Proposition de Stage

Indexation de documents pour un outil d'aide contextuelle

Sujet: Réalisation d'un outil d'aide contextuelle basé sur des technologies d'indexation de documents pour des portails d'accès à des données astronomiques,.

Durée	24 semaines 10/2016 02/2017
Lieu	Observatoire Astronomique de Strasbourg
Encadrement	Laurent MICHEL (SSC XMM-Newton)
	André Schaaff, François-Xavier Pineau, Gilles Landais (CDS)
Retribution	546,01 euros nets
Compétences requises	Linux
	POO Java
	HTML Javascript
	Anglais technique
Contact	Laurent Michel
	0 368 852 437
	laurent.michel@astro.unistra.fr

Détail du projet

Contexte

Un des aspects souvent délaissés du développement de services en ligne est l'aide apportée aux utilisateurs. Il est relativement aisé de documenter une interface simple, c'est à dire une interface dans laquelle chaque paramètre est représenté par un widget statique. Il suffit d'attacher à chacun de ces widgets une info-bulle ou un petit texte. Le nombre d'éléments étant par construction limité, l'intégration dans l'application de ces aides est réaliste, bien que souvent fastidieuse. Pour les utilisateurs chevronnés, il y a aussi la possibilté de mettre en ligne des PDFs avec une documentation complète.
Cette manière de faire ne correspond plus aux besoins actuels pour plusieurs raisons:

1. Le nombre de paramètres accessibles à l'utilisateur va croissant, ce qui alourdit le travail d'implémentation de l'aide.

2. Les interfaces proposent des interactions de plus en plus complexes telles que l'écriture de scripts ou de requêtes. L'affichage de l'aide ne peut plus être simplement attaché à un endroit précis de l'écran, mais il doit pouvoir être déclenché selon un contenu saisi par l'utilisateur.

3. Le mode de recherche imposé par Google est devenu la norme. L'utilisateur saisi des mots clés, et en retour il obtient une liste de suggestions dont il peut appréhender le contenu avant d'ouvrir le document. Il s'attend à pouvoir obtenir la description d'un paramètre à partir de son nom, mais aussi à retrouver le nom de ce paramètre à partir de quelques mots clés.

4. La durée considérée comme acceptable pour lire une aide va diminuant.

Ce constat rend très délicate la mise en forme manuelle d'éléments d'aide attachés à des actions précises sur l'interface.. L'objectif du stage est de développer une plateforme offrant un service compatible avec les 4 points énoncés précédemment.

Objectifs du stage

L'objet concret du stage est de mettre en oeuvre des technologies d'indexation de texte pour relier automatiquement les méta-données d'une base de données en ligne avec du texte libre décrivant sont contenu. Ce lien permettra d'apporter une aide contextuelle pertinente à l'utilisateur.

Le stage doit aboutir à la livraison d'une version fonctionnelle de l'outil décrit ci-dessous. Cela signifie que les implémentations doivent fonctionner, que le code doit être documenté ainsi que l'outil dans son ensemble.

Une première version sera livrée en cours de stage, elle n'implémentera pas toutes les fonctions mais elle sera utilisable dans un contexte de production. Ce sera un jalon important. Les fonctionnalités plus avancées seront implémentées dans un deuxième temps.

Outils utilisés

Le choix des outils finalement utilisés dépendra des tests d'évaluation réalisés en début de stage. Parmi les options, nous pouvons déjè citer MongoDB, ElasticSearch ou encore Sol-R.

Spécification des cas d'utilisation

La principale fonction du système est d'aider les utilisateurs à formuler des requêtes sur des bases de données en ligne. Le but est d'offrir un moyen aussi souple que possible pour identifier les paramètres adéquats pour formuler la requête répondant au besoin de l'utilisateur.

Dans le cas le plus simple, on suppose que l'utilisateur veut filtrer les lignes d'une table (table relationnelle par exemple). Pour cela, il doit placer des contraintes sur des colonnes à priori inconnues. Le problème à résoudre est de faciliter l'identification de ces colonnes. Cette identification requiert deux fonctions de base:

1. Retrouver un nom de colonne à partir d'une requête textuelle.

2. Accéder à une description complète d'un paramètre donné à partir soit de son nom soit d'une requête textuelle.

Dans la réalité, l'application Web héberge plusieurs ressources (tables). Chacune de ces ressources possède sa propre documentation.

