

Les aspects techniques

Introduction

Tous les processus pédagogiques développés pendant le projet ont été soutenus par une infrastructure technologique complexe et innovatrice. Les composants techniques soutiennent et intègrent en particulier trois fonctions principales:

- un environnement éducatif virtuel (EEV) complet, qui permet la création et la prise en charge des cours en ligne, et la transmission de ces cours à des groupes d'étudiants ;
- un stock d'études de cas, ou base de données d'études de cas (Case Study Database -CSDB), dans laquelle les travailleurs sociaux peuvent télécharger leurs expériences de façon structurée, et dont tous les étudiants et les autres travailleurs peuvent parcourir le contenu ;
- un sous-système d'indexation sémantique qui relie et connecte l'information dans les deux systèmes cités.

L'innovation technologique dans le projet de recherche concerne tout particulièrement l'infrastructure d'élaboration sémantique, qui permet d'intégrer le contenu didactique disponible dans le EEV de manière approfondie, et de réfléchir sur l'expérience pratique stockée dans le CSDB. Cette assimilation est visible par l'utilisateur sous grâce aux « liens » qui apparaissent lorsque les étudiants consultent les cours en ligne. Ces liens mènent à une liste d'études de cas intéressants en rapport avec le contenu du cours. L'innovation principale concerne la création d'un système automatisé et intégré qui permet d'identifier automatiquement, grâce aux technologies d'intelligence artificielle, quelles études de cas sont pertinentes.

Le choix d'associer l'automatique et l'intelligent donne quelques avantages : l'auteur du cours n'a pas besoin de connaître personnellement toutes les expériences associées au niveau européen, et les nouvelles expériences téléchargées dans le CSDB seront immédiatement connectées aux modules didactiques adéquats. Ceci implique notamment une mise à jour automatique du contenu des cours : le module n'est révisé que lorsque les auteurs ont besoin de le modifier, mais le contenu global présenté aux utilisateurs est constamment mis à jour et se réfère en permanence aux adjonctions les plus récentes du répertoire d'études de cas.

Le système est très complexe à construire et à gérer, puisque cela requiert l'intégration de fonctions interactives d'apprentissage par ordinateur, avec des algorithmes d'intelligence artificielle sophistiqués, basés sur une ontologie construite par une équipe d'experts internationaux. Néanmoins, cette complexité est masquée par un interface extrêmement simple pour les utilisateurs, qui consiste pour ces derniers (les étudiants) en un bouton '**Expériences associées**', alors que les créateurs du contenu (ici, les auteurs des modules didactiques et des études de cas) peuvent sélectionner les documents en naviguant simplement à travers un index thématique annoté.

