

Interface web

Introduction

Dans la première partie de ce chapitre, nous allons établir le cahier des charges dans lequel nous présenterons les besoins et les solutions que nous avons adoptés. Les deux autres parties seront consacrées à décrire en détail les menus du site web de consultation et de l'interface de travail collaboratif de Prolexbase. Nous présenterons des calculs de complexité sur quelques menus de l'interface de travail collaboratif.

7.1 Cahier des charges

Spécification des besoins

L'aspect coopératif du projet nécessite le développement d'une interface qui soit accessible sur Internet à tous les participants du projet et à toute personne extérieure au projet. L'interface web que nous devons développer doit respecter ces différentes contraintes :

- fonctionner sur tout système d'exploitation et tout navigateur Internet.
- présenter une interface conviviale qui permet aux utilisateurs de travailler efficacement à la fois sur des données simples et sur des listes de données. Pour cela, il faudra proposer un format simple de mise en page des fichiers de données qui permettra à tout utilisateur de travailler chez lui sans accès à l'interface.
- permettre aux utilisateurs de travailler dans des langues différentes.
- mettre en place un serveur de base de données accessible sur Internet.
- permettre la traduction des menus et des messages de l'interface de travail dans plusieurs langues.

Implémentation

Nous avons décidé de développer deux interfaces web. La première, beaucoup plus simple (voir section 7.2), permet aux visiteurs de consulter les données. La deuxième permettra aux participants du projet de travailler sur les données (voir section 7.3).

Il existe de nombreux systèmes de gestion de base de données. Parmi ceux-ci, nous avons décidé de créer notre base de données sous MySQL 4.1, car il s'agit d'un outil gratuit, très répandu et disponible au sein de notre laboratoire. MySQL est un système de gestion de bases de données libre possédant une architecture multiutilisateur et multitraitement. De nombreux langages de programmations (C, C++, Java, Perl, PHP, etc.) peuvent être utilisés pour créer des interfaces avec une base de données MySQL.

Travailler dans un environnement multilingue nécessite l'utilisation d'un format d'encodage universel : Unicode. Tous les mots stockés dans les tables, comme la table *PRO-LEXEME*, la table *ALIAS*, etc., de notre base de données sont codés avec la norme UTF-16. Nous avons choisi d'adopter UTF-16, car il s'agit de la norme utilisée par le logiciel UNITEX développé par le laboratoire de Marne-la-Vallée avec lequel nous avons plusieurs projets en commun. Nous espérons qu'une version de MySQL intégrant l'UTF-16 serait disponible avant la fin de la thèse. Comme la version 4.1 de MySQL n'est pas compatible avec UTF-16, nous avons été obligé de stocker provisoirement les noms propres sous format binaire (Blob).

L'interface de travail sur Prolexbase a été développée en Java sous forme d'applet, car le langage Java intègre la norme Unicode et une applet peut s'exécuter sur tout navigateur Internet et tout système d'exploitation possédant une machine virtuelle java.

Le site de consultation a été entièrement développé en PHP car celui-ci présente deux avantages : le script PHP est exécuté uniquement par le serveur et cette exécution ne réclame aucune installation sur l'ordinateur du visiteur.

7.2 Site de consultation de Prolexbase



FIG. 7.1 – Page d'accueil de l'interface de consultation.

Le site de consultation de Prolexbase est accessible au public à partir de l'adresse suivante : http://tln.li.univ-tours.fr/tln_prolex/prolex.php.

Sur la page d'accueil (figure 7.1), il est possible de sélectionner la langue de l'interface : français ou anglais.

7.2.1 Le menu *Recherche*

Après avoir choisi la langue, l'utilisateur arrive sur la page de recherche des noms propres (figure 7.2). Il suffit de taper une instance d'un prolexème, d'un alias ou d'un dérivé et de cliquer sur le bouton *Rechercher* pour que le résultat de la recherche soit affiché. La figure 7.2 montre l'exemple du résultat obtenu en recherchant le nom propre *Paris*. Le programme a trouvé dans la base de données un seul nom propre correspondant à *Paris* et affiche aussi les informations sur celui-ci. En dessous une liste de noms propres contenant la séquence

Paris est affichée. Il est possible d’avoir des informations sur ceux-ci en cliquant sur leur libellé.

