

# Interfaces dynamiques de fouilles textuelles Dynamic Tools to Rummage through Textual Data

**Javier Couto, Jean-Luc Minel**  
Laboratoire LALICC  
UMR 8139  
(CNRS, Université Paris-Sorbonne)  
96, bd Raspail, 75006- Paris France

Javier.Couto@paris4.sorbonne.fr  
Jean-Luc.Minel@paris4.sorbonne.fr

## Résumé

Dans cet article nous nous intéresserons essentiellement au développement des interactions entre l'utilisateur et un système de fouilles textuelles, en construisant des outils de navigation textuelle. Tout d'abord nous décrirons les modèles et des outils de représentation d'un texte sur lesquels nous nous appuyons. Ensuite nous présenterons les outils de navigation puis le concept de cartouche de navigation qui encapsule les connaissances navigationnelles. Les expérimentations menées à l'aide d'un premier prototype nous ont permis de constater l'intérêt de cette approche.

Mots Clés : Représentation formelle d'un texte, navigation textuelle.

## Abstract

In this article we will be essentially interested in the development of the interactions between the user and a system to rummage through textual data by building tools for textual navigation. We will describe first the text model and the tools on which we rely on. Then we will present navigation tools and the navigation cartridge concept that encapsulates the navigational knowledge. The experimentations with an initial prototype have shown this approach promising.

Keywords : Formal representation of a text, textual navigation.

## 1. Introduction

Dans le domaine de la recherche d'information textuelle, un grand nombre de travaux de recherche se sont focalisés soit :

- sur les outils qui permettent de retrouver un ou plusieurs documents dans une collection de documents (Chen & al. 1998), ou plus spécifiquement, de proposer une réponse à une requête d'un utilisateur. L'importance qu'ont pris ces dernières années les travaux dédiés aux systèmes de Question/Réponses en constitue une excellente illustration (Lin & Katz 2003) ;
- sur les systèmes qui permettent de sélectionner des parties saillantes dans un texte. Les systèmes de résumé et de filtrage automatique constituent un bon exemple de ce type d'approche (Kan & al. 1999, Mani 2001, Minel 2003).

Une des caractéristiques communes à ces différentes approches est le peu d'importance apporté aux modes de représentation visuelle du résultat : des phrases qui « répondent » à une question, un fragment textuel qui « résume » un texte. En fait, le résultat produit reste imprégné des contraintes imposées par une conception marquée par la prégnance technologique qui assimile texte et texte

imprimé (Vandendorpe 1999). Certains travaux, notamment (Hearst 1999, Jacquemin & al. 2002), ont proposé de développer des interfaces qui transforment les représentations visuelles, mais ces transformations ne s'appuient pas sur les marques linguistiques, c'est-à-dire plus précisément, sur les structures discursives construites par le scripteur.

Nous pensons que la fouille de textes peut être considérée comme un processus de recherche d'information guidé par les besoins spécifiques de l'utilisateur et par la prise en compte, contextualisée, des structures discursives dans lesquelles l'information recherchée est mise en discours, et que des outils de visualisation et de navigation textuelle qui exploitent ces structures peuvent être développés pour guider le lecteur dans sa fouille textuelle.

Le développement de ces outils de fouille de textes s'insère dans les travaux menés au sein du laboratoire LaLICC et englobe:

- le développement d'un cadre méthodologique fondé sur l'exploration contextuelle ( Desclés & al. 1997) ;
- la construction des modèles et des outils de représentation d'un texte (Minel & al. 2001, Crispino 2003);
- l'étude fine de certaines structures discursives (Porhiel 2001, Jackiewicz 2002) ;
- le développement d'une plate-forme informatique, ContextO, qui nous permet de valider, ou non, les modèles et les connaissances linguistiques associées (Crispino & al. 1999, Ben Hazez & al. 2001, Minel & al. 2001);
- le développement d'outils de navigation et de gestion des interactions entre l'utilisateur et le système de filtrage sémantique (Couto 2001, Minel 2002).

Dans cet article nous nous intéresserons essentiellement au dernier point. Néanmoins, nous rappellerons tout d'abord certains choix concernant la représentation d'un texte, choix qui influent sur les possibilités navigationnelles, puis nous présenterons les interfaces dynamiques développées dans le cadre du projet Regal (Ferret & al. 2001, Couto & al. 2004) pour illustrer nos hypothèses. Nous terminerons par la description du concept de *cartouche de navigation* qui est au centre de la plate-forme de navigation en cours de réalisation.