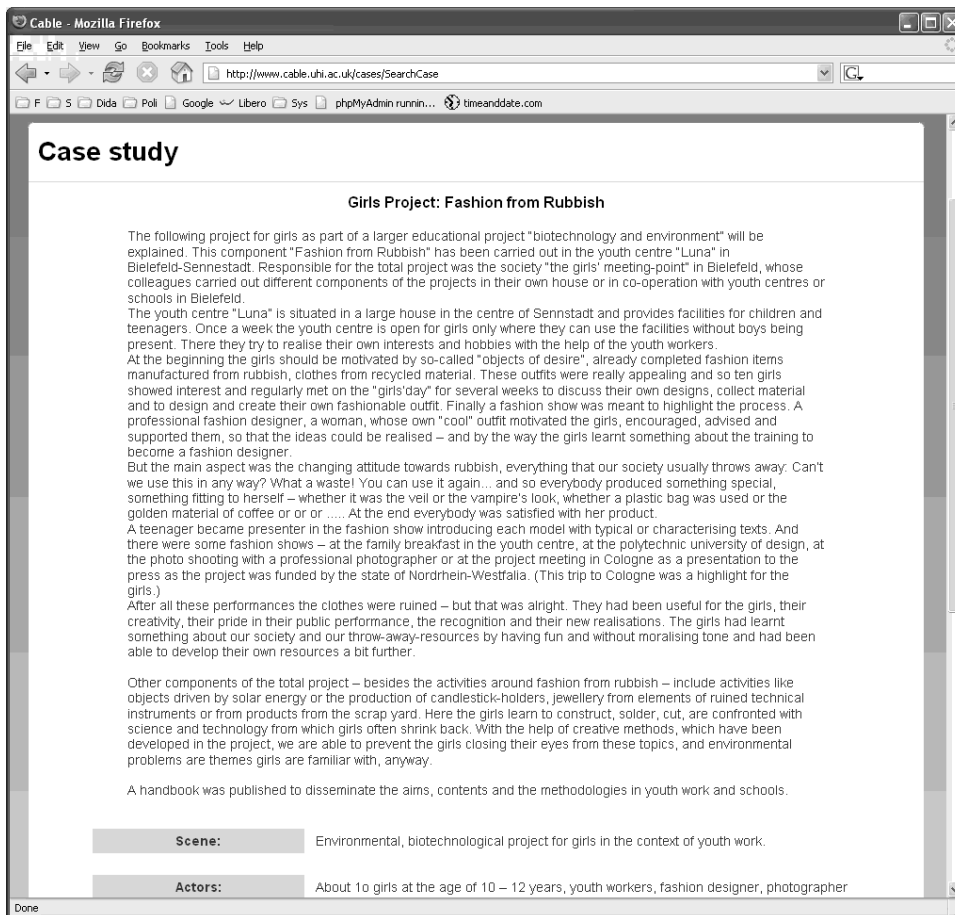


Figure 0.1: exemple d'une étude de cas

Le sous-système de la base de données des études de cas

La base de données des études de cas (BDEC) est un système de gestion de contenu, principalement conçu par YHSV (Yrkeshögskolan Sydväst), dont l'objectif est de fournir une application facile et simple à utiliser pour la publication d'études de cas en différentes langues.

Un système de gestion de contenu (*Content Management System - CMS*) est un dispositif dont le but est la récupération, le traitement et la publication de matière (contenu) au travers d'un ou plusieurs media de communication. Classiquement, dans la phase de récupération, la matière (contenu) est générée par la rédaction de nouveaux articles, ou par la récupération d'articles issus d'autres sources de données. Dans la phase de gestion, les articles sont traités et stockés pour une élaboration ultérieure, et sont étiquetés avec des méta données, c'est à dire, une information décrivant le contenu et son état. Le contenu est ensuite modifié suivant un protocole bien défini par des procédures manuelles ou automatiques. La phase finale est la publication, lors de laquelle les divers composants sont rassemblés en un format et une présentation appropriés, prêts à être livré à l'utilisateur final. Les CMS n'ont pas tous les mêmes fonctions ni ne peuvent gérer le même type de données, parce qu'ils sont destinés à des publications, à des organismes très différents et à des coûts variables.

L'architecture des études de cas

Dans le cas présent, une étude de cas correspond à un document décrivant une situation donnée et composée des unités de contenu suivantes :

- un récit ;
- une scène ;
- des acteurs ;
- des actions ;
- les réactions de l'entourage ;
- le temps, l'espace ;
- un objectif ;
- un contexte.

Les utilisateurs du système sont gérés par un protocole de LDAP²¹. Avec le LDAP, les utilisateurs ont le même nom et le même mot de passe que dans le EEV (environnement d'enseignement virtuel).

Les opérations possibles sont des classiques d'un CMS, aussi bien pour l'administration du système que pour la gestion du contenu, telles que :

- ajouter, éditer ou supprimer un lien vers un réseau sémantique ;
- ajouter, éditer ou supprimer une étude de cas ;
- traduire une étude de cas ;
- rechercher une étude de cas.

L'aspect polyglotte est géré en donnant l'identification de la langue de chaque document.

Avant d'écrire une étude de cas, l'auteur du contenu doit choisir la langue dans laquelle elle va être rédigée (ou la langue par défaut associée à l'utilisateur sera incorporée au contenu).

De plus, il y a un système de modification d'édition, dont le but est de faciliter la traduction d'une langue à une autre. Lors d'une traduction, les passages de l'étude de cas dans la langue d'origine apparaissent dans une fenêtre, côte à côte avec le passage correspondant de l'étude de cas dans la nouvelle langue.

Les études de cas dans différentes langues sont reliées ensemble pour faciliter le contrôle par les administrateurs et les auteurs du contenu.