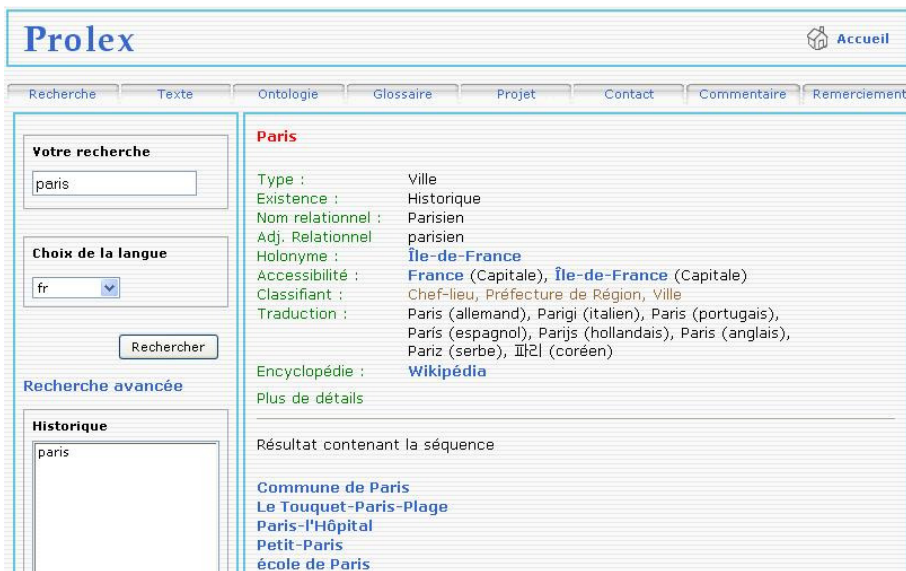


FIG. 7.2 – Recherche de noms propres.

Le visiteur peut aussi effectuer une recherche avancée (figure 7.3) dans la base de données en cliquant sur le lien *Recherche avancée*. Il est possible de faire une recherche sur des noms propres commençant, contenant et/ou finissant par une certaine séquence.

7.2.2 Le menu *Texte*

Ce menu (figure 7.4) permet au visiteur d’écrire, copier ou sélectionner depuis son ordinateur un texte puis de faire une recherche des noms propres dans ce texte en double-cliquant sur ceux-ci.

Par exemple, par un double-clic sur le nom propre *Paris* dans le texte, on obtient une fenêtre identique à celle de la figure 7.2 et en double-cliquant sur *luxembourgeois* la fenêtre de la figure 7.5 apparaît.

7.2.3 Autres menus

Le menu *Ontologie* fournit le schéma de l’ontologie de Prolexbase et le schéma d’un exemple. Le menu *Glossaire* donne les définitions des différents concepts utilisés pour décrire Prolexbase. Une description du projet est donnée dans le menu *Projet*. Les coordonnées du responsable du projet sont disponibles dans le menu *Contact*. Le menu *Commentaire* permet aux visiteurs de faire des remarques ou suggestions sur le site. Enfin, le menu *Remerciement* liste toutes les personnes ayant participé au projet.

7.3 Interface de travail

Elle est accessible à partir de l’adresse suivante : http://tln.li.univ-tours.fr/tln_prolexbase/.

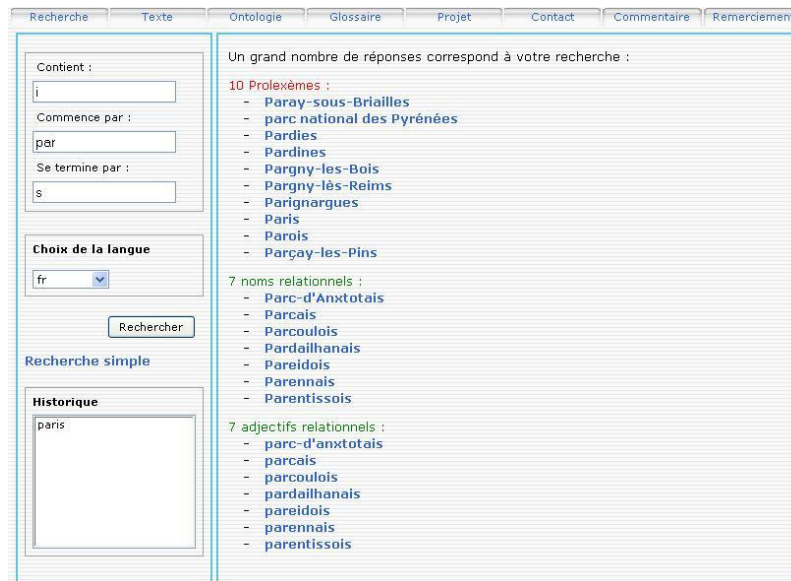


FIG. 7.3 – Recherche avancée.



FIG. 7.4 – Menu *Texte*.



FIG. 7.5 – Informations sur le mot *luxembourgeois*.

7.3.1 Les menus simples

Page d'accueil

Sur la page d'accueil (figure 7.6) l'utilisateur doit d'abord sélectionner la langue de l'interface. Trois langues sont disponibles pour l'interface : l'anglais, le français et le serbe.

Après avoir cliqué sur le bouton *OK*, les personnes ayant un compte doivent rentrer leur identifiant et leur mot de passe (figure 7.7), sinon ils peuvent cliquer sur le bouton *Visiter*. Seules les personnes ayant un compte peuvent travailler sur la base de données.

L'utilisateur doit choisir sa langue de travail, puis cliquer sur le bouton *Valider* (figure 7.8). Il peut à tout moment changer de langue de travail ou de consultation en revenant sur



FIG. 7.6 – Choix de la langue de l'interface.

The screenshot shows a web application interface with a menu bar at the top containing the following items: "Choix de la langue", "Consultation", "Ajout", "Modification", "Modification d'une liste", "Fichier", "Attributs et Notes", "Glossaire", "Compte", and "Suppression et fusion".

The main content area is divided into two sections:

- Accès Membre:** Contains two input fields labeled "Identifiant" and "Mot de passe", and a "Valider" button below them.
- Accès Visiteur:** Contains a "Visiter" button.

