fr/frantext> si vous êtes abonnés, ou sinon, au site <http://www.inalf.fr/atilf> , qui est le site d'accueil du laboratoire.

### **1.2. Le Trésor de la Langue Française informatisé**

Le TLFi (Dendien, 1996) est la suite logique du TLF (CNRS, 1976-1994) lequel avait démarré dans les années 60 avec Frantext (projet qui, à l'époque, lui était associé). Le TLFi peut être vu comme une base de données lexicales en même temps qu'une base de connaissances finement structurée. Son originalité est fondée sur des spécificités liées à :

#### **Son contenu :**

- Sa nomenclature qui est riche d'environ 100000 entrées (vedettes ou sous-vedettes) toutes présentes dans nos fonds et dictionnaires. Original aussi le traitement de morphèmes grammaticaux au même titre que les autres mots
- Sa liste d'objets métatextuels sur lesquels repose la structuration de l'article du dictionnaire : vedettes, codes grammaticaux, définitions, indicateurs stylistiques et/ou sémantiques, constructions , exemples etc.. En tout environ 40 objets métatextuels.
- Sa grande quantité d'exemples référencés : environ 430000.
- La grande diversité des rubriques proposées, concernant l'étymologie, la prononciation, la bibliographie etc.

#### **Sa structure :**

- Le dictionnaire a été balisé en XML et des balises spéciales ont été injectées pour pouvoir délimiter et atteindre chaque type d'objet textuel. Ainsi, si la recherche plein

texte sur la totalité du contenu du dictionnaire reste toujours permise, il est possible aussi d'éliminer le « bruit » inhérent à un tel type de requêtes. Il est en effet possible de restreindre cette recherche plein texte à une recherche plus « pointue » sur un type particulier d'objet textuel.

- De plus, une seconde dimension a été introduite : ces objets textuels ont été hiérarchisés, à l'aide d'une série de balises spécifiques et d'une grammaire de contrôle.

De par le traitement hautement élaboré qui a permis de décomposer le texte du TLF en objets élémentaires (définitions, exemples, indicateurs de domaine technique, indicateurs sémantiques, grammaticaux, stylistiques, etc.) et d'analyser les relations hiérarchiques liant ces objets, il résulte des possibilités d'interrogation virtuellement illimitées.

### **1.3. Les Dictionnaires de l'Académie**

Il existe à l'ATILF, parmi les Dictionnaires anciens, 5 versions des Dictionnaires de l'Académie Française. Trois ont été établies en coopération avec ARTFL (Académie 1<sup>e</sup> édition 1694, Académie 5<sup>e</sup> édition 1798, Académie 6<sup>e</sup> édition 1835). Les deux autres (Académie 8<sup>e</sup> édition 1932-35, et Académie 9<sup>e</sup> édition –volume paru en 1992) se trouvent à la fois sur le site de l'Académie Française et sur le site de l'ATILF <http://www.inalf.fr/academie8> et <http://www.inalf.fr/academie9>. Ces deux dernières ont été traitées et préparées à l'ATILF, et font partie des bases accessibles via le logiciel Stella. Par exemple, la huitième édition a été numérisée à partir de l'original, puis relue, puis balisée, le codage ayant suivi les spécifications élaborées au cours de l'Atelier de balisage de Limoges (Leroy-Turcan, DictA1998). Pour de plus amples renseignements, se reporter au site de l'ATILF/Dictionnaires anciens.

Ces importantes ressources sur la langue française exploitent les fonctionnalités du logiciel Stella, qui correspond à un véritable moteur de recherche dédié aux bases textuelles. Le logiciel Stella propose des formulaires de requêtes, et une riche aide en ligne pour accompagner l'utilisateur tout au long de son parcours dans les bases, et pour l'aider dans son hyper-navigation à travers ces richesses. (Pour tirer le meilleur parti de nos ressources, l'utilisateur est censé connaître les principes de base du World Wide Web ainsi que les spécificités du navigateur qu'il utilise. Il est donc inutile de chercher dans l'aide en ligne les informations relevant de ces connaissances)

## **2. Le Logiciel Stella**

### **2.1. Présentation générale :**

Stella (Dendien, 1991 ; Bernard et al. 2002) est le logiciel qui anime la base de données Frantext, le TLFi, ainsi que les versions informatisées de deux des éditions des Dictionnaires de l'Académie Française ( 8<sup>e</sup> édition de 1935 et 9<sup>e</sup> édition (en cours)). Ce logiciel qui a été intégralement développé à l'ATILF a longtemps travaillé sur un balisage « maison ». Il a évolué et consomme (et produit) maintenant du XML. Il peut s'appliquer à tout ensemble de données textuelles, structurées ou non. Le logiciel Stella se présente comme une boîte à outils (C++) comportant différents volets :

- Des utilitaires divers incluant tris, traitement des expressions régulières, et surtout une base de données fondée sur la nomenclature du TLF permettant des opérations de flexion ou de lemmatisation.
- Une interface Web permettant la mise en œuvre facile d'interfaces utilisateur, des fonctions de gestion de "sessions utilisateur" et de gestion d'espace de travail sur le serveur.
- Une solution permettant une hyper-navigation entre les différentes applications gérées par Stella, qu'elles résident ou non sur un même serveur. L'hyper-navigation permet une liaison dynamique entre les différentes ressources textuelles : une même requête via une des ressources peut ainsi être complétée par l'hyper-navigation dans les autres ressources ; il est ainsi possible, en cliquant sur n'importe quel mot d'une page affichée par l'une des bases gérées par Stella de déclencher l'apparition d'un menu déroulant proposant de le rechercher dans n'importe laquelle des autres bases.
- Un système complet de gestion de base textuelle assurant à la fois les fonctions de stockage et d'accès à l'information.

## **2.2. Les spécificités de Stella**

La conception de Stella, en tant que logiciel de gestion de base textuelle, repose sur des principes mathématiques rigoureux (Dendien, 1991) :

### ***2.2.1. Fonctions de "bas niveau"***

Ces fonctions permettent de fabriquer et de gérer les structures de stockage de l'information. Elles reposent sur un système d'indexation garantissant un stockage optimal des données en terme d'encombrement. Ce sont elles qui permettent de localiser rapidement les occurrences d'un objet textuel élémentaire (mot) au sein du corpus.

### ***2.2.2. Fonction de "haut niveau"***

Les fonctions de haut niveau permettent de s'attaquer au même problème pour la localisation d'objets textuels beaucoup plus complexes. Ces fonctions de haut niveau reposent sur une théorie des objets textuels dont les principes sont les suivants : tout objet textuel est manipulable par des méthodes standard permettant d'implémenter son moteur de recherche ; un ensemble de lois de composition (dont les plus simples sont les listes et les séquences) permet de fabriquer des objets textuels nouveaux, dits objets **composites**, à l'aide d'objets élémentaires ou d'autres objets composites.

De ces principes, on peut déduire les conséquences suivantes :

- Il est possible de fabriquer des objets composites d'une complexité arbitrairement élevée en appliquant successivement les lois de composition autant de fois que l'on veut.

- Tout objet composite est automatiquement muni de son moteur de recherche, qui se déduit des moteurs de recherche de ses composants, et, par récursivité, des moteurs de recherche des objets élémentaires à l'origine de sa construction.
- Il est possible de fabriquer des objets élémentaires nouveaux, dits objets **natifs**, de manière tout à fait arbitraire : il suffit pour cela de les doter des méthodes standard. Ces objets natifs pourront à leur tour rentrer comme éléments de fabrication d'objets composites. Par exemple, si on définit un objet "joker" muni de méthodes de localisation (particulièrement faciles à implémenter car elles expriment que ce joker est un objet situé n'importe où), il devient possible de définir un objet composite de type "séquence" comportant successivement le mot "un", le joker, et le mot "homme" pour localiser toutes les occurrences de "un XX homme", dans laquelle XX désigne un mot quelconque. Un autre exemple de constructeur d'objet natif admettant comme paramètre l'infinitif d'un verbe, consiste à fabriquer la liste composée des objets élémentaires correspondant à chacune de ses formes fléchies. Le moteur de recherche de cet objet localisera l'ensemble de formes fléchies du verbe donné en paramètre.

Ce mécanisme de création d'objets composites ou natifs confère à Stella une **architecture totalement ouverte** à bien des égards. L'ensemble des lois de composition est ouvert : l'implémentation de lois nouvelles permet la création d'objets composites nouveaux. Par exemple, si un texte a été segmenté et étiqueté à l'aide d'un analyseur morpho-syntaxique, il est possible de créer (comme cela est fait dans Frantext) un objet composite représentant à la fois le contenu textuel et les attributs grammaticaux associés (par ex. contenu "tire-bouchon" et attribut "substantif"). L'ensemble des objets natifs peut s'enrichir à l'infini. Tout nouvel objet natif peut entrer dans la fabrication d'objets composites, démultipliant ainsi leur combinatoire. Les objets natifs nouveaux (on l'a vu plus haut dans l'implémentation des formes fléchies d'un verbe) permettent d'introduire facilement un "savoir linguistique". Par exemple, rien n'est plus facile que d'introduire, si on le désire, les déclinaisons latines.

Ce qui permet à Stella de gérer des bases aussi diverses que le TLFi, Frantext, les Dictionnaires de l'Académie, et de permettre une hyper-navigation instantanée à travers toutes ces bases. Une aide en ligne est associée à chaque étape de la requête, pour expliquer à l'utilisateur les possibilités de choix, la « syntaxe » de la requête, la marche à suivre...

## **2.3. Stella et Frantext**

Les principes d'interrogation dans les deux bases sont les mêmes : vous entrez dans la base et vous cliquez sur *Menu de Frantext*, puis vous vous laissez guider.

### **3.3.1. Principes de base du travail sur Frantext :**

On peut distinguer 2 phases fondamentales dans le travail : choix des textes que l'on veut étudier, puis Série d'études portant sur les textes choisis. La recherche débute donc par la *création du corpus de travail*. Il est ensuite possible de le *visualiser*, d'*effectuer des recherches* puis de *télécharger les données*.

**1<sup>ère</sup> étape : La sélection du corpus de travail peut s'effectuer sur :**

- L'ensemble des textes de la base
  - Un ou plusieurs auteurs
    - Recherche d'un auteur : saisie en minuscule. (Exemple *zola,flaubert,hugo*)
    - Saisie de plusieurs auteurs : utiliser la virgule pour séparer les noms. Les noms peuvent être donnés de manière fragmentaire. (Exemple : *auber* suffit pour sélectionner Flaubert)
  - Une œuvre :
    - Recherche d'un titre En minuscule. (Exemple : La saisie de "*dame*" sélectionne "*Au bonheur des dames*", "*Notre Dame de Paris*", etc...)
  - Un ou plusieurs genres littéraires
  - Une tranche chronologique
  - La combinaison de plusieurs critères
- Pendant une session(12 heures), la recherche s'effectue sur le corpus sélectionné. Si l'on souhaite sélectionner un nouveau corpus, il est nécessaire de réactiver Frantext.*

## 2<sup>ème</sup> étape : les requêtes sur le corpus de travail

La nouvelle version du logiciel possède des fonctionnalités qui permettent à l'utilisateur de formuler des requêtes complexes. Certaines sont simples à mettre en œuvre, d'autres sont plus compliquées. Une aide en ligne accompagne chaque service.

Recherches simples : une graphie ; une phrase ; plusieurs graphies en cooccurrence dans une ou plusieurs phrases.

