L'indexation plein texte, technologie unificatrice de consultation en TAO

par Claude Bédard,

Version remaniée d'une communication présentée au Congrès de l'ACFAS en avril 2000

1. Introduction

De plus en plus, le traducteur a accès à des lexiques électroniques fournis par des clients, des collègues ou publiés sur Internet. L'interrogation de ces lexiques reste toutefois problématique avec un outil de bases de données classique, caractérisé par un processus rigide d'importation. Par ailleurs, avec la généralisation des documents électroniques et avec l'arrivée de nouvelles sources d'information traductionnelle (bitextes et mémoires de traduction), on assiste à l'étalement des ressources terminologiques le long d'un continuum. Désormais, l'information terminologique échappe à la fiche en bonne et due forme; elle déborde sur les lexiques de toute origine, les bitextes, les archives de traduction, etc. Dans ce nouveau contexte, l'indexation plein texte apparaît comme une technologie de choix pour unifier la consultation de ces différentes ressources.

En effet, la base de données terminologique classique de type SGBD (Access, FileMaker, etc.) apparaît de plus en plus comme une ressource limitée, pour plusieurs raisons :

- À cause du caractère contraignant du format des données qui s'y trouvent. C'est ce que nous examinerons à la section 2.
- À cause du caractère limité de la fiche en bonne et due forme telle qu'elle existe dans les SGBD. La section 3 propose un panorama des différentes sources de renseignements utiles.
- À cause de ses limitations face à des données textuelles de formes très variées, comme nous le verrons à la section 4.

Sur ces trois plans, la technologie de l'indexation plein texte vient combler ces lacunes.

Mais d'abord, qu'est-ce que l'indexation?

Pour chercher des mots dans un texte, deux moyens s'offrent à nous. Le premier, qui est aussi le plus simple, est de balayer le texte, caractère par caractère, à la recherche d'une chaîne de caractères identique à la chaîne recherchée. C'est la fonction Recherche du traitement de texte.

Le deuxième moyen consiste à construire d'abord un index de mots, puis à lancer notre recherche par l'entremise de cet index, et non par recherche directe sur le texte. Un index est en quelque sorte un dictionnaire de tous les mots du texte, avec pour chaque mot la position de toutes ses occurrences dans le texte. Supposons par exemple que dans un index, on ait les entrées :

L'index :

```
component  = 995, 2005...
contiguous = 34, 1089, 2004...
control    = 18, 243, 266, 851...
...
structure  = 244, 301, 559, 852...
subsection = 23, 50, 133...
```

Le contexte :

```
…with this control structure, you can…

  241 242 243 244  245 246
```

Lorsqu'on lance une requête sur le groupe de mots « control structure », le logiciel consulte d'abord l'index et constate que les deux mots sont présents dans le texte. Dans un deuxième temps, il consulte les positions de chacun des mots pour constater que ces deux mots se présentent de manière contiguë et dans l'ordre demandé aux positions 243-244 et 851-852. Enfin, le logiciel se rend à ces deux endroits et ramène les contextes.

Les SGBD offrent au départ la recherche par balayage ainsi que par indexation sur le contenu complet du champ, et non par mots séparés; on peut normalement créer des index par mots, mais limités à un champ. Des index multichamps sont possibles, mais leur réalisation est malaisée.

L'indexation plein texte, par contraste, porte d'emblée sur l'ensemble des mots d'un document, sans tenir compte a priori de sa structuration. Elle convient à merveille aux documents de texte, mais nous verrons qu'elle peut aussi faire très bon ménage avec des données structurées.

2. Les périls de l'importation

On a de moins en moins de temps pour s'en occuper et l'enrichir, en même temps que s'accroissent les ressources extérieures de toutes sortes. On a besoin de moyens « légers », sommaires et réversibles (qui ne portent pas à conséquence). Or les SGBD se prêtent mal à ce genre d'opérations.

Pour ajouter des données dans une base de données classique, on peut soit créer de nouvelles fiches une à une au moyen de l'interface prévue à cette effet, soit créer en une seule opération un grand nombre de fiches au moyen d'une opération appelée « importation ».

Or les sources de données potentiellement importables sont de formats, de fiabilité et de pertinence très inégaux, sans parler de la question de la redondance. Leur importation se heurte à des obstacles de plusieurs ordres.

Obstacles techniques

L'importation de données est une opération qui exige une extrême précision. Les habitués connaissent d'ailleurs bien la règle d'or : faire d'abord une copie de la base avant d'un importer le nouveau fichier, l'original de la base demeurant ainsi intact en cas de résultat insatisfaisant.