Below these sections, there is a text box containing the following text:

*Les données de Prolexbase sont disponibles sous une licence LGPL, la: Lesser General Public License For Linguistic Resources Le texte complet de cette licence se trouve à l'URL:
<http://www-igm.univ-mlv.fr/~unitex/lgplr.html>*

FIG. 7.7 – Login et mot de passe.

The screenshot shows a "Consultation" section with a dropdown menu currently displaying "fr". Below the dropdown is a "Valider" button.

FIG. 7.8 – Choix de la langue de travail.

l'onglet *Choix de la langue*.

Consultation

L'onglet *Consultation* (figure 7.9) comporte trois zones. Dans la zone gauche de l'onglet, l'utilisateur peut rechercher des prolexèmes de la base de données suivant différents critères. Les trois listes déroulantes permettent de faire une recherche de prolexèmes à partir de leur détermination, de leur flexion et de leur type. Il y a aussi la possibilité de faire une recherche de prolexèmes commençant par une certaine séquence, contenant une certaine séquence et/ou se terminant avec une certaine séquence. On peut interroger la base de données sur les derniers prolexèmes ajoutés en précisant un nombre dans le champs *derniers ajouts*. Enfin, une recherche suivant le numéro de pivot d'un prolexème est aussi possible.

Après avoir précisé les critères de recherche, l'utilisateur doit cliquer sur le bouton *Lancer la consultation* pour démarrer la recherche. Le programme affiche les résultats de la recherche dans la liste déroulante située en haut et au milieu de l'onglet. Par exemple, pour une recherche d'un prolexème commençant par *Franc*, le programme donnera les résultats suivants : *France*, *Franche-Comté*, *Francis Ford Coppola*, etc. En précisant que le type du prolexème est pays, on obtiendra seulement le prolexème *France*.

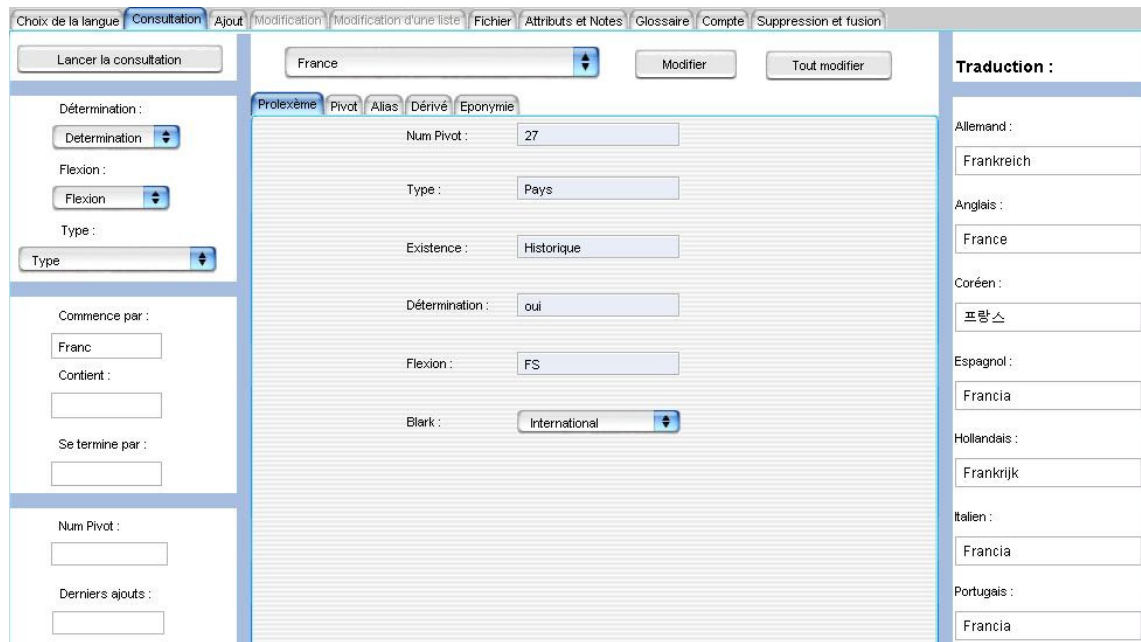


FIG. 7.9 – Onglet *Consultation*.

En sélectionnant un prolexème dans la liste déroulante en haut de la zone au milieu de la fenêtre, le programme affiche les informations sur celui-ci. Nous pouvons lire sur la figure 7.9 que le prolexème *France* a le numéro *27* comme pivot, son type est *pays*, son existence est *historique*, etc. L'onglet *Pivot* de la zone au milieu de la fenêtre renseigne sur les relations (méronymie, synonymie et accessibilité) que ce prolexème entretient avec d'autres prolexèmes. Les alias et les dérivés sont affichés dans l'onglet *Alias* et *Dérivés*. L'onglet *Eponymie* renseigne sur les expressions idiomatiques, les antonomases et la terminologie du prolexème.

Des traductions du prolexème sélectionné sont affichées dans la zone droite de l'onglet.

Dans cet exemple, la traduction du prolexème *France* en espagnol est *Francia*.