²¹ Lightweight Directory Access Protocol. Voir:
<http://www.kingsmountain.com/LDAPRoadmap/index.shtml>,
<http://www.redbooks.ibm.com/abstracts/sg244986.html> ou <http://www.cru.fr/ldap/> (en français)

Le sous-système de l'environnement d'enseignement virtuel

L'environnement d'enseignement virtuel (EEV) est un système de gestion de contenu (*Content Management System – CMS*) destiné à fournir un environnement dans lequel, à partir de blocs de contenu stockés dans un dépôt central, le contenu didactique peut être transmis à l'utilisateur par un système informatisé ; et le progrès de l'étudiant peut être contrôlé et sauvegardé. L'EEV possède une interface adéquate qui permet à l'utilisateur de s'inscrire et de suivre des cours pendant la durée de la formation, et une interface d'administration, qui permet aux enseignants de rédiger de nouveaux cours, de suivre les étudiants, de créer des exercices, de vérifier les résultats, de personnaliser l'aspect des cours, etc...

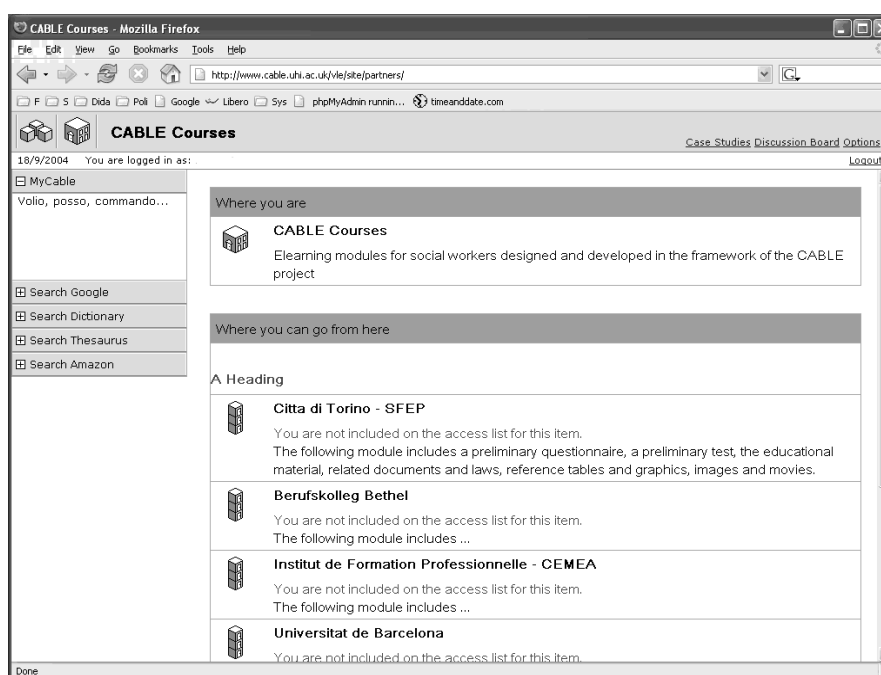


Figure 0.2: exemple d'une séance utilisant le EEV de Bodington

Le système de Bodington

Dans le contexte de ce projet, on a choisi le système d'EEV de Bodington²².

Développé à l'origine par l'université de Leeds pour créer leur propre environnement d'enseignement virtuel, Bodington est un logiciel Open Source que la communauté académique mondiale peut utiliser gratuitement et dont elle peut aider le développement. Bodington pourvoie à l'environnement d'enseignement virtuel en Grande Bretagne et à travers le monde.

Le système de Bodington est livré avec une licence Open Source : le logiciel a été créé à l'université de Leeds pour en faire profiter les étudiants et le corps enseignant et non pas pour une exploitation commerciale.

²² Voir : <http://www.bodington.org/index.html>

Le Bodington est utilisé actuellement par un grand nombre d'institutions pédagogiques, dont le UHI Millenium Institute, l'Université de Leeds, l'Université d'Oxford, l'Université de Manchester, et les collèges de Yorkshire Coast et de Eton.

Le système Bodington est un outil informatique qui peut être utilisé conjointement avec une base de données pour réaliser un site Web interactif. L'idée de départ était de pouvoir mettre en place un environnement d'enseignement virtuel à l'Université de Leeds - Bodington Common – mais de pouvoir l'utiliser aussi comme base pour d'autres types de sites interactifs. En fait, sa définition exacte serait celui d'un serveur d'application, puisque des modules peuvent y être ajoutés pour exécuter d'autres fonctions.