## **2. Modèles et outils de représentation d'un texte**

L'étude de certains phénomènes linguistiques implique la nécessité de prendre en compte la dimension textuelle, c'est-à-dire qu'il faut faire appel au contexte extérieur à la phrase pour les expliquer (Charolles 1997). Les connecteurs qui marquent la progression thématique, les mécanismes de prise en charge énonciative, les différentes formes de reprise référentielle (pronoms, déictiques, anaphores, cataphores), les chaînes de liage, parmi d'autres, exigent le traitement de structures du texte qui vont souvent au-delà de la phrase. Pour pouvoir exploiter dans les logiciels de TALN les recherches qui sont menées dans cette direction, il est nécessaire de disposer d'outils formels qui permettent de travailler de manière systématique et consistante dans une dimension textuelle.

Nous décrivons brièvement dans cette partie sur quelle représentation nous nous appuyons pour modéliser un texte, et quels sont les outils formels qui vont permettre à un linguiste de manipuler cette représentation. Notre préoccupation vise à rendre le travail linguistique indépendant de l'implémentation informatique, dans un langage qui fournisse un bon pouvoir expressif. Néanmoins, ce langage formel doit être facilement traduisible dans un langage de programmation.

Notre représentation d'un texte s'appuie sur les propositions de (Crispino 2003) afin de modéliser la structure logique de la manière la plus générale possible en incluant les éléments textuels qui peuvent être le mieux exploités dans notre projet. Notre vision d'un texte est celle d'une hiérarchie simple

d'éléments textuels. Bien qu'il soit fréquent de trouver des textes qui présentent une hiérarchie plus complexe, nous pensons que notre option fournit, d'une part, un cadre suffisant pour la formulation des connaissances linguistiques exprimables dans notre méthode et d'autre part, elle constitue un modèle de base, qui est facilement extensible pour traiter des phénomènes plus complexes.

Nous modélisons cette hiérarchie par une structure d'arbre dont la racine représente le texte complet. Les descendants directs de cette racine représentent les sections de premier niveau dans le texte. Ces sections peuvent avoir comme descendants d'autres sections de niveau inférieur (sous-sections) ou des paragraphes. Les descendants de ces paragraphes sont les nœuds qui modélisent les phrases et finalement, les descendants des phrases, les feuilles de l'arbre, sont les unités lexicales. Ainsi, le parcours de gauche à droite de cette hiérarchie correspond à la lecture linéaire du texte.

Pour manipuler cette structure, un langage (Crispino 2003) permet aux linguistes de spécifier des connaissances linguistiques (Ben Hazez & al. 2000, 2001). Enfin, rappelons que c'est cette structure hiérarchique qui est décorée par l'étiqueteur sémantique de la plate-forme ContextO (Minel & al. 2001). Cet étiquetage est réalisé de la manière suivante<sup>1</sup> : pour chacune des phrases qui composent le texte analysé, l'étiqueteur sémantique adresse une requête au gestionnaire des connaissances linguistiques afin que celui-ci identifie les positions des indicateurs et des indices et le nom des règles d'exploration contextuelle qui peuvent être déclenchées. Chaque règle, encodée comme une méthode d'une classe Java, est exécutée. L'ordre de leur déclenchement, pour une tâche donnée, est indifférent, les règles sont en effet considérées comme indépendantes. Ce mode de fonctionnement correspond à l'hypothèse, jusqu'ici vérifiée, que pour une tâche donnée certains marqueurs sémantiques ne sont pas exclusifs entre eux.

### **3. Outils de navigation**

Nos principes de navigation se distinguent de la navigation hypertextuelle. En effet, c'est sur les marques sémiotiques et linguistiques que vont s'appuyer nos opérations de navigation. L'identification de ces marques est le résultat d'une analyse automatique (Porhiel 2001, Jackiewicz 2002), analyse qui produit une ou plusieurs structures discursives qui viennent enrichir la représentation hiérarchique du texte (Jackiewicz & Minel 2003). La navigation proposée n'est donc pas guidée par l'auteur du texte, comme dans le cas de la navigation hypertexte où les hyperliens sont placés par cet auteur. Ainsi il y a potentiellement pour un même texte une multiplicité de parcours de navigation que l'on peut néanmoins comparer au cheminement déambulatoire, dans les hyperdocuments, proposé par (Géry 2002).

Notre conception des outils de navigation, dans le cadre de la fouille de textes, repose sur un principe général concernant l'objet textuel à produire : à partir de l'objet textuel source  $T$ , le résultat de la fouille de texte est un nouvel objet textuel  $T_f$  qui intègre l'objet  $T$  et une représentation décorée de  $T$  appelée  $T_d$ . Par conséquent, un déplacement, par l'utilisateur, dans  $T_d$  doit toujours être répercuté, par le système de navigation, dans  $T$ . A chacun de ces objets textuels va être associée une forme sémiotique  $S_d$  qui visualise sur l'écran l'objet  $T_d$ , et des opérations de navigation  $O_d$  qui vont permettre de se déplacer dans  $T_d$ .

$$T_f = f(T, T_d, S_d, O_d)$$

#### **3.1. Modèle général des outils**

La figure 1 illustre le modèle général de l'écran de navigation. On y retrouve les principes fondamentaux de toute Interface Utilisateur (GUI) (Shneiderman 1998, Hearst 1999) que nous supposons connus.