En cliquant sur le bouton *Modifier*, l'utilisateur pourra modifier des informations sur le prolexème sélectionné. Le bouton *Tout modifier* permet de changer les informations des prolexèmes contenus dans la liste déroulante de la zone du milieu.

Si le prolexème recherché n'existe pas dans la base de données, l'utilisateur pourra l'ajouter en utilisant l'onglet *Ajout* de l'applet.

Ajout

Ce menu (figure 7.10) permet d'ajouter un nom propre dans la base, ainsi que ses relations avec d'autres noms propres (figure 7.11), ses alias (figure 7.12), ses dérivés (figure 7.13), ses éponymes (figure 7.14), et la source d'où a été extrait celui-ci (figure 7.15).

FIG. 7.10 – Onglet *Ajout*.

FIG. 7.11 – Onglet *Pivot*.

Prolexème Pivot **Alias** Dérivé Eponymie Source

Alias : Retirer

Catégorie :

Flexion :

Règle d'alias : Retirer

Ajouter

Modifier

Ajouter

FIG. 7.12 – Onglet *Alias*.

Prolexème Pivot Alias **Dérivé** Eponymie Source

Prolexème

se rapportant à : Alias

Règle de dérivation : Retirer

Dérivé : français Retirer

Catégorie : Adjectif relationnel

Flexion : N/A61

Ajouter

Ajouter

Modifier

FIG. 7.13 – Onglet *Dérivé*.

Prolexème Pivot Alias Dérivé **Eponymie** Source

Antonomase : Retirer

Idiome : Retirer

Terminologie : Retirer

Ajouter

Ajouter

Ajouter

FIG. 7.14 – Onglet *Eponymie*.

Prolexème Pivot Alias Dérivé Eponymie **Source**

Source : Prolex ou Ajouter

FIG. 7.15 – Onglet *Source*.

Modification

Ce menu (figure 7.16) permet de modifier le libellé d'un prolexème et d'éditer ses informations et ses relations.

The screenshot shows the 'Modification' tab of a software interface. At the top, there is a navigation bar with tabs: 'Choix de la langue', 'Consultation', 'Ajout', 'Modification' (selected), 'Modification d'une liste', 'Fichier', 'Attributs et Notes', 'Glossaire', 'Suppression et fusion', and 'Administration'. Below the navigation bar, the 'Prolexème' field is set to 'France' with a 'Valider' button. Below this, there are sub-tabs: 'Prolexème' (selected), 'Pivot', 'Alias', 'Dérivé', 'Eponymie', and 'Source'. The main area contains several fields for editing the prolexeme: 'Prolexème' (France), 'Type' (Pays), 'Existence' (Historique), 'Détermination' (oui), 'Flexion' (FS), 'Classifiant' (empty), and 'Blark' (International). To the right of these fields, there are buttons for 'Retirer' and 'Ajouter'. At the bottom right, there are several 'Modifier' and 'Ajouter' buttons, each associated with a dropdown menu.

FIG. 7.16 – Onglet *Modification*.

Modification d'une liste

Ce menu (figure 7.17) permet d'éditer des informations et relations d'une liste de prolexèmes. Par exemple, si on sélectionne le classifiant *département* et le code flexionnel *FP*, ils seront associés à chaque prolexème de la liste.

The screenshot shows the 'Modification d'une liste' tab of a software interface. At the top, there is a navigation bar with tabs: 'Prolexème', 'Pivot', and 'Source' (selected). Below the navigation bar, the 'Prolexème' field is a list box containing 'Alpes-Maritimes', 'Bouches-du-Rhône', 'Côtes-d'Armor', and 'Deux-Sèvres'. To the right of the list box is a 'Retirer' button and a 'Valider' button. Below this, there are several fields for editing the list: 'Type', 'Existence', 'Détermination', 'Flexion' (FP), 'Classifiant' (Département), and 'Blark'. To the right of these fields, there are buttons for 'Retirer' and 'Ajouter'.

FIG. 7.17 – Onglet *Modification d'une liste*.

7.3.2 Le menu fichier

La plupart de nos collaborateurs possèdent des listes de noms propres sous forme de fichiers. Rentrer tous ces noms propres un par un en utilisant l'onglet *Ajout* risque de prendre beaucoup de temps. Pour éviter cela, nous avons développé dans l'onglet *Fichier* des outils pour permettre de travailler facilement et efficacement à partir de fichiers.

La première fonction proposée par cet onglet permet aux utilisateurs d'insérer un fichier contenant une liste de noms propres avec leurs informations. Ce fichier doit être enregistré au format Unicode UTF-16. La figure 7.18 donne un exemple de format de fichier. Ce fichier est organisé en colonnes. Une tabulation permet de séparer les données de chaque colonne. Dans cet exemple, la première colonne correspond aux prolexèmes, la seconde à la détermination, la troisième au type, la quatrième à la flexion, la cinquième aux dérivés et à la dernière la flexion des dérivés. La première ligne du fichier indique que le nom propre *France* se construit avec un article, son type est *Pays*, son code de flexion est *FS* et il a pour dérivé *Français*, dont le code de flexion est *N/A61*.

France	Oui	Pays	FS	Français	N/A61
Paris	Non	Ville	MFS	Parisien	N/A41
Belgique	Oui	Pays	FS	Belge	N/A31
Bruxelles	Non	Ville	MFS	Bruxellois	N/A61

FIG. 7.18 – Exemple de fichier.

