

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE**  
**UNIVERSITE M'HAMED BOUGARA – BOUMERDES**  
**FACULTE DES SCIENCES**



**Mémoire de Magister**

**Présenté par :**

**ADDOUR Drifa**

**Filière : Systèmes Informatiques et Génie des Logiciels**  
**Option : Spécification Logiciels et Traitement de l'Information**

**Développement d'une plate forme pour la  
découverte de services web.**

**Devant le Jury:**

Mr. MEZGHICHE Mohamed  
Mr. AHMED NACER Mohamed  
Mr. AIT BOUZIAD Ahmed  
Mr. AHMED OUAMER Rachid

Professeur, UMBB  
Professeur, USTHB  
M.C.B, UMBB  
M.C.A, UMMTO

Président  
Examineur  
Examineur  
Encadreur

# *Remerciements*

*Je remercie en premier lieu Monsieur AHMED-OUAMER Rachid pour la confiance qu'il m'a accordée en me proposant ce sujet. Comme je le remercie pour sa patience, sa disponibilité et ses conseils précieux.*

*Je tiens à remercier également les membres du jury pour l'honneur qu'ils m'ont attribué en acceptant d'examiner et d'évaluer mon travail. J'espère que ce travail sera à la hauteur de leurs exigences scientifiques.*

*Enfin, Que toute personne ayant contribué de près ou de loin à la réalisation de ce travail par une quelconque forme de contribution trouve ici le témoignage de ma plus profonde reconnaissance.*

**Résumé.** Dans le cadre de l'enseignement à distance, un des problèmes qui se pose est celui de la disponibilité des ressources pédagogiques. L'élaboration de ces ressources au fil de l'eau est possible mais ne permet pas de répondre aux besoins qui sont souvent très importants. Une variété de ressources pédagogiques est disponible sur le net, parmi ces ressources celles qui sont présentées sous forme de services web. Dans ce papier est proposée une infrastructure pour la découverte de ces services. Cette approche est basée sur l'utilisation d'une ontologie de l'e-learning et exploite les relations sémantiques entre concepts et connaissances du domaine à enseigner pour permettre de retrouver les plates formes d'e-learning disponibles et l'accès aux ressources pédagogiques dès lors que celles-ci sont décrites par des formalismes appropriés (WSDL, OWL-S ou tout autre langage standardisé).

**Mots clés.** e-Learning – Ressource pédagogique – Interopérabilité – Ontologie – Services Web – Web sémantique

**Abstract.** One of the problems that lands in e-learning is the availability of educational resources. The development of these resources is progressively possible but doesn't permit to answer the needs that are often very important. A variety of educational resources is available on the net; among these resources those that are presented as web services. In this paper is proposed an infrastructure for the discovery of these services. This approach is based on the use of ontology of e-learning and exploits the semantic relations between concepts and knowledge of the domain to be taught to permit to retrieve available e-learning platforms and the access to educational resources from the moment these are described by suitable formalisms (WSDL, OWL-S or all other standardized language).

**Keywords.** e-Learning – Educational Resource – Interoperability – Ontology – Web Services – Semantic Web

**ملخص.** في إطار التعليم عن بعد، من أهم المشاكل التي تطرح هو توفر الموارد التعليمية .تطوير هذه الموارد ممكن ولكن لا يلبي الاحتياجات التي غالبا ما تكون مهمة جدا و متنوعة . مجموعة متنوعة من الموارد التعليمية متوفرة على الشبكة الإنترنت ،و من بين هذه الموارد تلك التي قدمت في شكل خدمات ويب.نقترح في هذه الورقة البنية التحتية لاكتشاف هذه الخدمات. ويستند هذا النهج على استخدام بيانات معرفية من التعلم الإلكتروني ويستغل علاقات دلالية بين المفاهيم والمعارف في مجال التدريس للمساعدة في العثور على الموارد التعليمية و أنظمة التعليم الإلكتروني والوصول إلى الموارد المقترحة و المتاحة لتدريس عندما وصفت من قبل الشكليات المناسبة و المكتوبة في اللهجات المختلفة و المتنوعة.

**كلمات.** التعليم الإلكتروني - الموارد التعليمية - التوافقية - علم الوجود - خدمات الويب - الويب الدلالي

# *Table des Matières*

<b>Introduction générale</b> .....	1
<b>Chapitre 1 :L' e-learning</b>	
1 Introduction .....	5
2 Précurseurs de l'e-learning.....	5
2.1 L'enseignement assisté par ordinateur (EAO).....	5
2.2 Les didacticiels ou logiciels d'EAO.....	6
2.3 L'enseignement intelligemment assisté par ordinateur (EIAO).....	7
2.3.1 Les objectifs de l'EIAO .....	7
2.3.2 L'architecture d'un système d'EIAO.....	7
3 L' e-learning .....	9
4 Présentation de l'e-learning.....	10
5 Les LMS (Learning Management System) .....	12
6 Les LCMS (Learning Content Management System).....	12
7 Les acteurs d'une plate forme de l'e-learning.....	13
8 Les ressources pédagogiques.....	14
8.1 Définition d'une ressource pédagogique.....	14
8.2 Caractéristiques d'une ressource pédagogique .....	15
8.3 Description des ressources pédagogiques.....	15
8.3.1 Dublin Core (DC) .....	16
8.3.1.1 Les éléments du Dublin Core .....	16
8.3.1.2 Objectifs du Dublin Core.....	19
8.3.2 Learning Object Metadata (LOM).....	19
8.3.2.1 Les éléments du LOM .....	19
8.3.3 Sharable Content Object Reference Model (SCORM).....	22
8.3.3.1 Les spécifications du modèle SCORM.....	23
9 Conclusion .....	24
<b>Chapitre 2 : Les services web</b>	
1 Les services web.....	28
1.1 Définition.....	28
1.2 Les principaux acteurs.....	28
1.3 Les principales technologies des web services .....	29

1.3.1 SOAP (Simple Object Access Protocol) .....	29
1.3.2.1 Framework de messagerie SOAP .....	29
1.3.2.2 Ensemble de règles de codage.....	30
1.3.1.3 Fonctionnement en modèle client/ serveur (RPC).....	30
1.3.2 WSDL (Web Services Description Language).....	31
1.3.3 UDDI (Universal Description, Discovery and Integration).....	32
1.3.3.1 Langage de description .....	32
1.3.3.2 Mécanisme d'Interrogation.....	33
1.3.4 Vue globale des principales technologies.....	34
1.4 « learnServe » :plate forme e-learning orientée services .....	35
2 Les services web sémantique .....	37
2.1 Le web sémantique .....	38
2.2 Les langages du web sémantique.....	38
2.2.1 RDF (Resource Description Framework).....	39
2.2.2 OWL (Ontology Web Langage).....	39
2.3 Les ontologies .....	40
2.3.1 Composantes d'une ontologie.....	40
2.3.2 Types d'ontologie.....	41
3 La découverte de services web.....	43
3.1 Approches syntaxiques.....	43
3.1.1 Approche UDDI .....	43
3.1.2 Approche basée sur la qualité de service.....	44
3.1.3 Approche AASDU.....	45
3.2 Approches sémantiques.....	46
3.2.1 Ontologie des Services Web OWL-S.....	46
3.2.2 Approche basée sur les protocoles de conversation.....	47
4 Conclusion.....	49
<b>Chapitre 3 : La conception</b>	
1 Problématique .....	51
2 Approche pour la découverte des services web.....	52
2.1 Ontologie des Services Web OWL-S.....	52
3 La description ontologique de l'approche de découverte de services.....	54
3.1 Ontologie Associée à la pédagogie liée aux ressources pédagogiques.....	54
3.2 Ontologie du Domaine à Enseigner.....	57

3.2.1 Diagramme de classes de l'ontologie du domaine à enseigner .....	60
4 Les acteurs intervenants du système.....	60
5 Diagramme de cas d'utilisation principal.....	62
6 Description des Fonctionnalités du Système.....	62
6.1 Architecture de Publication de Services Web.....	63
6.2 Diagramme de séquence du module de publication de services web.....	64
6.3 Architecture de Découverte de Services Web.....	65
6.4 Diagramme de séquence du module de découverte de services web.....	66
7 La procédure de matching .....	67
8 Conclusion.....	72

## **Chapitre 4 : La réalisation**

1 Introduction .....	74
2 Technologies d'implémentation.....	74
2.1 Java .....	74
2.2 JSP et Servlet.....	75
2.3 Serveur Apache Tomcat.....	75
2.4 L'éditeur Protégé.....	76
2.5 API Jena.....	76
2.6 Registre UDDI.....	77
3 Architecture fonctionnelle du système.....	77
4 Développement des ontologies.....	79
4.1 Ontologie du domaine à enseigner.....	79
4.2 Ontologie des services web .....	79
5 Les interfaces du système.....	80
5.1 Interface de publication .....	81
5.2 Interface de découverte .....	83
6 Conclusion.....	84

<b>Conclusion générale.....</b>	<b>85</b>
---------------------------------	-----------

## **Références**

### **Annexe A : Ontologie associée à la pédagogie**

### **Annexe B : Ontologie du domaine à enseigner**

# Table des Figures

## Chapitre 1 :L' e-learning

Figure 1.1: composantes d'un système d'EIAO.....	8
Figure 1.2 : Système d'e-learning : répartition des rôles LMS/LCMS.....	11

## Chapitre 2 : Les services web

Figure 2.1 : Architecture des web services.....	29
Figure 2.2: Invocation d'objets distants avec SOAP.....	31
Figure 2.3 : L'annuaire UDD.....	34
Figure 2.4 : Vue globale : les technologies de base.....	35
Figure 2.5 : Architecture orientée services d'un système e-learning.....	36
Figure 2.6 : Les couches du web sémantique.....	38
Figure 2.7 : Exemple de graphe RDF.....	39
Figure 2.8 : Un modèle pour l'enregistrement et l'invocation de web services.....	45
Figure 2.9 : Approche basée sur les protocoles de conversation.....	49

## Chapitre 3 : La conception

Figure 3.1: Les classes principales de l'ontologie OWL-S .....	53
Figure 3.2 : L'ontologie OWL-S+.....	55
Figure 3.3 : Ontologie associée à la pédagogie.....	57
Figure 3.4 : Découpage des connaissances du domaine d'enseignement.....	58
Figure 3.5: L'ontologie du domaine à enseigner.....	59
Figure 3.6 : Diagramme de l'ontologie du domaine à enseigner.....	60
Figure 3.7 : Diagramme de contexte.....	61
Figure 3.8 : Diagramme de cas d'utilisation principal.....	62
Figure 3.9: Architecture du système proposé.....	63
Figure 3.10 : Diagramme de séquence de publication des services web.....	64
Figure 3.11 : Diagramme de séquence de découverte de services web.....	66
Figure 3.12: Les étapes du filtrage .....	68

## Chapitre 4 : La réalisation

Figure 4.1 : Architecture fonctionnelle du Système.....	78
Figure 4.2 : Ontologie du domaine à enseigner.....	79
Figure 4.3 : Ontologie des services web.....	80
Figure 4.4 : Interface principale du système.....	81



Figure 4.5 : Interface de publication des services web.....	82
Figure 4.6 : Interface de découverte de service web.....	84

## *Liste des Tableaux*

### **Chapitre 1 :L' e-learning**

Tableau 1.1 : tableau récapitulatif des éléments du Dublin Core.....	18
--	----

Tableau 1.2 : tableau récapitulatif des éléments du LOM.....	21
--	----

### **Chapitre 2 : Les services web**

Tableau 2.1 : tableau résumant les six éléments principaux d'un document WSDL.....	32
--	----

## Introduction générale

Dans le cadre de l'enseignement à distance, un des problèmes qui se pose est celui de la disponibilité des ressources pédagogiques et leurs échanges entre différents LMS (Learning Management System). Les ressources pédagogiques constituent ici l'ensemble des informations, documents, logiciels, programmes, banques de données, etc., et qui permettent de véhiculer, de transmettre ou d'appréhender des concepts et contenus d'enseignement.

L'élaboration de ces ressources au fur et à mesure par les auteurs concepteurs de cours au sein des plates formes d'e-learning est possible mais ne permet pas de répondre aux besoins des apprenants et enseignants qui sont souvent très importants, et Avec l'ouverture offerte par les plates formes actuelles d'apprentissage sur le web qui contient un volume considérable de ressources pédagogiques le développement des spécifications descriptives de ces ressources pour faciliter leurs localisation et réutilisation est devenu une nécessité.

Des initiatives ont été enregistrées comme le LOM (Learning Object Metadata), DC (Dublin Core) et le standard le plus adapté SCORM (Sharable Content Object Reference Model)... La diversité des standards pour la description des ressources pédagogiques a rendu la situation et la réutilisation de celles-ci insatisfaisantes par manque de compatibilité entre les spécifications adoptées par les différentes plates formes en plus de la cherté et la difficulté des corrections et mises à jour de fait qu'une simple correction ou mise à jour conduit à une rediffusion de la ressource sur les systèmes qui l'utilisent. Plusieurs solutions ont été proposées parmi elles on trouve : celles qui convergent vers l'utilisation d'un même standard qui ne résolvent que partiellement les problèmes cités, et celles qui proposent la conception des contenus pédagogiques (objets pédagogiques) sous forme de services web utilisables par les acteurs du système et des utilisateurs distants via un simple appel de service.

L'intérêt de développer une infrastructure pour la découverte de services liés aux ressources pédagogiques est manifeste. Celle-ci permettra, grâce aux moteurs de services publics, de retrouver ces ressources numériques. Elle permettra également de retrouver les plates formes d'e-learning disponibles dès lors que celles-ci sont décrites par des formalismes appropriés (WSDL, OWL-S ou tout autre langage standardisé). En outre, les ressources décrites sous forme de services web peuvent être intégrées dans n'importe quel système d'e-learning orienté services.

## **Organisation du document**

Ce papier s'intéresse au développement d'une infrastructure pour la découverte de services web liés aux ressources pédagogiques. Il est constitué des chapitres suivants :

- **Chapitre 1** : dans ce chapitre est traité le domaine de l'e-learning en présentant ses caractéristiques et particularités ainsi que les systèmes d'annotations des ressources pédagogiques existants.
- **Chapitre 2** : présente un état de l'art, qui a pour but de synthétiser les technologies des web services et le web sémantique, l'apport de ces deux technologies au domaine de l'e-learning et quelques stratégies de découverte de services.
- **Chapitre 3** : dans ce chapitre est présentée la modélisation et la conception de l'approche proposée pour la découverte des services web liés aux ressources pédagogiques en donnant les descriptions des deux ontologies et l'algorithme de matching proposé.
- **Chapitre 4** : Il comprend la mise en oeuvre du système de découverte des web services liés aux ressources pédagogiques et les outils de l'implémentation.

A la fin de ce papier nous trouverons une conclusion générale et deux annexes :

- **Annexe A** : Ontologie associée à la pédagogie.
- **Annexe B** : Ontologie du domaine à enseigner.