Ce système possède certaines caractéristiques telles qu'un système d'exploitation Multi-Utilisateur, mais accessible via une interface Web mondial. En outre, ce système, avec son approche orientée Objet, évite les pages de configurations complexes. On saisit le sujet qui nous intéresse et les pages de navigation ne listent que les options ayant trait au sujet choisi.

Les données et les pages d'un site sous système Bodington sont présentées sous forme d'arborescence, comme les documents et les fichiers dans un système de classement d'un disque dur. Mais, pour rendre la navigation plus facile, certains "documents" utilisent l'image d'un bâtiment, d'un étage ou d'une pièce. Cela veut dire qu'on peut indiquer à quelqu'un où trouver une ressource en langue courante au lieu de donner une adresse Web. A l'inverse de certains environnements d'enseignement, les outils, tels que les espaces communication, les questionnaires, etc. existent en tant qu'objets au sein de cette arborescence et lorsqu'on entre dans un questionnaire, par exemple, on obtient un menu simple de commandes et d'options qui sont en rapport spécifique avec le questionnaire, sans être encombré par du texte associé à d'autres outils dans le voisinage. Une barre de navigation simple permet de sortir facilement de l'outil en cours.

Les logiciels de EEV dictent souvent la structure d'une zone de cours, et peuvent manquer d'éléments nécessaires pour personnaliser certains cours. Avec le système Bodington on peut tout changer de la manière dont un cours s'affiche et fonctionne. On peut changer les icônes et les illustrations en remplaçant simplement les fichiers graphiques dans les dossiers patrons. De même, le changement des textes instructifs dans les formules et les pages est tout aussi simple puisqu'ils existent en format HTML dans les dossiers patrons. Des fragments de texte insérés de manière interactive dans les patrons sont générés par un logiciel en fonction, donc tout changement nécessite un travail sur le logiciel lui-même ; mais le code complet des sources est fourni et la licence permet toute modification utile.

La législation à travers le monde, y compris aux Etats Unis, en Grande Bretagne, en Suède, aux Pays Bas, etc., requiert de plus en plus, que les étudiants accèdent à un niveau d'éducation standardisé, sans tenir compte de leurs éventuels handicaps. Ceci implique que lorsque l'accès à un EEV fait partie de la formation, le EEV doit être accessible à tous. Le système Bodington s'efforce autant que possible d'adhérer étroitement aux recommandations du W3C. Le but de celles-ci est d'éviter l'utilisation inappropriée du HTML qui pourrait nuire aux technologies d'aide, telles que

les navigateurs audio sur le Web. Néanmoins, ceci ne rend pas les pages Web suffisamment accessibles. Pour aider les utilisateurs dyslexiques et ceux avec des problèmes de vue en particulier, le système Bodington peut leur fournir une liste de feuilles de styles. Le système enregistre la feuille de style choisie par chaque utilisateur et l'appliquera automatiquement lorsque l'utilisateur se connectera.

Le site Web peut donc s'afficher en couleurs très ou peu contrastées , en grands ou petits caractères etc., selon les besoins de l'utilisateur.

Le système Bodington correspond à un serveur d'application. Ce qui veut dire qu'il fournit des services de sauvegarde et de recherche dans une base de données, des systèmes pour gérer les utilisateurs, pour contrôler l'accès aux ressources, etc. Un programmeur peut tirer parti de cette fonction pour « ajouter » un code supplémentaire afin d'étendre cette fonctionnalité avec un minimum de programmation.

L'installation d'un seul système Bodington peut être divisé en plusieurs zones d'administration. Les administrateurs d'une zone peuvent créer les logins et les mots de passe, gérer les adhésions, etc. indépendamment des autres zones d'administration. Les administrateurs d'une zone peuvent donner accès à des ressources spécifiques à des utilisateurs spécifiques d'autres zones administratives, ce qui permet l'utilisation collaborative du site Web.

Au sein de la plupart des produits EEV se trouve une entité appelée *formation* dans laquelle différents outils peuvent être activés ou non. En essayant d'enfermer le concept de la *formation* dans le logiciel, on est prisonnier de la façon d'enseigner de l'époque de la création du logiciel. Le système Bodington est différent : il fournit des immeubles, des étages, des pièces, qui permettent d'organiser la substance pour s'adapter le mieux à la structure d'une formation spécifique et il donne des outils tels que des espaces de discussions, des questionnaires, des documents Web, un multiple choix d'articles, etc. qui peuvent être agencés librement.