The screenshot shows the 'Ajout Fichier' interface. At the top, there are menu items: 'Ouvrir Fichier', 'Ajout Fichier', 'Vérifier Fichier', 'Ajout Liste Pivot', 'Vérifier Fichier Multilingue', 'Ajout Fichier Multilingue', and 'Erreur'. Below this is a 'test.txt' file with a 'Valider' button. The main area displays a table with six columns: 'Colonne 1' (France), 'Colonne 2' (Oui), 'Colonne 3' (Pays), 'Colonne 4' (FS), 'Colonne 5' (Français), and 'Colonne 6' (N/A61). Below the table, there are dropdown menus for each column to specify their meaning: 'Colonne 1: Prolexème', 'Colonne 2: Détermination', 'Colonne 3: Type', 'Colonne 4: Flexion', 'Colonne 5: Dérivé', and 'Colonne 6: Flexion'. To the right of these are 'se rapportant à' dropdowns, with 'Colonne 1' selected for columns 2, 3, 4, and 5. At the bottom, there are tabs for 'Donnée fixe', 'Méronymie', 'Accessibilité', 'Synonymie', and 'Source'. The 'Donnée fixe' tab is active, showing a 'Donnée fixe 1:' dropdown set to 'Catégorie dérivé', a 'Nom relationnel' dropdown, and a 'se rapportant à' dropdown set to 'Colonne 5'. An 'Ajouter une donnée fixe' button is at the bottom left.

FIG. 7.19 – Ajout d'un fichier.

La figure 7.19 donne un aperçu de l'onglet *Ajout Fichier*. L'onglet comporte trois parties. La première partie, située en haut de l'onglet, affiche les deux premières lignes du fichier sur lequel on travaille. Dans la partie du milieu, l'utilisateur va pouvoir préciser la signification de chaque colonne. Si nous reprenons notre exemple, la colonne 1 correspond au prolexème, la colonne 2 à la détermination du prolexème de la colonne 1, la colonne 3 à son type, la colonne 4 à sa flexion, la colonne 5 au dérivé du prolexème et la dernière colonne à la flexion

du dérivé de la colonne 5. Pour éviter d’avoir une colonne avec une donnée unique, nous avons rajouté dans la dernière partie de cet onglet la possibilité d’affecter une valeur unique à une colonne. Par exemple, nous savons que tous les dérivés de la colonne 5 sont des noms relationnels. Nous allons déclarer qu’une donnée fixe correspondra à la catégorie des dérivés. Elle aura comme valeur *Nom relationnel* et se rapportera à la colonne 5. Il suffit, enfin, de cliquer sur le bouton *Valider* pour insérer les données dans Prolexbase.

L’onglet *Ajout Fichier* permet aussi d’ajouter des relations de méronymie, de synonymie ou d’accessibilité entre les noms propres d’un même fichier. La figure 7.20 donne un exemple d’ajout d’une relation d’accessibilité entre les prolexèmes de la colonne 2 avec ceux de la colonne 1 (*Berlin* est la capitale de l’*Allemagne*, *Lisbonne* est la capitale du *Portugal*, etc.).

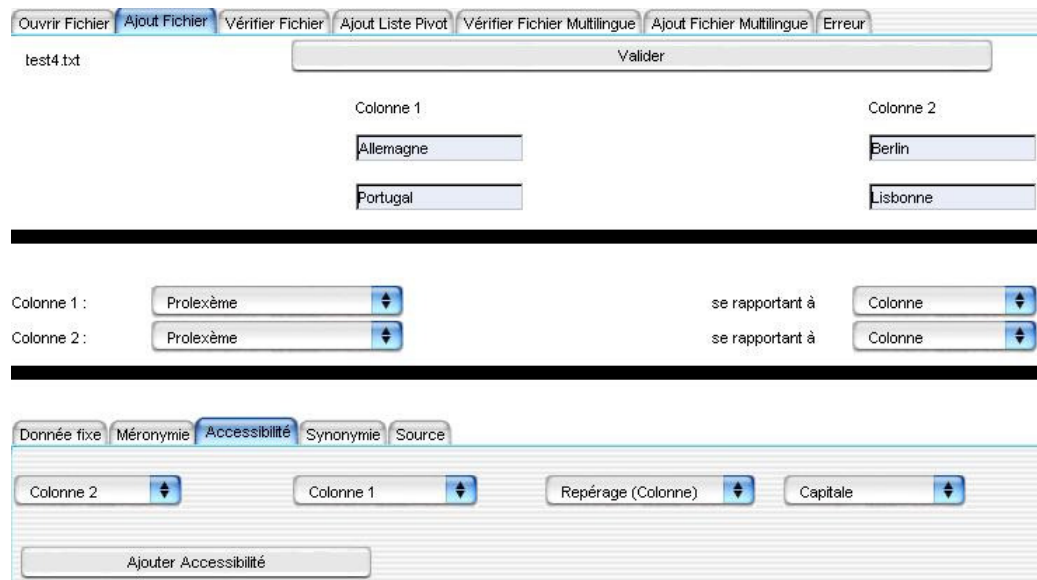


FIG. 7.20 – Relation de méronymie.