# Chapitre 1 :

## *L' e-learning*

### Sommaire

1	Introduction .....	5
2	Précurseurs de l'e-learning.....	5
2.1	L'enseignement assisté par ordinateur (EAO).....	5
2.2	Les didacticiels ou logiciels d'EAO.....	6
2.3	L'enseignement intelligemment assisté par ordinateur (EIAO).....	7
2.3.1	Les objectifs de l'EIAO .....	7
2.3.2	L'architecture d'un système d'EIAO.....	7
3	L' e-learning .....	9
4	Présentation de l'e-learning.....	10
5	Les LMS (Learning Management System) .....	12
6	Les LCMS (Learning Content Management System).....	12
7	Les acteurs d'une plate forme de l'e-learning.....	13
8	Les ressources pédagogiques.....	14
8.1	Définition d'une ressource pédagogique.....	14
8.2	Caractéristiques d'une ressource pédagogique .....	15
8.3	Description des ressources pédagogiques.....	15
8.3.1	Dublin Core (DC) .....	16
8.3.1.1	Les éléments du Dublin Core .....	16
8.3.1.2	Objectifs du Dublin Core.....	19
8.3.2	Learning Object Metadata (LOM).....	19
8.3.2.1	Les éléments du LOM .....	19
8.3.3	Sharable Content Object Reference Model (SCORM).....	22
8.3.3.1	Les spécifications du modèle SCORM.....	23
9	Conclusion .....	24

## *Table des Figures*

Figure 1.1: composantes d'un système d'EIAO.....	8
Figure 1.2 : Système d'e-learning : répartition des rôles LMS/LCMS.....	11

## *Liste des Tableaux*

Tableau 1.1 : tableau récapitulatif des éléments du Dublin Core.....	18
Tableau 1.2 : tableau récapitulatif des éléments du LOM.....	21

## **1 Introduction**

L'émergence des Technologies de l'Information et de la Communication (TIC), l'apparition des grands réseaux d'information (Internet, Intranet) et l'intégration des techniques d'intelligence artificielle (représentation des connaissances) ont permis à l'enseignement assisté par ordinateur (EAO) de quitter les centres informatiques et aux apprenants de se connecter au réseau, d'être plus actifs dans leurs processus de formation et participer à des situations d'apprentissage collectif, comme ils étaient à l'origine de l'enrichissement du contenu par des liens hypertexte et multimédia, ce qui a donné naissance à l'e-learning (e-formation) caractérisé par un échange pédagogique important.

Au début, les environnements d'e-apprentissage ont été des systèmes centralisés basés sur l'architecture client/serveur, mais le problème d'interopérabilité entre applications et de réutilisation du contenu et des fonctionnalités ont amené les concepteurs vers les solutions distribuées basées sur des architectures orientées services.

Dans ce chapitre est traité le domaine de l'e-learning en présentant ses caractéristiques et ses particularités.

## **2 Précurseurs de l'e-learning**

Le domaine de la formation a connu une évolution à travers les années précédentes.

Au début

### **2.1 L'enseignement assisté par ordinateur (EAO)**

L'EAO (Enseignement Assisté Par Ordinateur) est un terme qui désigne l'utilisation de matériels informatiques et des logiciels pour assurer tout ou une des parties du processus d'enseignement ou de formation. Malgré, les ambitions initiales de ces systèmes à fournir un enseignement de qualité, ceux-ci avaient des limites, et négligent certains aspects importants du processus d'apprentissage tels que le comportement pédagogique de l'apprenant, le test de ses réponses, etc. En effet, ils ne font souvent, que des comparaisons plus ou moins fines de chaînes de caractères et les erreurs non prévues sont généralement renvoyées à un traitement unique qui n'est pas toujours le plus approprié.

Les logiciels issus de l'EAO sont appelés didacticiels.

## 2.2 Les didacticiels ou logiciels d'EAO

Un didacticiel est un logiciel d'enseignement assisté par ordinateur, spécialisé dans une ou plusieurs matières, mettant un apprenant en interaction avec la matière enseignée dans le but de lui faire acquérir des connaissances. Il existe plusieurs sortes de didacticiels [16] :

- Didacticiel de test (questionnaire).
- Didacticiel informatif.
- Didacticiel d'entraînement (ou drill) : qui combine les deux types précédents.
- Didacticiel de simulation.
- Le tutorial : qui utilise toutes les formes précédentes.
- Le didacticiel intelligent ou pédagogique : possédant en plus les caractéristiques suivantes :
  - Une adéquation pédagogique qui prime tout raffinement concernant les canaux de communication (tels que la qualité des graphismes, de la mise en page, du son...).
  - Les questions « ouvertes » le sont réellement (il n'y a pas de QCM caché) et le traitement de réponses en langue naturelle se fait intelligemment.
  - Les modules d'aides sont appuyés par des filtres logiciels pédagogiques accédant à une encyclopédie thématique.

Malgré l'essor qu'a connu l'EAO dans plusieurs domaines d'apprentissage, il représente néanmoins des insuffisances qui sont :

- **La non prise en compte de l'individualité de l'apprenant** (son niveau, sa psychologie).
- **La non prise en compte de la progression individuelle de l'apprenant** : en effet, le contrôle de l'apprentissage se fait par des questions figées (quel que soit l'apprenant), et la réponse est utilisée pour passer à la question suivante.

Selon Labat [24], ces systèmes souffrent de deux inconvénients majeurs :

- Les connaissances nécessaires à la résolution des problèmes ne sont pas représentées dans la machine sous une forme utilisable par le système (structure figée de la base de connaissances). En conséquence les réponses des exercices doivent être préenregistrées et l'apprenant ne peut pas poser ses propres exercices.
- L'adaptation à l'apprenant est généralement inexistante une seule stratégie pédagogique, qui de plus est implicite, y est mise en œuvre, les exercices, les explications,

les corrections et les leçons sont, le plus souvent les mêmes pour tous (le parcours du didacticiel est fixe).

L'intégration des techniques d'intelligence artificielle dans l'EAO a permis à ce dernier de franchir ses limites et a donné naissance à l'EIAO (enseignement intelligemment assisté par ordinateur).

### **2.3 L'enseignement intelligemment assisté par ordinateur (EIAO)**

L'intégration des techniques de l'intelligence artificielle dans l'EAO a donné naissance à l'EIAO (enseignement intelligemment assisté par ordinateur). Cette nouvelle discipline a été définie par J.F.Nicaud et M.Vivet [30] comme étant une science interdisciplinaire faisant appel aux domaines suivants :

Pédagogie générale, didactique des disciplines, psychologie cognitive, ergonomie et informatique.

#### **2.3.1 Les objectifs de l'EIAO**

L'objectif de l'EIAO est la production facile et souple de logiciels éducatifs capables de remplacer l'enseignement humain c'est-à-dire :

- Qui maîtrise le domaine enseigné, donc peuvent proposer et résoudre des problèmes de complexité variable, et peuvent suivre et comprendre la démarche de résolution de l'apprenant pour détecter éventuellement ses erreurs et les lui expliquer.
- Qui disposent de stratégies tutorielles (méthodes d'enseignement) explicites non enfouies dans les connaissances du domaine enseigné.
- Qui sont applicables à plusieurs domaines, donc utilisant des principes pédagogiques généraux.
- Qui sont capables d'évaluer les performances de l'apprenant pour pouvoir suivre sa progression.
- Qui permettent une adaptation dynamique et individualisée du système à son interlocuteur.

#### **2.3.2 L'architecture d'un système d'EIAO**

Les systèmes d'EIAO présentent les caractéristiques suivantes :



- Une représentation explicite des connaissances du domaine et des mécanismes de raisonnement qui dotent ces systèmes de la capacité de répondre à des questions, de résoudre des exercices dans la solution n'a pas été explicitement prévue et dont l'énoncé peut être proposé par l'apprenant.
- Un processus de « modélisation de l'apprenant », visant à disposer explicitement d'informations telles que son degré de maîtrise des connaissances du domaine.
- Un processus de « modélisation de l'apprenant », vise à disposer explicitement d'informations telles que son degré de maîtrise des connaissances du domaine.
- L'explication de stratégies tutorielles, pour permettre au système d'engendrer dynamiquement ses interventions en fonction de la situation d'apprentissage, d'objectifs pédagogiques et du modèle de l'apprenant.
- La recherche de capacité de communication souple et variée, avec des possibilités d'intervention et de prise d'initiative de l'apprenant.

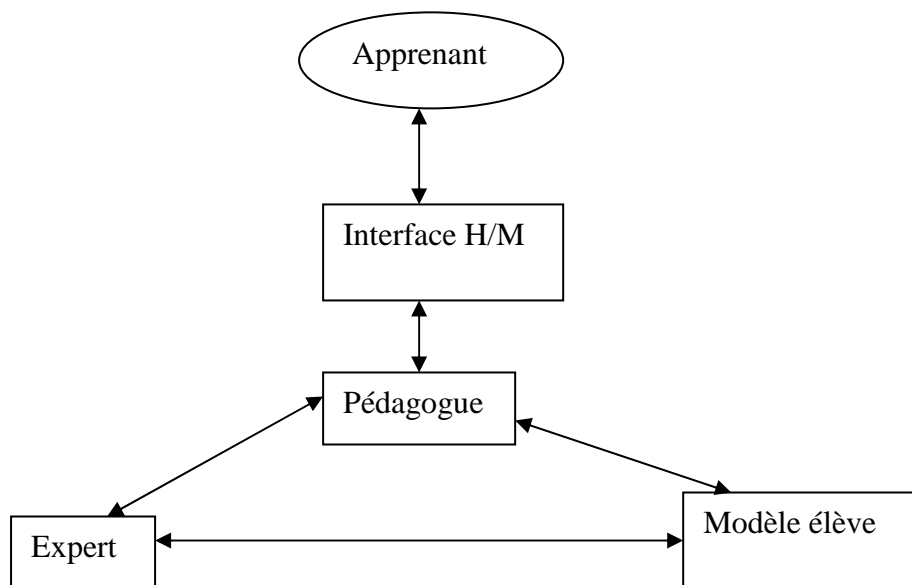


Figure 1.1 : composantes d'un système d'EIAO

L'architecture générale d'un système d'EIAO comporte donc les composantes suivantes : un module expert, un pédagogue, un modèle élève et une interface.

*-le module expert* : Il contient les structures de connaissances du domaine enseigné ainsi que les méthodes permettant de rechercher les informations dans ces structures.

L'expert a la charge de quatre tâches principales [5] :

1. la généralisation de problèmes
2. la résolution de problèmes
3. le test des réponses de l'élève.
4. et l'explication portant sur : les erreurs, les énoncés d'exercices et la résolution de problèmes.

*-Le module pédagogique :* Il est constitué d'un ensemble de spécifications sur la manière dont le système doit construire ses interventions. Ce module interagit avec l'élève plus ou moins directement en sélectionnant les problèmes qu'il doit résoudre en le guidant vers la solution, en critiquant ses performances, en fournissant une aide appropriée lorsque l'élève le lui demande, en montrant des exemples.

*-Le modèle élève :* Le modèle élève est constitué d'un profil de compétence de l'élève pour les divers éléments de connaissance.

Le modèle de l'élève regroupe les connaissances liées au domaine propre à chaque apprenant, les connaissances indépendantes du domaine enseigné telles que : les mécanismes d'apprentissage préférés par l'apprenant.

### **3 L'e-learning**

L'e-learning est la diffusion d'une formation à travers un réseau (Internet, Intranet,...). Toute formation quel que soit le domaine enseigné repose essentiellement sur les acteurs intervenants (apprenants, formateurs, auteurs, etc.), le domaine d'enseignement sous-jacent et les ressources pédagogiques utilisées pour l'apprentissage [3]. Selon Bouthry [9] les plates formes d'e-learning sont avant tout conçues comme des outils pour la gestion des cours par correspondance électronique. Elles offrent plusieurs possibilités comme l'autoformation (les contenus des cours sont disponibles en ligne, en libre service, à tout moment) et l'accompagnement en ligne qui personnalise la formation à l'occasion du tutorat synchrone (le formateur et l'apprenant échangent en temps réel leurs points de vue).

En e-learning, les fonctionnalités principales recherchées sont : une distribution plus large et plus efficace de la formation, l'individualisation et l'adaptation du parcours de formation aux besoins individuels des apprenants, les possibilités de suivi et de tutorat, et enfin la démultiplication de l'offre grâce à la mise en commun de ressources et de modules de formation.

Donc l'e-learning peut être défini de plusieurs manières.

1. l'e-learning est l'utilisation d'Internet dans la gestion et la diffusion de la formation. il comprend l'ingénierie pédagogique, les contenus de formation et une plate forme technologique. Il permet aussi une formation « juste à temps» grâce à la gestion des connaissances, et une gestion administrative via l'e-learning management system (LMS).
2. l'e-learning est un dispositif de formation utilisant Internet ou des intranets comme vecteurs de diffusion des formations. En ce sens, l'e-learning est une forme d'apprentissage à distance.
3. l'e-learning est une formation ouverte, flexible, individualisée, à distance et en réseau, alliant travail en autoformation et accompagnement, communautés d'apprentissage sur le lieu de travail ou chez soi, avec des objectifs de formation identifiés et mesurables.

#### **4 Présentation de l'e-learning**

L'e-learning peut se présenter sous trois formules qui peuvent être complémentaires:

- *autoformation avec contenus en ligne avec tutorat asynchrone* : l'apprenant gère son emploi du temps, suit sa formation de façon autonome, et passe ses tests via un site Internet ou un intranet. Il communique à des moments définis avec son tuteur,
- *formation individuelle en ligne avec tutorat asynchrone* : chaque apprenant a accès à ses besoins de formation, définis à l'avance par des tests. Le tuteur communique avec chaque apprenant pour répondre à ses questions et commenter ses résultats,

On remarque que ces deux formules permettent : 1) une flexibilité en terme de temps et lieu. 2) un apprentissage adapté au profil apprenant et 3) un accès illimité au support du cours, mais en présentant des contenus peu interactifs et en renforçant le sentiment d'isolement.

- *Classes virtuelles en ligne avec tutorat synchrone* : le tuteur intervient en même temps auprès des élèves, il fixe des horaires pour réunir les apprenants autour d'un forum en ligne. Il propose des travaux dirigés et répond aux questions tout en utilisant des outils de collaborations comme le chat, forum, etc. mais cette technique présente un inconvénient qui

est le fait que les apprenants partagent le même tuteur, en plus des contraintes techniques de mise en œuvre.

D'une façon générale, une plate forme de l'e-learning est un logiciel qui comprend un ensemble d'outils spécialisés (LMS : Learning Management System et LCMS : Learning Content Management System) dont les fonctions principales permettent de : produire puis intégrer, gérer et administrer des contenus de formation et des ressources pédagogiques, diffuser et permettre l'accès à ces contenus et ressources, présenter des programmes de formation, permettre un positionnement puis un suivi d'apprenants, construire des parcours de formation individualisés, gérer ces parcours individualisés ou des parcours communs, encadrer des individus et animer des groupes, administrer financièrement ou du point de vue technique.