Le sous-système d'indexation sémantique

Cette section expose les principes à la base du sous-système sémantique, qui relie le sous-système des cours virtuels à celui des études de cas. Le système sémantique est d'abord un service d'indexation et travaille en arrière plan ; sans être directement visible par les utilisateurs, il est de fait un élément clé et central du système.

Ce sous-système n'est pas un système de recherche de document classique, car il n'utilise pas des mots et des phrases comme élément de base de recherche, mais il a une approche sémantique. Dans cette approche l'unité de recherche de base est le concept.

Les systèmes de recherche de documents et leurs inconvénients

L'idée initiale d'associer automatiquement les cours et les études de cas (que nous appellerons ci-après 'documents') était d'ajouter des mots clés à la description de tous les documents au sein du système, en utilisant un moteur de recherche classique pour indexer les mots clés ou indexer un texte entier sur le corps de documents. Ensuite, en introduisant des mots ou phrases clés, on devait pouvoir retrouver des documents associés pertinents. Les

moteurs de recherche sont utilisés dans le domaine de la technologie informatique depuis les années 70 ; beaucoup de bibliothèques et de programmes sont disponibles, gratuits ou payants, et la littérature sur le sujet est croissante.

Malheureusement, cette solution a des inconvénients qui ont rendu son utilisation inappropriée pour notre application.

Tout d'abord, deux mots ou plus pouvaient être homographes, c'est à dire, avoir la même orthographe, mais être des dérivés et avoir des sens différents. Par exemple, le mot *chair* veut dire une chaise pour une personne, le rang du professeur, l'officier qui préside aux réunions d'un organisme ou un bloc de fer utilisé pour supporter les rails du chemin de fer. Le mot *lounge* désigne une banquette rembourrée pour plusieurs personnes, ou une salle d'attente équipée en sièges. Si on cherche des informations sur les meubles de salon en utilisant un moteur de recherche basé sur des mots clés, comme Google, le mot clé '*lounge*' donnera en première page de résultats des labels de disques et des personnalités de la radio ou de la télévision. Bien sûr, il y a toujours moyen de peaufiner la recherche en ajoutant d'autres mots clés, mais des résultats intéressants pourraient être ignorés.

A l'inverse, le problème des synonymes. Par exemple, le mot *sofa* (*canapé en français*) a le même sens que *lounge*, mais Google ne l'indexe pas ; on doit donc renouveler la recherche en utilisant *sofa*, en ayant toujours les mêmes sites de labels de disques qui encombrant les résultats. Un autre problème se soulève lorsque la recherche utilise un terme général, par exemple *furniture* (*meubles en français*), mais qu'on s'intéresse aussi à des termes spécifiques tels que chaise (*chair*), table (*table*) et meuble de rangement (*cabinet*).

D'autres problèmes surviennent encore quand un document en différentes langues est utilisé; dans ce cas un moteur de recherche ne peut associer un mot dans une langue avec le même mot dans une autre, sans sous-programme supplémentaire. Il est très difficile de relier un document en une langue au même document traduit dans une autre langue, même si ces documents concernent les mêmes choses, mais qu'ils ont différentes 'étiquettes' pour les décrire.

L'approche sémantique

La première étape pour résoudre ce problème est d'assigner un seul sens classique à chaque mot clé : utilisant les mots cités ci-dessus, on peut choisir de définir, dans notre contexte d'application, *chair* comme un siège pour une seule personne et *lounge* comme une banquette rembourrée pour plusieurs personnes. Nous pouvons aussi définir des synonymes des mots, impliquant que plusieurs mots peuvent représenter le même concept.

De plus, nous pouvons définir des concepts qui ne correspondent pas qu'à un seul mot dans notre langue, mais qui conserve un sens dénué d'ambiguïté et clairement défini. Dans notre exemple nous aurions pu définir un concept de « fonction d'assise » comme un objet adapté pour s'asseoir.

L'assignation d'un seul sens à un mot, ou à un groupe de mots clés, nécessite la définition d'un nouveau concept. Un concept, dans le sous-système sémantique est défini par :

- un identifiant unique (id), qui est utilisé au sein du système ;

- une courte description (sofa [*divan*]);
- une définition (un siège long, dont le fond, le dos et les côtés sont habituellement rembourrés, couramment utilisé comme meuble confortable).

