

PROSOP : un système de gestion de bases de données prosopographiques

Introduction :

Ce document présente l'outil en développement PROSOP qui permet la gestion d'une base de donnée prosopographique de la phase de création jusqu'à la phase de publication.

1. Architecture logiciel :

Le système de gestion de bases de données prosopographiques PROSOP est une application web développé en J2EE utilisant plusieurs technologies récentes. C'est une application n-tiers qui contiens les couches suivantes :

- Couche persistance : Contiens les composants assurant le « mapping » entre les objets définis dans la couche Métier et les composants de stockage définis dans la base de données. Cette couche ne peut référencer que les couches « Référence » et la base de données. Concrètement, il s'agit de la seule couche ayant un lien avec la base de données.

Pour cette couche les technologies utilisées sont Hibernate qui assure le mapping surmonté par Spring, ceci est surtout utilisé pour la gestion des utilisateurs, des droits d'accès et la gestion des versions.

- Couche référence : Cette couche contient les objets de données pures qui transitent entre toutes les autres couches. Ces objets sont aussi parfois nommés DataValues ou DataObjects.

Dans PROSOP, on utilise JAXB permettant de créer des classes [Java](#) à partir de [schémas \(.XSD\)](#) et inversement.

- Couche métier : Contient tous les composants métier dont le module. Ces composants métier ont en charge la gestion du cycle de vie des objets métier géré par le module. Cette couche ne peut référencer que les couches « Référence » et « Persistance »

- Couche application : Contiens tous les contrôleurs de cas d'utilisation du module. Cette couche assure le lien entre les composants graphiques et les composants métier. Cette couche ne peut référencer que les couches « Métier », « Persistance » et « Référence ».

- Couche présentation : Contiens tous les composants graphiques du module composant l'interface homme-machine (fenêtres, contrôle utilisateur...) avec le code propre à l'affichage de leur représentation et de leur contenu. Cette couche ne peut référencer que les couches « Référence » et « Application ». Mettre le moins de code possible dans cette couche. On utilise pour cette partie deux technologies, d'une part Java Server Faces avec l'implémentation riche en javascript PrimeFaces, et d'autre part XSLT pour la transformation des fiches XML en pages XHTML.

2. Choix de la structuration des bases de données :

Modèle relationnel :

Pour créer une base prosopographique, les outils informatiques jouent un rôle important, puisque c'est ce choix qui détermine la manière de création, d'utilisation et d'exploitation de la base. On peut citer comme outils les éditeurs de texte comme Word ou les tableurs comme Excel. Or, ce modèle n'est pas compatible avec une telle utilisation du fait de sa faible structuration, les erreurs de rédaction sont plus fréquentes, puisqu'il n'y a pas de contrainte sur la structuration, alors que la diffusion, l'exploitation et la mise à jour des données restent complètement manuelles et difficiles. L'utilisation des systèmes de gestion de base de données relationnelle comme MySQL corrige un grand nombre de ces problèmes par l'utilisation des tables pour bien structurer les données. Ils

permettent l'ajout de contraintes d'intégrité. Et le couplage avec PHP et d'autres langages de programmation simplifie l'affichage en Web, ces bases de données utilisant un langage de manipulation de données universel SQL (Structured Query Language). Ces bases de données relationnelles ont aussi des limites et ne répondent pas totalement à nos besoins. En particulier on peut citer les limitations suivantes :

- Structuration forte qui rend l'évolution du schéma très difficile.
- Pas d'interopérabilité.
- Pas de gestion des données manquantes.
- Utilisation des métadonnées non triviale.
- La multiplicité des tables rend la gestion et la mise à jour très techniques.

Modèle semi-structuré XML :

Le modèle semi-structuré est un modèle de base de données qui permet de représenter des structures :

- Irrégulières : on peut comparer des données avec des formats différents (une chaîne de caractères avec un ensemble)
- Implicites : données et structures (grammaire, schéma) sont mélangées
- Partielles : coexistence de données structurées et non-structurées

Le langage le plus utilisé pour ce type de base est XML¹. La qualité des modèles et des schémas XML est beaucoup plus importante par son incidence sur les systèmes d'information. En plus de la disponibilité d'une syntaxe standard et de beaucoup d'outils de manipulation facilitant l'interopérabilité et la mise à jour, ce métalangage est idéal pour représenter, modéliser, décrire et échanger des informations (voir schéma 1).