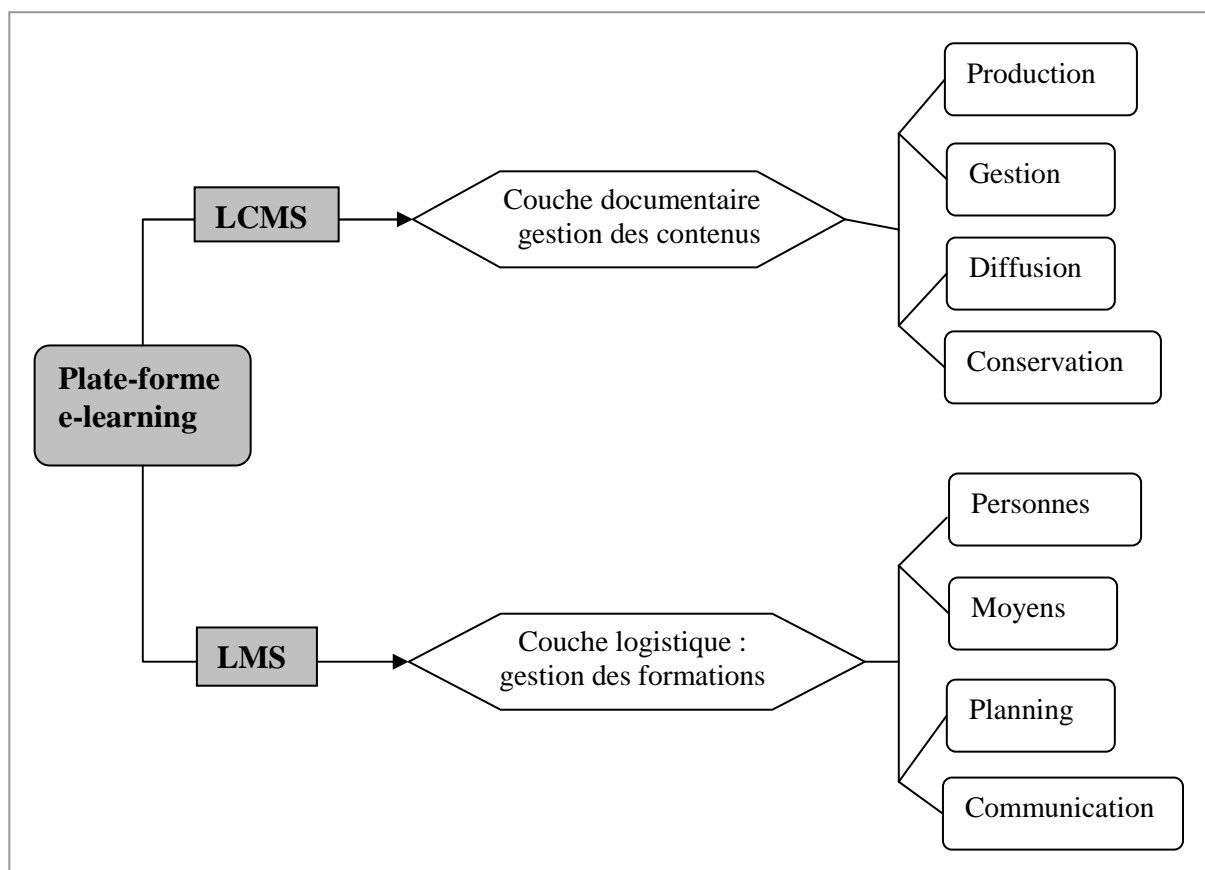


Figure 1.2 : Système d'e-learning : répartition des rôles LMS/LCMS [12]

## **5 Les LMS (Learning Management System)**

Une plate forme LMS désigne une application logicielle utilisée comme outil d'aide à la gestion du déroulement des formations ainsi que leur administration, la gestion et le suivi des apprenants pour fournir des apprentissages individualisés [13].

En principe un LMS fournit les fonctionnalités suivantes :

- l'administration et l'identification des utilisateurs de la plate forme (formateurs, concepteurs...), contrôle et la gestion des inscriptions des apprenants aux formations offertes ainsi que leurs affectation aux différents groupes en prenant en compte leurs profils, leurs pré-requis et leurs rythmes d'avancement.
- le suivi et l'évaluation des apprenants par le tuteur en fonction des résultats des tests, la durée de la résolution des problèmes et le temps globale consacrés à la formation en utilisant les outils de « reporting ».
- intégration d'outils de communication permettant une interactivité entre les apprenants et entre apprenants et le tuteur de la formation. Ces outils peuvent être des outils synchrones (La messagerie, Le forum, Le Chat, Les documents pédagogiques...) ou asynchrones (tableau blanc, audio/vidéo conférence...)
- la conception des cours en assemblant les objets pédagogiques créés par les formateurs et les créateurs concepteurs de cours et la mise à disposition un ensemble d'outils qui facilitent la recherche des ressources pédagogiques.

Les LMS n'offrent pas des outils complets pour la gestion et la création des contenus d'apprentissage, pour cela ils font appel aux LCMS (Learning Content Management Systems).

## **6 Les LCMS (Learning Content Management System)**

Les LCMS ont été développés réellement à partir de 2001. Les fonctions principales d'un LCMS sont la création, le stockage et la diffusion personnalisée des contenus d'apprentissage pour faciliter la réutilisation des objets pédagogiques et minimiser la durée de leurs création ce qui réduit le coût du développement.

Les acteurs d'un LCMS sont les auteurs concepteurs d'objets pédagogiques et les tuteurs utilisateurs de ces objets dans leurs cours. Chaque LCMS est caractérisé par :

- Un ensemble d'outils collaboratifs (créateurs) de créations et de diffusion personnalisée de cours et des tests d'évaluation tout en gardant la séparation entre le contenu et la présentation.

- Un dispositif pour le stockage des contenus d'apprentissages associés à des spécifications de description (méta-données) pour faciliter la recherche, ce dispositif peut être une base de données ou un entrepôt de données.

Selon [3] toute formation quel que soit le domaine enseigné repose essentiellement sur les acteurs, le domaine d'enseignement, et les ressources pédagogiques utilisées pour l'apprentissage. L'utilisation des nouvelles technologies et de l'Internet ont permis l'amélioration de la qualité de formation à travers l'accès à distance à des ressources et services ainsi qu'à des collaborations et des échanges, d'où on conclut que l'e-learning est basé sur des technologie fiables, mais il est orienté vers la pédagogie. Une pédagogie qui réside et s'exprime dans les techniques et stratégies d'enseignement adaptées, et l'intégration et la gestion des contenus de formation et des ressources pédagogiques.

De nombreuses définitions de ressources pédagogiques ont été proposées. Ces définitions ont été enrichies avec le développement des technologies du e-learning et plusieurs standards et normes ont été développés pour la description de ces ressources comme le DublinCore, LOM et SCORM.

### **7 Les acteurs d'une plate forme de l'e-learning**

Plusieurs acteurs interviennent dans le déroulement du processus d'apprentissage offert par les plates formes d'apprentissage.

**Apprenant** : il transforme les informations en connaissances, il exploite les ressources pédagogiques mises à disposition comme il communique et échange des informations avec les autres apprenant et formateur.

**Présentateur** : son rôle est de rendre les informations disponible sous une forme présentable avec un contenu clair et d'analyser et évaluer le contenu des documents.

**Concepteur** : c'est lui qui développe les systèmes d'apprentissage et mit en place les différentes fonctionnalités.

Il modélise les connaissances à des fins pédagogiques et construit les scénarios pédagogiques.

**Formateur** : il facilite l'apprentissage sur le plan pédagogique en réalisant les diagnostics de l'apprentissage, il oriente et assiste les apprenants et évalue leurs travaux.

**Gestionnaire (administrateur)** : il gère les acteurs, les groupes et les évènements, il planifie le déroulement des activités et dirige les opérations de diffusion des contenus et des ressources et administre les réseaux de communication.

## 8 Les ressources pédagogiques

Parmi les fonctions principales d'une plate forme e-learning : la production puis l'intégration, la gestion et l'administration des contenus de formation et des ressources pédagogiques, diffusion et le contrôle de l'accès à ces contenus et ressources.

### 8.1 Définition d'une ressource pédagogique

De nombreuses définitions ont été proposées ce qui rend la notion de ressource pédagogique floue :

Selon Gérard Puimatto [19]

« Les ressources pédagogiques constituent l'ensemble des informations, documents, logiciels, programmes, banques de données, et qui permettent de véhiculer, de transmettre ou d'appréhender des concepts et contenus d'enseignements »

et d'après Robert Bibeau [42] « une ressource d'enseignement et d'apprentissage (REA) est toute entité numérique ou non susceptible d'être utilisée, réutilisée ou évoquée dans un contexte d'apprentissage à support technologique. Le contenu numérique, les applications et les didacticiels utilisés dans un contexte d'apprentissage à support technologique. »

En apprentissage automatique, la forme fondamentale d'une ressource pédagogique est l'« objet pédagogique » (OP).

La définition donnée par le standard d'annotation LOM est : « Toute entité (numérique ou non) utilisée dans un processus d'enseignement, de formation ou d'apprentissage. »

Le groupe de travail des IEEE-LTSC (Learning Technology Standards Committee) a proposé une définition d'un OP qui est la suivante : «any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning».

Donc un objet pédagogique est défini comme toute entité numérique ou non qui peut être utilisée, réutilisée ou référencée pendant des activités d'apprentissage assistées par ordinateur.

D'après ces définitions on remarque que la définition des objets pédagogiques englobe celle des ressources pédagogiques. En plus de la composante « contenu », un objet pédagogique peut avoir une composante processus comme le cas des didacticiels. Chaque objet pédagogique doit satisfaire un ensemble prédéfini de caractéristiques et propriétés.

## 8.1 Caractéristiques d'une ressource pédagogique

Selon [19] les caractéristiques principales d'un objet ou d'une ressource pédagogiques sont les suivantes :

- Indépendance (autonomie)**: l'objet pédagogique ne doit pas s'appuyer sur aucun autre matériel pour avoir du sens.
- Réutilisation** : les objets pédagogiques sont des composants indépendants qui peuvent être utilisés par différentes applications dans des différents contextes à des différentes fins.
- Interopérabilité** : un objet pédagogique doit pouvoir être utilisé dans des environnements technologiques différents. Donc il doit être conçu pour répondre aux différents standards.
- Découverte** : Les objets doivent être décrits par un ensemble d'éléments constituant des métadonnées descriptives pour faciliter leurs repérage et découverte.
- Contexte** : Afin de maximiser leurs réutilisabilités, la quantité d'informations spécifiques à un contexte donnée doit être minimisée.
- Granularité** : Un objet pédagogique peut aussi être une simple image qu'un cours complet.

Selon l'emplacement des ressources pédagogiques on peut distinguer deux types de ressources : celles qui sont internes et celles qui sont externes

**Les ressources externes** : celles qui proviennent du Web, typiquement ce sont des liens vers d'autres sites, des services, des images, vidéos, logiciels outils, plates-formes de formation, moteurs de recherche, applications éducatives etc, qui ne sont que des URLs, et non pas directement stockées sur le serveur.

**Les ressources internes** : elles sont directement stockées sur le serveur de l'application

## 8.3 Description des ressources pédagogique

Un nombre important de ressources est disponible sur le web et pour qu'une ressource pédagogique puisse être accessible facilement, réutilisable et partagée elle doit être décrite par un ensemble d'informations qui facilitent son exploitation. Ces informations doivent être structurées, utilisables, descriptives de la ressource et de son utilisation et non nécessaire pour l'accès à cette ressource, c à d la ressource en question peut être utilisée indépendamment de ces informations.

Dans la littérature, il existe deux termes pour décrire ces informations : méta-données et annotation [42].

**Méta-données** : une méta-donnée « est une donnée sur une donnée ». Elle présente une description formalisée de la ressource ajoutée par son propriétaire, traitable par un système



informatique. Contrairement aux autres données qui ne sont qu'interprétables par un être humain, la méta-donnée permet le traitement automatique et enregistrée séparément de la ressource.

**Annotation :** une annotation est une quelconque marque de lecture critique ou explicative portée sur un document textuel ou image, postée par un indexeur qui n'est pas le propriétaire de la ressource.

### **Modèles de description des ressources pédagogiques :**

L'association de méta-données descriptives et standardisées aux ressources pédagogiques a amélioré le processus de découverte de celle-ci, en permettant une recherche basée sur les champs( titre, auteur, ...). Dans ce but, plusieurs modèles et normes ont été développés pour la description des ressources pédagogiques parmi eux on trouve :

#### **8.3.1 Dublin Core (DC)**

La norme de métadonnées Dublin Core (DC) a été défini par des équipes impliquées dans la sémantique sur le web à Dublin en 1995 pour faciliter la découverte et la gestion des ressources numériques. Le DC est un ensemble d'éléments simples mais efficaces pour décrire une grande variété de ressources en réseau. La norme du Dublin Core comprend 15 éléments dont la sémantique a été établie par un consensus international de professionnels provenant de diverses disciplines telles que la bibliothéconomie, l'informatique, le balisage de textes, la communauté muséologique et d'autres domaines connexes. Chaque élément est facultatif et peut être répété sans limitation.[21]

##### **8.3.1.1 Les éléments du Dublin Core :**

Chaque élément du Dublin Core est défini par un ensemble de dix attributs provenant de la description standard ISO/IEC 11179 dont six sont communs à tous les éléments (Version, Registration Authority, Language, Obligation, Datatype, Maximum Occurrence)

Le tableau suivant résume les 15 éléments en donnant les valeurs des différents attributs :

Elément	Attributs communs	Name	Identifiant	Definition	Comment
Title	<b>*Version:</b> 1.1  <b>*RegistrationAuthority:</b> DublinCore  MetadataInitiative  <b>*Language:</b> Anglais  <b>*Obligation:</b> Optional  <b>*Datatype:</b> Chaîne de caractère  <b>*MaximumOccurrence:</b> non limité	titre	Title	Le nom donné à la ressource	Un titre est le nom par le quel la ressource est officiellement connue
Creator		créateur	Creator	L'entité responsable de la création du contenu de la ressource	Le créateur peut être une personne, une organisation ou un service
Subject		Sujet et mot clefs	Subject	Le sujet du contenu de la ressource	Le sujet est décrit par un ensemble de mots clefs ou de phrases ou un code de classification
Description		description	description	Description du contenu de la ressource	Une description peut contenir : un résumé, table des matières, une référence à une présentation graphique du contenu, un texte libre qui décrit le contenu
Publisher		éditeur	Publisher	L'entité responsable de la publication et la diffusion de la ressource	L'éditeur peut être une personne, une organisation (département universitaire, entreprise) ou un service. Le nom d'une maison d'édition est souhaitable.
Contributor		Contributeur	Contributor	Entité qui a contribué à la création de la ressource	Personne, organisation ou service
Date		Date	Date	date associée avec un évènement dans le cycle de vie de la ressource	La date de la création ou la publication de la ressource. Le format utilisé est AAAA-MM-JJ

## Chapitre 1: L'e-learning

Type		Type de la ressource	Type	La nature ou le genre du contenu de la ressource	
Format		Format	Format	Le caractère physique ou numérique de la ressource	Le format inclut le média, la taille et la durée de la ressource et il précise les logiciels nécessaires pour le fonctionnement de la ressource
Identifiant		Identifiant de la ressource	Identifiant	La référence de la ressource	L'utilisation d'une chaîne de caractères ou un système d'identification est recommandée
Source		Source	Source	La source originale de la ressource	Référencée par une chaîne de caractères ou un système d'identification.
Language		Langue	Language	La langue du contenu de la ressource	Par exemple : 'en' pour l'anglais et 'fr' pour le français.
Relation		Relation	Relation	La référence d'une autre ressource qui est en relation avec la ressource	L'utilisation d'une chaîne de caractères ou un système d'identification est recommandée
Coverage		Couverture	Coverage	La couverture spatio-temporelle de la ressource	Inclut une position géographique et une période de temps
Rights		Gestion des droits	Rights	Information sur les droits	Exemple: droits de propriété intellectuelle , copyright...

Tableau 1.1 : tableau récapitulatif des éléments du Dublin Core

### 8.3.1.2 Objectifs du Dublin Core:

L'ensemble des éléments du DC a été défini dans le but d'atteindre un certain nombre d'objectifs qui sont les suivants :

-Simplicité de création et de gestion des notices descriptives simples pour les ressources web par des utilisateurs non spécialistes d'une façon économique.

- Aider le chercheur non spécialiste à retrouver l'information par l'utilisation d'un ensemble commun d'éléments ayant une sémantique communément comprise et supportée.

-Envergure internationale en développant des versions dans plusieurs langues pour donner à la norme DC la nature multilingue et multiculturelle de l'univers de l'information électronique.