Une fois les concepts définis, il est possible de les relier avec des objets de la même famille. Un concept pourrait couvrir une signification plus précise, respecter une signification plus large ou représenter une qualité ou une partie d'un autre concept. Cette technique engendre une méthode plus aisée pour classer les documents, mais permet surtout une recherche plus puissante, en associant des documents qui concernent des concepts similaires.

L'utilisation de l'approche sémantique avec les concepts, aide à résoudre le problème des langues multiples.

Par définition, les concepts, ne sont pas liés à une seule langue, il est donc possible avec cette approche de rechercher des arguments similaires parmi des documents en différentes langues. Il est possible d'avoir en effet, des descriptions courtes, des définitions et des mots en plusieurs langues, tous reliés à un concept unique. Un document en italien qui parle de *sedia* pourrait être relié à un document en anglais qui parle de *chair*.

Avec une série de concepts donnés et leurs associations appropriées, il est possible de créer un modèle théorique qui décrit le domaine de connaissance dans lequel l'application fonctionne. Ce genre de modèle s'appelle une ontologie. Une ontologie simplifiée ne peut contenir qu'une classification hiérarchique et qu'un seul type d'association— plus précisément appelée une taxonomie – et pourrait être représentée comme un arbre avec des nœuds correspondant aux concepts et des arcs représentant leurs associations. Une ontologie pourrait aussi être représentée en graphique dans lequel les associations seraient représentées par des arcs étiquetés, sans structure hiérarchique.