Recherches plus complexes :

- les formes fléchies d'un verbe, d'un substantif ou d'un adjectif ;
- des formes tronquées : les mots suffixés en "*ette* " par ex. ;
- des listes de mots ;
- des expressions à choix multiples en une seule requête. Par ex. "*maison (blanche | verte | jaune)*" recherchera *maison blanche, maison verte, maison jaune* ;
- des statistiques sur la fréquence des mots ;
- des études de vocabulaire autour d'un mot pivot (par ex. le vocabulaire employé dans les phrases contenant le mot "*abeille*") ;
- des phénomènes linguistiques complexes tels que la quantification, les constructions pronominales, les temps composés, etc. ; ce type de recherche est rendu possible grâce à l'écriture de grammaires formelles constituées de règles paramétrables. Ces grammaires permettent de rechercher des contextes arbitrairement complexes sur le corpus.

Il est possible au terme de sa recherche de télécharger les résultats. La procédure dépend du navigateur Internet utilisé par le chercheur. Il suffit d'enregistrer les résultats dans un fichier sur son disque dur. On peut ensuite ouvrir ce fichier à l'aide d'un éditeur de texte. Les résultats sont encodés suivant la norme ISO 8859-1. Cette norme définit la grille de codes des caractères. Elle est compatible avec les usages en vigueur sur les principaux systèmes (MS-Windows, Unix, Mac-OS, ...).

*La longueur des contextes obtenus dans la base FRANTEXT est limitée à trois pages pour les textes libres de droits, et à une courte citation de 300 signes pour les textes protégés.*

### 2.3.2. Les principes de recherche :

Les critères de recherches peuvent être :

- saisis au cours d'une session (recherche simple avec saisie d'**expressions de séquence**),
- saisis dans une liste préparée avant d'effectuer une recherche : "**création/édition de listes de mots**", ou en utilisant une "**grammaire**" prédéfinie

#### 2.3.2.1. Les règles de saisie d'une expression de séquence :

Une expression de séquence a pour but de rechercher des contextes contenant une suite de mots consécutifs (d'où le terme séquence) tels que *maison blanche* ou *le chien noir de la ferme* ou des choses beaucoup plus élaborées.

La saisie d'une expression de séquence est effectuée sur le formulaire disponible sous l'option "**recherche dans les textes**".

La saisie d'une séquence : consiste à saisir une suite de mots consécutifs. Une expression de séquence est constituée de N sous-expressions.

#### Pour écrire une expression de séquence :

- **Opérateur ET**, matérialisé par la juxtaposition de deux mots :

Ex. : *maison blanche* cherche *maison blanche*

- **Opérateur OU**, matérialisé par la barre verticale | et les parenthèses)

Ex. : *(maison|palais) d'un (blanc|bleu)sale* va chercher *maison d'un blanc sale, maison d'un bleu sale, palais d'un blanc sale, palais d'un bleu sale.*

- Verbe fléchi : **&c**

Ex. : *&caimer* va chercher toutes les formes du verbe *aimer*, conjuguées ou non.

- Substantif ou Adjectif fléchi : **&m**

Ex. : *&mbocal* va chercher *bocal bocal*

*&mgrand* va chercher *grand, grande, grandes, grands*

- Sous-expressions optionnelles : **& ?** et éventuellement des parenthèses.

Ex. : *un& ?grand homme* cherche un homme ou un grand homme

*un &(très grand) homme* cherchera un homme ou un très grand homme

- Mot quelconque : **&q** introduisant une discontinuité dans la séquence.

Ex. : *Un &qhomme* cherchera une séquence *Un (...) homme* avec un mot entre *un* et *homme*

*Un &q(0,2) homme* cherchera une séquence avec 0, 1 ou 2 mots entre *un* et *homme*.

- La négation: **^**

Ex. : *homme ^très grand qui* est une expression de séquence qui va chercher tous les contextes du genre *homme XXX grand qui* tels que XXX ne soit pas égal à **très**.

*homme ^(très grand)* est une expression de séquence qui va chercher tous les contextes du genre *homme XXX* tels que ou bien XXX est différent de **très**, ou bien XXX est égal à **très** et le mot suivant est différent de **grand** (autrement dit XXX n'est pas le point de départ d'une séquence **très grand**).

*&carriver ^&e(g=A) à* est une expression de séquence qui va sélectionner des contextes tels que *arriveront vite à*, mais pas un contexte tel que *arrivaient en même temps à*. En effet **^&e(g=A)** désigne une **graphie** qui n'est pas un adjectif, mais ne désigne pas une **entité catégorisée** qui n'est pas un adjectif : cette dernière désignation se noterait en effet **&e(g!=A)**. On prendra donc bien garde à ne pas confondre **^&e(g=A)** et **&e(g!=A)**.

#### Exemples de remplissage de formulaire :

**Exemple 1** : On désire rechercher le mot **cheval** au singulier. Il suffit de taper **cheval** dans le cadre séquence 1 et de soumettre la demande.

**Exemple 2 :** On désire rechercher le mot **cheval** au singulier ou au pluriel. Il suffit de taper **&mcheval** dans le cadre séquence 1 et de soumettre la demande.

**Exemple 3 :** On désire rechercher le mot **cheval** au singulier ou au pluriel lorsque ce mot apparaît dans la même phrase que le mot **selle**. Il suffit :

de taper **&mcheval** dans le cadre séquence 1

de taper **selle** dans le cadre séquence 2

**Exemple 4 :** On désire chercher les phrases qui contiennent le mot **femme**, sans que le mot **homme** n'apparaisse à droite du mot femme à moins de 10 mots. Il suffit de :

taper **femme** dans le cadre séquence 1

taper **homme** dans le cadre séquence 2 et de choisir sur la même ligne l'option **Exclu**

choisir dans "Positions relatives des séquences 1 et 2 " l'option **1 AVANT 2** et sur la même ligne de remplir le cadre "distance maximale" avec la valeur **10**

En ce qui concerne les **expressions catégorisées, les différentes syntaxes sont :**

**&e(g=yyy)** désigne une entité ayant le code grammatical précisé par yyy, où yyy est une chaîne de caractères formée par un ou plusieurs codes grammaticaux.

Exemples :

**&e(g=S)** désigne un substantif.

**&e(g=S A)** désigne un substantif ou un adjectif. (On se reportera à la liste des codes disponibles pour connaître l'ensemble des codes existants).

**&e(g!=yyy)** désigne une entité n'ayant pas le code grammatical précisé par yyy.

Exemples :

**&e(g!=S)** désigne une entité autre que substantif.

**&e(g!=S A)** désigne une entité autre que substantif ou adjectif.

**&e(c=expression)** désigne une entité catégorisée ayant un contenu textuel conforme à expression. Une expression de contenu textuel est une expression de séquence qui ne comporte aucune expression **&e(xxx)** : en effet, les graphies contenues dans une entité catégorisée ne sont pas elles-mêmes catégorisées.

Exemples :

**&e(c=tire-)** recherche les entités catégorisées contenant tire-

**&e(c=(&mdemi &q|&q &mdemi))** recherche les entités catégorisées contenant au moins deux mots, dont demi, demis, demie ou demies

**&e(c!=expression)** est une entité n'ayant pas le contenu indiqué par expression.

Expressions d'entités composites : toutes les possibilités ci-dessus peuvent se combiner entre elles.

Exemple :

**&e(c=(&mdemi &q|&q &mdemi) c!=&mheure g=S)** désigne une entité comportant au moins deux mots (dont demi fléchi), mais ne contenant ni heure, ni heures, et qui de surcroît soit un substantif.

On notera que l'ordre des expressions **c=...**, **g=...** au sein d'une même entité catégorisée est indifférent. On aurait pu écrire l'entité ci-dessus sous la forme strictement équivalente :

**&e(g=S c!=&mheure c=(&mdemi &q|&q &mdemi))**

### 2.3.2.2. Les listes

Il est possible de définir au préalable une liste de mots qui servira ensuite de critère de recherche sur le corpus sélectionné. Les listes de mots sont des fichiers stockés sur un espace disque temporaire du centre serveur. **Les listes sont disponibles tant que votre session FRANTEXT est en cours.**

*La création de listes :*

Plusieurs possibilités sont offertes :

- La création manuelle d'une liste
- La création d'une liste par flexion d'un lemme
- La création d'une liste extraite de l'ensemble des graphies du corpus de travail

Vous pouvez relire ou éditer une liste déjà créée.

*Création d'une liste par flexion d'un lemme*

Il suffit de saisir l'infinitif s'il s'agit d'un verbe, le singulier s'il s'agit d'un substantif, le masculin singulier s'il s'agit d'un adjectif.

*Création d'une liste à partir des graphies du corpus de travail*

Le critère de sélection vous permet de définir quelles graphies du corpus de travail seront sélectionnées. Par exemple, le critère *.ismes?* sélectionne tous les mots se terminant par isme ou ismes.

**La syntaxe générale d'un critère est la suivante :**

- Le symbole . (point) désigne n'importe quel caractère.  
Ex. : *.oule* sélectionnera les mots *boule, coule, foule, etc.*
- Une expression telle que *[abcd]* désigne un caractère qui est soit a, soit b, soit c, soit d.  
Ex. : *coule[sr]* sélectionnera les mots *coules* ou *couler*.
- Une expression telle que *[^abcd]* désigne un caractère quelconque, à condition qu'il soit différent de a ,b,c ou d.  
Ex. : *[^cr]oule* sélectionnera les mots *foule, houle, poule, etc.* mais pas les mots *coule* et *roule*.
- Les parenthèses servent à délimiter un groupe de caractères. Un groupe peut être subdivisé en plusieurs sous-groupes par le symbole |. Ce symbole signifie qu'il y a alternative.  
Ex. : *cheva(|lux)* sélectionnera *cheval* ou *chevaux*.
- Les parenthèses peuvent être imbriquées à plusieurs niveaux :  
Ex. *(crainti(f|ve)|peureu(x|se))* sélectionnera *craintif, craintive, peureux, peureuse*.
- Symboles de modification.  
Le symbole \* placé derrière un caractère ou un groupe signifie que ce caractère ou groupe peut être absent ou se répéter un nombre quelconque de fois. Ce symbole peut se placer derrière n'importe quel caractère ou groupe, mais il est en fait très pratique lorsqu'il est associé au symbole . (point) désignant n'importe quel caractère.  
Ex. : *.\*form.\** sélectionnera les mots contenant la chaîne "form" tels que *formée, réforme, informel* etc.  
Ex. : *bla(bla)\** sélectionnera *bla, blabla, blablaba, ...*
- Le symbole + placé derrière un caractère ou un groupe signifie que ce caractère ou groupe peut se répéter un nombre quelconque de fois (mais au moins une fois).

Ex. : .+form.\* sélectionnera les mots contenant la chaîne "form" précédée d'au moins un caractère.

Ex. : bla(bla)+ sélectionnera blabla, blablaba, ...

- Le symbole ? placé derrière un caractère ou un groupe signifie que ce caractère ou groupe est optionnel.  
Ex. : a?politiques? sélectionnera les mots politique, politiques, apolitique et apolitiques.  
Ex. cinéma(tograph(e|ique))s? sélectionnera : cinéma, cinémas, cinématographe, cinématographes, cinématographique, cinématographiques.

*L'utilisation des listes pour la recherche sur le corpus :*

Sur le formulaire de recherche saisir : **&lxxx**, où xxx désigne le nom de la liste créée.

Exemple : Si, dans une liste de nom couleur on a les mots rouge, vert, jaune, on peut écrire l'expression de séquence suivante : **volet &lcouleur** pour chercher *volet rouge, volet vert, volet jaune*.

### **2.3.2.3. Les grammaires :**

Les grammaires permettent de formuler des expressions de recherche capables de localiser dans un corpus des occurrences de phénomènes multiformes : par exemple, une référence à une période peut prendre la forme dès 1954, ou en juillet dernier/prochain, la semaine dernière/prochaine, etc. Les grammaires (au sens de la théorie des langages) ne sont pas à confondre avec les grammaires des langues naturelles. En revanche, elles permettent de rechercher des phénomènes (par exemple une construction pronominale de verbe) qui peuvent se manifester sous des formes très variables en raison des règles syntaxiques du français.