27	France	France	Frankreich	Francia	Francia	França	Frankrijk
38558	Paris	Paris	Paris	Parigi	Paris	Paris	Parijs
45883	Belgique	Belgium	Belgien	Belgio	Bélgica	Bélgica	België
45884	Bruxelles	Brussel	Brüssel	Brussel	Bruselas	Bruxelas	Brussel

FIG. 7.21 – Exemple de fichier multilingue.

Il arrive souvent que plusieurs personnes travaillent dans une même langue. Pour éviter qu’une personne ne rentre des noms propres qui existent déjà dans la base de données, nous avons créé l’onglet *Vérifier fichier* pour contrôler si une liste de prolexèmes existe déjà dans la base de données puis ajouter uniquement ceux qui ne sont pas présents. Cet onglet est pratiquement identique à l’onglet *Ajout Fichier*.

Une autre fonction du menu fichier permet de récupérer les numéros de pivot d’une liste de prolexèmes que l’on a déjà rentrée dans la base de données. Le programme effectue une recherche sur les prolexèmes de la base et renvoie le numéro de pivot si le prolexème existe. Pour les prolexèmes s’écrivant de la même manière, le programme renverra plusieurs numéros de pivot. L’utilisateur récupère les résultats du programme dans un fichier.

Les utilisateurs peuvent modifier les informations relatives aux prolexèmes de la base

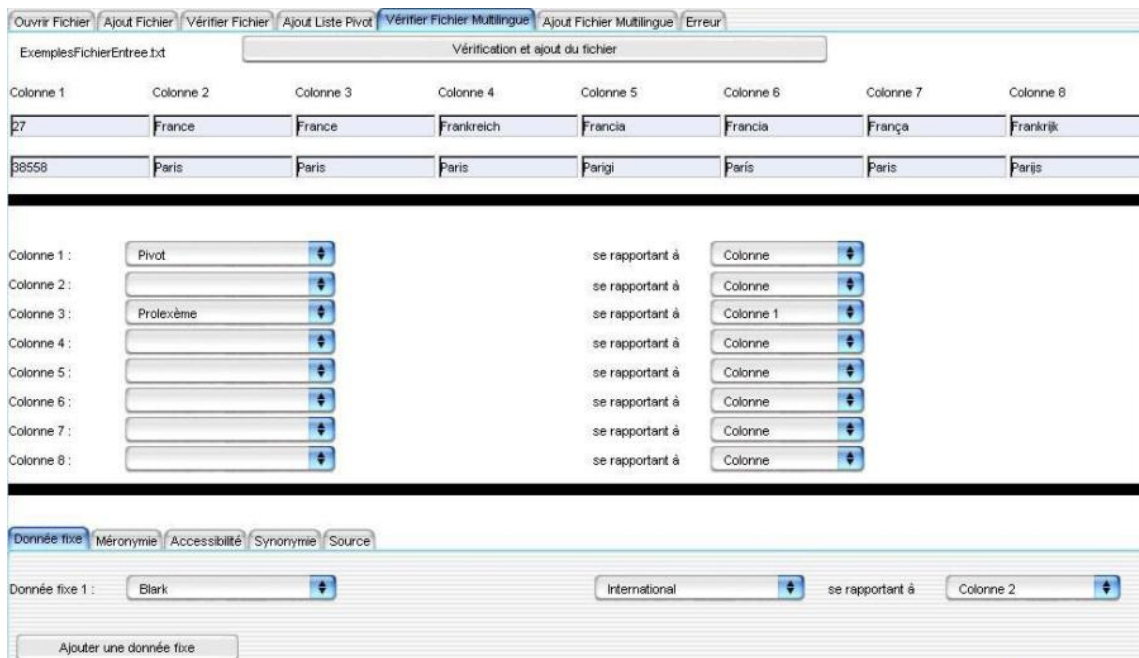


FIG. 7.22 – Traduction de prolexèmes dans une autre langue.

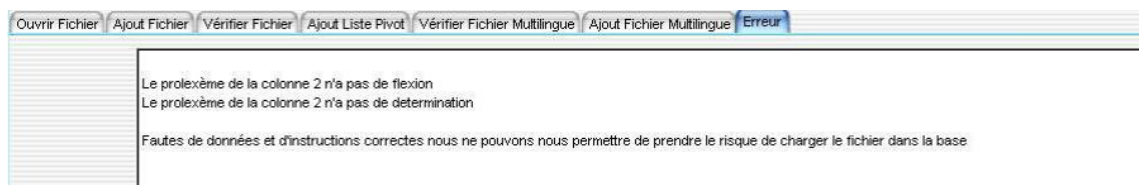


FIG. 7.23 – Les erreurs.

à partir d'une liste de numéros de pivot. Pour cela, ils doivent indiquer au programme la colonne du fichier correspondant aux numéros de pivot et sélectionner les informations qu'ils souhaitent modifier.