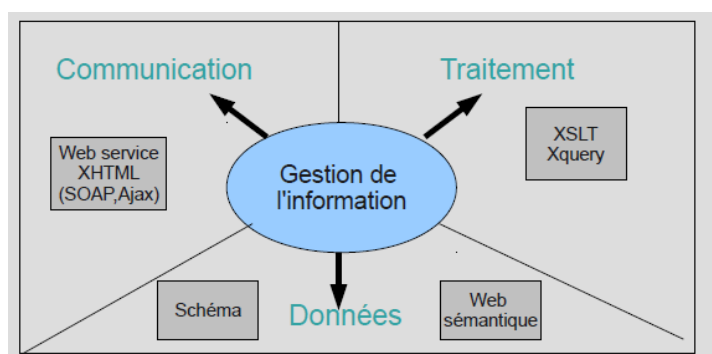


Schéma 1

Le formalisme XML peut-il remplacer les bases de données relationnelles telles que nous les connaissons ? La réponse est clairement non, et c'est même le danger d'une mauvaise utilisation du formalisme XML. Un document XML est un fichier texte ; il n'est optimisé ni en espace ni pour les manipulations que l'on peut opérer sur ce type de fichiers. Un document XML pourrait être davantage perçu comme une partie d'un système d'information, car il résout un problème de circulation de l'information à un moment donné. Il n'y a pas de raison que les bases de données relationnelles ne soient pas gérées à l'avenir comme aujourd'hui.

Le système actuellement en développement utilise ainsi d'une part une base MySQL pour la gestion des utilisateurs pour contrôler l'accès aux données avec des droits différents, et d'autre part une base XML où chaque fiche prosopographique est stockée dans un fichier XML valide par rapport à un schéma prédéfini.

¹ Standards W3C depuis 1998 d'échange et de structuration des données (<http://www.w3.org/XML/>).

3. Fonctionnalités de l'outil PROSOP :

Les droits d'utilisateurs (voir schéma 1)

L'application PROSOP contient plusieurs modules qui requiert des droits différents selon les utilisateurs :

Le Module gestion :

On distingue quatre types d'utilisateurs suivant leur rôle vis à vis du système :

- L'administrateur : son rôle consiste à gérer la création des comptes utilisateurs et leur affecter des droits spécifiques. C'est lui aussi qui a le droit de création d'une nouvelle base.

- Le rédacteur : peut créer, compléter ou supprimer une fiche.
- Le collaborateur : a les mêmes droits que le rédacteur vis à vie des fiches
- Le superviseur : fait partie d'un comité d'expert qui a la charge de la publication et la dépublication des fiches sur le Web.