L'intérêt des grammaires est triple :

Une grammaire est un recueil d'expression de séquences tapées une fois pour toutes par l'utilisateur.

Les expressions de séquences peuvent être paramétrées, ce qui permet de les utiliser pour des recherches multiples.

Une grammaire permet d'élaborer facilement des expressions de séquences d'une complexité arbitrairement élevée.

Une grammaire est un fichier qui peut être saisi par l'utilisateur grâce au service de création de grammaire.

*Qu'est-ce qu'une grammaire ?*

Une grammaire est un fichier qui contient est une suite de règles. Chaque règle de grammaire a un nom et un corps.

Le *nom* d'une règle est une chaîne de caractères composée des caractères de a à z (majuscule ou minuscule), de chiffres de 0 à 9, et des caractères - (moins) et \_ (tiret de soulignement). Il doit être suivi d'un blanc puis du caractère : puis d'un retour à la ligne. Le *corps* de la règle peut s'étendre sur plusieurs lignes : un retour à la ligne dans le corps d'une règle n'a aucune signification.

**a) Exemple de grammaire appelée "essai"** : (en italique : le nom de la règle, et en gras : le corps de la règle )

*quantifieur* : très|assez|plus|excessivement

*qualifieur* : grand|gros||petit

*qualification* : &rqualifieur|&rquantifieur &rqualifieur

La grammaire est ensuite utilisée (ou invoquée) pour rechercher sur le corpus prédéfini.

Dans notre exemple, il suffit de saisir sur le formulaire de recherche : *&rqualifieur,essai homme*, ce qui permet d'invoquer la grammaire appelée "essai" et de rechercher des expressions comme "grand homme, petit homme, gros homme".

On peut aussi saisir sur le formulaire de recherche : *&rqualification,essai homme*, ce qui permet d'invoquer la grammaire appelée "essai" et de rechercher des expressions comme "très grand homme, petit homme, plus gros homme".

**b) Elaboration d'une grammaire** : nous voyons ci-dessous en détail l'élaboration d'une grammaire capable de rechercher les contextes contenant des **constructions pronominales de verbes**. Par simplification, les phrases interrogatives sont laissées de côté et nous supposons que le sujet du verbe est un pronom.

**Les usages pronominaux sont :**

**temps simple** : je me/m' (ou tu te/t', il/elle/ils/elles/on/qui se/s', nous nous, vous vous) + une forme verbale

**temps composé** : je me/m' (ou tu te/t', il/elle/ils/elles/on/qui se/s', nous nous, vous vous) + auxiliaire être +participe passé

**temps simple avec négation** : je ne me/m' (ou tu ne te/t', il/elle/ils/elles/on/qui ne se/s', nous ne nous, vous ne vous) + une forme verbale + pas/plus/jamais/guère/point/mie/etc.

**temps composé avec négation** : je ne me/m' (ou tu ne te/t', il/elle/ils/elles/on/qui ne se/s', nous ne nous, vous ne vous) + auxiliaire être + pas/plus/jamais/guère/point/mie/etc. + participe passé

On constate que **temps simple** et **temps composé** (sans négation) commencent de la même façon, que nous pouvons décrire avec une première règle de grammaire de nom affirmatif :

*affirmatif* :

**je (me|m') | tu (te|t') | (il|elle|ils|elles|on|qui) (se|s') | nous nous | vous vous**

Remarque : le corps d'une règle peut s'étendre sur plusieurs lignes (un retour à la ligne n'a aucune signification).

Il est maintenant facile d'écrire la règle *temps\_simple\_affirmatif* pour les temps simples :

*temps\_simple\_affirmatif* :

**&raffirmatif &e(g=V)**

De même nous pouvons écrire la règle *temps\_compose\_affirmatif* pour les temps composés :

*temps\_compose\_affirmatif* :

**&raffirmatif &cêtre &e(g=Ps)**

De la même façon, nous écrivons la suite de la grammaire :

*negatif* :

**je ne (me|m') | tu ne (te | t') | (il|elle|ils|elles|on|qui) ne (se|s') | nous ne nous | vous ne vous**

*temps\_simple\_negatif* :

**&rnegatif &e(g=V) &rfin\_negation**

Remarque : nous précisons plus loin que la règle *fin\_negation* correspond à pas/plus/jamais/guère/point/mie/etc.

*temps\_compose\_negatif* :

**&rnegatif &cêtre &rfin\_negation &e(g=Ps)**

*fin\_negation* :

**pas|plus|jamais|guère|mie|point**

Enfin, il reste à réunir les quatre cas de figure en une seule règle :

*usage\_pronominal :*

**&rtemps\_simple\_affirmatif | &rtemps\_compose\_affirmatif |&rtemps\_simple\_negatif |  
&rtemps\_compose\_negatif**

**Voici la grammaire construite :**

*affirmatif :*

**je (me|m') | tu (te|t') | (il|elle|ils|elles|on|qui) (se|s') | nous nous | vous vous**

*temps\_simple\_affirmatif :*

**&raffirmatif &e(g=V)**

*temps\_compose\_affirmatif :*

**&raffirmatif &cêtre &e(g=Ps)**

*negatif :*

**je ne (me|m') | tu ne (te|t') |(il|elle|ils|elles|on|qui) ne (se|s') | nous ne nous | vous ne vous**

*temps\_simple\_negatif :*

**&rnegatif &e(g=V) &rfin\_negation**

*temps\_compose\_negatif :*

**&rnegatif &cêtre &rfin\_negation &e(g=Ps)**

*fin\_negation :*

**pas|plus|jamais|guère|mie|point**

*usage\_pronominal :*

**&rtemps\_simple\_affirmatif | &rtemps\_compose\_affirmatif |&rtemps\_simple\_negatif |  
&rtemps\_compose\_negatif**

Cette grammaire a été écrite pour fonctionner avec tous les verbes. Elle donne 501 résultats sur le corpus Hugo/Travailleurs de la Mer.

**Supposons maintenant que nous voulions une grammaire similaire, mais qui ne s'appliquerait qu'au verbe *dire*.**

A chaque expression **&e(g=V)** (forme verbale quelconque) , il convient de substituer **&cdire** (forme verbale du verbe *dire*).

De même, à chaque expression **&e(g=Ps)** (participe passé d'un verbe quelconque), il convient de substituer **&e(g=Ps c=&cdire)** (participe passé du verbe *dire*).

Nous obtenons donc la grammaire suivante :

*affirmatif :*

**je (me|m') | tu (te|t') | (il|elle|ils|elles|on|qui) (se|s') | nous nous | vous vous**

*temps\_simple\_affirmatif :*

**&raffirmatif &cdire**

*temps\_compose\_affirmatif :*

**&raffirmatif &cêtre &e(g=Ps c=&cdire)**

*negatif :*

**je ne (me|m') | tu ne (te|t') |(il|elle|ils|elles|on|qui) ne (se|s') | nous ne nous | vous ne vous**

*temps\_simple\_negatif :*

**&rnegatif &cdire &rfin\_negation**

*temps\_compose\_negatif :*

**&rnegatif &cêtre &rfin\_negation &e(g=Ps c=&cdire)**

*fin\_negation :*

**pas|plus|jamais|guère|mie|point**

*usage\_pronominal :*

**&rtemps\_simple\_affirmatif | &rtemps\_compose\_affirmatif |&rtemps\_simple\_negatif |  
&rtemps\_compose\_negatif**

Cette grammaire fournit 13 résultats sur le même corpus Hugo/Travailleurs de la Mer.

La première objection qui vient à l'esprit est la suivante : vais-je devoir écrire une grammaire différente pour chaque verbe que je veux étudier ? Ce serait évidemment très fâcheux si un nouveau mécanisme n'était pas introduit : les règles avec paramètres.

**c) Les règles avec paramètres :**

Nous allons élargir la syntaxe des expressions d'invocation de règles avec la possibilité suivante :  $\&rXXX(P1,P2,\dots,Pn),YYY$

Cette expression invoque la règle XXX de la grammaire YYY.

P1,P2,...,Pn sont des chaînes de caractères qui sont transmises à la règle XXX, c'est-à-dire que, dans l'écriture du corps de XXX, nous pourrions faire référence à ces paramètres.

**c1) Le plus facile est d'étudier un exemple très simple :**

Supposons que nous voulions successivement chercher les usages de toute une série de verbes donnés (aimer, danser, boire, ...) à la première personne du pluriel.

Il serait possible de formuler la requête suivante : **(nous &caimer | nous n'&caimer | nous &cavoir &e(g=Ps c=&caimer) | nous n'&cavoir &q &e(g=Ps c=&caimer) | &caimer-nous)** autant de fois qu'il y a de verbes à étudier.

Une autre possibilité serait d'écrire une grammaire pour le verbe *aimer*, puis de la modifier pour *danser*, etc.

L'une et l'autre de ces solutions sont également fastidieuses. Au lieu de cela nous allons écrire la grammaire G suivante

*auxiliaire :*

**&cavoir | &cêtre**

*verbe :*

**nous &c&1 | nous (ne|n')&c&1 | nous &rauxiliaire &e(g=Ps c=&c&1) | nous (ne|n') &rauxiliaire &q &e(g=Ps c=&c&1) | &c&1-nous**

On remarquera dans cette grammaire, un nouvel élément de syntaxe : **&1**. Cette expression signifie que la règle verbe doit être appelée avec au moins un paramètre, et que ce paramètre se substituera à **&1**. Autrement dit, **&1** est une manière générique de désigner le premier paramètre avec lequel la règle verbe est invoquée.

Si maintenant, nous tapons l'expression de recherche :  $\&r\text{verbe(aimer),G}$  la règle *verbe* est équivalente à :

**nous &caimer | nous (ne|n')&caimer | nous &rauxiliaire &e(g=Ps c=&caimer) | nous (ne|n') &rauxiliaire &q &e(g=Ps c=&caimer) | &caimer-nous** et on procédera donc à la recherche des emplois de *aimer* à la première personne du pluriel.

Bien entendu, une règle peut avoir plusieurs paramètres. Ils seront respectivement désignés dans le corps de la règle par **&1, &2, &3**, etc.

Une règle peut transmettre à une autre règle un ou plusieurs des paramètres qu'elle a reçus.

**c2) Pour illustrer ceci, voyons comment nous pouvons modifier la grammaire cherchant les usages pronominaux pour la faire fonctionner sur un verbe passé en paramètre.**

La règle *usage\_pronominal* doit recevoir nécessairement un paramètre : le verbe à étudier. Elle devra retransmettre ce paramètre à la règle *temps\_simple\_affirmatif* qui en a besoin pour savoir quel verbe il doit conjuguer. De même, la règle *usage\_pronominal* devra transmettre son paramètre aux autres règles qu'elle invoque.

La grammaire devient ainsi :

*affirmatif* :

**je (me|m') | tu (te|t') | (il|elle|ils|elles|on|qui) (se|s') | nous nous | vous vous**

*temps\_simple\_affirmatif* :

**&raffirmatif &c&1**

*temps\_compose\_affirmatif* :

**&raffirmatif &cêtre &e(g=Ps c=&c&1)**

*negatif* :

**je ne (me|m') | tu ne (te|t') |(il|elle|ils|elles|on|qui) ne (se|s') | nous ne nous | vous ne vous**

*temps\_simple\_negatif* :

**&rneгатif &c&1 &rfin\_negation**

*temps\_compose\_negatif* :

**&rneгатif &cêtre &rfin\_negation &e(g=Ps c=&c&1)**

*fin\_negation* :

**pas|plus|jamais|guère|mie|point**

*usage\_pronominal* :

**&rtemps\_simple\_affirmatif(&1) | &rtemps\_compose\_affirmatif(&1)**

**|&rtemps\_simple\_negatif(&1) | &rtemps\_compose\_negatif(&1)**

**c3) Essayons enfin de faire encore plus fort** en réécrivant la grammaire ci-dessus pour qu'elle fonctionne, au choix, ou bien sur tous les verbes ou bien sur un verbe passé en paramètre. Pour obtenir ceci, il faut que certaines règles changent de comportement en fonction des paramètres qui lui sont passés.