- la possibilité d'étendre l'ensemble des éléments du DC par d'autres éléments pour des besoins d'une découverte précise de ressources. Donc l'assurance d'une extensibilité tout en gardant l'équilibre entre le besoin et la simplicité.

Les 15 éléments du modèle DC offrent une description de base d'une ressource numérique, mais des fois les utilisateurs ont besoin d'une description plus précise qui prend en considération les caractéristiques spécifiques de leurs ressources. Dans cet objectif que Learning Object Metadata (LOM) a été conçu pour la description des ressources pédagogiques.

### 8.3.2 Learning Object Metadata (LOM)

Le LOM est un schéma de méta-dada qui permet la description des principales caractéristiques d'une ressource pédagogique pour faciliter leur découverte, échange et réutilisation donc leur interopérabilité. Il a été publié en 2002 par le Learning Technology Standards Committee (LTSC) de l'IEEE (Institute of Electrical and Electronics Engineers).

#### 8.3.2.1 Les éléments du LOM :

LOM est le schéma de méta-données le plus détaillé qui existe. Il comprend 80 éléments de données regroupés en **neuf catégories** et ils sont tous facultatifs. Les types de données utilisées par le LOM sont : la chaîne de caractères (CharacterString), la chaîne de langue (LangString), le vocabulaire (vocabulary), la date et le temps (DateTime) et la durée (Duration)

Le tableau suivant donne un aperçu général de ce modèle :

<b>Catégorie</b>	<b>description</b>	<b>Eléments</b>
1. General (généralités)	regroupe les caractéristiques générales qui décrivent la ressource dans son ensemble	-Identifiant (Catalogue, Entrée)  -Titre  -Langue  -Description  -Mots-clés  -Couverture  -Structure  -Niveau d'agrégation
2. Life Cycle (cycle de vie)	Cette catégorie décrit l'histoire et l'état actuel de l'objet pédagogique et des entités qui ont eu des répercussions sur l'objet pédagogique lors de son évolution.	-Version  -Statut  -Contribution (Rôle, Entité, Date)
3. MetaMetadata (annotation)	Caractéristiques de la description elle-même : la date de contribution, les différents contributeurs à l'établissement des métadonnées.	-Identifiant  -Contribution (Rôle, Entité, Date)  -Schéma  -langue
4. Technical (technique)	définit les exigences techniques en terme de navigateur, de système d'exploitation, et les caractéristiques de l'objet pédagogique	-Format  -Taille  -Localisation  -Exigence (Type, Nom, Version minimale, Version maximale)  -Remarque d'installation  -Autres exigences  -Durée

<p>5. Educational (pédagogique)</p>	<p>Constituée de treize sous-catégories pédagogiques dans le but d'améliorer l'exploitation du contenu pédagogique.</p>	<ul style="list-style-type: none"> <li>-Type d'interactivité</li> <li>-Type de ressource pédagogique</li> <li>-Niveau d'interactivité</li> <li>-Densité sémantique</li> <li>-Public ciblé</li> <li>-Contexte</li> <li>-Tranche d'âge</li> <li>-Difficulté</li> <li>-Durée d'apprentissage</li> <li>-Description</li> <li>-Langue de l'utilisateur</li> </ul>
<p>6. Rights (droits)</p>	<p>Caractéristiques exprimant les conditions d'utilisation et coût de la ressource pédagogique.</p>	<ul style="list-style-type: none"> <li>-Coût</li> <li>-Copyright</li> <li>-Description</li> </ul>
<p>7. Relation (relation)</p>	<p>Définit les relations de ressources avec d'autres ressources pédagogiques en précisant la nature de la relation (« ...est requis par... », « ...est une partie de... »).</p>	<ul style="list-style-type: none"> <li>-Sorte [Identifiant (Catalogue, Entrée), Description]</li> <li>-Ressource</li> </ul>
<p>8. Annotation (commentaire)</p>	<p>Annotations ou commentaires sur l'utilisation pédagogique de la ressource.</p>	<ul style="list-style-type: none"> <li>-Entité</li> <li>-Date</li> <li>-Description</li> </ul>
<p>9. Classification (classification)</p>	<p>Caractéristiques de la ressource décrites par des entrées dans des systèmes de classification.</p>	<ul style="list-style-type: none"> <li>-Objectif</li> <li>-Chemin taxum [Source (Identifiant, Entrée), Taxum]</li> <li>-Description</li> <li>-Mots-clés</li> </ul>

Tableau 1.2 : tableau récapitulatif des éléments du LOM

Le LOM est utilisé dans plusieurs pays qui ont adapté ce standard selon leurs besoins soit en limitant les éléments du LOM par la sélection d'un sous ensemble d'éléments jugés nécessaires, soit en enrichissant ces éléments par d'autres reliés à un contexte d'utilisation spécifique, ce que on appelle un profil d'application. Aujourd'hui, plusieurs profils d'application existent comme le CANCORE (Canada), le LOM- Normetic(Québec) et le LOMFR (France).

### 8.3.3 Sharable Content Object Reference Model (SCORM)

Le standard SCORM a été publié par Le consortium ADL (Advanced Distributed Learning) dans sa première version en janvier 2000 soutenue par Advanced Distributed Learning du département de la défense américain [2]. Il propose un modèle pour l'empaquetage de contenus, leur structuration et leur navigation. Sharable Content Object Reference Model 2004 (SCORM 2004 : la troisième édition de SCORM publiée en 2006) regroupe un ensemble de spécifications et standards (LOM, IMS-SS...) qui vise les mêmes objectifs de partage, de réutilisabilité et d'interopérabilité des objets pédagogiques. En effet SCORM permet le partage et l'échange de contenus entre différents environnement d'apprentissage multiples (LMS : Learning Management System, LCMS : Learning Content Management System) en offrant des moyens techniques et des méthodes qui permettent l'agrégation des contenus et la création des séquences d'activités de formation.

SCORM possède trois niveaux de granularité de fait qui 'il partage le contenu d'apprentissage en un ensemble d'actifs, d'objets de contenu partageables et des agrégations de contenu.

**Actifs (asset) :** Ils constituent les éléments atomiques d'un contenu d'apprentissage. Il s'agit du niveau concernant principalement les ressources numériques élémentaires (texte, images, vidéo, sons, ...).

**Objet de contenu partageable (SCO : Shareable Content Object) :** est une collection d'actifs. Parmi ces éléments atomiques on trouve un actif spécifique qui utilise l'environnement d'exécution de SCORM pour communiquer avec les systèmes de gestion de l'apprentissage (LMS). Un SCO constitue le niveau le plus fin des ressources d'apprentissage pouvant être suivi et contrôlé par un LMS.

**Agrégations de contenu ( CA : Content Aggregation) :** Ce niveau et celui de plus haut niveau, où des OCP et des Actifs sont assemblés pour former un cours, un chapitre ou un module, et pour organiser diverses activités pédagogiques. Chaque agrégat du contenu est présenté sous la forme d'un fichier XML (imsmanifest.xml) qui décrit les ressources et l'enchaînement pédagogique des activités.

### 8.3.3.1 Les spécifications du modèle SCORM

Le modèle SCORM regroupe un ensemble de spécifications et standards qui sont organisés selon les trois grandes sections suivantes :

**Le modèle d'agrégation du contenu (CAM : Content Aggregation Model) :** il constitue un moyen neutre sur le plan pédagogique et il permet aux concepteurs d'assembler les ressources pédagogiques pour créer un cours ou une séquence d'activités d'apprentissage. Le CAM se compose de quatre structures :

- *Le modèle de contenu SCORM :* il présente les types de composants de contenu SCORM nécessaires à la création d'une activité d'apprentissage. Ces composants sont : les Actifs, les Objets de contenu partageables, les Agrégations de contenu, les activités (unité significative d'instruction utilisée par l'apprenant pour demander à un LMS de lancer un Actif ou un OCP donné) et les organisations de contenu (représentation de l'ordonnancement pédagogique des activités réalisées au moyen des règles de séquence).
- *Le Séquencement et la navigation :* la spécification IMS Simple Sequencing (IMS-SS) utilisée par SCORM permet la personnalisation du parcours d'un apprenant dans le cours et sa navigation, en choisissant l'ensemble des activités et ressources qui seront accessibles par l'apprenant, en fonction de son avancement, de ses résultats et compétences acquises.

Toutes ces informations sont offertes par le IMS-SS qui permet la représentation du parcours pédagogique d'un apprenant sous forme d'une arborescence d'activités.

- *les métadonnées IEEE-LOM :* pour faciliter la recherche et l'accès aux différents composants ressources, SCORM utilise la spécification de méta-données LOM en sélectionnant juste les éléments jugés obligatoires pour obtenir la compatibilité SCORM.
- *Le conditionnement du contenu (packaging) :* dans le but de rendre les contenus d'apprentissage disponibles et accessibles par les apprenants et faciliter leur échange (interopérabilité) entre différents LMSs, SCORM utilise la spécification IMS Content



Packaging (IMS-CP) qui fournit un format bien défini pour importer et exporter des contenus, de façon standardisée.

**L'environnement d'exécution ( RE : Run-time Environment) :** il assure la communication en la ressource (contenu d'apprentissage) constituée d'un contenu pédagogique et un mécanisme de contrôle, et les IMSs pour avoir l'état de manipulation de la ressource par l'apprenant et son avancement.

**Le séquençement et la navigation ( SN : Sequencing and Navigation) :** cette spécification définit les règles à suivre par un IMS pour déduire l'enchaînement d'une activité d'apprentissage à partir de l'état de manipulation de la ressource et de l'avancement de l'apprenant.

Actuellement SCORM est le standard le plus utilisé par les plates formes d'apprentissage (Claroline, Ganesha, Moodle, WebCT...) ce qui traduit son efficacité pour la réutilisation, l'exploitation et l'interopérabilité des contenus d'apprentissage en ligne.

Malgré sa large utilisation, SCORM présente quelques limites :

- n'offre pas une couverture sémantique du contenu d'apprentissage : aucun élément de la catégorie « Educational » du standard LOM n'est sélectionné comme étant élément obligatoire.
- N'adapte pas les activités collaboratives car la spécification IMS-SS est adaptée à l'apprentissage individuel.
- la seule philosophie éducative supportée est la philosophie behavioriste. L'apprentissage consiste à faire des lectures et à passer des tests, le tout dans un processus fortement contrôlé par la machine.

## 9 Conclusion

Il existe sur le web un volume considérable de ressources pédagogiques, les plates formes actuelles d'apprentissage offrent une certaine ouverture sur le web, d'où la nécessité de développement des spécifications descriptives de ces ressources pour faciliter leurs localisation et réutilisation. Des initiatives ont été enregistrées comme le LOM, DC et le standard le plus adopté SCORM... La diversité des standards pour la description des ressources pédagogiques a rendu la situation et la réutilisation de celles-ci insatisfaisantes par manque de compatibilité entre les spécifications adoptées par les différentes plates formes en plus de la cherté et la difficulté des correction et mises à jour de fait qu'une simple correction ou mise à jour conduit à une rediffusion de la ressource sur les systèmes qui l'utilisent.

L'avènement de la technologie des services web a apporté des solutions à ces difficultés en proposant des architectures orientées services (SOA). Ces architectures proposent la conception des contenus pédagogiques (objets pédagogiques) sous forme de services web utilisables par les acteurs du système et des utilisateurs distants via un simple appel de service.

Dans le chapitre qui suit nous aborderons les technologies des services web, puis nous présenterons un exemple d'architecture orientée services e-learning et nous terminerons par l'exposition de la problématique de la découverte de services web.

# Chapitre 2 :

## Les services web

### Sommaire

1 Les services web.....	28
1.1 Définition.....	28
1.2 Les principaux acteurs.....	28
1.3 Les principales technologies des web services .....	29
1.3.1 SOAP (Simple Object Access Protocol) .....	29
1.3.2.1 Framework de messagerie SOAP .....	29
1.3.2.2 Ensemble de règles de codage.....	30
1.3.1.3 Fonctionnement en modèle client/ serveur (RPC).....	30
1.3.2 WSDL (Web Services Description Language).....	31
1.3.3 UDDI (Universal Description, Discovery and Integration).....	32
1.3.3.1 Langage de description .....	32
1.3.3.2 Mécanisme d'Interrogation.....	33
1.3.4 Vue globale des principales technologies.....	34
1.4 « learnServe » :plate forme e-learning orientée services .....	35
2 Les services web sémantique .....	37
2.1 Le web sémantique .....	38
2.2 Les langages du web sémantique.....	38
2.2.1 RDF (Resource Description Framework).....	39
2.2.2 OWL (Ontology Web Langage).....	39
2.3 Les ontologies .....	40
2.3.1 Composantes d'une ontologie.....	40
2.3.2 Types d'ontologie.....	41
3 La découverte de services web.....	43
3.1 Approches syntaxiques.....	43
3.1.1 Approche UDDI .....	43
3.1.2 Approche basée sur la qualité de service.....	44

3.1.3 Approche AASDU.....	45
3.2 Approches sémantiques.....	46
3.2.1 Ontologie des Services Web OWL-S.....	46
3.2.2 Approche basée sur les protocoles de conversation.....	47
4 Conclusion.....	49

## *Table des Figures*

Figure 2.1 : Architecture des web services.....	29
Figure 2.2: Invocation d'objets distants avec SOAP.....	31
Figure 2.3 : L'annuaire UDD.....	34
Figure 2.4 : Vue globale : les technologies de base.....	35
Figure 2.5 : Architecture orientée services d'un système e-learning.....	36
Figure 2.6 : Les couches du web sémantique.....	38
Figure 2.7 : Exemple de graphe RDF.....	39
Figure 2.8 : Un modèle pour l'enregistrement et l'invocation de web services.....	45
Figure2.10 : Approche basée sur les protocoles de conversation.....	49

## *Liste des Tableaux*

Tableau 2.1 : tableau résumant les six éléments principaux d'un document WSDL.....	32
--	----

## 1 Les services web

### 1.1 Définition

*“A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards”* est la définition d’un service web donnée par le consortium W3C [40].

D’une façon générale les services web sont des applications logicielles autos descriptives et modulaires qui fournissent un modèle simple de déploiement d’applications basé sur des normes, identifiables par des URIs (Uniform Resource Identifier). Ils réalisent au travers de l’infrastructure web, des fonctions allant de simples requêtes aux processus métiers très sophistiqués. Ils permettent d’assurer l’interaction entre applications, et d’accéder d’une manière uniforme, à partir d’un site à plusieurs services applicatifs distants. Ce nouveau modèle de déploiement d’applications assure l’interconnexion de services en se basant sur les standards facilitant le transport, l’invocation, la description et la recherche de services web. Les services web permettent également l’échange de données. Cette interopérabilité de données est due à l’utilisation de normes ouvertes (Service Oriented Architecture : SOA). D’après Kadima et al. [22], les services web sont le résultat de la combinaison de http, SOAP (Single Object Access Protocol) un protocole de transmission de messages, WSDL (Web Services Description Language) permettant de décrire un service web et comment l’invoquer, UDDI (Universal Description Discovery and Integration) annuaire pour l’enregistrement des services, ainsi que de composants logiciels.

### 1.2 Les principaux acteurs

Une architecture de référence a été définie pour les web services dans le but d’identifier les composants fonctionnels et définir les relations entre eux. Cette architecture de référence comporte trois acteurs:

**Le fournisseur de service :** c’est le propriétaire du service. D’un point de vue technique, il est constitué par la plateforme d’hébergement du service.