- Les requêtes utilisateur ne porteront que sur la ressource courante. Ce confinement de la portée de la requête sera porté par un paramètre caché à l'utilisateur.

A un niveau supérieur on peut supposer que l'utilisateur cherche à identifier une table particulière au sein de la ressource.

- Le moteur de recherche possédera un index global permettant de localiser une table particulière parmi toutes celles hébergées.

Définition des requêtes utilisateur à implémenter

L'utilisateur peut utiliser l'outil soit pour récupérer un nom de colonne, soit pour récupérer les méta-données de cette colonne, soit pour récupérer des informations littéraires sur la grandeur portée par cette colonne.
Ce ne sera pas à l'utilisateur de choisir le type de requêtes qu'il souhaite exécuter mais à l'outil de faire en sorte que les données retournées répondent à ses attentes.
Pour ce faire, le résultat des requêtes sera toujours structuré de la même manière.
Ce sera un ensemble de descripteurs attachés chacun à une seule colonne.
Chaque descripteur contiendra à la fois les méta-données issues de la base et la description littéraire issue des textes indexés.
L'implémentation des requêtes supportera un mécanisme d'autocomplétion proposant les noms de colonnes correspondant au filtre en cours de saisie.

La table ci-dessous montre un exemple de recherche à partir du filtre *[veloc]*. Dans cet exemple, deux colonnes correspondent à ce filtre. Pour chacune d'elles, l'outil retourne à la fois les méta-données et la description textuelle de la grandeur associée.

<table>
<tr><th>Filtre saisi</th><th colspan="2">Colonnes trouvées</th></tr>
<tr><td rowspan="4">> veloc</td><td rowspan="2">#1 Proper motion</td><td>Description de la colonne (méta données)
Dbname : _tvel
Name: TVel
Unit: km/sec
This is a tangential velocity.</td></tr>
<tr><td>Description de la grandeur
Fragment 1 ….
Fragment 2….</td></tr>
<tr><td rowspan="2">#2 Stellar velocity</td><td>Description de la colonne (méta données)
Dbname : _vc
Name: VC
Unit: km/sec
…...</td></tr>
<tr><td>Description de la grandeur
Fragment 1 ….
Fragment 2….</td></tr>
</table>

Base Documentaire

La base documentaire contient des données provenant de deux origines:

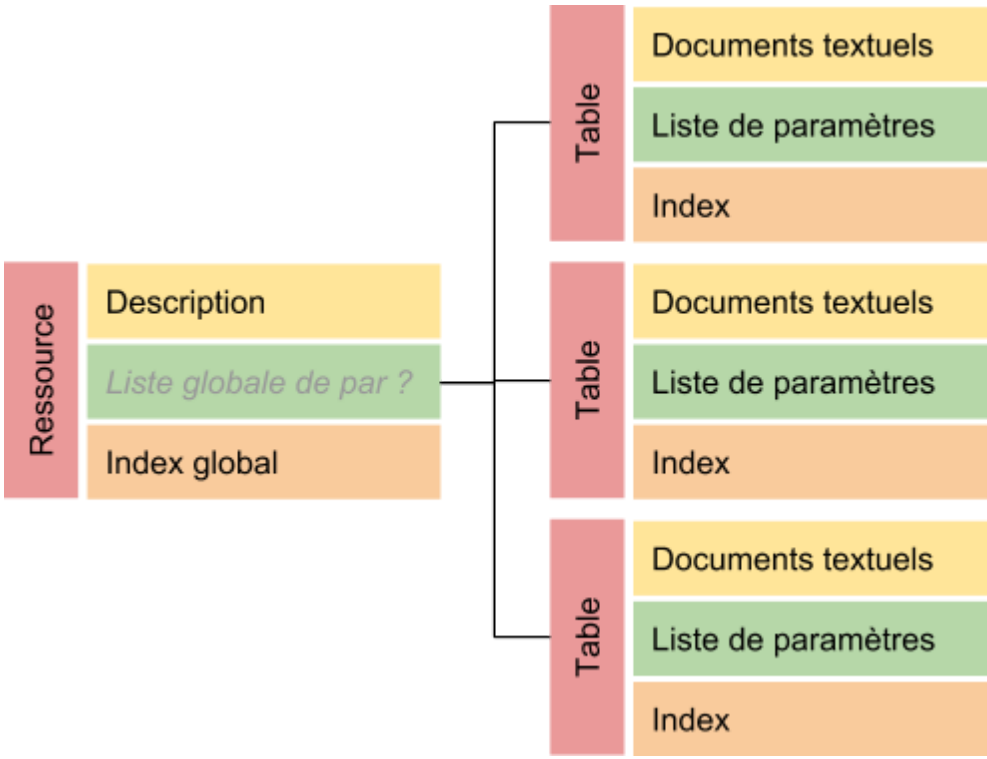
1. Les données textuelles qui sont fournies sous la forme de fichiers PDF, texte ou HTML. Ces fichiers ne pourront être ni modifiés ni adaptés avant indexation. Ce peut être des publications scientifiques, des notices de pipelines de traitement de données, des descriptions de tables relationnelles ou encore des documents de spécifications.