Par exemple, si nous considérons la règle *temps\_simple\_affirmatif*, son corps devrait être :

tantôt du genre **&raffirmatif &e(g=V)** tantôt du genre **&raffirmatif &cXXX** avec XXX=verbe à conjuguer.

Supposons que nous écrivions cette règle sous la forme :

*temps\_simple\_affirmatif* :

**&raffirmatif &rverbe\_&1(&2)**

Si maintenant, nous invoquons cette règle en lui passant deux paramètres particulier et *laver*, alors son corps devient équivalent à :

**&raffirmatif &rverbe\_particulier(laver)**

Maintenant nous ajoutons à la grammaire la règle :

*verbe\_particulier* : **&c&1**

Alors, la chaîne *laver* est transmise à la règle *verbe\_particulier* et nous obtenons un fonctionnement correct pour un verbe particulier.

Si au contraire, nous invoquons la règle *temps\_simple\_affirmatif* en lui passant deux paramètres *general* et *n'importe quoi*, son corps devient équivalent à : **&raffirmatif &rverbe\_general(n'importe quoi)**

Maintenant, ajoutons à la grammaire la règle :

*verbe\_general* :  
**&e(g=V)**

Le paramètre *n'importe quoi* sera ignoré par cette règle et nous obtenons un fonctionnement correct pour le cas général.

Nous voyons ainsi qu'il est possible de passer en paramètre absolument tout ce que l'on veut, y compris des noms ou des parties de noms de règles. Ceci permet d'écrire des règles qui vont invoquer telle ou telle règle en fonction des paramètres avec lesquels elles sont invoquées, et d'obtenir ainsi des grammaires d'une très grande souplesse.

En systématisant ce raisonnement, nous obtenons finalement la grammaire suivante :

*affirmatif* :  
**je (me|m') | tu (te|t') | (il|elle|ils|elles|on|qui) (se|s') | nous nous | vous vous**  
*temps\_simple\_affirmatif* :  
**&raffirmatif &rverbe\_&1(&2)**  
*temps\_compose\_affirmatif* :  
**&raffirmatif &cêtre &rparticipe\_&1(&2)**  
*negatif* :  
**je ne (me|m') | tu ne (te|t') |(il|elle|ils|elles|on|qui) ne (se|s') | nous ne nous | vous ne vous**  
*temps\_simple\_negatif* :  
**&rnegatif &rverbe\_&1(&2) &rfin\_negation**  
*temps\_compose\_negatif* :  
**&rnegatif &cêtre &rfin\_negation &rparticipe\_&1(&2)**  
*fin\_negation* :  
**pas|plus|jamais|guère|mie|point**  
*usage\_pronominal* :  
**&rtemps\_simple\_affirmatif(&1,&2) | &rtemps\_compose\_affirmatif(&1,&2)  
|&rtemps\_simple\_negatif(&1,&2) | &rtemps\_compose\_negatif(&1,&2)**  
*verbe\_general* :  
**&e(g=V)**  
*participe\_general* :  
**&e(g=Ps)**  
*verbe\_particulier* :  
**&c&1**  
*participe\_particulier* :  
**&e(g=Ps c=&c&1)**

Si maintenant, nous lançons une recherche avec les invocations :

*&rusage\_pronominal(particulier,agir)*, alors nous recherchons des usages pronominaux du verbe agir. (5 résultats avec le verbe *agir*, 1 résultat avec le verbe *moquer*, etc.).

*&rusage\_pronominal(general,)* alors nous recherchons des usages pronominaux de n'importe quel verbe. (501 réponses sur le corpus Hugo/Travailleurs de la mer)

Remarque : on notera dans l'invocation ci-dessus que le second paramètre est une chaîne vide (ce paramètre est en effet ignoré si le premier paramètre est *general*. Mais ne pas oublier la virgule derrière *general*).

#### *Invocation d'une règle de grammaire :*

La syntaxe à respecter pour invoquer des grammaires :

&rXXX (invocation de la règle XXX de la grammaire courante)

&rXXX,GGG (invocation de la règle XXX de la grammaire GGG)

&rXXX(P1,P2,...,Pn) (invocation de la règle XXX de la grammaire courante avec passage de paramètres)

&rXXX(P1,P2,...,Pn),GGG (invocation de la règle XXX de la grammaire GGG avec passage de paramètres)

Les règles peuvent apparaître dans n'importe quel ordre. En particulier, le corps d'une règle peut invoquer une autre règle qui ne sera écrite que plus loin dans la grammaire.

&1,&2,&3,... manière générique de désigner, dans une règle de grammaire, le premier, le second, le troisième,... paramètre.

#### **2.3.2.4. Calcul de fréquences**

La fréquence d'un mot est égale à son nombre d'occurrences dans le corpus de travail. Pour calculer des fréquences, vous disposez de services vous permettant :

- d'extraire du corpus de travail les mots conformes à un profil (suffixe, préfixe, infixes donnés) et de calculer la fréquence de chacun d'eux.. Les principes de saisie de votre requête sont identiques à ceux décrits au chapitre concernant la création d'une liste à partir des graphies du corpus de travail.
- de calculer la fréquence d'un mot ou de chaque mot d'une liste. Si vous choisissez cette dernière possibilité, il vous faudra fabriquer la liste de mots avant de lancer le service de calcul de fréquences. (voir le service de création de listes de mots).
- de calculer la distribution des fréquences d'un mot ou de chaque mot d'une liste.

Ce service vous permet de découper votre corpus de travail en "tranches" et d'obtenir un tableau comparatif de la fréquence d'un mot donné (ou de la somme des fréquences des mots d'une liste donnée) dans chaque tranche.

Vous pouvez définir les tranches de différentes façons :

1) Une tranche = l'oeuvre d'un auteur. Les résultats vous permettent alors de comparer l'emploi que font d'un mot (ou des mots d'une liste) les différents auteurs.

2) Une tranche = une référence bibliographique. Les résultats vous permettent alors de comparer l'emploi de mots donnés dans les différentes oeuvres.

3) Une tranche = une période de X années (vous pouvez choisir la valeur de X). Les résultats vous permettent alors de comparer l'emploi de mots donnés à différentes époques.

#### **Présentation des résultats :**

Suivant les cas, les résultats peuvent être triés :

Cas 1) par ordre décroissant des fréquences (les auteurs utilisant le plus fréquemment les mots donnés sont sortis en premier) ou par ordre alphabétique des auteurs.

Cas 2) par ordre décroissant des fréquences (les oeuvres attestant le plus fréquemment les mots donnés sont sorties en premier) ou par ordre alphabétique des références.

Cas 3) par ordre décroissant des fréquences (les tranches chronologiques pendant lesquelles les mots donnés sont le plus fréquemment employés sont sorties en premier) ou par ordre chronologique des tranches.

Les résultats sont présentés sous forme d'un histogramme.

L'étude de la distribution des fréquences peut être, au choix, réalisée en utilisant les fréquences absolues (fréquence absolue d'un mot dans une tranche = le nombre d'occurrences du mot dans la tranche) ou en utilisant les fréquences relatives (fréquence relative d'un mot dans une tranche = le nombre d'occurrences du mot dans la tranche divisé par le nombre total d'occurrences dans la tranche).

### **2.3.2.5. Etude du vocabulaire au voisinage des occurrences d'un mot**

Ce service vous permet d'étudier, dans le corpus de travail, le vocabulaire employé dans le voisinage des occurrences d'un mot donné ou des mots d'une liste donnée.

Appelons pivot une occurrence du mot donné (ou d'un des mots de la liste donnée). Le voisinage peut être défini, au choix, comme la phrase contenant le pivot (en y ajoutant éventuellement plusieurs phrases à gauche ou à droite) ou comme une portion de texte contenant X mots avant et Y mots après le pivot (X et Y à votre choix).

Par exemple, il est possible d'étudier le vocabulaire employé dans les phrases contenant le mot "abeille".

Le résultat d'une telle étude est d'établir la liste de tous les mots trouvés dans les voisinages du pivot. Chaque mot est fourni avec sa fréquence. La liste peut être classée au choix par ordre alphabétique ou par ordre croissant ou décroissant des fréquences.

#### **TEMPS D'EXECUTION :**

*Ce service met en jeu des moyens de traitement très importants. Il doit donc être utilisé avec mesure. Pour cette raison, ce programme s'arrête automatiquement lorsque de nombre d'occurrences dépasse 10000. Dans ce cas, les résultats sont quand même obtenus, mais ne correspondent qu'au dépouillement des 10000 premiers voisinages.*

*Il est donc prudent de s'assurer de la fréquence du pivot grâce au service calcul de la fréquence d'un mot ou de chaque mot d'une liste.*

## **2.4. Stella et le TLFi**

L'accès aux possibilités les plus raffinées exige à la fois la connaissance de la structure du TLF et la compréhension des concepts permettant de mettre en oeuvre des recherches très puissantes.

**Trois niveaux d'utilisation** de puissance croissante sont proposés :

**La consultation simple des articles**, qui apporte d'emblée des facilités de lecture très intéressantes. Sa mise en oeuvre ne demande aucune connaissance particulière et ne nécessite la lecture d'aucune documentation.

**Les recherches assistées**, qui fournissent des possibilités étonnantes de consultation du TLF, et dont la mise en oeuvre ne demande que l'examen de quelques exemples types très simples.

**Les recherches complexes**, qui offrent les possibilités maximales d'investigation. Elles demandent la consultation d'une documentation présentant de manière imagée les différents objets rencontrés dans le TLF et les concepts (l'équivalent de 4 à 5 pages dactylographiées) permettant d'exprimer les recherches les plus raffinées.

L'utilisateur pressé pourra se contenter des deux premiers niveaux d'utilisation. Qu'il sache cependant qu'il a en réserve un potentiel considérable. Il pourra aussi jouer avec la nouvelle fonctionnalité qu'est la **recherche phonétique** s'il est brouillé avec l'orthographe. Il pourra

aussi tester *l'hyper-navigation* à travers les bases gérées par Stella, ce qui apporte un plus réel à la connaissance.

### 2.4.1. Interrogation simple

Il s'agit de la « visualisation simple » d'un article, qui se fait en deux étapes :

- Choix de la tranche alphabétique dans laquelle se situe le mot cherché  
Exemple : **DISSOLU** à **DOMINOTERIE**
- Choix du mot dans la tranche retenue  
Exemple **DITON**

Important : Ce type d'interrogation permet d'atteindre toutes les « vedettes » d'articles. Mais attention, le mot cherché peut se trouver à l'intérieur de l'article, dans un bloc de hiérarchie inférieure. Il se peut aussi qu'il soit un « mot caché » apparaissant sous une autre vedette que la sienne. Si la visualisation sélective ne donne pas de résultat, essayer l'une des interrogations plus avancées :

Par exemple : on peut chercher l'article concernant LOTOPHAGE.

La réponse est « aucune solution trouvée ».

Or il existe une solution, cachée, et accessible par la recherche assistée.