La situation classique, en amont de l'importation, est le transcodage des données en vue de leur transfert d'une base à une autre. Le schéma est le suivant :

Base 1 => Exportation => Transcodage => Importation => Base 2

Il est rare que la structure des données à importer soit compatible avec celle de la base réceptrice. Une restructuration s'impose donc, soit après l'exportation, soit (mieux encore, si la possibilité existe) à l'exportation.

Signalons deux grands principes de structuration en importation-exportation.

- Le premier est celui des champs balisés : chaque champ de chaque fiche porte un code d'identification, qui permet de le repérer malgré l'absence de champs dans certaines fiches et malgré une séquence différente de champs. Ainsi, dans le document d'exportation, les champs vides peuvent ou non être présents, l'ordre des champs peut ou non varier, etc.
- Le deuxième est celui des champs séquentiels : les champs ne sont pas balisés, mais séparés par un symbole uniforme, la séquence des champs étant censée fournir implicitement des renseignements sur leur nature. Dans le document d'exportation, tous les champs, même vides, sont présents.

Sans compter les incompatibilités entre fichiers qui relèvent d'un même principe de structuration, on constate des problèmes évidents de transcodage entre deux fichiers structurés selon des principes différents. Ainsi par exemple :

```
[a]word processor [f]logiciel de traitement de texte [s]Fichier
personnel [r]Terme à privilégier [d]Informatique[@]
```

```
[a]word processing [f]traitement de texte [s]Fichier personnel
[d]Informatique [@]
```

Pour importer ces fiches dans une base selon un traitement par champs séquentiels, il faut transcoder les balises de champ en séparateur standard (par exemple une tabulation), puis spéficier que le premier champ est celui de la vedette anglaise, etc. Premièrement, on espère que la séquence des champs est uniforme dans l'ensemble du fichier à importer. Deuxièmement, on espère aussi qu'en cas de champ vide, la balise est quand même inscrite afin que le synchronisme soit maintenu avec les champs du fichier récepteur. Or on constate que dans la deuxième fiche le champ Remarques est absent; en conséquence, le mot Informatique tombera dans le champ Remarques et le champ Domaine restera vide.

À l'inverse, le problème consiste à créer des balises différentes ([a], [f], [s], etc.) à partir d'un symbole (la tabulation) ambigu. Dans l'exemple suivant, il faut une bonne dose d'habileté pour créer une macrocommande qui remplace la première tabulation de chaque paragraphe par [f], la deuxième par [s], etc.

```
word processing<tab>traitement de texte<tab>Fichier personnel
<tab>Informatique
```

L'autre grande situation est celle où l'on souhaite importer un document de texte ordinaire, qui n'est pas préstructuré selon la logique d'une base de données. Avec de l'habileté et de la chance, on peut obtenir des résultats raisonnables par une série de remplacements globaux. Toutefois, dans la plupart des cas, la situation se révèle trop complexe. Prenons le cas extrême d'un lexique disposé visuellement non pas au moyen de cellules, mais de tabulations ligne à ligne :

```
word processing<tab>traitement de<tab>Informatique<retour>
<tab>                texte<retour>
```

Les problèmes à résoudre pour convertir ces fichiers, on l'imagine, peuvent être considérables, sinon insurmontables sans une forte dose de travail manuel. D'autant que l'importation est une opération qui n'a aucune tolérance à l'erreur. En cas d'imprécision, certaines données sont perdues ou tombent dans le mauvais champ, ou encore la fiche au complet est rejetée (avec ou sans possililité de l'identifier).

Or, comme nous le verrons, la tolérance à l'imprécision est justement une des vertus de l'approche plein texte. Lorsqu'on intègre un document de fiches à une base plein texte, il n'y a aucun phénomène de rejet ou de perte de données : si les données sont mal disposées, elles restent malgré tout accessibles en recherche globale sur tous les champs.

Obstacles qualitatifs

On ne veut pas « polluer » sa précieuse base en y important n'importe quoi. En effet, une fois des données incorporées à la base, il n'est pas évident de pouvoir faire machine arrière.

- Certains logiciels permettent de supprimer des blocs de fiches qui répondent à certains critères d'interrogation. Encore faut-il avoir inscrit dans ces fiches, avant de les importer, une information (par exemple un nom de source) bien spécifique et non utilisée dans les autres fiches de la base, qui pourra servir à cette fin.
- Ou encore, on pourra créer une base séparée pour ces fiches. Toutefois, à moins que le SGBD utilisé permette l'interrogation simultanée de plusieurs bases, il faudra se résoudre à interroger séparément cette nouvelle base.

Obstacles temporels

Corollaire des deux points précédents, on se voit condamné à « avoir le temps » de restructurer ou de valider les données avant l'importation – temps que bien souvent on ne trouve jamais. Ce qui arrive le plus souvent, c'est qu'on abandonne l'idée dès le départ puisqu'on sait qu'on ne trouvera pas le temps d'effectuer la phase qualitative : c'est le principe du seuil d'inaction, sur lequel nous reviendrons plus loin. En somme, on se prive de sources terminologiques secondaires potentiellement fort utiles.