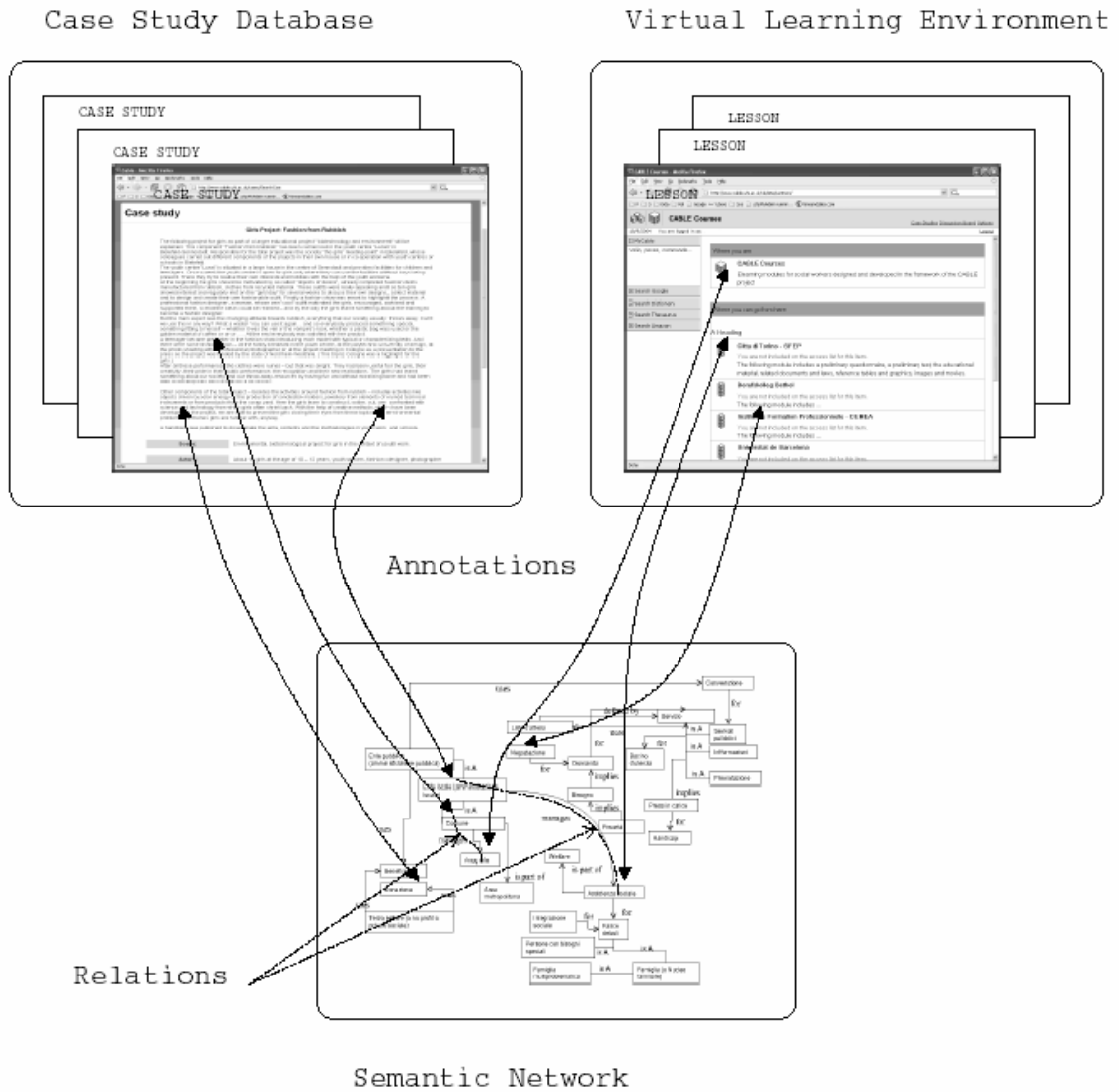


Figure 0.3: Assemblage des études de cas avec le EEV grâce au système sémantique

H-Dose

Le système utilisé pour l'indexation sémantique est le H-DOSE. L'association de Web Services et de Multi Agent Systems fournit un accès simple à la fonction sémantique, avec une attention particulière sur l'indexation, la recherche et la fonction de recherche étendue.

Puisque le H-DOSE a été développé en tant que projet de recherche à l'université polytechnique de Turin, c'est une source d'accès libre : le projet est distribué sous licence gratuite et fonctionne avec des technologies gratuites : un serveur Apache Tomcat, un système d'administration de la banque de données PostgreSQL, le réseau Apache Axis SOAP, l'accès et l'administration à l'ontologie Jena API de Hewlett-Packard et le cadre logiciel TILAB²³ Jade pour le déploiement des agents.

²³ anciennement CSELT

Le H-DOSE possède une architecture de distribution, c'est à dire qu'on peut étendre divers sous-systèmes à différents ordinateurs, communiquant entre eux par des messages SOAP²⁴, permettant plus de souplesse dans la forme du système et améliorant ainsi la performance globale.

La plateforme peut gérer des ressources de formats et de langues variés, bien que son objectif actuel soit les ressources textuelles. Le but du H-DOSE est de fournir une fonction sémantique à des applications Web à travers une interface facile d'accès, permettant l'intégration rapide des services fournis dans le flux de travail en cours. Ceci en essayant d'optimiser le rapport bénéfice/coût de la procédure. Les services de la plateforme sont dirigés vers des tâches d'indexation sémantique et de recherche, dans le contexte d'un domaine de connaissance spécifique, défini par l'application qui utilise la plateforme.

Plus précisément, le H-DOSE offre des moyens de classer les ressources textuelles en les associant avec les concepts d'une ontologie. De plus il fournit une fonction de recherche d'associations entre les ressources et les concepts, permettant la recherche d'information dont le concept est en rapport avec les demandes d'applications. La recherche exploite la structure ontologique de la base de connaissance et prend en compte des relations non explicites entre les concepts, par un système de navigation simple.

Le H-DOSE a été conçu pour minimiser l'impact et l'effort nécessaires à l'intégration des services de la plateforme dans les applications courantes. Ainsi, les choix de conception révèlent cet objectif en séparant, autant que possible, le domaine du modèle conceptuel des ressources. Il définit aussi les associations entre les ressources et les concepts, qui sont basés sur les standards du Web. Grâce à la notion "d'impact minimal", l'approche choisie pour résoudre le problème des langues multiples consiste en une seule approche ontologique basée sur l'idée que celles-ci partagent au niveau conceptuel bien plus de ressemblances que de différences.

Les concepts sont considérés comme des entités indépendantes de la langue, situées à un niveau de spécification qui fait abstraction de la définition ontologique des éléments propres à la langue. L'information spécifique à une langue est gérée en associant à chaque concept indépendant d'une langue une série de noms, de définitions et de mots clés, un par langue utilisée dans l'application. Le nom du concept, sa définition et les mots clés s'appelle "Synset" puisque les mots clés sont presque tous des synonymes que les humains adoptent pour repérer les sémantiques des documents tels que les ressources textes. La structure de l'ontologie et le "synset" sont sauvegardés sous fichiers XML, rendant le lien plus aisé à modifier avec des programmes *ad hoc*, tels que '*Protégé*' ou des éditeurs tout venant.