### 2.4.2. Interrogation assistée

Encore appelée Requête assistée, elle se fait par l'intermédiaire d'un formulaire de recherche, ce formulaire propose de restreindre la recherche

- Aux vedettes contenant une graphie  
Exemple *carnage*
- Aux vedettes contenant un code grammatical  
Exemple *onomatopée*
- A un domaine technique particulier  
Exemple *horticulture*
- A un indicateur d'emploi  
Exemple *péjoratif*
- A un objet textuel de type à choisir dans un menu déroulant. Si l'on se sert de cette proposition, il faut impérativement spécifier quelque chose de son contenu et remplir une ou plusieurs des cases Mot1, Mot2, Mot3  
Exemple *indifférent* avec mot 1 contenant *lotophage* et petite case cochée (pour sing. et plur.) permet de trouver ce mot caché sous un autre, accompagné de sa définition

La recherche peut être restreinte à l'une de ces possibilités, ou en croiser plusieurs :

- Vedette contenant *age* et code grammatical = *substantif*
- Vedette contenant *ée* et code grammatical = *substantif* et domaine = *horticulture*
- Code = *substantif* et indicateur = *populaire* et objet4 = *auteur d'exemple* avec mot1 = *hugo*. (Ce qui permet de savoir quels sont les substantifs qui dans un emploi populaire, ont été illustrés par un exemple de Hugo)

### 2. 4.3. Interrogation avancée

encore appelée **Requête complexe**, elle se fait par un formulaire de recherche spécifique.

#### 2.4.3.1. La Requête complexe

Les requêtes peuvent être élémentaires (un seul objet recherché) ou multi-objets.

##### *Requêtes élémentaires*

Les requêtes les plus simples portent sur un seul objet.

Vous avez **impérativement** à spécifier le type de l'objet que vous recherchez, et **éventuellement** à lui imposer un contenu textuel. (Voir ci-dessous le § 4.3.2 concernant les objets textuels et leurs types, le § 4.3.3. concernant la notion de contenu, et le § 4.3.5. concernant la notion de listes de mots)

Ainsi, vous pouvez rechercher les définitions qui contiennent le mot "*instrument*". Pour cela, il suffit de remplir un formulaire vous permettant de définir le type et le contenu de l'objet. Cliquez alors sur le bouton "EXECUTION" pour exécuter la recherche.

Le formulaire est accessible depuis le menu principal (rubrique Recherche complexe)

##### *Requêtes multi-objets*

Il est possible de faire porter une requête sur plusieurs objets. (Voir ci-dessous le § 4.3.2. concernant les objets textuels et leurs types, le § 4.3.3 concernant la notion de contenu, et le § 4.3.5 concernant les listes de mots).

Par exemple, on désire rechercher les définitions empruntées au dictionnaire de l'Académie qui contiennent le mot *instrument*.

Une telle requête met en jeu **deux objets** :

- 1) **Un objet** de type "*Définition empruntée*" ayant le contenu textuel "*instrument*"
- 2) **Un objet** de type "*Source*" ayant le contenu textuel "*Académie*".

On comprend aisément que pour exprimer une requête complexe, il ne suffit pas d'énumérer les objets qu'elle met en jeu. Il faut également indiquer quels sont les liens logiques entre les objets.

Dans l'exemple ci-dessus, il ne suffit pas d'exprimer que l'on s'intéresse d'une part aux définitions contenant "*instrument*" et, d'autre part, à l'ensemble des sources contenant "*Académie*".

Il reste à exprimer un lien logique essentiel : l'objet 2 est **la** source de l'objet 1. Autrement dit, il faut exprimer **la nature du lien** entre les objets 1 et 2. (Voir le § 4.3.4 concernant la notion de liens).

#### 2.4.3.2. Les objets textuels et leur type

Les articles d'un dictionnaire sont constitués d'objets élémentaires de différents **types** (définitions, exemples, indicateurs de domaine technique, indicateurs grammaticaux, etc.).

Le TLF a donc été décomposé en de tels **objets textuels**. Ceci va vous permettre d'effectuer des recherches portant sur tel ou tel **type** d'objet.

Par exemple, il vous sera possible de chercher les objets de type "définition" qui contiennent le mot *instrument*.

Pour obtenir une description complète des types d'objets du TLF, [voir l'aide en ligne](#) qui, en plus du tableau ci-dessous, renvoie à des exemples pour chacun des objets.

|  |  |
|--|--|
| <a href="#">Article</a>                | Toute portion de texte consacrée à l'étude d'un mot. Un "article" est un article principal ou un article dérivé ou un article mot formé.   |
| <a href="#">Article principal</a>      | Article correspondant à une vedette principale du TLF.   |
| <a href="#">Article dérivé</a>         | Article correspondant à un mot cité dans une rubrique "remarque" ou une rubrique "dérivé".   |
| <a href="#">Article mot formé</a>      | Article correspondant à un mot cité dans un article principal dont la vedette est un élément formant ou de composition, un préfixe ou un suffixe.  |
| <a href="#">Auteur d'exemple</a>       | Auteur d'un exemple cité.  |
| <a href="#">Code grammatical</a>       | Code grammatical (verbe, substantif, etc.) associé à un mot cité en vedette.   |
| <a href="#">Construction</a>           | Indication des conditions syntaxiques d'emploi de la vedette.  |
| <a href="#">Crochets</a>               | Condition d'emploi de la vedette (du genre "le complément d'objet est un inanimé", etc.)   |
| <a href="#">Date d'exemple</a>         | Date d'un exemple cité.  |
| <a href="#">Définition</a>             | Définition de la vedette, elle peut être empruntée à un autre dictionnaire ou être "maison".   |
| <a href="#">Définition empruntée</a>   | Définition empruntée à un autre dictionnaire.  |
| <a href="#">Définition TLF</a>         | Définition "maison"  |
| <a href="#">Discours</a>               | Portion d'article débutant derrière un sigle de structure (I,A,a, etc...) et se terminant au sigle de structure suivant.   |
| <a href="#">Domaine technique</a>      | Indication du domaine technique dans lequel s'emploie la vedette.  |
| <a href="#">Domaine général</a>        | Indication de domaine non technique (par ex. Domaine des sens, de l'esthétique, etc...).   |
| <a href="#">Élément formant</a>        | Partie de la vedette correspondant à l'élément formant (ou au préfixe/suffixe) ayant servi à la forger.  |
| <a href="#">Entrée</a>                 | Couple vedette-code grammatical d'un article principal ou dérivé ou formé.   |
| <a href="#">Entrée principale</a>      | Couple vedette-code grammatical d'un article principal.  |
| <a href="#">Entrée de mot dérivé</a>   | Couple vedette-code grammatical d'un article dérivé.   |
| <a href="#">Entrée de mot formé</a>    | Couple vedette-code grammatical d'un article formé.  |
| <a href="#">Exemple</a>                | Exemple cité. Il peut être détaché ou enchaîné.  |
| <a href="#">Exemple détaché</a>        | Exemple cité particulièrement illustratif (apparaissant dans le TLF dans un paragraphe séparé du reste du texte).  |
| <a href="#">Exemple enchaîné</a>       | Exemple cité autre que exemple détaché.  |
| <a href="#">Indicateur</a>             | Indication de type grammatical, stylistique ou sémantique.   |
| <a href="#">Langue empruntée</a>       | Mention de la langue dont vient la vedette principale de l'article.  |
| <a href="#">Localisation d'exemple</a> | Localisation de l'exemple dans l'oeuvre (page, chapitre, etc...).  |
| <a href="#">Mot vedette</a>            | Mot vedette d'un article principal ou dérivé ou formé.   |
| <a href="#">Paragraphe</a>             | Paragraphe quelconque du TLF.  |
| <a href="#">Paragraphe syntagme</a>    | Suite de syntagmes faisant l'objet d'une mise en évidence sous forme d'un paragraphe séparé du reste du texte. Le paragraphe syntagme est un sous-cas du type "syntagme" (voir plus loin). |
| <a href="#">Sigle hiérarchique</a>     | Sigle de structure(I,A,I,a,etc...).  |
| <a href="#">Source</a>                 | Source bibliographique d'une définition empruntée ou d'un syntagme cités dans d'autres ouvrages.   |
| <a href="#">Synonyme/antonyme</a>      | Liste des synonymes ou antonymes de la vedette.  |
| <a href="#">Syntagme</a>               | Paragraphe syntagme (voir plus haut) ou syntagme enchaîné (voir ci-dessous).   |
| <a href="#">Syntagme enchaîné</a>      | Syntagme formé à partir de la vedette, ne se trouvant pas dans un paragraphe syntagme.   |

|                                 |  |
|---------------------------------|--|
| <a href="#">Titre d'exemple</a> | Titre de l'oeuvre dont est tiré un exemple cité. |
|---------------------------------|--|

### 2.4.3.3. La notion de contenu :

Vous pouvez imposer aux objets que vous recherchez de contenir tel ou tel élément textuel.

Par exemple, vous pouvez rechercher les objets de type "Définition" contenant le mot "instrument", mais pas le mot "optique".

Nous allons voir que vous pouvez définir un contenu avec une grande richesse de possibilités.

#### 2.4.3.3.1. Contenu unique

##### a) Description du contenu

Chaque mot d'un contenu peut être spécifié sous plusieurs formes :

- **Forme 1**-Par sa graphie exacte
- **Forme 2**-Par une indication de conjugaison sous la forme *&cINFINITIF*.  
**Exemple** :*&caimer*
- **Forme 3**-Par une indication de flexion pour les adjectifs et substantifs *&mMOT*, où MOT désigne le mot au singulier (pour les substantifs) et au masculin singulier (pour les adjectifs).  
**Exemple** :*&mvert*
- **Forme 4**-Par une indication de liste de mots sous la forme *&INOM\_DE\_LISTE*. (Voir le service Gestion de listes de mots)
- **Forme 5**-Par une indication de mot quelconque sous la forme *&q*

##### b) Négation d'un élément de contenu

Chacune de ces manières de spécifier un mot (sauf la forme 5 = *&q*) peut être précédée du symbole de négation *^*. Le sens de cette négation sera illustré par les exemples ci-dessous.

##### c) Exemples de contenus

Exemple 1 : Le contenu *&caimer la nature* recherche les occurrences d'une forme flexionnelle du verbe aimer, suivie des mots "la" et "nature".

Exemple 2 : Le contenu *^&cruer dans les brancards* recherche les occurrences d'un mot qui n'est pas une forme du verbe ruer, suivi des mots "dans", "les" et "brancards".

Exemple 3 : Le contenu *un &q jour* recherche les occurrences du mot "un", suivi d'un mot quelconque, puis du mot "jour".

Exemple 4 : Le contenu *&credouter un &lexemple* recherche les occurrences d'une forme du verbe "redouter", suivie du mot "un", suivi d'un mot appartenant à la liste "exemple". (Voir le service Gestion de listes de mots).

Exemple 5 : Le contenu *&ccouper &q &mtête* recherche les occurrences d'une forme du verbe "couper", suivie d'un mot quelconque, suivi de tête ou têtes.

**Note importante** : Un contenu, comme on le voit dans les exemples ci-dessus, est une suite d'éléments (éventuellement un seul) spécifiés sous une des formes 1, 2, 3, 4, 5. Une expression de contenu qui ne contiendrait que la spécification *&q* n'a aucun sens (Que signifierait en effet de rechercher les définitions contenant un mot quelconque ?). Il en va de même pour un contenu qui ne contiendrait que des spécifications niées avec le symbole *^* (Que signifierait en effet de rechercher les définitions contenant un mot qui n'est pas "homme" ?).

Nous en déduisons la règle suivante : *pour qu'un contenu soit acceptable, il faut qu'il contienne au moins une spécification de la forme 1,2,3 ou 4 non précédée du symbole de négation ^.*

*d) Exclusion d'un contenu*

Le fait d'indiquer **&n** en tête d'une expression de contenu signifie que l'on **élimine** les objets textuels qui le contiennent.