2. Les méta-données extraites de la base et qui décrivent pour chaque colonne les noms, unités, types… La génération de la liste des méta données fera partie des fonctionnalités de l'outil.

Le contexte d'utilisation de l'outil contraint l'organisation de la base documentaire:

- Une table est décrite dans un petit nombre de documents textuels.
- Ces documents sont locaux; ils sont attachés à la ressource en ligne.
- La description des paramètres figure dans un fichier à part.
- La base de texte est dans un format facile à afficher (PDF, texte, HTML)
- Un texte donné contient à priori la description de plusieurs (ou de tous les) paramètres, mais sa structure est assez rigoureuse pour considérer que les éléments relatifs à un paramètre donné sont groupés dans le texte (paragraphes, section…)
- On suppose que dans le texte, rien ne permet d'identifier les mots qui sont aussi des noms de colonne.
- La liste de paramètres est considérée comme une ressource fournie par un mécanisme spécifique implémenté par l'outil.

Le schéma ci-dessous montre la structure logique de la base documentaire attachée à l'application Web.



- **Documents textuels:** ensemble de textes décrivant un noeud particulier de l'application (une table)
- **Liste des paramètres:** Liste des paramètres utilisables pour contraindre les requêtes portant sur la table
- **Index:** index généré par l'outil pour une seule table
- **Description:** Description globale de la ressource. Cette description ne doit pas être indexée, mais doit être accessible d'une manière ou d'une autre depuis l'outil
- **Index global:** Indexation de l'ensemble des tables
- **Liste globale de paramètres:** Liste de tous les paramètres supportés par la ressource. La gestion de cette dernière est à la marge du st age

Indexation

L'indexation se fera d'abord au niveau table. L'administrateur fournit les fichiers texte au système puis enclenche le processus suivant:

1. Etablissement des listes de paramètres. Cette opération est très dépendante de la ressource sur laquelle on travaille
2. Segmentation des textes: Il s'agit de fragmenter les textes en morceaux relatifs à un paramètre donné
3. Indexation des fragments de texte
4. Déclenchement de la mise à jour de l'index global
5. Déploiement

Feuille de route du stage

Le stage se déroulera en trois phases:

1. Réalisation d'un démonstrateur
2. Réalisation d'une implémentation répondant aux exigences de base
3. Implémentation des exigences étendues