**Le client** : c'est le demandeur de service. Techniquement, il est constitué par l'application qui va rechercher et invoquer un service. Une application cliente peut être elle-même un web service

**L'annuaire des services** : c'est un registre de descriptions de services offrant des facilités de publication de services pour les fournisseurs de services ainsi que des facilités de recherche de services pour les clients.

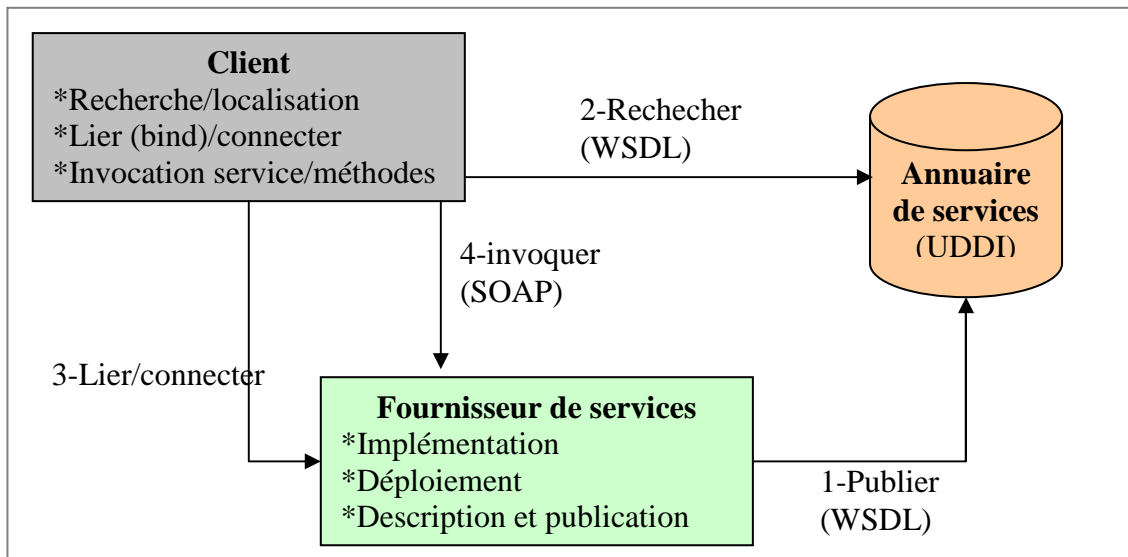


Figure 2.1 : Architecture des web services.[23]

### 1.3 Les principales technologies des web services

Pour permettre aux fournisseurs de définir leurs services et de les publier dans un annuaire de service, et de faciliter aux clients la recherche et la sélection d'un service souhaité puis l'invocation de son fournisseur, des technologies ont été proposées :

**1.3.1 SOAP (Simple Object Access Protocol)** : est un protocole léger bâti sur XML.

Il permet la transmission des informations structurées (message) entre objets distants dans un environnement distribué et décentralisé (SOA). Le transfert se fait à l'aide des protocoles http et SMTP (W3C) [40]

Le protocole SOAP est particulièrement utile pour exécuter des dialogues requête-réponse RPC (Remote Procedure Call). Il définit :

#### 1.3.1.1 Framework de messagerie SOAP :

Un message SOAP est un document XML ordinaire contenant les éléments suivant :

**SOAP Envelope :** est l'élément racine d'un message SOAP. Cet élément identifie le document XML comme un message SOAP

**SOAP Header :** L'élément SOAP Header contient des informations d'en-tête spécifiques (comme l'authentification, paiement, etc)

**SOAP Body :** L'élément SOAP Body comporte le message SOAP lui-même c'est-à-dire des appels et des informations de réponse.

**SOAP Fault :** L'élément SOAP Fault est optionnel. Il est utilisé pour indiquer les messages d'erreur et les informations d'état pour un message SOAP.

### **1.3.1.2 Ensemble de règles de codage :**

Ces règles permettent l'encodage des objets dans des messages SOAP d'une manière standard et leurs décodages au niveau du destinataire.

### **1.3.1.3 Fonctionnement en modèle client/ serveur (RPC) :**

L'un des objectifs principaux du SOAP est l'exécution des dialogues requête/réponse RPC à l'aide de XML qui permet le transport des appels de procédure et leur résultat.

Pour invoquer un objet distant le client crée un document XML contenant les informations nécessaires pour l'invocation. Avant sa transmission sous forme d'une requête HTTP le document doit être inséré dans une enveloppe SOAP. Une fois le serveur a reçu le message, il l'envoie vers l'objet distant, ce dernier traite la requête puis il envoie la réponse au serveur SOAP. Une réponse sous forme d'un message SOAP va être transmise au client puis le résultat au demandeur initial [22].



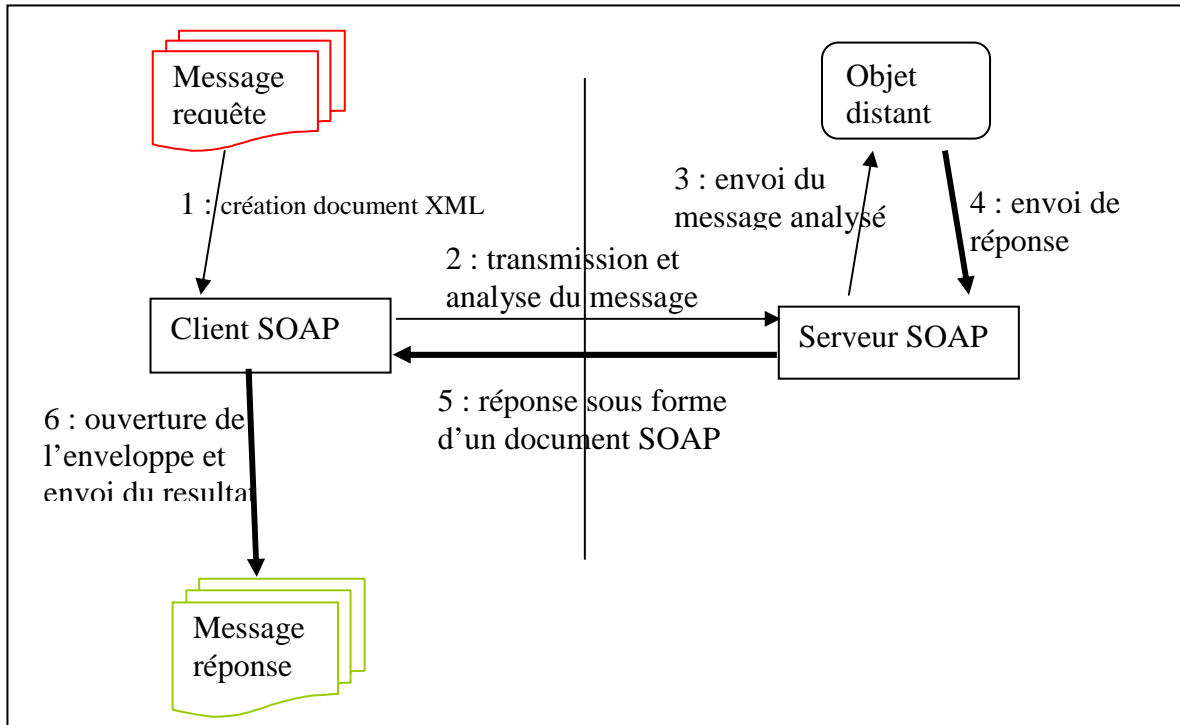


Figure 2.2: Invocation d'objets distants avec SOAP. [22]

### 1.3.2 WSDL (Web Services Description Language)

WSDL est un langage basé sur XML qui permet de décrire un service web et comment l'invoquer. Il est devenu une recommandation du W3C 26 Juin 2007.

WSDL introduit une grammaire commune pour la description des services en précisant les méthodes pouvant être invoquées, le point d'accès (URL, port, etc.), et la description des liaisons qui décrivent comment utiliser WSDL en conjonction avec SOAP, HTTP, et MIME en utilisant ces éléments majeurs [40] [22]:

Élément	Définition
<types>	Les types de données utilisées pour décrire les messages échangés
<message>	La définition abstraite de la donnée en cours de transmission. Un message comporte des parties logiques, chacune étant associée avec une définition dans un système de type.
<portType>	un ensemble d'opérations abstraites. Chaque opération se réfère à un message d'entrée et à des messages de sortie. Les portTypes sont utilisés pour définir les traitements offerts par un web service.

<binding>	Les protocoles de communication utilisés par le service Web, un binding spécifie un protocole réel et les spécifications de format de données pour les opérations et les messages définis par un type de port donné.
<port>	Une adresse pour une liaison définissant un simple point terminal de communication.
<service>	Un ensemble de ports associés.

Tableau 2.1 : tableau résumant les six éléments principaux d'un document WSDL

### 1.3.3 UDDI (Universal Description, Discovery and Integration)

UDDI est un annuaire qui permet l'enregistrement des services web et facilite leur découverte en offrant une structure de données basée sur XML et une API d'intégration.

L'usage de l'annuaire UDDI permet au fournisseur de se présenter et publier ses services pour accélérer leurs échanges par l'intermédiaire d'un opérateur sur le web. Il se comporte lui-même comme un web service dont les méthodes sont appelées via le protocole SOAP.

Le but premier d'un registre UDDI est de fournir une infrastructure de base pour la publication, la découverte et l'invocation de services.

**La publication :** la publication d'un service web consiste à publier sa description WSDL générée manuellement ou automatiquement et de lui créer une entrée dans le registre UDDI.

**La découverte :** consiste à retrouver et rechercher dans l'annuaire UDDI le service qui répond à l'ensemble de caractéristiques souhaitées (nom fournisseur, adresse,..), donc l'acquisition de la description WSDL.

**L'invocation :** le client exploite la description WSDL pour générer des requêtes SOAP et invoquer le service.

#### 1.3.3.1 Langage de description :

Le modèle de donnée de l'annuaire UDDI comporte cinq types principaux de structures de données XML pour aider à la localisation et compréhension rapide des informations qui constituent un enregistrement UDDI [22] [8].

**BusinessEntity** : BusinessEntity est une structure de donnée de haut niveau qui donne les informations sur le fournisseur offrant le service. Les «businessEntities» sont en quelque sorte les pages blanches d'un annuaire UDDI identifiables par des « businesskey »

**ServiceEntity** : ce sont en quelque sorte les pages jaunes d'un annuaire UDDI, qui décrivent de manière non technique les services proposés par les différents fournisseurs. On y trouvera essentiellement le nom et la description textuelle des services ainsi qu'une référence à l'organisation proposant le service et un ou plusieurs «bindingTemplates».

**businessService** : donne les informations descriptives ( nom, description textuelle, classification, ..) sur le service offert comme elle peut contenir plusieurs « bindingTemplate ». Ce sont en quelque sorte les pages jaunes d'un annuaire UDDI

**BindingTemplate**: définit les coordonnées des services, les informations requise pour les invoquer via différents protocoles (HTTP, SMTP,..) et « tModels » associés.

**tModel** : donne les informations techniques permettant de connaître les normes que respecte le service web (le format des messages, le protocole de transport, la classification,..) en utilisant le WSDL comme un vocabulaire de publication.

**publisherAssertion** : Description de la relation qu'une businessEntity entretient avec une autre businessEntity.

### 1.3.3.2 Mécanisme d'Interrogation

UDDI fournit une API d'interrogation qui permet de diviser les données de l'annuaire UDDI en trois catégories :

**Les pages blanches** : contiennent les informations sur les fournisseurs de services telles que le numéro contact, le nome et l'adresse.

**Les pages jaunes** : classification des services et des compagnies suivant une taxonomie standardisée. Elles comportent les descriptions des services au format WSDL publiées par les fournisseurs.

**Les pages vertes** : offrent les indications techniques détaillées pour faire appel au service et joindre le fournisseur (processus métier, les descriptions,...).

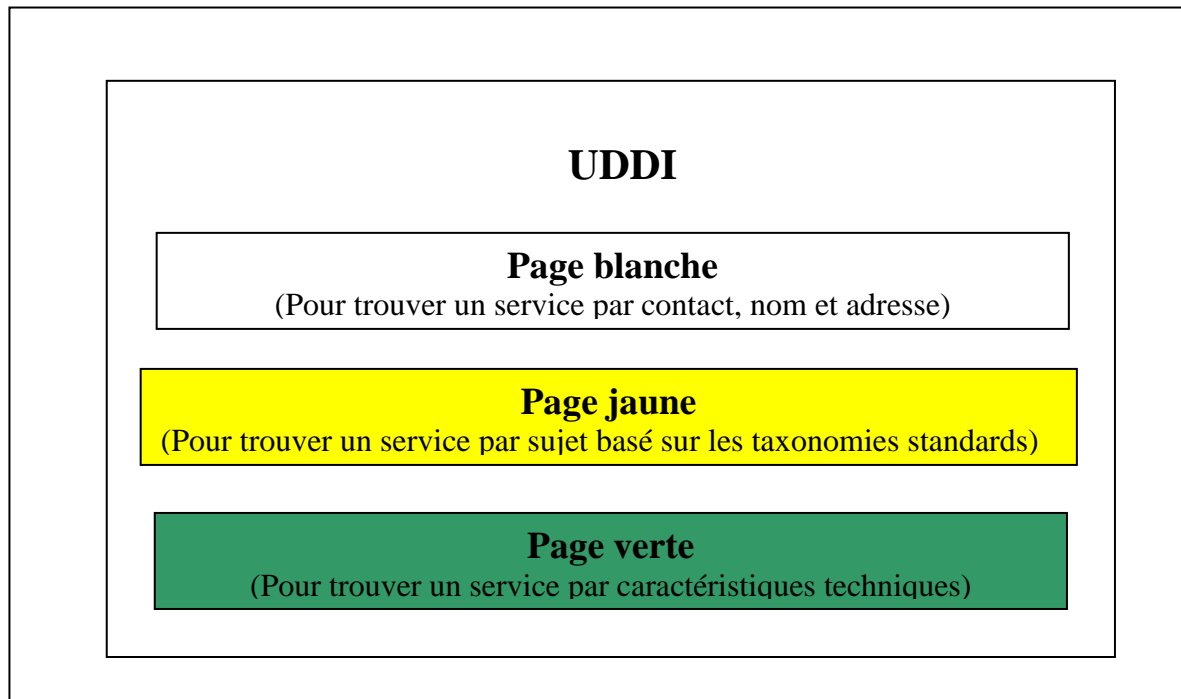


Figure 2.3 : L'annuaire UDDI

### 1.3.4 Vue globale des principales technologies

Le schéma suivant résume le mécanisme d'accès aux services de l'annuaire UDDI grâce aux technologies de base : SOAP, WSDL et UDDI.