Par exemple, si l'objet textuel est une définition, et si on s'intéresse aux seules définitions qui ne contiennent pas le mot "homme", on écrira le contenu suivant : **&n homme**

On prendra garde à ne pas confondre "**^homme**" qui signifie "je veux que l'objet contienne un mot qui soit différent de "homme" avec "**&n homme**" qui signifie "je veux que tous les mots du contenu soient différents de "homme".

*e) Position du contenu textuel dans l'objet*

Il est possible d'imposer que le contenu soit à une distance maximale (exprimée en nombre de mots) du début et/ou de la fin de l'objet.

Exemple 1 : **&d3 un homme** signifie que "un homme" doit être trouvé à 3 mots au plus du début de l'objet textuel.

Exemple 2 : **&f4 un homme** signifie que "un homme" doit être trouvé à 4 mots au plus de la fin de l'objet textuel.

Exemple 3 **&d3 &f4 un homme** signifie que "un homme" doit être trouvé à 3 mots au plus du début de l'objet textuel et à 4 mots au plus de sa fin.

Exemple 4 : **&n &d3 un homme** signifie que "un homme" **ne doit pas** être trouvé à moins de 3 mots du début de l'objet textuel.

*2.4.3.3.2. Contenu multiple*

Il est possible de spécifier qu'un objet doit, ou ne doit pas avoir simultanément plusieurs contenus. Il suffit pour cela d'écrire plusieurs contenus simples (sous n'importe laquelle des formes décrites ci-dessus) en les séparant par le caractère |

Exemple 1 : **homme|femme** signifie que l'objet doit contenir homme et femme

Exemple 2 : **homme|&n femme** signifie que l'objet doit contenir homme, mais pas femme

Exemple 3 : **&d3 homme|&n &f4 femme|enfant** signifie que l'objet doit contenir homme à moins de 3 mots de son début, ne pas contenir femme à moins de 4 mots de sa fin, et contenir enfant.

*2.4.3.4. La notion de liens*

Nous avons dit qu'il est possible de faire porter une requête sur plusieurs objets. (Voir plus haut, § 4.1.)

Pour exprimer une requête complexe, il ne suffit pas d'énumérer les objets qu'elle met en jeu. Il faut également indiquer quels sont les liens logiques entre les objets. Nous allons voir que les liens entre deux objets sont de deux natures distinctes.

#### 4.3.4.1. Lien "objet X inclus dans objet Y".

Si on considère une définition empruntée, on constate qu'elle a la structure suivante :

▶ Sorte d'instrument dont on se sert pour enfoncer les pilotis en terre et qu'on appelle autrement mouton ▶(Ac. 1935).◀◀

▶  
avec :

▶  
▶et ◀marquant le début et la fin de la définition.

▶  
▶et ◀marquant le début et la fin de la source.

On voit que l'objet définition contient à la fois l'énoncé de la définition et la source. Autrement dit l'objet "Source" est **inclus dans** l'objet "Définition empruntée".

Dans notre exemple, il suffit donc de spécifier que l'objet 2 est **inclus dans** l'objet 1 pour expliciter le fait que l'objet 2 est bien **la source de** l'objet 1.

Pour indiquer que l'objet 2 est à l'intérieur de l'objet 1, il suffit de cliquer dans la fenêtre "Liens" associée à l'objet 2 du formulaire sur l'entrée "inclus dans l'objet 1" qui signifie **est inclus** dans l'objet 1.

#### *Autre exemple expliquant comment remplir le formulaire de requête :*

##### **Chercher les exemples de Zola contenant le mot "amour"**

**Objet 1 :** *type* "Exemple", *contenu* "amour"

**Objet 2 :** *type* "Auteur d'exemple", *contenu* "Zola", *lien* "inclus dans l'objet 1"

ou, ce qui est strictement équivalent:

**Objet 1 :** *type* "Auteur d'exemple", *contenu* "Zola", *lien* "inclus dans l'objet 2"

**Objet 2 :** *type* "Exemple", *contenu* "amour"

NOTA : le document donnant la description des objets trouvés dans le TLF permet de savoir quel type d'objet est susceptible de contenir tel ou tel autre type d'objet.

#### 2.4.3.4.2. Lien "objet X dépendant de l'objet Y".

Prenons l'exemple d'un passage du TLF tel que celui-ci :

A)MARINE, rare  
1)Emploi singulier  
....  
Définition 1  
....  
2)Emploi pluriel  
...  
Définition 2  
...  
B)ASTRONOMIE  
....

Définition 3

....

Etc.

Il est clair pour le lecteur que les propriétés suivantes sont vérifiées:

- a) La définition 1 et la définition 2 sont valables dans le domaine technique "Marine".
- b) La définition 1 est relative à un emploi singulier de la vedette.
- c) La définition 2 est relative à un emploi pluriel de la vedette.
- d) La définition 3 n'est pas relative à la marine.

Ceci découle du fait qu'une mention telle que "MARINE" est *valable* pour le niveau hiérarchique où elle apparaît (donc "MARINE" est valable pour l'indicateur "rare") et pour les niveaux hiérarchiques inférieurs (donc pour les indicateurs "emploi singulier" et "emploi pluriel" ainsi que pour les définitions 1 et 2)

Nous dirons qu'un objet X est **dépendant** d'un objet Y si X est hiérarchiquement inférieur ou égal à Y. Il est clair que les propriétés a, b, c, d ci-dessus sont vérifiées parce que :

- a) L'objet "Définition" 1 et l'objet "Définition" 2 sont dépendants de l'objet "Domaine technique" MARINE.
- b) L'objet "Définition" 1 est dépendant de l'objet "Indicateur" emploi singulier.
- c) L'objet "Définition" 2 est dépendant de l'objet "Indicateur" emploi pluriel.
- d) L'objet "Définition" 3 n'est pas dépendant l'objet "Domaine technique" MARINE.

Le nouveau lien "**l'objet X est dépendant de l'objet Y**" est exprimable dans le formulaire de requêtes. Il va apporter à l'interrogation du dictionnaire une **puissance considérable**.

NOTE : La vedette d'un article, ainsi que toute information qui apparaît au début de l'article avant que ne commence la numérotation des niveaux hiérarchiques, sont de toute évidence hiérarchiquement supérieures à tout autre objet de l'article. Par convention, tout objet de l'article est donc considéré comme dépendant de la vedette, ou des informations de début.

Voyons, à travers quelques exemples, la puissance que confère aux recherches la notion de dépendance hiérarchique des objets :

**1) Quelles sont les définitions contenant le mot "mets", utilisées dans le domaine de l'art culinaire, et illustrées par des exemples de Zola ?**

**Objet 1 :** *type* "définition", *contenu* "mets", *lien* "dépendant de l'objet 2"

**Objet 2 :** *type* "Domaine technique", *contenu* "art culinaire"

**Objet 3 :** *type* "Auteur d'exemple", *contenu* "Zola", *lien* "dépendant de l'objet 1" (on estime raisonnablement qu'un exemple dépendant d'une définition illustre cette définition)

**2) Quels sont les substantifs empruntés à une langue étrangère (non précisée) et qui, lorsqu'ils sont employés dans le domaine de l'art culinaire, sont illustrés par une définition empruntée à l'Académie ?**

**Objet 1 :** *type* "Entrée"

**Objet 2 :** *type* "Code grammatical", *contenu* "substantif", *lien* "inclus dans l'objet 1"

**Objet 3 :** *type* "Domaine technique", *contenu* "art culinaire", *lien* "dépendant de l'objet 1"

**Objet 4 :** *type* "Définition empruntée", *lien* "dépendant de l'objet 3"

**Objet 5 :** *type* "Source", *contenu* "Académie", *lien* "inclus dans l'objet 4"

**Objet 6 :** *type* "Langue empruntée", *contenu* (non précisé) *lien* "dépendant de l'objet 1"

Explications :

- Le lien "inclus dans l'objet 1" de l'objet 2 exprime que l'entrée est un substantif.
- Le lien "dépendant de l'objet 1" de l'objet 3 exprime que l'indication de domaine technique est dans la portée de l'objet 1 (l'entrée), et se trouve donc dans l'article consacré à cette entrée.
- Le lien "dépendant de l'objet 3" de l'objet 4 exprime que la définition est valable dans le domaine de la géographie.
- Le lien "inclus dans l'objet 4" de l'objet 5 exprime que la source de la définition est l'Académie.
- Le lien "dépendant de l'objet 1" de l'objet 6 exprime que l'objet "langue empruntée" est dans l'article dont l'entrée est l'objet 1.

NOTE : Plusieurs liens peuvent être attachés à un même objet, comme le montre l'exemple ci-dessous :

**Quelles sont les locutions populaires ayant trait à la religion ?**

**Objet 1 :** *type* "Indicateur", *contenu* "populaire"

**Objet 2 :** *type* "Domaine", *contenu* "religion"

**Objet 3 :** *type* "Indicateur", *contenu* "locution", *liens* "dépendant de l'objet 1" et "dépendant de l'objet 2"

Le double lien "dépendant de l'objet 1" et "dépendant de l'objet 2" indique que la locution est à la fois sous l'influence de "populaire" et de "religion".

### ***Méthodologie pour remplir un formulaire***

Les exemples exposés ci-dessus peuvent donner l'impression que le remplissage d'un formulaire est d'une grande complexité. **Il n'en est rien** : il suffit de se laisser guider par les termes de la question posée.

Reprenons par exemple la question :

***Quels sont les substantifs empruntés à une langue (non précisée) et qui, lorsqu'ils sont employés dans le domaine de l'art culinaire, sont illustrés par une définition empruntée à l'Académie ?***

Laissons-nous guider par l'énoncé :

- ***Quels sont les substantifs*** se traduit par :

**Objet 1 :** *type* "Code grammatical", *contenu* "substantif".

- ***empruntés à une langue quelconque (non précisée)*** se traduit par :

**Objet 2 :** *type* "Langue empruntée", *contenu* (non précisé)

Puisqu'on introduit un deuxième objet, il convient de se demander par quel lien il se rattache à un objet précédemment défini. La réponse à cette question est toujours très simple : ou bien le nouvel objet est inclus dans un objet précédemment défini, ou bien il en est dépendant, ou bien c'est un objet précédemment défini qui est dépendant du nouvel objet. Dans notre cas; il est clair que l'objet 2 est dépendant de l'objet 1 (l'indication "verbe" étant valable pour tout l'article).

- ***qui, lorsqu'il sont employés dans le domaine de l'art culinaire*** se traduit par :

**Objet 3 :** *type* "Domaine technique", *contenu* "art culinaire".

En ce qui concerne l'établissement d'un lien avec l'un des objets précédents 1 ou 2, il ne peut s'agir que d'une dépendance hiérarchique par rapport à la mention "substantif" : le lien à établir pour l'objet 3 est donc "dépendant de l'objet 4".

- ***sont illustrés par une définition empruntée*** se traduit par :

**Objet 4** : *type* "Définition empruntée"

Lien : Objet 4 "dépendant de l'objet 1".

- *à l'Académie* se traduit par :

**Objet 5** : *type* "Source", *contenu* "Académie".

Lien : objet 5 "inclus dans l'objet 4".

Pour montrer que le remplissage du formulaire est uniquement guidé par les termes de la question, reprenons la même question, formulée autrement :

Pour moi qui m'intéresse aux définitions de l'Académie, je voudrais savoir celles qui sont utilisées pour des verbes empruntés à l'anglais et utilisés en physique.

Reprenons la démarche :

- *Pour moi qui m'intéresse aux définitions* se traduit par :

**Objet 1** : *type* "Définition empruntée"

- *de l'Académie* se traduit par :

**Objet 2** : *type* "Source" *contenu* "Académie"

En ce qui concerne le lien à établir : objet 2 "inclus dans l'objet1".