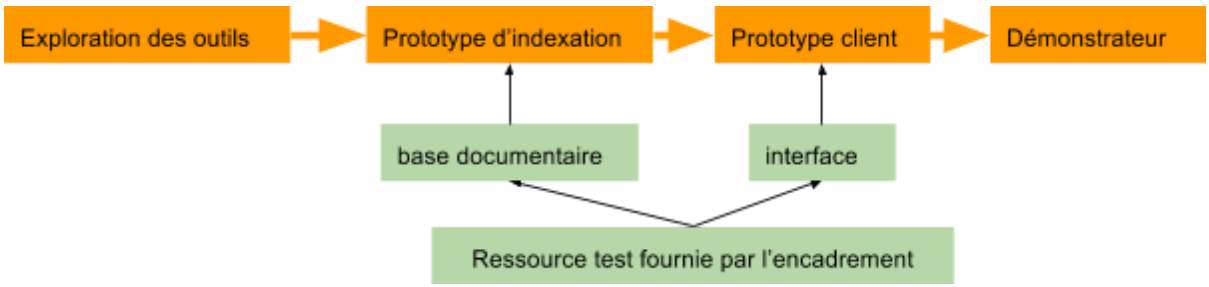
La phase 2 doit nécessairement être terminée au cours du stage car elle est indispensable à la mise en production de l'outil.

Chacune de ces phases correspond à un jalon du projet et fera l'objet d'une validation avec les encadrants.

Jalon 1: Réalisation du démonstrateur

Lors de cette première phase, le stagiaire devra s'appuyer sur des travaux réalisés précédemment à l'Observatoire pour choisir les bons outils.
Il devra choisir les outils et démontrer la faisabilité des fonctions de base:

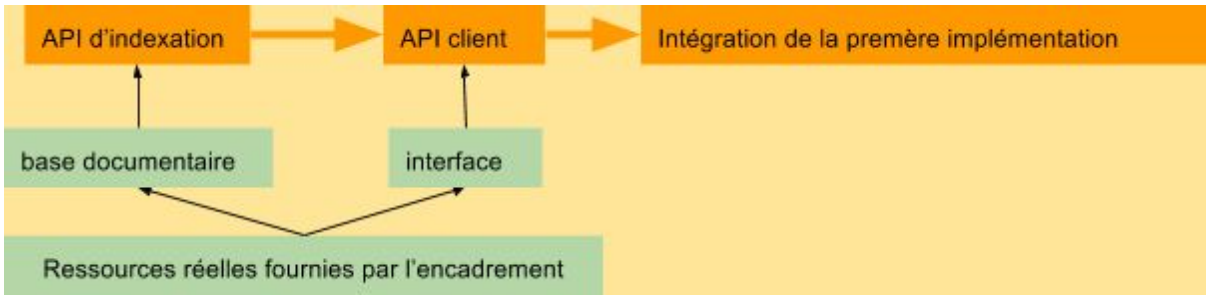
- Segmentation des textes à indexer
- Liens entre textes et paramètres



A partir de là, les outils seront définitivement choisis et un premier prototype pourra être présenté. Il servira à définir l'architecture finale

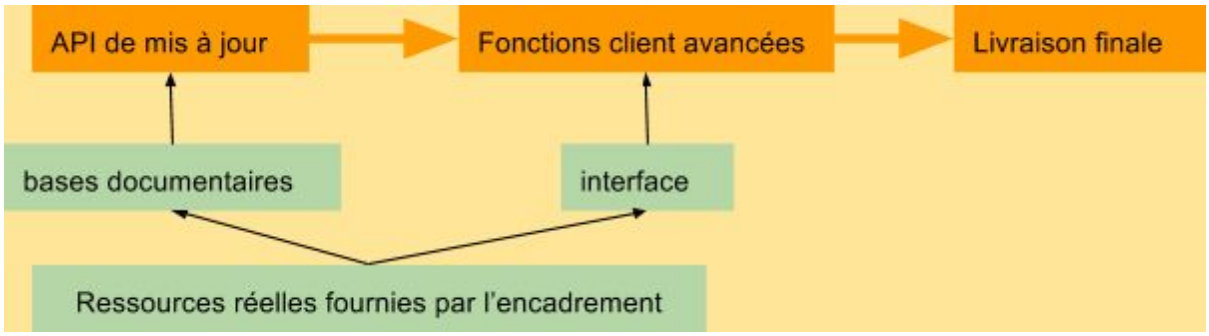
Jalon 2: Implementation de la première version fonctionnelle

Cette phase ne peut commencer qu'une fois les outils définitivement sélectionnés et l'architecture générale bien définie. Il s'agira de réaliser une première version fonctionnelle, implémentant aussi bien les fonctions d'installation que les services clients.
Ce **β** prototype sera validé sur au moins deux jeux de données (XMM et une table Vizier). Sa livraison s'accompagnera d'un jeu de tests mettant en avant aussi bien les requêtes traitées avec succès que celles qui échouent.