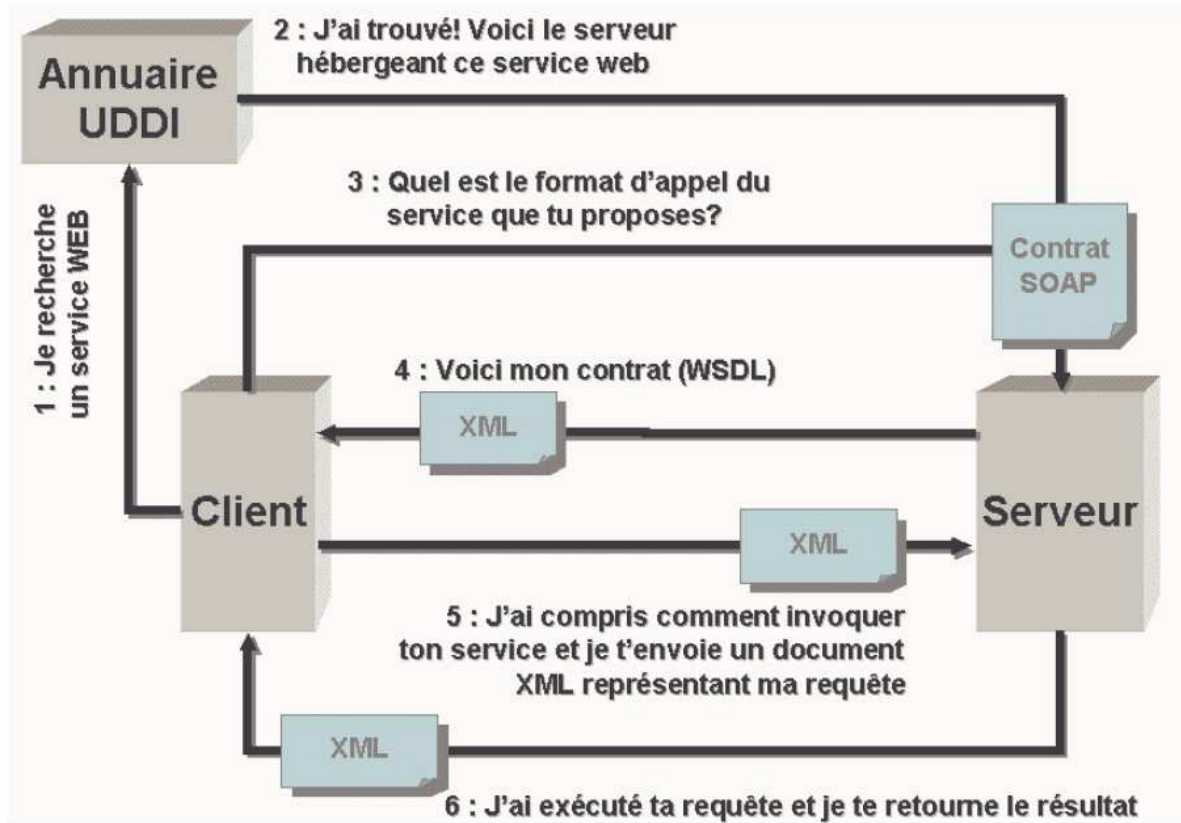


Figure 2.4 : Vue globale : les technologies de base [15]

#### 1.4 « learnServe » : plate forme e-learning orientée services

Pour avoir plus de compatibilité dans l'échange de données entre les systèmes e-learning, plusieurs standards ont été développés. L'existence d'un nombre important de standards pour le même et unique problème pose des difficultés. « LearnServe » [39] attaque ce problème par l'utilisation du paradigme orienté services (SOA : architecture Orientée Services) qui offre un nouveau modèle pour construire des systèmes d'apprentissage évolutifs et rapidement adaptables.

Les composants système et le contenu (Learning objects) « LearnServe » sont tous les deux fournis sous forme services web par plusieurs et déferents fournisseurs.

learnServe est utilisé en conjonction avec d'autre système, donc le transport du contenu d'un système à un autre est le point le plus intéressant et l'aspect le plus important de l'approche LearnServe, c'est que tout à l'exception de la plate-forme d'intégration est mis en œuvre sous la forme d'un service. Cela signifie, notamment, que le contenu e-learning est également

implémenté comme un service qui peut être intégré dans une plateforme d'intégration. Il n'y a pas de paquets de contenu physique.

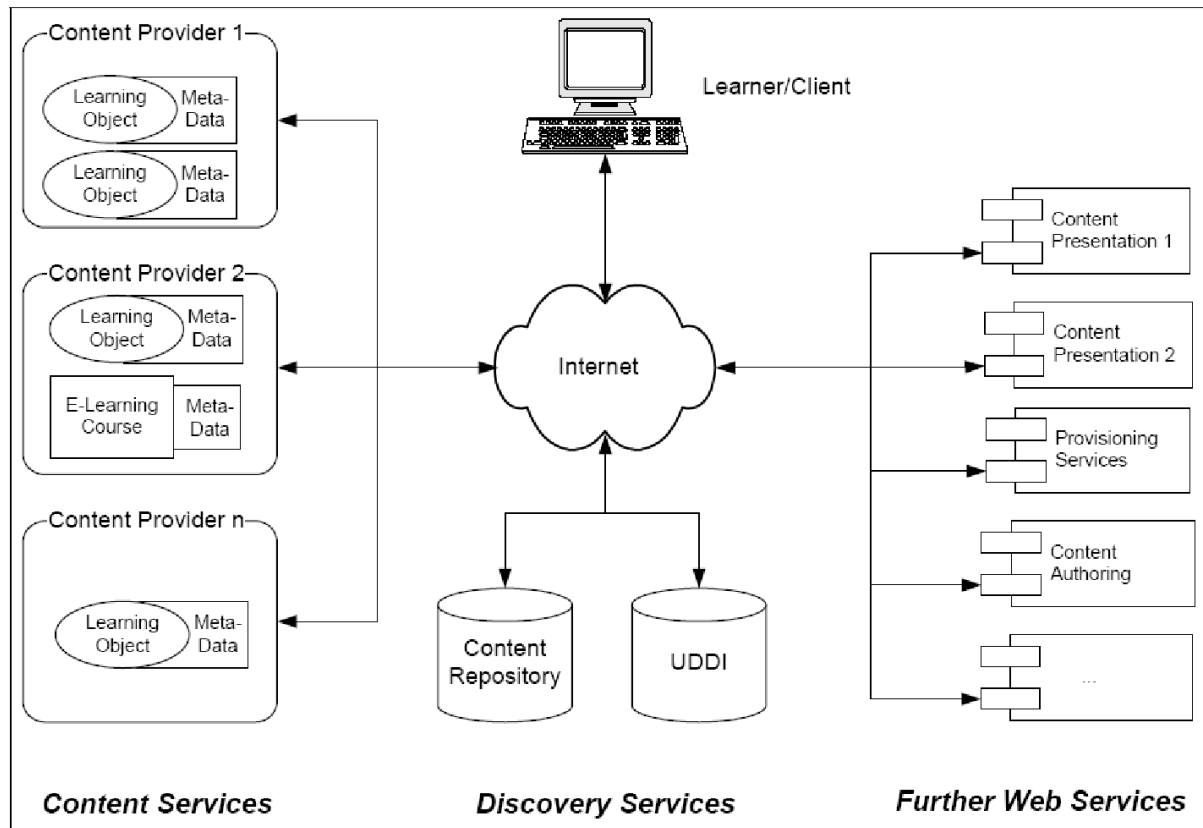


Figure 2.5 : Architecture orientée services d'un système e-learning [39]

- Les systèmes comme LearnServe utilisent un registre UDDI pour rechercher des services communs. Toutefois, UDDI n'est pas approprié pour les services de contenu puisque le stockage des métadonnées complémentaires sur le contenu n'est pas suffisamment pris en charge. Donc le processus de découverte est soutenu par le repository LearnServe et adapte essentiellement le framework UDDI au contexte d'e-learning. Il se distingue par le fait que le repository contient lui-même des données centralisées sur le contenu, c'est à dire, toutes les méta-informations couvertes par la norme LOM. Pour utiliser certains contenu, la plate-forme sous-jacente appelle le contenu souhaité, qui est ensuite exécuté par un service de présentation pour le remettre à l'apprenant.

- LearnServe services sont décrits par Web Services Description Language (WSDL) documents et utilise Simple Object Access Protocol (SOAP) pour les messages de leurs interactions. Toutefois, ni, WSDL ni SOAP permettent de manipuler les données utilisateur d'une manière standardisée et uniforme. Pour être en mesure d'offrir des informations sur le

matériel d'apprentissage déjà présenté et de vérifier le droit d'accès Provisioning Server est utilisé.

-LearnServe système utilise la norme WSRP pour fournir du contenu et les exercices services web. Les données binaires comme les animations peut être fixées au moyen de pièces jointes aux services web. Comme il utilise la notion de Sharable Content Services (SCSs) selon SCORM Sharable Content Object (SCO). Comme un SCO, une SCS est toujours composé de plusieurs actifs et peut être interprété comme contenu à afficher à la fois, c'est à dire, une présentation complète y compris les activités de navigation du contenu. Chaque activité de navigation dans le contenu est un appel d'une SCS et donc une communication avec le fournisseur distant.

L'utilisation de WSRP conforme des services web, permet une utilisation plug-and-play des SCS pour tous les partenaires connectés et fait SCSs hautement réutilisable.

L'exploitation des architectures orientées services, conduit des avantages :

- le contenu peut être intégré dans un système en tant que service.
- les plates-formes peuvent être configurés en fonction des besoins utilisateurs.
- la maintenance de contenu est plus facile, puisque il est stocké une fois sur le serveur du fournisseur.
- Chaque apprenant se connecte, exactement au même contenu. Les Mises à jour sont effectuées une seule fois, et une distribution physique du contenu à tous les LMS, qui ont importé le contenu, n'est pas nécessaire.
- les fonctionnalités peuvent être réutilisés dans différentes plates-formes.

## **2 Les services web sémantique**

Les services web sont des composants logiciels qui offrent un nouveau modèle pour construire des systèmes évolutifs et rapidement adaptables. Ce nouveau modèle de déploiement d'applications assure l'interopérabilité en se basant sur les standards SOAP(un protocole de transmission de messages), WSDL(un langage de description de services), et UDDI (un annuaire pour l'enregistrement des services). Ces standards facilitent le transport, l'invocation, la description et la recherche de services web mais ne permettent pas leurs automatisations.

L'émergence des technologies du web sémantiques telle que les ontologies ont permis d'ajouter des annotations sémantiques à la description des services web afin d'automatiser leurs utilisation : Ce que on appelle les services web sémantiques. [23]

## 2.1 Le web sémantique

Le web sémantique, inventé par Tim Berners-Lee (directeur du W3C), consiste à définir une nouvelle génération du web. Il désigne un ensemble de technologies visant à rendre le web d'aujourd'hui un vaste espace d'échange de ressources humains et machines grâce à une représentation explicite de la sémantique des données, programmes, pages web et services web.

Cette infrastructure permet l'utilisation de connaissances formalisées en plus du contenu informel actuel du Web et de localiser, d'identifier et de transformer des ressources de manière robuste et saine en s'appuyant sur un système de métadonnées formelles, des langages de représentation développés par le W3C et les ontologies qui représentent la technologie clé du web sémantique.

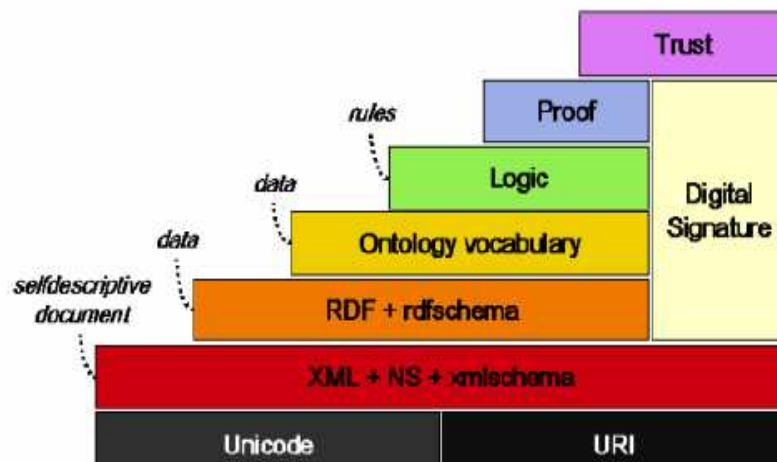


Figure 2.6 : Les couches du web sémantique

## 2.2 Les langages du web sémantique

Pour que le web sémantique puisse être manipulé par les machines, il est nécessaire de disposer de langages pour exprimer les données et les métadonnées, définir les ontologies et décrire les services. Il existe trois sortes de langages : les langages d'assertion pour définir les relations entre objets et exprimer les annotations associées aux ressources web (RDF et Cartes topiques), les langages de définition d'ontologies (OWL) et des langages pour la description des services web (WSDL et autres).



### 2.2.1 RDF (Resource Description Framework)

RDF est un langage formel développé par le W3C destiné à décrire les ressources Web et leurs métadonnées, de façon à permettre le traitement automatique de telles descriptions et assurer l'interopérabilité entre applications échangeant de l'information non formalisée et non structurée sur le Web.

Un document RDF est un ensemble de triplets de la forme < sujet, prédicat, objet >. Les éléments de ces triplets peuvent être des URIs (Universal Resource Identifiers) des littéraux ou des variables. Cet ensemble de triplet peut être représenté de façon naturelle par un graphe plus précisément un multi-graphe orienté étiqueté où les éléments apparaissant comme sujet ou objet sont des sommets, et chaque triplet est représenté par un arc dont l'origine est son sujet et la destination est son objet. Ce document sera codé en machine par un document RDF/XML, mais est souvent représenté sous une forme graphique.

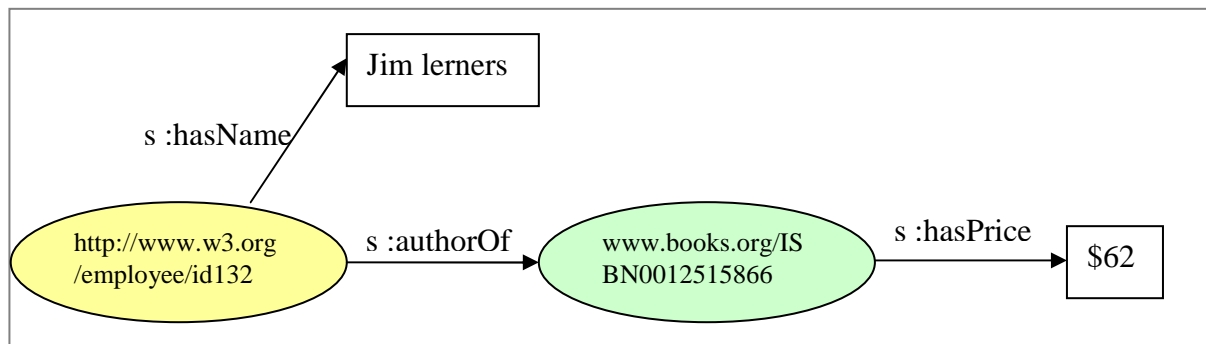


Figure 2.7 : Exemple de graphe RDF

### 2.2.2 OWL (Ontology Web Langage)

OWL offre un moyen d'écrire des ontologies web et intègre des outils de comparaison de propriétés et de classes. Il offre aussi une plus grande capacité d'interprétation de contenus web, grâce à un vocabulaire plus large et à une sémantique formelle appropriée [7] [20]. Ceci a permis de développer des outils pour la description sémantique de services web comme OWL-S (anciennement DAML-S basé sur le langage DAML+OIL). Il dispose d'un grand nombre de constructeurs permettant d'exprimer les propriétés des classes définies. C'est pour cette raison que OWL est divisé en sous langages qui sont les suivants :

**OWL Lite** : destiné aux utilisateurs qui ont besoin des hiérarchies de classifications et des caractéristiques de contraintes simples. Il ne contient qu'un sous ensemble réduit de constructeurs disponibles

**OWL DL** : Supporte les utilisateurs qui demandent un maximum d'expressivité tout en maintenant la complétude et la décidabilité. OWL DL contient tout les constructeurs du langage OWL mais sont utilisables avec des restrictions (par exemple, lorsqu'une classe peut être une sous classe de plusieurs autre classes, une classe ne peut être une instance d'une autre classe).