- *je voudrais savoir celles qui sont utilisées pour des substantifs* se traduit par :

**Objet 3** : *type* "Code grammatical" *contenu* "substantif"

Lien à établir : objet 1 (mais variante tout aussi valable : objet 2) "dépendant de l'objet 3".

- *empruntés à une langue* se traduit par :

**Objet 4** : *type* "Langue empruntée" *contenu* (non précisé)

Lien : objet 4 "dépendant de l'objet 3"

- *et utilisés en géographie* se traduit par :

**Objet 5** : *type* "Domaine technique" *contenu* "art culinaire"

Lien : objet 5 : "dépendant de l'objet 3".

En résumé : les deux expressions de recherche ci-dessus sont sémantiquement équivalentes. Elles correspondent à deux façons différentes de remplir le formulaire, mais donneront strictement les mêmes résultats. Le remplissage du formulaire ne vous impose pas de penser dans tel ou tel ordre : vous ne faites que décrire une collection d'objets, dans l'ordre où ils vous viennent à l'esprit, en énonçant des liens logiques simples qui les unissent.

#### 2.4.3.4.3. Connexité d'une requête.

Imaginons une requête comme une collection d'objets ponctuels pesants (chaque objet de la requête est un point) reliés par des fils (il y a un fil entre le point X et le point Y si on a exprimé un lien entre les objets X et Y correspondants).

Supposons que l'on soulève au hasard un des objets. Deux cas peuvent se présenter :

a) On soulève l'ensemble des autres objets.

b) Un objet, ou un paquet d'objets reste sur la table.

Dans le cas b) il existe au moins deux "paquets" d'objets P1 et P2 sans aucun lien entre eux. Cette situation est anormale, car elle signifie qu'aucun lien logique n'a été spécifié entre P1 et P2 et que, par conséquent, la requête n'a aucun sens.

Nous dirons dans ce cas que l'ensemble des objets est **non connexe**. Cette anomalie est détectée lors de la soumission de la requête qui sera alors rejetée.

Examinons un exemple de non connexité :

Supposons que l'on veuille savoir quels sont les verbes cités en dérivés dans le TLF, dérivant de substantifs. On élabore la requête suivante :

**Objet 1** : *type* "Entrée principale"

**Objet 2** : *type* "Code grammatical", *contenu* "substantif", *lien* "inclus dans l'objet 1"

**Objet 3** : *type* "Entrée dérivée"

**Objet 4** : *type* "Code grammatical", *contenu* "verbe", *lien* "inclus dans l'objet 3"

On crée alors 2 paquets indépendants : d'une part les objets 1 et 2, et d'autre part les objets 3 et 4. La requête est non connexe. Il manque le lien "dépendant de l'objet 1" à l'objet 3 pour exprimer que l'objet 3 est une entrée dérivée de l'entrée principale et rétablir ainsi la connexité.

#### 2.4.3.5. La notion de liste de mot

Les listes de mots peuvent être créées manuellement, ou créées par flexion de lemmes, ou créées à partir de l'ensemble des graphies du TLF. Elles peuvent être modifiées elles peuvent être détruites.

Pour la création à partir des graphies du TLF, le critère de sélection permet de définir quelles graphies seront sélectionnées. Par exemple, le critère *.isme* sélectionne tous les mots se terminant par *-isme*, et le critère *.isme ?* sélectionne tous les mots se terminant par *-isme* ou *-ismes*.

*La syntaxe générale d'un critère est la suivante*

- Le symbole *.* (point) désigne n'importe quel caractère.  
Ex. : **.ou** sélectionnera les mots *boule, coule, foule, etc.*
- Une expression telle que *[abcd]* désigne un caractère qui est soit a, soit b, soit c, soit d.  
Ex. : **coule[sr]** sélectionnera les mots *coules* ou *couler*.
- Une expression telle que *[^abcd]* désigne un caractère quelconque, à condition qu'il soit différent de a, b, c ou d.  
Ex. : **[^cr]ou** sélectionnera les mots *foule, houle, poule, etc.* mais pas les mots *coule* et *roule*.
- Les parenthèses servent à délimiter un **groupe** de caractères. Un groupe peut être subdivisé en plusieurs sous-groupes par le symbole *|*. Ce symbole signifie qu'il y a alternative.  
Ex. : **cheva(l|ux)** sélectionnera *cheval* ou *chevaux*.  
Les parenthèses peuvent être imbriquées à plusieurs niveaux :  
Ex. **(crainti(f|ve)|peureu(x|se))** sélectionnera *craintif, craintive, peureux, peureuse*.
- Symboles de modification.
- Le symbole *\** placé derrière un caractère ou un groupe signifie que ce caractère ou groupe peut être absent ou se répéter un nombre quelconque de fois. Ce symbole peut se placer derrière n'importe quel caractère ou groupe, mais il est en fait très pratique lorsqu'il est associé au symbole *.* (point) désignant n'importe quel caractère.  
Ex. : **.\*form.\*** sélectionnera les mots contenant la chaîne "form" tels que *formée, réforme, informel, etc.*  
Ex. : **bla(bla)\*** sélectionnera *bla, blabla, blablabla, ...*
- Le symbole *+* placé derrière un caractère ou un groupe signifie que ce caractère ou groupe peut se répéter un nombre quelconque de fois (mais au moins une fois).  
Ex. : **.\*form.\*** sélectionnera les mots contenant la chaîne "form" précédée d'au moins un caractère.  
Ex. : **bla(bla)+** sélectionnera *blabla, blablabla, ...*
- Le symbole *?* placé derrière un caractère ou un groupe signifie que ce caractère ou groupe est optionnel.  
Ex. : **a?politiques?** sélectionnera les mots *politique, politiques, apolitique et apolitiques*.

Ex. **cinéma(tograph(e|ique))s?** sélectionnera : *cinéma, cinémas, cinématographe, cinématographes, cinématographique, cinématographiques.*

## 2.5. Stella et l'Académie

Nous parlons ici des Dictionnaires de l'Académie Française, 8<sup>ème</sup> et 9<sup>ème</sup> éditions, qui sont gérés par Stella, et que l'on peut atteindre, soit directement, soit par hyper-navigation.

L'aide en ligne est en cours d'harmonisation avec l'aide en ligne de Frantext et du TLFi. Le principe d'interrogation repose sur le fait que l'on peut rechercher des articles comprenant un champ ayant un contenu textuel imposé, ou contenant simultanément plusieurs champs ayant un contenu textuel imposé. Pour interroger efficacement le dictionnaire, il convient de choisir soigneusement le (ou les) champs que l'on veut interroger, et de lui (ou leur) imposer le contenu textuel voulu. Voir pour cela les aides à la recherche qui expliquent comment procéder (expressions Stella, expressions régulières, sélections multiples, expressions de fourchette). Une recherche phonétique est possible, tout comme dans Frantext et le TLFi.

La personnalisation de l'environnement est la même pour les deux éditions.

## 3. Exemples d'exploitation de nos ressources

### 3.1. Dans les deux bases Frantext :

#### 3.1.1. Recherche d'occurrences

##### 3.1.1.1. Recherche simples

Dans Frantext intégral, on peut rechercher un mot, une séquence de mots. Par exemple : les occurrences du mot *mer* chez Balzac ou la séquence *oreille de lièvre* dans la totalité de la base (demander séquence 1 = *&moreille de &mlièvre* pour rechercher ces occurrences au singulier ou au pluriel) Il y a des solutions ! Dans Frantext catégorisé, on peut rechercher un mot associé à un code grammatical donné. Par exemple, demander les occurrences du substantif *déjeuner* (séquence 1 = *&e(g=S,c=déjeuner)*, ce qui amène 3338 réponses

##### 3.1.1.2. Recherches par liste de mots

###### *Recherche des formes d'un verbe*

On peut vouloir par ce biais rechercher toutes les formes du verbe *gésir*, puis, visualiser cette liste, et faire une recherche dans Frantext à partir de cette liste pour confronter les résultats à la liste théorique...

###### *Recherche des adjectifs de couleur*

A partir d'une liste manuelle contenant les adjectifs de couleurs..

##### 3.1.1.3. Recherche par grammaire

La grammaire G4 est la reprise de la grammaire détaillée au §2.3.2.3, pour retrouver toutes les formes du verbe "*se plaindre*" dans les phrases affirmatives et négatives chez *Hugo*

1) sélectionner le corpus Hugo

2) écrire la grammaire suivante, et la sauvegarder sous le nom de G4

*preamb\_aff :*

**je(me|m')|tu(te|t')|(se|s')nous nous|vous vous**

*t\_simpl\_aff* :  
&rpreamb\_aff&cplaindre  
*t\_comp\_aff* :  
&rpreamb\_aff&cêtre&rpart\_pass  
*part\_pass* :  
plaint|plainte|plaintes|plaints  
*preamb\_neg* :  
je ne(me|m')|tu ne(te|t')|ne (se|s')nous ne nous|vous ne vous  
*t\_simpl\_neg* :  
&rpreamb\_neg&cplaindre&rfin\_neg  
*t\_comp\_neg* :  
&rpreamb\_neg&cêtre&rfin\_neg&rpart\_pass  
*fin\_neg* :  
pas|plus|jamais|mie|guère|point  
*usage\_pronominal* :  
&rt\_simpl\_aff|&rt\_comp\_aff|&rt\_simpl\_neg|&rt\_comp\_neg

Lors de la requête, demander en séquence 1 : &rusage\_pronominal,G4

### 3.1.2. Recherche de cooccurrences

#### 3.1.2.1. S'intéresse-t-on aux fruits ?

On peut demander les cooccurrences de *abricot* et *cerise* dans une même phrase (avec séquence 1 = &mabricot et séquence 2 = &mcerise )

#### 3.1.2.2. S'intéresse-t-on aux constructions factitives ?

Dans Frantext catégorisé, on peut demander les cooccurrences du verbe *faire* avec un verbe infinitif (séquence 1 = &cfaire &e(g=Inf), ce qui amène plus de 50000 réponses.

On peut aussi le faire en demandant comme suit :

Définition de la séquence 1 : &cfaire

Définition de la séquence 2 : &e(g=Inf, c=faire)

Définition de la séquence 3 : &e(g=Adv)

Position relative des séquences 1 et 2 : 1 avant2      Distance maximale : 2

Position relative des séquences 1 et 3 : 1 avant3      Distance maximale : 2

Position relative des séquences 2 et 3 : indifférente Distance maximale : 1

### 3.1.3. Extraction de sous-lexiques

L'utilisateur pourra s'essayer à utiliser le service *Calcul de fréquence* pour étudier le vocabulaire utilisé par Hugo dans "Les Travailleurs de la mer".

Après avoir défini le corpus de travail (auteur et titre), passer par le service **liste de mots**, "création à partir des graphies de la base", avec le critère .\* (= point suivi d'astérisque). Donner à cette liste un nom (mot-mer) et la sauvegarder (il y a 16600 graphies dans cette liste)..

Ensuite, retourner au menu principal pour aller vers **Calculs de fréquence**/"calcul de fréquence", et remplir le cadre2 du formulaire, en sélectionnant le nom de la liste à prendre en compte, et en choisissant une des options (par exemple par fréquence décroissante). En

cliquant sur le bouton *Calcul de fréquence*, on obtient la liste des mots utilisés, accompagnés de leur fréquence.