Jalon 3: Implementation des exigences étendues

Cette dernière phase vise à améliorer l'outil livré au terme du jalon 2. Elle devra permettre d'entendre l'usage de l'outil à un environnement multi-table et de l'utiliser pour localiser une table et non plus seulement un composant d'une table donnée.
D'autres fonctions pourront être explorées telles que l'intégration d'un système de notation de la pertinence des résulats agissant de manière rétroactive sur l'indexation.



Allocation de ressources-temps

La table ci-dessous donne une estimation de la durée allouée à chaque jalon.

Phase	**Début**	**Fin**	**Points clés**
#1	01/10/2016	15/11/2016	Exploration de l'existant
			Choix des outils
			Prototype d'indexeur
			Prototype d'API client
			Document d'architecture

#2	15/11/2016	01/01/2017	API finale d'indexation
			API finale pour le client
			Livraison d'un outil packagé
#3	01/01/2017	15/02/2017	Indexeur multi table
			Indexation incrémentale
			Notation de pertinence
			Client final

Livrables

L'outil délivré devra être déployable en l'état.
Il devra être packagé de manière à pouvoir être facilement déployé dans différents contextes.
Il devra impérativement implémenter les fonctions suivantes:

1. Indexation d'un corpus de données de manière persistante. Si la réalisation de certaines fonctions requiert des interventions manuelles, ces dernières devront être correctement implémentées (interfaces, APIs) dans le module d'indexation.
2. Traitement de requêtes utilisateurs via l'API
3. Mécanisme d'auto-complétion pour la saisie des paramètres

Il est souhaitable qu'il implémente les fonctions suivantes:

1. Indéxation incrémentale
2. Mécanisme d'évaluation de la pertinence des résultats
3. Mise à jour de l'indexation selon des notes de pertinence.
4. Widget JS configurable permettant une intégration facile de l'outil dans une application WEB

Toutes les interfaces devront être correctement documentées

Data Corpus

- XMM data
    - Pages Web
    - Publication scientifique
    - Documents de spécification
- Arches data
    - Publication scientifique
    - Documents internes du projet
    - Pages WEB

- Catalogue Vizier
    - Fichier README
    - Publication associée

Déroulement du stage

Le stage déroulera de manière présentielle à L'Observatoire.
L'avancement des travaux sera suivi quotidiennement via une plateforme collaborative (Wiki ou Google Drive). L'étudiant devra y consigner ses choix, les problèmes rencontrés ainsi que les solutions testées.
Le code sera versé sur le gestionnaire de version de l'Observatoire.
L'environnement de travail sera défini en collaboration avec les encadrants de manière à ce que le support futur du projet soit facilité.
L'étudiant aura la liberté d'organiser son travail comme il l'entend. Il aura une certaine liberté dans les choix techniques. Il lui sera demandé en retour de rendre compte par écrit de ses options de travail.
L'étudiant aura la charge de récupérer toutes les informations qu'il juge utiles auprès des responsables des deux bases de données les plus concernées: XCatDB et Vizier. Cette liste peut être étendue selon l'intérêt sucsité par ces travaux. Il sera également amené à prendre contact avec des futurs utilisateurs astronomes
Le travail se fera sur un ordinateur de bureau Linux. Un accès à des serveurs plus puissants ou à des volumes de stockage plus importants peut être envisagé.
L'étudiant devra présenter l'avancement de son travail à l'ensemble des personnels techniques de l'Observatoire.
L'ordinateur privé de l'étudiant n'aura pas accès au réseau privé de l'Observatoire.

Liens

- Mission XMM-Newton: http://www.cosmos.esa.int/web/xmm-newton
- SSC XMM-Newton: http://xmmssc.irap.omp.eu/
- XCatDB: http://xcatdb.unistra.fr
- CDS: http://cdsweb.u-strasbg.fr/index-fr.gml
- Vizier: http://vizier.u-strasbg.fr/viz-bin/VizieR