**OWL Full** : destiné aux utilisateurs qui demandent un maximum d'expressivité avec la liberté syntaxique de RDF sans aucune garantie de calculs. OWL Full permet aussi à une ontologie d'augmenter le sens du vocabulaire prédéfini (RDF et OWL).

## 2.3 Les ontologies

Le mot Ontologie vient du mot grec ontos pour être et de logos pour univers. C'est un terme philosophique introduit au XIXeme siècle qui caractérise l'étude des êtres

Dans les milieux de l'intelligence artificielle, l'ontologie ait été abordée pour la première fois par John McCarthy qui reconnut le recouplement entre le travail fait en Ontologie philosophique et l'activité de construire des théories logiques de systèmes d'intelligence artificielle. Avec l'émergence de l'ingénierie des connaissances, de la modélisation conceptuelle, et de la modélisation du domaine, la signification du terme a évolué

Au début des années 1990, plusieurs définitions ont été proposées dans chacun des sous-domaines de l'intelligence artificielle.

La définition de Neeches et ses collègues: "An ontology defines the basic terms and relations to define extensions to the vocabulary". [28]

En 1993, Gruber propose sa définition qui est la définition plus citée : "An ontology is an explicit specification of a conceptualization". [18]

Dans le cadre du web sémantique une ontologie est une l'organisation hiérarchique des concepts d'un domaine particulier. Elles servent pour le vocabulaire, la structuration et l'exploitation des méta-données, comme représentation pivot pour l'intégration de sources de données hétérogènes et pour décrire les services Web.

### 2.3.1 Composantes d'une ontologie

Les connaissances traduites par une ontologie sont à véhiculer à l'aide des éléments suivants :[33]

**Les concepts :** aussi appelés termes ou classes de l'ontologie, correspondent aux abstractions pertinentes d'un segment de la réalité (le domaine du problème), retenues en fonction des objectifs qu'on se donne et de l'application envisagée pour l'ontologie. Ces concepts peuvent être classifiés selon plusieurs dimensions : niveau d'abstraction (concret ou abstrait) ; atomicité (élémentaire ou composée) ; niveau de réalité (réel ou fictif).

**Les relations :** traduisent les associations (pertinentes) existant entre les concepts présents dans le segment analysé de la réalité. Ces relations incluent les associations suivantes: Sous-classe-de (généralisation – spécialisation), Partie-de (agrégation ou composition), Associée-à, Instance de, etc. Ces relations nous permettent d'apercevoir la structuration et l'interrelation des concepts, les uns par rapport aux autres.

**Les fonctions :** constituent des cas particuliers de relations, dans laquelle un élément de la relation, le nième est défini en fonction des n-1 éléments précédents.

**Les axiomes :** constituent des assertions, acceptées comme vraies, à propos des abstractions du domaine traduites par l'ontologie.

**Les instances :** constituent la définition extensionnelle de l'ontologie ; ces objets véhiculent les connaissances (statiques, factuelles) à propos du domaine du problème.

### 2.3.2 Types d'ontologie

Les ontologies peuvent être classifiées selon plusieurs dimensions. Nous présenterons celle basée sur l'objet de conceptualisation [33].

#### **Ontologie de représentation des connaissances**

Ce type d'ontologies regroupe les concepts impliqués dans la formalisation des connaissances. Un exemple est l'ontologie de Frame qui intègre les primitives de représentation des langages à base de frames : classes, instances, facettes, propriétés/slots, relations, restrictions, valeurs permises, etc.

### **Ontologie supérieure ou de Haut niveau**

Cette ontologie est une ontologie générale. Son sujet est l'étude des catégories des choses qui existent dans le monde, soit les concepts de haute abstraction tels que: les entités, les événements, les états, les processus, les actions, le temps, l'espace, les relations, les propriétés. L'ontologie de haut de niveau est fondée sur : la théorie de l'identité, la méréologie (theory of whole and parts role) et la théorie de la dépendance.

### **Ontologie Générique**

Cette ontologie aussi appelée, méta-ontologies ou core ontologies, véhicule des connaissances génériques moins abstraites que celles véhiculées par l'ontologie de haut niveau, mais assez générales néanmoins pour être réutilisées à travers différents domaines. Elle peut adresser des connaissances factuelles (Generic domain ontology) ou encore des connaissances visant à résoudre des problèmes génériques (connaissances procédurales) appartenant à ou réutilisables à travers différents domaines (Generic task ontology).

### **Ontologie du Domaine**

Cette ontologie régit un ensemble de vocabulaires et de concepts qui décrit un domaine d'application ou monde cible. Elle permet de créer des modèles d'objets du monde cible. L'ontologie du domaine est une méta-description d'une représentation des connaissances, c'est-à-dire une sorte de méta-modèle de connaissance dont les concepts et propriétés sont de type déclaratif. La plupart des ontologies existantes sont des ontologies du domaine. Selon Mizoguchi, l'ontologie du domaine caractérise la connaissance du domaine où la tâche est réalisée. Dans le contexte de la formation à distance, un domaine serait par exemple : le téléapprentissage.

### **Ontologie de Tâches**

Ce type d'ontologies est utilisé pour conceptualiser des tâches spécifiques dans les systèmes, telles que les tâches de diagnostic, de planification, de conception, de configuration, de tutorat, soit tout ce qui concerne la résolution de problèmes. Elle régit un ensemble de vocabulaires et de concepts qui décrit une structure de résolution des problèmes inhérente aux tâches et indépendante du domaine.

### **Ontologie d'Application**

Cette ontologie est la plus spécifique. Les concepts dans l'ontologie d'application correspondent souvent aux rôles joués par les entités du domaine tout en exécutant une certaine activité.

## **3 La découverte de services web**

Les services web ont connu un grand succès due à l'utilisation des normes ouvertes (Service Oriented Architecture : SOA) et leurs capacités dans l'intégration des applications d'une façon plus rapide et moins coûteuse et avec des perspectives d'évolution et de réutilisation réelles. Ainsi le web devient un dispositif distribué où les services peuvent interagir en étant capables de se découvrir automatiquement, de négocier entre eux et de se composer en des services plus sophistiqués.

La découverte de services web se réfère à la possibilité de localiser automatiquement un Web service qui répond à des besoins particuliers des utilisateurs. Différentes approches ont été proposées pour réaliser la découverte dynamique de services.

Les premières approches de découverte de services proposées dans la littérature étaient des approches syntaxiques. Avec l'émergence de la nouvelle génération du web, des approches sémantiques ont vu le jour. Celles-ci présentent une description sémantique de services web interprétables par une application dans le but d'assurer un degré d'automatisation élevé.

### **3.1 Approches syntaxiques**

Les premières approches proposées ont été des approches syntaxiques basées sur la comparaison de mots clé.

#### **3.1.1 Approche UDDI**

L'approche UDDI [29] est basée sur l'utilisation d'un registre de descriptions de services Web. la publication et la découverte des services web est basée sur la comparaison des mots clés.

Lors de la recherche l'utilisateur ou le programme envoie une requête constituée de mots clés, cette requête est ensuite traitée et découpée puis comparée avec les mots clés du registre UDDI. Un ensemble de descriptions des services Web est ensuite donné comme résultat de recherche, l'utilisateur sélectionne le service Web qui répond au mieux à ses exigences.

L'origine de cette approche est issue du domaine de recherche d'information. Malgré sa simplicité cette approche présente des limites : la méthode renvoie un nombre important de résultats ou au contraire peu de résultats.

Pour rendre la découverte de services Web basée sur les mots clés plus efficace, une technique issue du domaine de IR(Recherche d'Information) a été adoptée. Elle consiste à représenter les descriptions des services Web sous forme de vecteurs, tel que chaque vecteur contient un ensemble de mots issus des termes utilisés dans toutes les descriptions des Services Web. Les vecteurs de description sont ensuite organisés sous forme de matrice (terme £ description). La deuxième étape consiste à appliquer, sur cette matrice, la technique LSI (Latent Semantic Indexing). Cette méthode permet de renvoyer toute description de service Web qui a une relation sémantique avec la requête de recherche.

### **3.1.2 Approche basée sur la qualité de service**

Cette approche [36] prend la qualité de service comme contrainte pendant la recherche d'un web service. Elle comporte quatre éléments qui sont :

- Le fournisseur de service : offre le service en l'enregistrant dans l'annuaire UDDI,
- Le consommateur de service : c'est lui qui découvre et invoque les services,
- L'annuaire UDDI : c'est l'annuaire UDDI doté d'informations sur la description fonctionnelle du web service et des informations sur les QoS associées à ce service.
- Et le certificateur : son rôle est de vérifier les revendications de qualité du service d'un web service avant son enregistrement.

Le fournisseur de service transmet la QoS de son service au certificateur. Ce dernier vérifie et certifie ou pas la revendication : dans le cas favorable l'information d'identification de certification est envoyée au fournisseur et enregistrée au niveau du certificateur et identifiée par un ID.

Après la publication de la certification de QoS, le fournisseur peut alors enregistrer la description fonctionnelle du service et la qualité de service dans UDDI qui va vérifier l'existence de la certification auprès du certificateur.

Le consommateur recherche dans UDDI un web service avec des contraintes de QoS à la recherche. S'il n'y a aucun service avec les qualités exigées le système demande au consommateur de réduire les contraintes de qualité de service.

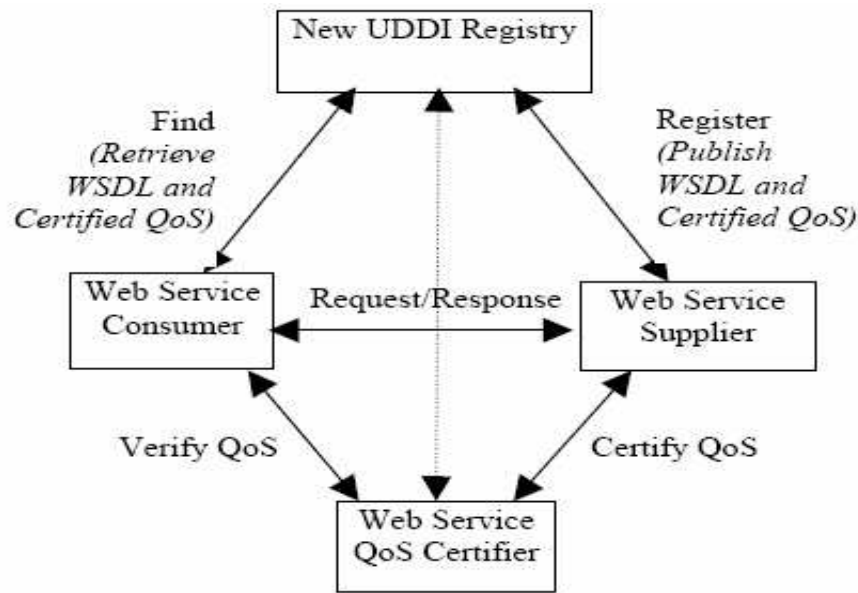


Figure 2.8 : Un modèle pour l'enregistrement et l'invocation de web services. [36]

### 3.1.3 Approche AASDU

Le système appelé AASDU (Agent Approach for Service Discovery and Utilization) [32] propose une approche multi agents pour la découverte de services Web. Il est composé d'une interface utilisateur graphique (Graphical User Interface GUI), un agent analyseur de requête (Query Analyzer Agent QAA) et un système référentiel des domaines d'expertises des agents de service.

Ce système permet de référencier les agents selon leur expertise donc chaque agent a juste connaissance des services relatifs à son domaine d'expertise.

Dans ce système, chaque agent a un profil déterminant ses intérêts et son expertise. L'expertise de l'agent est représentée par un vecteur de mots clés tel que chaque mot clé représente un domaine donné. Pour chaque mot clé un score est assigné indiquant le degré d'expertise de l'agent dans ce domaine.

De plus, chaque agent a une liste d'agents voisins (Neighborlist) qui indique les voisins de l'agent ainsi que leurs domaines d'expertise.

Le module de services offre trois sous services :

- un service permettant aux fournisseurs de services de publier les descriptions de leurs services Web. Pour chaque fournisseur de service Web un agent lui est assigné.
- un service consiste en un agent de négociation permettant la sélection de service.

- et un service qui est offert par l'agent composition, le rôle de cet agent est d'invoquer un des services issus de l'étape de sélection ou d'invoquer un service similaire lors de la défaillance du service sélectionné en premier.

Dans ce système l'utilisateur entre sa requête de recherche sous forme de chaîne de caractères via l'interface GUI. La requête est ensuite envoyée à l'agent QAA, le rôle de cet agent est de faire ressortir de cette requête les mots clés pertinents qui seront utilisés pour sélectionner des agents du système référentiel des domaines d'expertises des agents service. Pour ce faire l'agent QAA utilise une simple variante de la technique TFIDF (Term Frequency Inverse Document Frequency) pour ressortir les mots clés pertinents, sur la base de ces mots clés l'agent QAA sélectionne un ensemble d'agents experts. Les agents sélectionnés transmettent par la suite les paramètres des services avec lesquels ils sont liés à l'agent composition. Ce dernier invoque un des services selon le choix de l'utilisateur.

## 3.2 Approches sémantiques

La principale approche de découverte qui avait conduit le développement des cadres sémantiques des services Web est l'approche OWL-S.

### 3.2.1 Ontologie des Services Web OWL-S

L'ontologie des services web OWL-S donne des informations sur les fournisseurs de services web avec un ensemble de classes décrivant les propriétés et les caractéristiques de ces services. OWL-S facilite l'automatisation de l'utilisation des services web, y compris la découverte et la sélection d'un service, l'invocation de ce service, l'interopération et la composition de services.

OWL-S s'appuie sur le langage OWL qui est une recommandation du Web Ontology Working Group du World Wide Web Consortium (W3C) [20].

Les classes principales suivantes de l'ontologie OWL-S sont illustrées dans la figure suivante :

**ServiceModel.** Le "ServiceModel" sert à expliquer comment le service fonctionne. Il comprend deux aspects :

- Le processus qui est décrit par l'ontologie de processus "ProcessOntology". Cette ontologie permet de décrire le processus en termes de "inputs", "outputs", "preconditions" et "effects".
- Le module de contrôle du processus défini par l'ontologie de contrôle de processus



"Process Control Ontology".

**ServiceProfile** : Le "ServiceProfile" décrit le service en fonction de ce qu'il fait pour permettre au client de voir si le service proposé lui convient. Le "ServiceProfile" propose une présentation descriptive du service web et sert de support à la découverte de services web et à leur sélection. Le "ServiceProfile" est composé de trois sections de description : "Profile", "Profile non functional properties" et "Service Profile Functionality Description".

**ServiceGrounding** : Le "ServiceGrounding" traite le niveau concret des deux spécifications abstraites définies dans le "ServiceProfile" et le "ServiceModel" afin de faciliter l'accès au service web.

**Resource** : Les services web ont besoin de ressources pour pouvoir s'exécuter. On en distingue deux types de ressources : celles qui sont consommables "ConsumableAllocation", et celles qui restent réutilisables "ReusableAllocation".