²⁴ Simple Object Access Protocol : Protocole simple d'accès à un objet – un protocole de conventions minimales pour invoquer le code en utilisant XML sur HTTP, dans un environnement de distribution décentralisé. Voir <http://www.w3.org/TR/soap/>

Les associations entre les ressources et les concepts ontologiques sont appelées 'annotations'. Les annotations peuvent être définies soit manuellement, en sélectionnant des concepts adaptés à un document donné, ou à une partie de ce document, et en envoyant l'ensemble avec le document pour être indexé ; soit automatiquement, utilisant des algorithmes d'extraction de données à partir du texte à indexer.

Dans le premier cas, les annotations sont plus fiables car elles ont été définies par des *experts* ; néanmoins il y en a peu car leur création coûte cher et elles pourraient aussi être appliquées à des données non textuelles, comme des films, des sons ou des images. En revanche, une machine extrait beaucoup d'informations, mais il est probable qu'elles soient moins précises que celles générées par des êtres humains.

Dans les deux cas, les annotations ont un poids qui permet de prendre en compte les différents degrés de fiabilité et de constance, favorisant la définition de diverses puissances d'association entre les ressources et les concepts ontologiques. En d'autres mots, puisqu'une ressource étayera des arguments plus largement qu'un seul concept, cela sera indiqué par une quantité non négligeable d'annotations lourdes, chacune associant la ressource avec un concept ontologique donné.

Les annotations permettent à la plateforme de prendre en charge la confusion intrinsèque des ressources, en exploitant le poids des annotations. Quoi qu'il en soit, les contraintes ontologiques demeurent, et le modèle du domaine des concepts et des relations est toujours pris en compte pour fournir des services sémantiquement riches.

Les annotations suivent la corrélation sémantique entre une ressource et un concept, y compris l'information concernant la force de cette corrélation. Les données concernant les concepts, la référence aux documents et au groupe d'annotations, sont sauvegardées en forme de tableau dans une base de données relationnelle, représentant, à un niveau inférieur, les arguments qui se trouvent dans le document.

En fait, le H-Dose a besoin d'un URI²⁵ pour référencer le contenu, mais il suffit d'avoir une procédure qui associe un document à un URI ; et cette procédure peut être externe aux applications existantes. Bien sûr il est possible de stocker des annotations dans le document en cours d'indexation ; cela peut être utile quand les annotations sont faites manuellement, mais ceci n'est pas une obligation. Avec cette approche, les données sont stockées à l'extérieur du document ce qui permet l'utilisation du système sans avoir besoin de modifier les structures et le contenu du document. Le H-DOSE pourrait donc être utilisé sans modifier l'application existante.

Comme nous l'avons mentionné plus haut, le H-DOSE utilise le PostgreSQL comme système DBMS, mais il serait facile d'adapter celui-ci pour utiliser

²⁵ Le Uniform Resource Identifier (URI) est un fil compact de caractères pour l'identification d'un résumé ou d'une ressource physique. Le Uniform Resource Locator (URL) est une forme de URI qui fournit aussi le moyen de localiser la ressource en décrivant son mécanisme d'accès primaire. Voir : <http://www.w3.org/Addressing/>

d'autres systèmes de bases de données, libres ou sous licence, en se servant d'une classe de JDBC différente et en adaptant la syntaxe des recherches SQL à une autre cible.

Des applications externes pourraient interagir directement avec les messages SOAP, mais un portail XML-RPC à SOAP existe et donne plus de flexibilité pour écrire le code qui lie les services du H-DOSE aux systèmes existants.

Les applications existantes pourraient utiliser le H-DOSE pour fonctionner comme moteur de recherche de texte, tant que les demandes d'indexation de nouveaux documents, la suppression d'anciens et les recherches sont modifiés pour solliciter les méthodes du H-DOSE appropriés, au lieu de ceux trouvés dans le moteur de recherche de textes.

En conclusion, le H-DOSE favorise une approche sémantique de la recherche de documents, avec un minimum d'effort nécessaire pour adapter les logiciels existants. Avec son approche distribuée, il est possible de mettre en route un système plus souple et, puisque le logiciel est gratuit, il est facile de l'adapter à des environnements et des recherches variés.