La terminologie actuelle n'est plus l'entreprise patiente et appliquée de thésaurisation que'elle était. De nos jours, on veut pouvoir se servir de ces apports extérieurs, on veut s'en servir tout de suite, et on a peu de temps à consacrer à la gestion.

Une base « sans importation »

Une fois l'importation terminée, le contenu du document d'importation réside matériellement dans la base. Par contraste, dans le cas de l'indexation plein texte, les données une fois intégrées à la base n'y résident pas matériellement; les documents de données ne sont pas « dans » la base. On pourrait plutôt dire que la base existe « par-dessus » les documents, qu'elle vient les chapeauter.

Conséquence, les documents-sources de la base continuent d'exister dans leur format d'origine. Il sont manipulables (on peut les modifier, les renommer, les déplacer, les supprimer) à volonté. Ainsi, lorsqu'on ne veut plus d'un de ces documents, il suffit de le retirer du répertoire d'indexation et de mettre à jour les index.

L'autre façon consiste à créer des sous-bases à inclure-exclure dans la recherche.

Quant à la performance d'interrogation, on peut interroger tout document directement en texte pur. Si on prend la peine de créer des balises de champ, le document devient interrogeable par champ et affichable dans la grille de résultats.

3. Variété des sources d'information terminologique

Nous assistons, depuis une décennie, à une progression foudroyante de l'information textuelle disponible sous forme électronique. Alors que naguère la base terminologique représentait un des seuls supports d'information interrogeables, les progrès de la bureautique et d'Internet mettent à notre disposition d'innombrables documents électroniques, dont la consultation réclame des outils de recherche plus souples.

Les frontières entre la fiche traditionnelle et les autres sources d'information terminologique se brouillent de plus en plus. Voici un survol des principales formes que peut prendre aujourd'hui l'information à potentiel terminologique (IPT).

- **Fiche terminologique en bonne et due forme**

Il suffit qu'elle ait, en plus de (au moins 2) champs vedette, des champs annexes comme la source, le domaine, etc.

- **Lexique bilingue minimal**

De plus en plus, des listes « sèches » constituées d'équivalences bilingues (sans indication de source ni de domaine) circulent entre traducteurs et terminologues. Les donneurs d'ouvrage, souvent, en fournissent. Sans compter les listes qu'on peut trouver sur Internet.

- **Bitexte**

Un bitexte est un document mixte constitué des deux versions d'un même texte dans deux langues différentes. Il contient non seulement un abondant réservoir d'équivalences terminologiques en contexte, mais aussi d'innombrables exemples de solution de traduction. Pour plus de détails, voir l'article Éloge du bitexte.

- **Glossaire unilingue**

Les documents constitués de termes accompagnés d'une définition pullulent désormais sur Internet. Il suffit de lancer une requête incluant un nom de domaine et le mot « glossary » pour s'en convaincre. Ces sources sont précieuses pour le traducteur et le terminologue, car faute de mettre la main sur un équivalent, une connaissance précise de la notion permet de proposer un équivalent qui est quand même minimalement motivé.

Une fois le glossaire affiché dans le navigateur Web, on a le choix de l'enregistrer sur disque au moyen de la commande Enregistrer sous, ou de le sélectionner au complet avec la souris et de le recopier vers un document de traitement de texte. Dans le premier cas, le document conserve son

codage HTML; dans le deuxième cas, il le perd (mais est ponctué de retours de chariot de fin de ligne).

- **Documents de renvoi**

Une nouvelle catégorie de sources d'information mérite d'être davantage exploitée : les documents de renvoi. Ces documents ne donnent pas telle quelle l'information recherchée, mais elles signalent son existence et renvoient à la source appropriée. Il s'agit en somme d'un outil de « préconsultation ». La façon classique de créer de tels documents consiste à passer au lecteur optique des index alphabétiques d'ouvrages :

- terminologiques (en général, le terme suivi de son numéro de page ou d'entrée);
- unilingues spécialisés (pour accéder éventuellement à des contextes définitoires).

L'avantage de tels documents, c'est d'attirer l'attention sur l'existence d'une ressource, qu'elle soit électronique ou, le plus souvent, sur papier. Une consultation de la source elle-même est nécessaire. Cette démarche est malgré tout très avantageuse, sans quoi on est condamné à consulter ces sources manuellement, l'une après l'autre, processus long et fastidieux – avec la tentation constante de renoncer (toujours notre principe du seuil d'inaction).