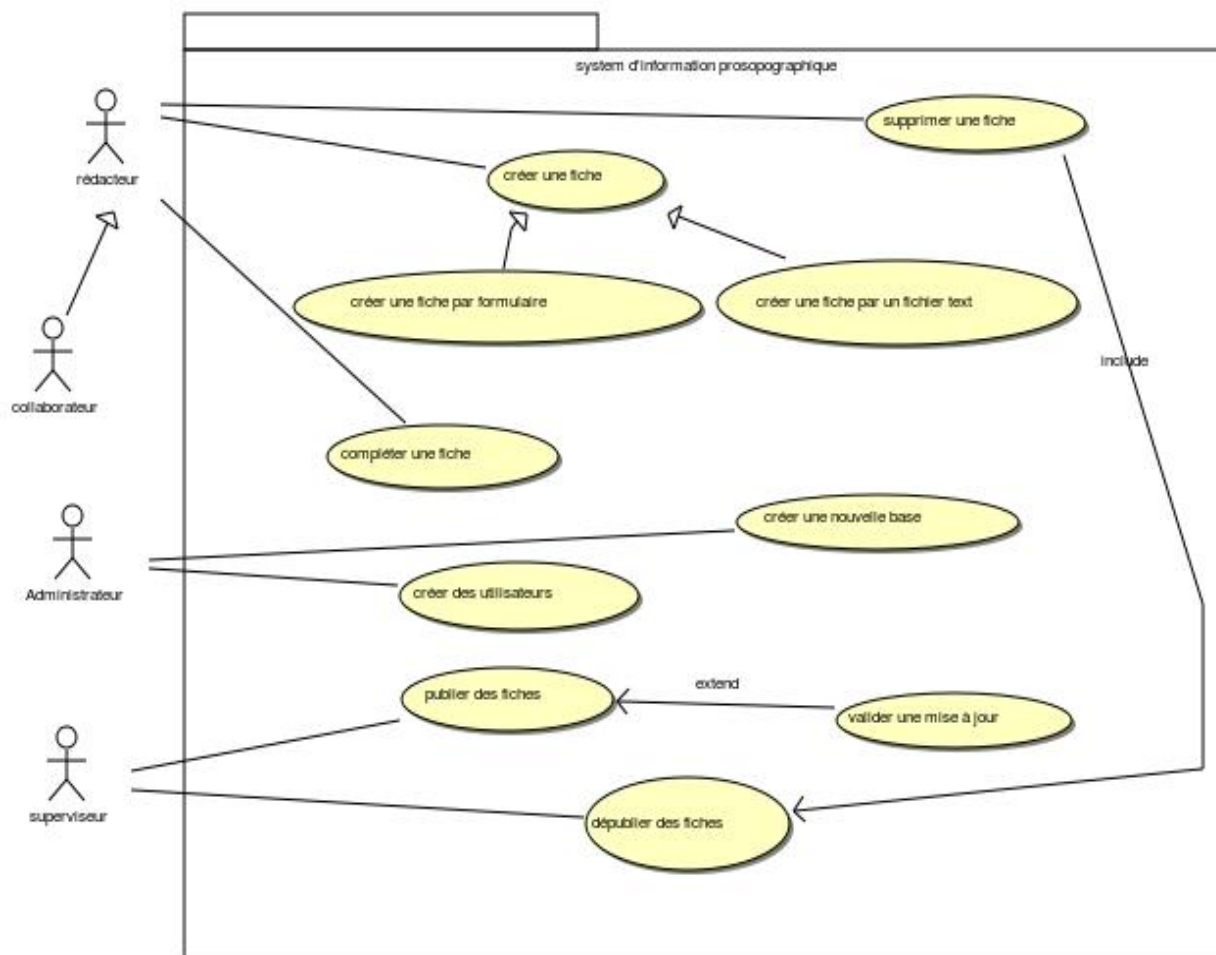


Schéma 1

La création des fiches :

On distingue ici cette phase de création des fiches de la phase de création de la base qui sera traité ultérieurement dans ce document. L'application Prosop est utilisé actuellement comme socle pour

la base Studium Parisiense, Cette base est constitué d'environ 17000 fiches prosopographiques consacrée aux membres de l'Université de Paris entre le XIIème et le XVIème siècle. Les quelques 17 000 fiches, qui doivent la composer à terme, comportent des informations biographiques et bibliographiques relatives entre autres aux étudiants, aux professeurs et aux suppôts de l'université. Cette base de données a été commencée dans un but pédagogique pour former les étudiants d'histoire au travail collectif de construction des bases de données historiques sous la direction de Jean-Philippe Genet. Au cours de ce premier travail, le dépouillement de répertoires biographiques et de travaux de recherche a permis de réunir une somme de données considérable.

Afin de rendre consultable et utilisable ces données, un travail d'harmonisation et de restructuration est mené actuellement.

En parallèle, un travail de dépouillement des répertoires biographiques et des travaux récents permet d'enrichir la base.

De ce fait, on peut créer des fiches selon deux méthodes différentes :

1 . Introduction de fiches dictionnaires : cette méthode suit la logique de construction originale de la base Paris qui consiste à rédiger à l'extérieur de l'application PROSOP des fiches prosopographiques sous format textuelle semi-structurées en utilisant des règles de rédaction prédéfinies (voir le document sur les règles de rédaction).

Dans l'application, pour introduire les fiches dans la base de donnée, on utilise une interface interactif qui permet de choisir les fichiers à charger ainsi que le fichier de configuration des règles de rédaction.

Pour la base paris ce fichier se présente sous la forme d'un fichier XML (variabe.xml) qui fait le lien entre les codes qui existent dans les règles de rédaction et les balises associées qui sont définies dans le schéma XSD. Ce fichier lie aussi chaque balise à un label suivant les langues.