---

<sup>1</sup> Le lecteur trouvera une explication détaillée du processus d'étiquetage sémantique dans (Crispino 2003).

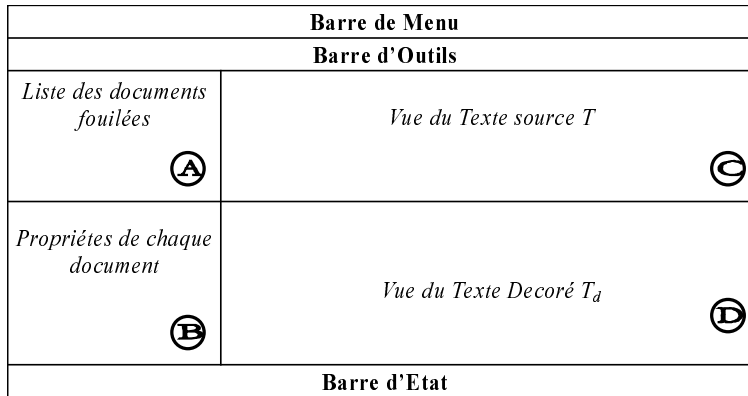


Figure 1 : Modèle général de présentation

L'écran de navigation se compose, en sus des barres standards, de quatre fenêtres, A, B, C, D. La fenêtre A visualise la liste des documents choisis par l'utilisateur. La fenêtre B visualise les propriétés du texte source {T}. Ce sont des propriétés structurales (nombre de titres, paragraphes, phrases, etc.) ou issues de la fouille de texte (nombre de phrases décorées, nombre d'indicateurs repérés, etc.). La fenêtre C visualise le texte source en cours de fouille et la fenêtre D visualise le fragment textuel qui correspond au couple {T<sub>f</sub>, S<sub>d</sub>} construit à partir de l'application d'une tâche et d'un profil de fouille choisis par l'utilisateur. Un profil correspond au choix de différentes décorations pour une tâche donnée (Minel & al. 2001).

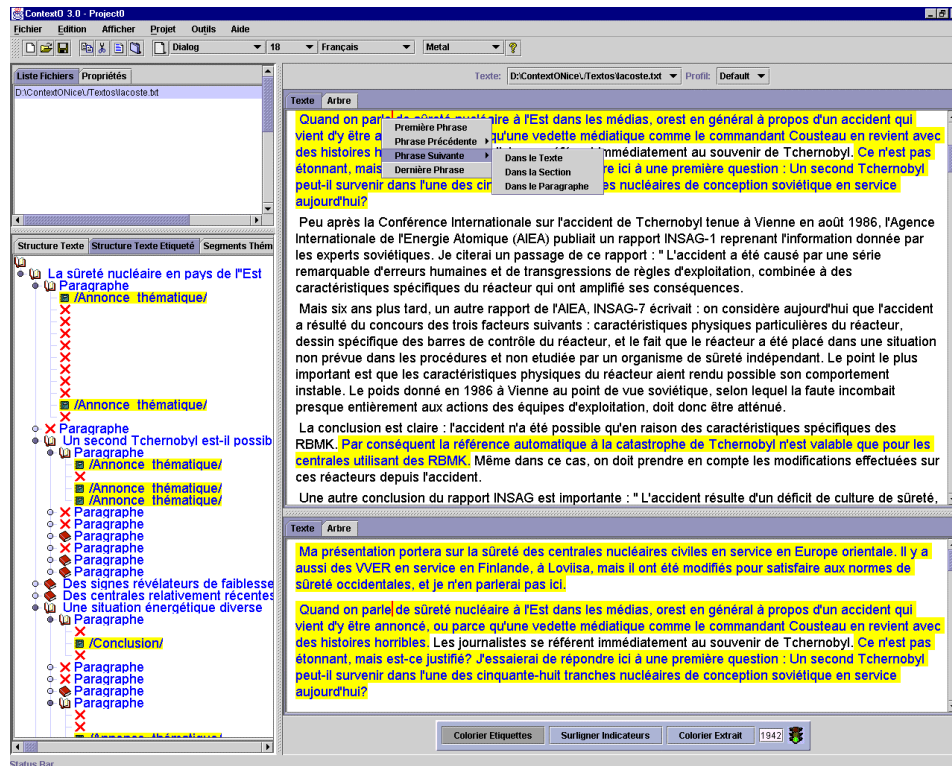


Figure 2 : Exemple d'interfaces dynamique

Des options de visualisation (les deux fenêtres intitulées *Texte* et *Profil* sur la figure 3, §3.2) sont offertes pour permettre à l'utilisateur de visualiser plusieurs documents avec pour chaque document,