Les utilisateurs possédant un fichier de prolexèmes avec leur numéro de pivot peuvent ajouter dans la base la traduction de ceux-ci vers une autre langue. La figure 7.21 donne un exemple de prolexèmes français (colonne 2) avec leur numéro de pivot (colonne 1) traduit en anglais (colonne 3), en allemand (colonne 4), etc. Ils doivent d'abord sélectionner dans le menu fichier la langue vers laquelle ils souhaitent traduire et ensuite indiquer au programme le fichier de travail. Dans la figure 7.22, l'utilisateur doit préciser que la colonne 1 correspond aux numéros de pivot et la colonne 3 est la traduction des prolexèmes français en anglais. Le programme vérifie s'il n'existe pas de prolexème anglais en lien avec ces numéros de pivot et crée dans ce cas-là les prolexèmes anglais correspondants. Pour ajouter la traduction vers l'allemand, l'utilisateur doit revenir au menu fichier et changer la langue de traduction.

Les messages d'erreurs sont affichés dans l'onglet Erreur (figure 7.23).

7.3.3 Les menus d'administration

Attributs et notes

Ce menu (figure 7.24) comporte trois onglets. L'onglet *Variable* permet aux utilisateurs de définir les libellés des données qui sont spécifiques à chaque langue (comme les règles de flexions, les libellés de Blark, les catégories d'alias...). L'onglet *Fixe* permet de définir les données qui sont communes pour toutes les langues (comme les types, l'existence...). L'onglet *Interface* permet à l'utilisateur de traduire la langue de l'interface, c'est-à-dire les messages et textes affichés dans l'applet, vers une autre langue.

The screenshot shows a web interface with three tabs: 'Variable', 'Fixe', and 'Interface'. The 'Variable' tab is active. It contains three sections:

- 1) Sélectionner une table :** A dropdown menu with 'Flexion' selected.
- 2) Liste des éléments :** A dropdown menu with 'FS' selected. Below it, a 'Libellé :' label is followed by a text input field containing 'FS'. Below that, a 'Note :' label is followed by a large text area containing the text 'Nom ou adjectif uniquement féminin singulier'.
- 3) Que souhaitez-vous faire ? :** Three radio buttons labeled 'Ajouter', 'Modifier', and 'Supprimer'. The 'Ajouter' button is selected. Below these are a 'Libellé :' label with an empty text input field, and a 'Note :' label with an empty large text area.

At the bottom of the form is a 'Valider' button.

FIG. 7.24 – Onglet *Attributs et Notes*.

Compte

L'onglet *Compte* (figure 7.25) permet de créer des comptes pour les personnes qui travaillent sur Prolexbase. Il faut préciser pour chaque compte le nom, le prénom, l'identifiant, le mot de passe, les dates de validité du compte, les droits du compte et la langue dans laquelle le compte peut travailler. Nous avons défini trois types de droits :

- *A* : il s'agit d'un compte administrateur qui possède tous les droits sur Prolexbase. L'administrateur peut travailler dans toutes les langues de la base de données. Il peut créer tout type de compte.
- *C* : ce compte est destiné au chef de projet. Un chef de projet peut travailler dans une seule langue, dans laquelle il possède tous les droits y compris celui de créer des comptes utilisateurs sans pouvoir.
- *U* : ce compte est destiné aux utilisateurs sans droits, qui ne peuvent pas créer de compte.

FIG. 7.25 – Onglet *Compte*.

7.3.4 Suppression et fusion

L'onglet *Suppression et fusion* permet de supprimer ou de fusionner des prolexèmes de la base de données. La suppression est possible :

- dans toutes les langues pour les administrateurs.
- dans leur langue pour les chefs de projet.
- dans leur langue de travail pour les utilisateurs. Tant qu'un autre utilisateur ne s'est pas connecté dans sa langue de travail, l'utilisateur pourra supprimer les données qu'il a créées. Cela permet d'éviter qu'il ne supprime des données qui ont été utilisées par d'autres.

Lors de la fusion de deux prolexèmes P_1 et P_2 d'une langue L , deux cas peuvent survenir :

- Il n'existe pas de langue L' dans laquelle P_1 et P_2 se traduisent par P'_1 et P'_2 . Dans ce cas, la fusion se fait automatiquement.
- Les deux prolexèmes P_1 et P_2 possèdent tous les deux une traduction dans une autre langue L' (P'_1 et P'_2). La fusion nécessite l'accord du chef de projet de la langue L' , sauf si les deux prolexèmes P'_1 et P'_2 et si toutes les informations associées sont identiques.

7.4 Calcul de complexité

Nous présentons dans cette partie quelques calculs de complexité sur le menu *Consultation* et le menu *Ajout fichier* de l'interface de travail collaboratif.

Menu Consultation

Soit $\alpha(T)$ le coût d'une requête SQL sur la table T . La complexité d'une requête dans le menu *Consultation* se calcule de la façon suivante :

$$O(\alpha(P) + a \alpha(A) + d \alpha(D) + m \alpha(M) + s \alpha(S) + c \alpha(C))$$

Avec :

- $\alpha(P)$: coût d’une requête SQL de type *SELECT* de recherche d’un prolexème suivant différents critères (flexion, commence par, type, etc.).
- a : le nombre d’alias que possède ce prolexème.
- $\alpha(A)$: coût d’une requête SQL de type *SELECT* pour récupérer des informations sur les alias.
- d : le nombre de dérivés.
- $\alpha(D)$: coût d’une requête SQL de type *SELECT* pour récupérer des informations sur les dérivés.
- m : le nombre de pivots en relation de méronymie avec le prolexème de la recherche.
- $\alpha(M)$: coût d’une requête SQL de type *SELECT* pour récupérer des informations sur les relations de méronymie.
- s : le nombre de pivots en relation de synonymie avec le prolexème de la recherche.
- $\alpha(S)$: coût d’une requête SQL de type *SELECT* pour récupérer des informations sur les relations de synonymie.
- c : le nombre de pivots en relation d’accessibilité avec le prolexème de la recherche.
- $\alpha(C)$: coût d’une requête SQL de type *SELECT* pour récupérer des informations sur les relations d’accessibilité.