Pour ces sources d'information, les termes sont souvent présentés de façon inversée, ou même disjointe. En voici deux exemples classiques :

```
loops
      conditional              loops, conditional
      endless                  loops, endless
      exiting                  loops, exiting
      nested                   loops, nested
```

Un indexeur plein texte retrouvera sans difficulté ces termes. Par exemple, si l'on lance la requête « nested loop », on obtiendra une réponse dans les deux cas, même si les mots figurent sur des lignes différentes, sont inversés et non contigus. (On pourrait faire de même dans un SGBD, mais comment importer convenablement de telles données?)

Pour revenir à la saisie optique, souvent nécessaire pour ce genre de source, il est intéressant de constater que l'approche « indexation plein texte » est gagnante dans ce cas. En effet, une lecture optique est plus ou moins émaillée d'erreurs, qu'on a plus ou moins le temps de corriger, du moins dans l'immédiat. Or on peut très bien indexer immédiatement le résultat brut, sans validation aucune. Le résultat est imparfait, mais :

- Les termes qui ont été lus sans faute seront bel et bien retrouvés à l'interrogation.
- On pourra à tout moment corriger le document en question; et même procéder en plusieurs étapes. Il suffira ensuite de mettre à jour les index.

En somme, là encore, on fait échec au principe du seuil d'inaction.

Signalons en passant, pour terminer, qu'à mesure que se construira l'habitude de créer des hyperliens, l'utilisateur sera amené (tout comme dans certains sites Web) à créer des documents constitués exclusivement de liens qui mènent à d'autres documents. Toutefois, cela sort de notre propos, car il s'agit d'un processus de création manuel.

- **Textes simples**

Même dans les textes dits « simples » (c'est-à-dire des textes suivis, non structurés en champs), il y a beaucoup à découvrir. On dispose de textes de référence fournis par les clients ou par des collègues, mais il existe aussi des ressources souvent insoupçonnées :

- Des pages Web enregistrées sur disque. On peut prendre l'habitude, chaque fois qu'une page Web nous intéresse, de l'enregistrer sur disque (si possible en la classant dans un système de répertoire significatif). Par la suite, il suffira d'indexer ces documents pour qu'ils soient

consultables. À la limite, on peut maintenant enregistrer ainsi des sites complet, au moyen de logiciels comme WebWacker).

- Des messages électroniques; on peut repérer et indexer les documents INBOX de son logiciel de courrier électronique – ou mieux, des boîtes à courrier créées pour des correspondants particuliers. Ainsi, tous les échanges de courriel avec tel ou tel client, notamment les réponses à des demandes terminologiques, sont consultables.
- Ses propres archives de traduction. Parfois elles n'existent qu'en langue cible et l'intérêt en est plus limité, mais bien réel. Interrogeable par nom de personne, de produit, etc. Ou encore vérifier si tel terme en langue cible a déjà été employé.
- Dans le cas d'archives bilingues (on dispose aussi de la version en langue source), on peut s'arranger pour aller assez rapidement d'une version à l'autre. Soit en plaçant les deux documents côte à côte dans un même répertoire et sous des noms jumeaux, soit en créant au début de chaque document un hyperlien qui renvoie à l'autre. Ou encore, la formule du bitexte (voir ci-après).

En somme, on peut utiliser le texte simple pour répondre à deux questions :

- Qu'est-ce que ça signifie? On recherche des contextes explicatifs pour un terme en langue source.
- Est-ce que ça se dit? On imagine ce que pourrait être la traduction d'un terme, puis on lance une recherche pour constater si ce terme existe, et dans quel contexte. Une telle recherche peut ainsi confirmer la validité d'un équivalent.

4. Conclusion

En somme, l'approche Indexation plein texte a le discret mais inestimable avantage de présenter le plus petit dénominateur commun en matière de recherche : si c'est du texte, c'est interrogeable.

Quand on va au bout de l'exercice, on est surpris de constater à quel point des données « plates » peuvent acquérir un comportement structuré.

Nous sommes habitués à la culture SGBD. Pourtant, la plupart d'entre nous utilisons abondamment Internet, dont l'outil de recherche privilégié n'est autre que l'indexeur plein texte. Les SGBD ont acquis des lettres de noblesse et une image de sérieux. Mais…

Il faut passer d'une culture de rigueur (qualitative) à une culture d'utilité immédiate, d'opportunisme (quantitative).

Autre conséquence frappante : cela permet d'abaisser considérablement le « seul d'inaction ». Les opérations de saisie étant (beaucoup) moins exigeantes, on accepte de les faire même si on n'est pas certain de la qualité ni de la performance du résultat.

L'approche préconisée ici, selon moi, permet de débloquer la situation, caractérisée par un écart grandissant entre les données disponibles et les données consultables, et ce sous le parapluie d'un seul outil.