des profils différents, suivant le principe du “mille-feuilles”. La dynamique entre les deux fenêtres permet ainsi de visualiser constamment le contexte d’une phrase du fragment textuel, indépendamment du type de la vue affichée (linéaire ou structurée). D’autre part, la visualisation des informations saillantes repose sur des principes d’ergonomie qui utilisent les possibilités offertes par la colorisation (Murch, 1985). La saillance et les décorations sémantiques sont traduites par des couleurs choisies par l’utilisateur. Un menu contextuel, qui correspond aux opérations  $\{O_d\}$ , permet d’afficher la ou les étiquettes attribuées et de se déplacer dans le texte, en précisant la portée relative (*dans un paragraphe, dans une section, dans le texte*) ou absolue (*première ou dernière phrase du texte*) de la recherche, afin de rechercher les phrases qui sont annotées de manière identique. Ces opérations permettent ainsi de rechercher des segments textuels en fonction de leur finalité discursive (annonce thématique, conclusive, etc.).

Enfin, l’utilisateur pouvant étudier le résultat de la fouille de plusieurs textes, nous avons développé un ensemble de formes sémiotiques  $\{S_d\}$  qui exploitent les résultats quantitatifs calculés à partir des décorations. Celles-ci peuvent permettre de comparer différents genres textuels ou plus spécifiquement la distribution de certaines marques sémantiques (Couto 2001).

Toutes ces fonctionnalités ont été développées dans un composant autonome en langage Java, en s’appuyant sur les principes de conception d’un Bean<sup>2</sup>.

### 3.2. Spécialisation des outils

Dans le cadre du projet Regal<sup>3</sup> (Ferret & al. 2001, Couto & al. 2004), développé en collaboration avec le LATTICE, le LIMSI et le CEA, nous avons pu mettre à l’épreuve nos choix de visualisation. Les informations à exploiter sont le résultat d’un processus d’intégration entre deux approches, l’une fondée sur un calcul de cohésion lexicale et l’autre sur le repérage de marques linguistiques. L’écran de navigation (cf. figure 3) se compose des barres standards et de quatre fenêtres, A, B, C, D décrites précédemment, mais il offre également de nouvelles fonctionnalités.

La fenêtre B offre différentes vues décorées  $\{T_d, S_d\}$  du texte source  $\{T\}$ , accessibles par des onglets spécifiques. Ces décorations sont le résultat des calculs effectués (repérage des segments thématiques, fréquence des termes saillants, étiquette discursive attribuée à une phrase, etc.) et offre des points d’accès aux différentes parties du texte. Chaque segment thématique identifié est représenté sous la forme d’un rectangle coloré, les segments liés à un même thème étant de la même couleur. Sa taille en nombre de phrases est disponible ce qui permet à l’utilisateur d’évaluer l’importance de ce segment ou de ce thème par rapport aux autres thèmes du texte. La visualisation de la structure permet de mieux indiquer l’organisation des segments entre eux. L’inclusion d’un segment dans un autre signifie qu’il y a un lien thématique entre eux et permet de visualiser ainsi sa place par rapport aux autres thèmes. Pour chaque segment, il est possible d’afficher la liste des termes saillants avec leur fréquence. D’autres onglets permettent d’obtenir une vue hiérarchique du texte, organisée en arbre ou une vue globale des annotations attribuées aux phrases. Cette fenêtre offre donc différentes formes sémiotiques du texte : selon les thèmes, selon les marques discursives ou selon sa structure logique. Il est ainsi possible de prendre connaissance globalement du contenu du texte sans avoir à le lire et d’approfondir seulement certains points si des parties semblent intéressantes.

---

<sup>2</sup> Néanmoins ce composant ne constitue pas un JavaBean au sens strict.

<sup>3</sup> Le projet RÉGAL a reçu le soutien de l’ACI Cognitive (LAC038).