Cette méthode de rédaction des fiches manuellement, même si elle s'avère nécessaire pour un nombre d'historiens, génère beaucoup d'erreur de saisie et sort du choix de la collaboration du fait de la possibilité du double rédaction d'une même fiche.

2. Rédaction par formulaire : Pour éviter ce problématique la création et la mise à jour des fiches par un formulaire qui permet de :

- utiliser un système d'indexation plus fin qui permet d'identifier les noms de personnes, les lieux, les institutions, les titres et les dates.
- Utiliser un champs commentaire qui peut être aussi indexé

La consultation des fiches : (voir schéma 2)

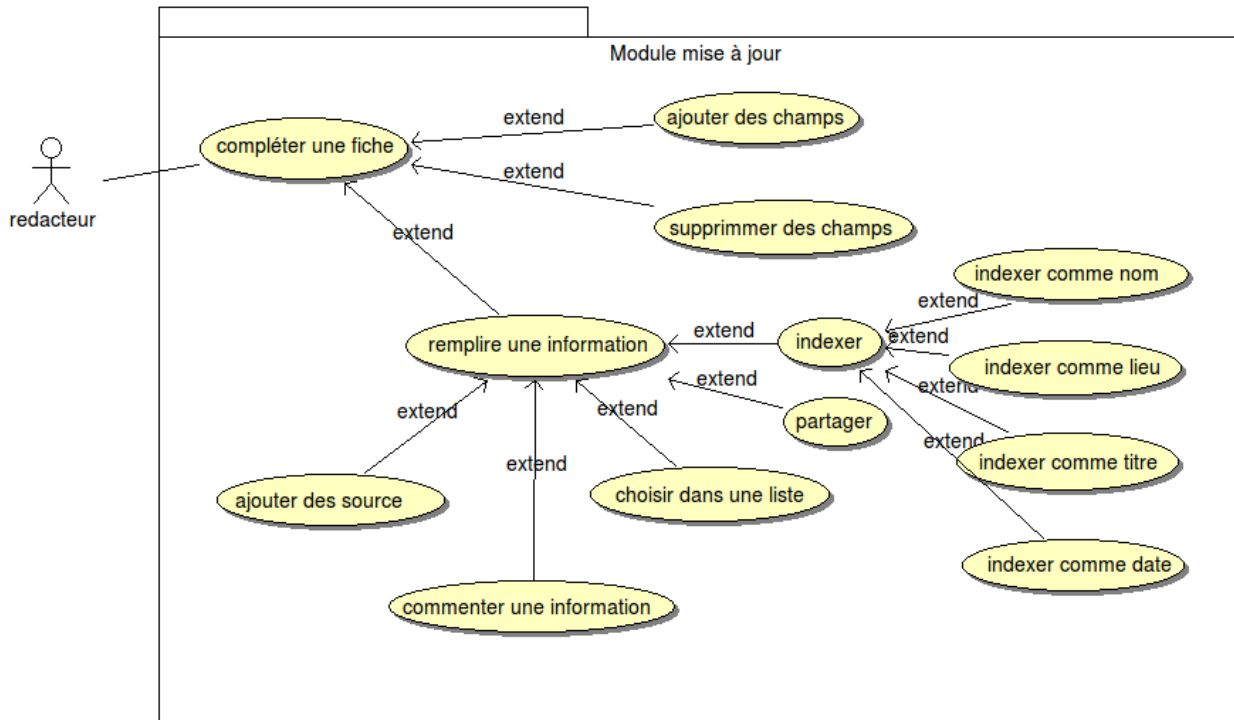
la consultation des fiches est rendu public sans aucune authentification, par ailleurs si on identifie on peut enregistrer une recherche. Les fonctionnalités principale sont :

- La recherche simple sur le nom, sur les œuvres ou sur tous les champs.
- La recherche avancée qui combine l'ensemble des champs importants de la fiche.
- La recherche sur un index alphabétique.
- L'exportation en PDF
- L'exportation Tabulaire
- Faire une analyse.
- Une exportation personnalisée par choix de d'attribut sur une liste.



Schéma 2

La mise à jour des fiches : (voir schéma 3)



La création de nouvelle base : (voir schéma 3)