### 3.1.4. Etudes morphologiques

Par les listes de mots, on pourra repérer tous les mots de la base suffixés par *-ette* ou préfixés par *anti-*. Ou encore tous les mots composés à partir de *homme-*

### 3.1.5. Etudes de syntaxe locale

L'utilisateur pourra essayer d'écrire des grammaires pour rechercher :

#### 3.1.5.1. Des contextes/concordances

Par exemple des concordances mot -2 à mot+2 pour un mot donné par exemple le mot *effroi* :

a) *La grammaire G1 qui cherche les concordance -2 +2 d'un mot donné en paramètre*

*reg1 :*

**&q &q &1 &q &q**

et demander dans le formulaire de requêtes, en séquence 1 : *&rreg1(effroi),G1*  
(plus de 2200 réponses sur Frantext catégorisé.)

b) *La grammaire G2 qui sort les contextes du mot effroi suivi d'un adjectif....*

*reg3 :*

**&1 &e(g=&2)**

*reg4 :*

**&q &q &rreg3(&1,&2)&q &q**

demander dans le formulaire de requêtes, en séquence 1 *&rreg3(effroi, A),G2*

#### 3.1.5.2. Des schémas énumératifs

Il s'agit d'une grammaire G3 pour une étude d'énumérations (dans Frantext catégorisé). On pourra par exemple l'utiliser pour reconnaître des séquences itératives chez Hugo ou chez Zola.

La règle "**item**" définit l'item textuel qui va se répéter. Elle admet deux paramètres &1 et &2, remplacés par leurs valeurs effectives correspondantes lors de l'invocation de la règle.

Par exemple:

1) *&ritem(en,S)* invoque la règle item en passant "en" et "S" en paramètre. La règle item devient alors équivalente à "en &e(g=S)" ce qui signifie "en" suivi d'un substantif.

2) *&ritem(A)* invoque la règle item en passant un premier paramètre vide et "A" en second paramètre. La règle item devient alors équivalente à "&e(g=A)" ce qui désigne un adjectif.

La règle "**repetition**" exprime que l'item textuel doit se répéter deux fois ( sous-expression "*&ritem(&1,&2)* , *&ritem(&1,&2)*" plus un nombre (supérieur ou égal à 1) quelconque de fois (sous expression "*&+(, &ritem(&1,&2))*" ). L'item doit donc se répéter 3 fois ou plus. Elle permet de rechercher un item textuel se répétant au moins 3 fois avec une virgule intercalaire.

*item :*

**&1 &e(g=&2)**

*repetition :*

**&ritem(&1,&2) , &ritem(&1,&2) &+(, &ritem(&1,&2))**

Pour utiliser cette grammaire G3, il suffit d'invoquer la règle "repetition" en lui passant les deux paramètres qu'elle transmettra à la règle "item". Voici à titre d'exemple quelques résultats fournis par cette grammaire sur des textes de Zola :

&rrepetition(A),G3 détecte : "beau, jeune, amoureux, écervelé" (La curée);  
"liberticide, anticonstitutionnel, révolutionnaire" (Le ventre de Paris) ; "rouges, jaunes, mauves, blanches" (La faute de l'abbé Mouret), etc.

&rrepetition(en,S),G3 détecte : "en rumeurs, en rires, en applaudissements" (Nana) ;  
"en coeurs, en festons, en médaillons" (L'œuvre) ; "en vertu, en génie, en crime, en ivrognerie, en sainteté" (Le docteur Pascal), etc.

&rrepetition(V),G3 détecte : "chancelait, glissait, tombait" (La conquête de Plassans) ; "débattait, mordait, égratignait" (La joie de vivre) ; "rampait, descendait, remontait" (Germinal), etc.

### 3.1.5.3. Des schémas syntaxiques

Voici par exemple une grammaire G4, pour repérer toutes les séquences du genre *avoir* (*peur, faim, chaud, etc.*), avec possibilité d'avoir un adverbe ou un adjectif inséré entre les deux :

*avoir* :

**&e(g=V c=&cavoir)**

*inser* :

**&e(g=Adv A)**

*loc* :

**&ravoir &rinser &1**

et sur le formulaire de requête, demander en séquence 1 *&rloc(peur)*, G4 où le paramètre &1 est remplacé par la valeur *peur*

## 3.2. dans le TLFi

### 3.2.1. Recherche phonétique

L'utilisateur pourra rechercher à partir des deux moyens proposés, des séquences phonétiques telles que **étik**, ou **psikiatre**, ou **ouagon**

### 3.2.2. Visualisation simple

On pourra rechercher par exemple le mot **indulgence**. L'article n'est pas visible dans sa totalité. Il faut donc, de là, partir explorer le *Panneau de réglage*, puis, n'y ayant pas trouvé la solution, retourner au menu principal pour accéder au service *Personnalisation du fonctionnement*, où l'on trouve ce qui nous intéresse, c'est à dire la *Taille des contextes*.

A partir de la visualisation de l'article, on pourra aussi jouer avec l'*hyper-navigation*.

### 3.2.3. Interrogation assistée

On peut demander par exemple quelles entrées ont reçu un code de *verbe transitif et* dans un emploi *populaire*, ont une *définition* qui contient le mot *manger*. 10 réponses instructives !

### 3.2.4. Interrogation avancée

#### 3.2.4.1. Etude de vocabulaire chez un auteur :

Existe-t-il des exemples de Hugo contenant le mot « **contemplation** » (au singulier ou au pluriel), mais **PAS** extraits des Contemplations ?)

Objet 1 = exemple, contenu = &mcontemplation

Objet 2 = auteur, lien=inclus dans objet 1, contenu = hugo

Objet 3 = titre d'exemple, lien = inclus dans objet 1, contenu= &ncontemplations

5 réponses. Visualiser la 3, puis lancer l'**hyper-navigation** vers Frantext domaine public.

Ce mot a été utilisé par 133 auteurs dont la liste s'affiche.

Si on clique sur Hugo => œuvres dans lesquelles ce mot a été employé, dont *Les Contemplations*. Les contextes s'affichent dans la fenêtre.

#### 3.2.4.2. Etude « morpho-sémantique » :

On cherche les entrées terminées en **-oir**, dont les définitions contiennent le mot « **outil** » en position 1 ou 2.

On voudrait les entrées en **-oir**, contenant &d2 outil. Mais il est impossible de demander directement Objet 1 = **Entrée**, contenu = **.\*oir**. Il faut passer par les listes de mots.

Donc **Gestion de liste de mots** à partir du TLFi, critère = **.\*oir**, nom de liste = **oir** (593 graphies).

Puis ensuite **Recherche complexe** avec

Objet 1 = **entrée**, contenu = **&loir**

Objet 2 = **définition**, lien = **dépendant** de objet 1, contenu = **&d2 outil** (54 réponses)

#### 3.2.4.3. Etude de vocabulaire chez un auteur (bis) :

On cherche les entrées dont les exemples enchaînés contiennent une **forme conjuguée** du verbe aimer

Attention : &caimer donne toutes les formes du verbe, conjuguées ou non. Une requête faite simplement sur **&caimer** amène 4036 réponses.

Pour restreindre la requête aux formes conjuguées, on passe par les **requêtes complexes**. Il y a plusieurs façons d'afficher le contenu : soit en listant les formes non désirées, ce qui est un peu fastidieux, soit en utilisant un raccourci.

Voici trois variantes :

- On peut écrire la **requête complexe** de la façon suivante :

Objet 1 = **entrée**

Objet 2 = **exemple enchaîné**, lien = **dépendant** de objet 1, avec

contenu = **&caimer | &naimer | &naimant** suivant les formes à éliminer

(l'objet 2 doit contenir les formes de aimer, **et** non les formes précisées = l'infinitif et la forme en **-ant**. Dans ce cas précis, on obtient 3296 réponses (= les formes conjuguées et les participes passés).

- On peut aussi écrire cette **requête complexe** de la façon suivante :  
Objet 1 = *entrée*  
Objet 2 = *exemple enchaîné*, lien = *dépendant* de objet 1, avec contenu = *&caimer | &n &maimer*, ce qui élimine toutes les formes non conjuguées.  
On obtient 2600 résultats.
- Variante de ci-dessus : recherche des formes conjuguées du verbe aimer dans les exemples de Colette :  
Objet 1 = *exemple enchaîné*, contenu = *&caimer | &n &maimer*  
Objet 2 = *auteur d'exemple*, lien = *inclus* dans objet 1, avec contenu = *colette*  
Il y a 56 réponses.

#### 3.2.4.4. Recherche de schémas syntaxiques :

*Par exemple, on cherche les verbes qui se construisent avec la préposition « de ».)*

Voici deux variantes pour repérer tous les verbes se construisant avec une complémentation en *de*

1<sup>ère</sup> variante, qui amène 29 solutions

objet 1 = *entrée*

objet 2 = *code grammatical*, lien = *inclus* dans objet 1, contenu = *verbe pronominal*

objet 3 = *construction*, lien = *dépendant* de 2, contenu = *de*

2<sup>ème</sup> variante, qui amène 9 solutions

objet 1 = *entrée*

objet 2 = *code grammatical*, lien = *inclus dans objet 1*, contenu = *verbe pronominal*

objet 3 = *construction*, lien = *dépendant de 2*, contenu = *de+*

Il est intéressant de comparer les deux résultats, pour comprendre comment raffiner ses requêtes.

## 4. Perspectives

Frantext, TLFi, Dictionnaires de l'Académie informatisés et logiciel d'exploitation Stella forment un ensemble, mais ne sont pas fermés sur eux-mêmes. Cet ensemble de ressources (corpus et dictionnaires) ouvert à toutes sortes d'utilisations dans tous les domaines, ne représente qu'une partie des richesses de l'ATILF dont on peut trouver une présentation plus générale dans (Bernard et al., 2001).

Cet ensemble de ressources est mis à la disposition de la Communauté scientifique car il peut être le fondement de projets divers et variés, en particulier la constitution de lexiques dérivés, à partir d'études portant sur les synonymes, ou sur les collocations. Plusieurs projets de collaboration sont à l'étude, entre l'ATILF et des partenaires intéressés, visant de nouveaux objectifs scientifiques à partir des ressources offertes. Conscients que ces ressources constituent un patrimoine important financé essentiellement par le CNRS, nous avons le souhait de les mettre le plus possible à disposition de la communauté de recherche à travers des projets coopératifs de recherche. Toute nouvelle proposition de coopération, portant tant

sur le contenu de ces ressources que sur l'interface utilisateur des systèmes proposés est vivement souhaitée et sera étudiée avec soin.

## Références

Bernard P., Bernet C., Dendien J, Pierrel J.M., Souvay G et Tucsnak Z (2001) « Un serveur de ressources linguistiques informatisées via le Web », *Actes de TALN 2001*, Tours, Juillet 2001, pages 333-338.

Bernard P., Dendien J, Lecomte J., Pierrel J.M. (2002) Les ressources de l'ATILF pour l'analyse lexicale et textuelle : TLFi, Frantext et le logiciel Stella, *8èmes Journées Internationales d'Analyse Statistique des Données Textuelles JADT 2002*, pages 137-149

Bernard P., Lecomte J., Dendien J, Pierrel J.M. (2002) Computerized linguistic resources of the research laboratory ATILF for lexical and textual analysis: TLFi, Frantext and the software Stella, *LREC-2002*.

Dendien J. (1991) *Access to information in a textual database: access functions and optimal indexes*, Oxford, Clarendon press.

Dendien J. (1996). Le projet d'informatisation du TLF, in *Lexicographie et informatique*, pages 25-34.

Leroy-Turcan I.(1998). Balisage formel ou balisage fin pour les dictionnaires anciens informatisés: objectifs et implications méthodologiques. *DictA1998*, at <http://www.inalf.fr/academie8>, sous *Atelier de balisage de Limoges*

CNRS (1976-1994) *TLF, Dictionnaire de la langue du 19e et 20e siècle*, CNRS, Gallimard, Paris.