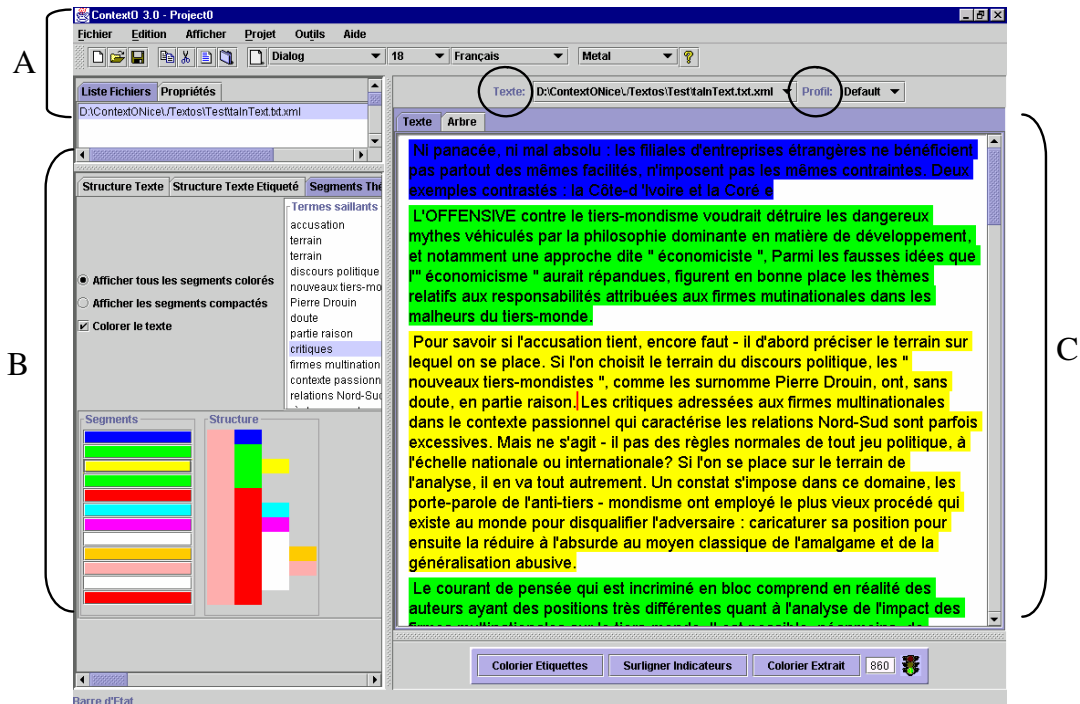


Figure 3 : Exemple d'interface dynamique spécialisée (extrait de Couto & al. 2004)

#### 4. Cartouche de navigation

Nous avons décrit précédemment des outils dynamiques développés, afin de gérer les interactions avec un lecteur. Si ces outils proposent certaines formes sémiotiques que l'on peut qualifier de génériques, par contre les opérations de parcours du texte sont plutôt spécifiques, c'est à dire liées à un type de structure discursive, comme, par exemple, les segments thématiques du projet Regal.

A partir de ce constat, nous cherchons à développer une plate-forme informatique qui offre, pour un concepteur d'outils de navigation, des possibilités de déployer des connaissances navigationnelles. La seule contrainte est que les trajectoires de navigation doivent s'appuyer soit :

- sur l'étiquetage de segments textuels, les nœuds les feuilles de la représentation hiérarchique du texte ;
- sur l'étiquetage d'arcs entre les segments textuels qui appartiennent à cette représentation hiérarchique.

La solution proposée par J. Couto (Couto 2003) consiste à encapsuler les connaissances navigationnelles dans des composants logiciels appelés *cartouches de navigation*. Chaque *cartouche de navigation* identifie un ensemble de N vues avec les paramètres et les contraintes de création et les opérations de navigation possibles. Une vue est définie par :

- sa représentation visuelle (texte plat, arborescente, etc.) ;
- ses paramètres (selon le type de représentation) ;
- ses contraintes (par exemple: conditions sur des unités textuelles et/ou des attributs) ;
- des opérations de navigation.

Le contenu de ces cartouches est défini par un langage formel, fondé sur une syntaxe de type langage objet, qui permet d'exprimer les connaissances navigationnelles. Ces connaissances s'appuient sur le modèle du texte décrit dans un format XML et sur un ensemble de possibles opérations de visualisation et navigation. Dans le texte, chaque unité textuelle est balisée et son type est précisé, comme le montre l'exemple ci-dessous.

```

<UT Type="Paragraphe" Nro="9">
  <UT Type="Phrase" Nro="29">
    <Chaîne>Une remarque essentielle est que parmi les énergies actuellement
    utilisées à grande échelle seul le nucléaire et l'hydraulique ne sont pas
    concernés.</Chaîne>
  </UT>
  <UT Type="Phrase" Nro="30">
    <Attribut Type="EtSem">Recapitulation</Attribut>
    <Chaîne>Ceci est notre première conclusion.</Chaîne>
  </UT>
</UT>

```

Figure 4 : Exemple d'un morceau de texte étiqueté

Dans cet exemple, les unités textuelles correspondant à une feuille dans l'arborescence textuelle possèdent un élément nommé *chaîne* qui représente la chaîne de caractères typographiques. De plus, chaque unité textuelle peut avoir un ou plusieurs attributs. Ces attributs sont typés et valués. Dans l'exemple ci-dessus, la phrase numéro 30 est annotée comme une «*récapitulation*».