Actuellement le nombre d’éléments des tables alias, synonymie et accessibilité est négligeable par rapport au nombre d’éléments des autres tables (voir section 8.4 page 144). La complexité devient alors :

$$O(\alpha(P) + d \alpha(D) + m \alpha(M))$$

Supposons que $\alpha = \alpha(P) \approx \alpha(D) \approx \alpha(M)$; comme d est négligeable par rapport à m , nous obtenons alors :

$$O(m \alpha)$$

Menu Ajout fichier

Nous allons calculer la complexité de ce menu sur un fichier de n prolexèmes (sans alias, ni dérivés). Ce fichier comporte donc n lignes et cinq colonnes (prolexème, type, existence, flexion et détermination).

Lors de l’ajout d’un fichier, l’utilisateur doit obligatoirement préciser au programme le contenu de chaque colonne. Par exemple, la colonne numéro un correspond à un prolexème, la colonne numéro deux, à son type, etc. Le programme commence par parcourir chaque colonne pour récupérer sa signification. La complexité de cette boucle est $O(1)$.

Il vérifie si les données du fichier (type, existence, flexion, etc.) correspondent bien à celles de la base de données. Par exemple, si l’utilisateur précise que la colonne numéro deux correspond à des types, le programme vérifie si la valeur de la colonne existe déjà dans la base de données. En cas de valeur non valide, c’est-à-dire non présente dans la base de données, le programme avertit l’utilisateur de l’erreur. Étant donné la taille des tables (30 types, 3 existences, 38 flexions et 2 déterminations), nous supposons que le coût de ces vérifications dans la base de données est quasiment identique pour chaque table et égal à β . La complexité de cette vérification est donc $O(n \beta)$.

Une fois que la vérification a été faite, le programme ajoute ligne par ligne les données du fichier dans la base de données. Le programme lance une requête SQL qui ajoute le prolexème et ses informations. Soit γ le coût de cette requête. La complexité de cette partie est $O(n \gamma)$.

La complexité totale du menu ajout fichier est donc de :

$$O(n \beta + n \gamma)$$

Si nous supposons que $\beta \approx \gamma$, la complexité totale du menu ajout fichier devient alors $O(n \beta)$.

Quatrième partie

Synthèse

Chapitre 8

Évaluation

8.1 Le modèle

Dans cette partie, nous présentons une évaluation de la modélisation des noms propres que nous avons proposée et nous en discutons quelques limites.

8.1.1 Prolexème et forme vedette

Théoriquement, nous aurions dû considérer le prolexème comme un identificateur : le couple pivot-langue. La table des alias aurait alors regroupé toutes les formes possibles du nom propre. Dans la pratique nous avons préféré définir le prolexème par une forme vedette, ce qui facilite la manipulation des données par des linguistes, en simplifiant l'accès au dictionnaire.

Il n'est cependant pas évident de la choisir. Parmi les noms propres *Organisation des Nations Unies*, *Nations Unies* et *ONU*, lequel devons-nous prendre comme forme vedette ? Et sur quel critère devons-nous le choisir ?

L'utilisation future de règles d'aliasation et de dérivation nous a conduit à prendre la forme la plus longue comme prolexème. Nous pensons qu'il sera plus facile ainsi d'établir les règles de formation d'alias et de dérivés. Par exemple, il est plus évident de concevoir des règles pour former les noms propres *Nations Unies* et *ONU* à partir du nom propre *Organisation des Nations Unies* que d'utiliser les deux autres formes pour retrouver la première. Nous pourrions créer une règle qui consiste à effacer la partie générique *Organisation* pour obtenir le nom propre *Nations Unies* et une autre règle qui consiste à prendre les premières lettres de chaque mot plein pour générer le nom propre *ONU*. A la fin de cette thèse, ce travail sur la création de règles d'alias reste un projet.

Suivant les applications, les stratégies pourront être différentes. Dans la traduction de l'anglais vers le français, il faudrait probablement proposer *ONU* pour *UNO* alors que dans la recherche d'information le prolexème et tous ses alias seront également intéressants.

8.1.2 Date

Nous nous sommes posé la question de savoir si nous devons ajouter une date pour certaines relations, comme la synonymie et la méronymie.

Faut-il préciser la date dans une relation de synonymie diachronique ? Nous avons longtemps hésité sur cette question.

La ville de *Saint-Petersbourg* a été fondée vers 1703. Elle a pris le nom de *Petrograd* de 1914 à 1924, puis de 1924 à 1991 elle a porté le nom de *Leningrad*. Elle a repris le nom