Les opérations de parcours sont ensuite décrites comme un chemin dans un graphe. Ce graphe est constitué des sommets qui sont les segments textuels décorés :

**Sommet** *n1* = **Sommet**({{Element = "UT", Type = "Phrase"}}Avec{Element = "Attribut", Type = "EtSem", Valeur = "Annonce\_thématique"}));

Le langage offre la possibilité de guider ce parcours en spécifiant l'orientation du cheminement dans le graphe :

**Ensemble** *ops\_navigation* = **Ensemble**(*graphe\_nav\_ex1*);  
**Graphe** *graphe\_nav\_ex1* = **Graphe**(*sommets*, *arcs*);  
**Ensemble** *sommets* = **Ensemble**(*n1*, *n2*, *n3*);  
**Ensemble** *arcs* = **Ensemble**(**Arc**(*n1*, *n2*), **Arc**(*n1*, *n3*), **Arc**(*n2*, *n3*), **Arc**(*n3*, *n2*));

Cette suite d'instructions commence d'abord par construire un objet nommé *ops\_navigation* qui est un ensemble constitué d'un seul élément, nommé *graphe\_nav\_ex1*. Ce graphe est décrit par ses sommets et ses arcs. Les sommets sont énumérés et chaque arc décrit le lien entre deux sommets.

La figure 5 illustre une utilisation de ce langage, dans une cartouche de navigation qui spécifie un cheminement pour des segments textuels étiquetés *Annonce\_thématique*, *Conclusion* et *Récapitulation*.

```

Vue texte_arbre = Vue(ARBRE_CONTENU,
    « Vue arborescente du texte »,
    params,
    contraintes,
    ops_navigation);

Ensemble params = Ensemble({Montrer_Racine, faux},
    Montrer_toutes, vrai);

Ensemble contraintes = Ensemble({{Element = "U" } Avec
    {Element = "Attribut", Type = "Filtrat", Valeur = "Ex1"}});

Ensemble ops_navigation = Ensemble(graphe_nav_ex1);

Graphe graphe_nav_ex1 = Graphe(sommets, arcs);

Ensemble sommets = Ensemble(n1, n2, n3);
Ensemble arcs = Ensemble(Arc(n1, n2), Arc(n1, n3), Arc(n2, n3), Arc(n3, n2));

Sommet n1 = Sommet({{Element = "UT", Type = "Phrase"} Avec {Element = "Attribut", Type =
    "EtSem", Valeur = "Annonce_thématique"}});
Sommet n2 = Sommet({{Element = vUT", Type = "Phrase"} Avec {Element = "Attribut", Type =
    "EtSem", Valeur = "Conclusion"}});
Sommet n3 = Sommet({Element = "UT", Type = vPhrase"} Avec { Element = "Attribut", Type =
    "EtSem", Valeur = "Récapitulation"});

```

Figure 5 : Exemple de contenu d'une cartouche de navigation

Un premier prototype développé en langage Java, dont l'architecture est illustrée par la figure 6, nous a permis de vérifier la validité de nos hypothèses. La plate-forme est ainsi composée de différents modules dont :

- un module qui se charge de construire à partir d'un texte balisé produit, par exemple par la plate-forme ContextO, la représentation décorée  $\{T_d\}$  du texte ;
- un module de visualisation qui gère les interactions avec l'utilisateur en chargeant et en compilant à la demande une ou plusieurs cartouches de navigation. Le résultat de la compilation est un graphe de parcours qui est projeté sur la représentation décorée  $\{T_d\}$ . La visualisation effective est assurée par la sélection dans la base des modèles d'une ou de plusieurs formes sémiotiques  $\{S_d\}$  spécifiées dans la cartouche.

Nous travaillons actuellement à un langage qui possède des fonctions de navigation prenant en compte des organisations textuelles plus complexes.

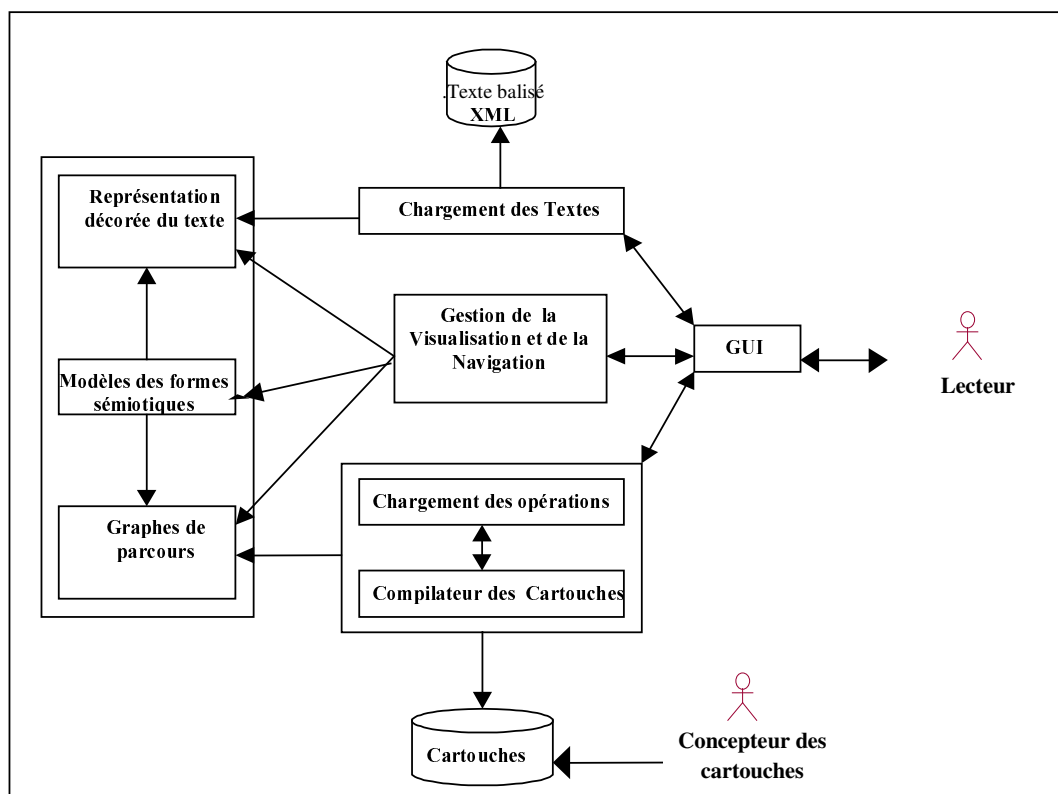


Figure 6 : Architecture du prototype de la plate-forme

#### 4. Conclusion

Les principes de navigation et de visualisation que nous avons décrits mettent en avant le principe de l'interaction plutôt que celui d'une focalisation par granularité de plus en plus fine comme dans (Boguraev & al., 2001). Nous pensons en effet que l'activité consistant à sélectionner des informations à l'intérieur d'un texte donné est une démarche qui mobilise de multiples fonctions cognitives et qui varie en fonction des sujets, de leur expertise du domaine traité ainsi que de leur degré d'attention au moment où ils en prennent connaissance. Si la modélisation de ces fonctions n'est pas actuellement possible, du moins dans toute leur complexité, en revanche, nous pouvons viser la construction de représentations d'un texte qui fournissent à l'utilisateur des repères sur les organisations discursives qui structurent le texte.

Ce repérage permettra à l'utilisateur de déployer ses propres stratégies de **recherche d'informations** et de ne pas subir la linéarité du texte (Harris 1993) imposée, entre autre, par la technologie de l'imprimé. Notre projet vise ainsi le développement d'une boîte à outils permettant la navigation selon des critères conçus par un concepteur (linguiste, documentaliste, analyste) de cartouches de navigation. Les connaissances exprimées dans ces cartouches doivent donc être considérées comme une aide à la lecture, voire une suggestion de lecture. Étant donné que la conception des cartouches s'appuie sur certaines informations déjà présentes dans les documents la qualité de la visualisation et de la navigation repose sur la compétence du linguiste mais aussi sur la richesse du langage de représentation.

Nous avons développé un premier prototype informatique afin de tester les idées proposées. Les expérimentations menées nous ont permis de constater l'intérêt de cette approche. Néanmoins nous avons aussi pu constater que le gain, pour l'utilisateur, ne se mesure pas nécessairement par un gain en temps de recherche de l'information, mais plutôt en termes de qualité de lecture. Ce constat rend le

processus d'évaluation difficile car les critères d'évaluation sont multiples. Deux aspects sont en effet à évaluer : le *langage* des cartouches offert aux concepteurs et les connaissances des *cartouches* développées. Néanmoins nous travaillons à l'élaboration d'un protocole d'évaluation concernant à la fois le processus de conception des cartouches par les concepteurs et l'utilité vis-à-vis de la lecture par l'utilisateur.

## Références

- Ben Hazez, S., J.-P. Desclés, J.-L. Minel. (2001). Modèle d'exploration contextuelle pour l'analyse sémantique des textes, *TALN 2001*, Tours, p.73-82.
- Ben Hazez S., J.-L. Minel. (2000). Designing Tasks of Identification of Complex Patterns Used for Text Filtering, *RIA0'2000*, Paris, p. 1558-1567.
- Boguraev B. C. Kennedy, R. Bellamy, S. Brawer, Y. Wong, J. Swartz. (2001). Dynamic Presentation of Document content for Rapid On-Line Skimming. *AAAI Spring Symposium on Intelligent Text Summarisation, Stanford, CA, March 1998*
- Charolles M. (1997). L'encadrement du discours : univers, champs, domaines et espaces, *Cahier de Recherche Linguistique, LANDISCO*, Université Nancy 2, p. 1-73.
- Chen H., Houston A., Sewell R., Schatz B. (1998). Internet browsing and searching: user evaluations of category map and concept space techniques, *Journal of the American Society for Information Science*, Special Issue on "AI Techniques for Emerging Information Systems Applications" 49 (7), p. 582-603.
- Couto J. (2001). ContextO, Los sistemas de exploracion contextual de cara al usuario, Mémoire de Master, Université de la République, Uruguay.
- Couto J. (2003). Développement d'une plate-forme de navigation textuelle, *Journée Doctorale de LaLICC*, Octobre 2003.
- Couto J., Ferret O., Grau B., Hernandez N., Jackiewicz A., Minel J.-L., Porhiel S (2004). RÉGAL, un système pour la visualisation sélective de documents, *Revue d'Intelligence Artificielle*, 32 pages, à paraître.
- Crispino G., S. Ben Hazez., J.-L. Minel. (1999). ContextO, un outil de la plate-forme d'ingénierie linguistique Filtext, *VEXTAL 99*, Venise, Italie, p. 361-367.
- Crispino G. (2003). Une plate-forme informatique de l'Exploration Contextuelle : modélisation, architecture et réalisation (ContextO). Application au filtrage sémantique de textes. Thèse de Doctorat. Université Paris-Sorbonne.
- Desclés, J.-P., Cartier E. Jackiewicz A., Minel J.-L (1997). Textual Processing and Contextual Exploration Method, *Context 97*, Brésil, p. 189 -197.
- Ferret O., Grau B., Minel J.-L., Porhiel S. (2001). Repérage de structures thématiques dans des textes, *TALN 2001*, Tours, p. 163-172.
- Gery M. (2002). Un modèle d'hyperdocument en contexte pour la recherche d'information structurée sur le Web. *Revue des Sciences et Technologies de l'Information*, 7/2002, Hermès, Paris, p. 11-44.
- Harris R. (1993). *La sémiologie de l'écriture*. CNRS Editions, Paris, 377 p.
- Hearst M. (1999). User Interfaces and Visualization. In *Modern Information Retrieval* R. Baeta-Yates, B. Ribeiro-Neto (eds), Addison-Wesley, p. 257-322.
- Jackiewicz A. (2002). Repérage et délimitation des cadres organisationnels pour la segmentation automatique des textes, *CIFT'02*, pp. 95-107, Hammamet, Tunisie.
- Jackiewicz A., Minel J.L. (2003). L'identification des structures discursives engendrées par les cadres organisationnels, *TALN 2003*, pp. 155-164, Bats-sur Mer.
- Jacquemin C., Jardino M. (2002). *Multi-dimensional and Multi-scale Visualizer of Large XML Documents*, Proceedings of EUROGRAPHICS, Saarbrücken, Germany.
- Kan M.Y., Mckeown K.R. (1999). Information extraction and Summarization : Domain Independence through Focus Types. *Computer Science Technical Report, Columbia University*.
- Lin J., Katz B. (2003). Question Answering Techniques for the Word Wide Web. *EACL 2003, Budapest*.
- Mani I. (2001). *Automatic Summarization*, John Benjamins Publishing Company, Amsterdam.
- Minel J.-L., J.-P. Desclés. (2000) *Résumé Automatique et Filtrage des textes*, in *Ingénierie des langues*, (sous la direction de J-M. Pierrel) Paris, Editions Hermès, p. 253-270.
- Minel J.-L., E. Cartier, G. Crispino, J.-P. Desclés, S. Ben Hazez, A. Jackiewicz. (2001). Résumé automatique par filtrage sémantique d'informations dans des textes, Présentation de la plate-forme FilText, *Technique et Science Informatiques*, n° 3, Hermès, Paris, p. 369-396.
- Minel J.-L. (2002). *Filtrage sémantique de textes. Problèmes, conception et réalisation d'une plate-forme informatique*. Habilitation à diriger des recherches, Université Paris-Sorbonne.
- Minel J.-L. (2003). *Filtrage sémantique. Du résumé à la fouille de textes*. 200 p., Hermès, Paris.
- Murch G. J (1985). Color Graphics - Blessing or Ballyho, in Baecker & al. (eds) *Readings in Humanities Computer Interaction : Toward the Year 2000*. Morgan Kaufman Publishers.

- Porhiel S. (2001). Linguistic expressions as a tool to extract thematic information, *Corpus Linguistic 2001*, Lancaster University.
- Shneiderman B. (1998). Designing the User Interface : strategies for effective Human-Computer interaction. *Addison-Wesley*.
- Vandendorpe, C.(1999). *Du papyrus à l'hypertexte*. 271 p., Editions la Découverte, Paris.