### 3.2.2 Approche basée sur les protocoles de conversation

L'approche proposée est basée sur le comportement des services (en particulier leurs protocoles de conversation). Elle utilise un appariement sémantique des graphes de processus, pour délivrer aussi bien des résultats exacts que des résultats approchés en appliquant des opérations d'édition permettent de modifier le graphe requête pour le rapprocher le plus possible des graphes cibles selon une distance associée à chaque requête. [17]

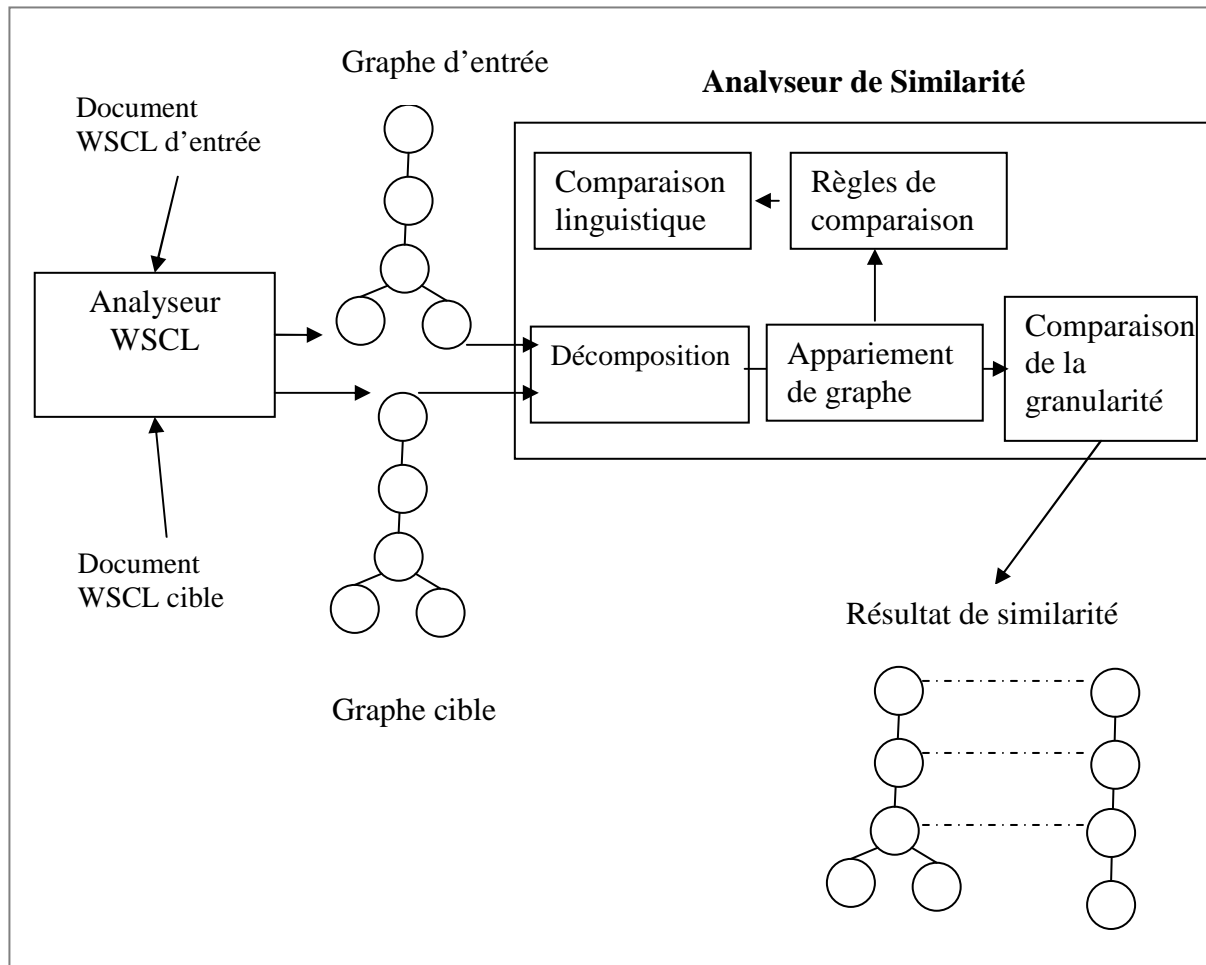


Figure2.10 : Approche basée sur les protocoles de conversation [17]

Le système réalisant l'appariement basé sur le comportement se compose de :

-un analyseur qui transforme un document WSCL en graphe et un module d'analyseur de similarité qui évalue la similarité entre les graphes.

L'analyseur de similarité est composé des éléments suivants :

- Module de décomposition qui applique l'opération de décomposition afin d'avoir le même niveau de granularité dans les deux graphes.
- Appariement de graphes qui prend comme entrées les graphes produits par le module de décomposition et trouve l'isomorphisme de sous-graphes avec tolérance d'erreurs avec le coût minimal.
- Module des règles de comparaison qui regroupe les fonctions de coût pour les opérations d'édition de graphe.
- Comparaison linguistique qui implante différents algorithmes pour trouver la similarité entre deux chaînes de caractères.

- Comparaison de la granularité qui vérifie si des opérations de composition/jointure sont nécessaires et ajoute leurs coût à la distance totale.

## 4 Conclusion

Les services web sont des applications auto descriptives et modulaires qui fournissent un modèle simple de déploiement d'applications basé sur les standards SOAP, WSDL et UDDI. Ces technologies semblent suffisantes pour la mise en place de ces services, mais insuffisantes pour l'automatisation de leurs exploitation. Les premières approches proposées sont basées sur des mots clés.

L'ajout de la couche sémantique à la description des services web joue un rôle très important dans l'amélioration de la tâche de découverte. L'une des approches qui avaient conduit le développement des cadres sémantiques des services Web, l'approche OWL-S (Ontology Web Language for Web Service)

Cependant dans le domaine d'apprentissage OWL-S ne répond pas aux besoins des utilisateurs en termes de services qui offrent des fonctionnalités et des services liés aux ressources pédagogiques et ne satisfait leurs exigences. OWL-S ne prend pas en considération ni l'aspect pédagogique du service, ni la sémantique de son contenu.

A cette fin nous proposerons dans le chapitre suivant une approche de découverte de services liés aux ressources pédagogiques.

Celle-ci est basée sur deux ontologies : une ontologie associée à la pédagogie liée aux ressources pédagogiques qui donne une description sémantique et pédagogique des services web associés aux ressources pédagogiques, et une ontologie du domaine à enseigner relative au domaine d'apprentissage.

# *Chapitre 3 :*

## *La conception*

### **Sommaire**

1	Problématique .....	51
2	Approche pour la découverte des services web.....	52
2.1	Ontologie des Services Web OWL-S.....	52
3	La description ontologique de l'approche de découverte de services.....	54
3.1	Ontologie Associée à la pédagogie liée aux ressources pédagogiques.....	54
3.2	Ontologie du Domaine à Enseigner.....	57
3.2.1	Diagramme de classes de l'ontologie du domaine à enseigner .....	60
4	Les acteurs intervenants du système.....	60
5	Diagramme de cas d'utilisation principal.....	62
6	Description des Fonctionnalités du Système.....	62
6.1	Architecture de Publication de Services Web.....	63
6.2	Diagramme de séquence du module de publication de services web.....	64
6.3	Architecture de Découverte de Services Web.....	65
6.4	Diagramme de séquence du module de découverte de services web.....	66
7	La procédure de matching .....	67
8	Conclusion.....	72

*Table des Figures*

Figure 3.1: Les classes principales de l'ontologie OWL-S .....	53
Figure 3.2 : L'ontologie OWL-S+.....	55
Figure 3.3 : Ontologie associée à la pédagogie.....	57
Figure 3.4 : Découpage des connaissances du domaine d'enseignement.....	58
Figure 3.5: L'ontologie du domaine à enseigner.....	59
Figure 3.6 : Diagramme de l'ontologie du domaine à enseigner.....	60
Figure 3.7 : Diagramme de contexte.....	61
Figure 3.8 : Diagramme de cas d'utilisation principal.....	62
Figure 3.9: Architecture du système proposé.....	63
Figure 3.10 : Diagramme de séquence de publication des services web.....	64
Figure 3.11 : Diagramme de séquence de découverte de services web.....	66
Figure 3.12: Les étapes du filtrage .....	68

## **1 Problématique de la découverte de Services Web Liés aux Ressources Pédagogiques**

De nos jours les environnements d'e-learning sont distribués et basés sur des architectures orientées services. Les composants et ressources de ces systèmes sont implémentés à l'aide de services web. Certaines ressources pédagogiques disponibles sur le web sont liées aux services web mis à disposition du public par des fournisseurs de services afin d'aider l'apprenant dans son apprentissage en lui permettant l'accès à des contenus d'apprentissage, et l'enseignant auteur-concepteur dans la création et la publication de contenus sous forme de services web pouvant par la suite être à leur tour utilisés... Cependant leur exploitation pose des problèmes de découverte et de localisation des services qui répondent le mieux aux besoins et spécificités des différents usagers. Les premières approches de découverte de services proposées dans la littérature étaient des approches syntaxiques. Avec l'émergence de la nouvelle génération du web, des approches sémantiques ont vu le jour. Celles-ci présentent une description sémantique de services web interprétables par une application dans le but d'assurer un degré d'automatisation élevé. Ceci a donné naissance à une nouvelle génération de langages pour le web, tels que RDF, OWL. Ceci a permis de développer des outils pour la description sémantique de services web comme OWL-S. Dans le cas des services pédagogiques l'annuaire UDDI n'est pas approprié pour leur découverte de fait qu'il ne permet pas le stockage des informations relatives à la sémantique du contenu pédagogique d'où la nécessité d'une approche sémantique pour la description de ces services comme le OWL-S. Cependant OWL-S ne prend en considération ni l'aspect pédagogique du service, ni les besoins et spécificités du demandeur de service. Notre proposition consiste à développer une approche de découverte de services adaptés aux besoins et spécificités des usagers. Celle-ci est basée sur deux ontologies : une ontologie associée à la pédagogie liée aux ressources pédagogiques qui donne une description sémantique et pédagogique des services web associés aux ressources pédagogiques, et une ontologie du domaine à enseigner relative au domaine d'apprentissage.

## **2 Approche pour la découverte de Services Web Liés aux Ressources Pédagogiques**

L'un des objectifs principaux de OWL-S est la découverte automatique de services web. Cet objectif n'est que partiellement atteint dans le domaine de l'e-learning. En effet OWL-S ne

décrit pas parfaitement les services web associés à l'e-Learning. Le "ServiceProfile" de l'ontologie des services web OWL-S donne une description du service et de son fournisseur, une description du comportement fonctionnel du service et une description des attributs fonctionnels utiles à la sélection automatique des services mais il ne donne pas une description pédagogique du service qui prend en considération les concepts du domaine d'apprentissage, les relations sémantiques qui existent entre eux, et les aspects pédagogiques liées aux ressources numériques.

L'objectif de notre approche est d'améliorer la découverte et la pertinence des résultats obtenus à l'aide de OWL-S en sélectionnant les services qui répondent le mieux aux besoins et spécificités des différents usagers (auteur-concepteur, apprenant, formateur et administrateur), et d'offrir à l'apprenant un apprentissage personnalisé selon ses besoins [1]. Cette approche est basée sur :

- La définition d'une sous ontologie dans le "ServiceProfile" de l'ontologie des services web OWL-S, qui permet une description sémantique et pédagogique des services et qui assure le lien avec l'ontologie du domaine à enseigner.
- L'utilisation de l'ontologie du domaine à enseigner qui définit les concepts du domaine d'apprentissage et les relations sémantiques qui existent entre eux.

## 2.1 Ontologie des Services Web OWL-S

L'ontologie des services web OWL-S donne des informations sur les fournisseurs de services web avec un ensemble de classes décrivant les propriétés et les caractéristiques de ces services. OWL-S facilite l'automatisation de l'utilisation des services web, y compris la découverte et la sélection d'un service, l'invocation de ce service, l'interopération et la composition de services.

OWL-S s'appuie sur le langage OWL qui est une recommandation du Web Ontology Working Group du World Wide Web Consortium (W3C) [20].

Les classes principales suivantes de l'ontologie OWL-S sont illustrées dans la figure 3.1 :

**ServiceModel.** Le "ServiceModel" sert à expliquer comment le service fonctionne. Il comprend deux aspects :

- Le processus qui est décrit par l'ontologie de processus "ProcessOntology". Cette ontologie permet de décrire le processus en termes de "inputs", "outputs", "preconditions" et "effects".
- Le module de contrôle du processus défini par l'ontologie de contrôle de processus "Process Control Ontology".

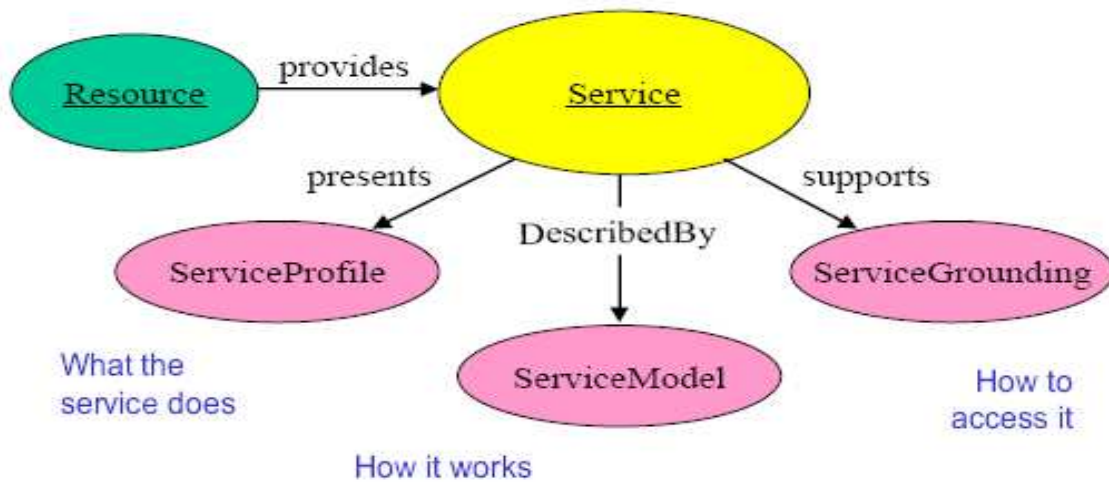


Figure 3.1: Les classes principales de l'ontologie OWL-S [31]

**ServiceProfile** : Le "ServiceProfile" décrit le service en fonction de ce qu'il fait pour permettre au client de voir si le service proposé lui convient. Le "ServiceProfile" propose une présentation descriptive du service web et sert de support à la découverte de services web et à leur sélection. Le "ServiceProfile" est composé de trois sections de description : "Profile", "Profile non fonctional properties" et "Service Profile Functionality Description".

**Profile**. Cette section fournit les superclasses de chaque section de la description. Elle ne nécessite aucune représentation des services, mais exige l'information de base pour lier n'importe quelle instance de profil avec une instance de service. Il y a une relation bidirectionnelle entre un service et un profil :

- **Presents** : décrit une relation entre une instance de service et une instance de profil, et indique que le service est décrit par le profil.
- **PresentedBy** : c'est l'inverse de "presents", elle indique qu'un profil donné décrit un service
- Profile non functional properties** : Cette section fournit une description du service et de son fournisseur.
- **ServiceName** : fait référence au nom du service qui est offert. Il peut être utilisé comme un identifiant du service.
- **TextDescription** : fournit une brève description du service, résume ce que le service offre et décrit ce que le service exige pour fonctionner.
- **ContactInformation** : fournit un mécanisme de référence aux personnes responsables du service.



**Service Profile Functionality Description** : Cette section fournit la spécification des fonctionnalités que le service offre et la spécification des conditions qui doivent être remplies pour un résultat réussi. Elle définit les propriétés suivantes :

- **HasParameter** : elle prend comme valeur les instances de la classe "Parameter" de l'ontologie du processus.
- **HasInput** : elle prend comme valeur les instances de la classe "Inputs" de l'ontologie du processus.
- **HasOutput** : elle prend comme valeur les instances de la classe "ConditionalOutput" définie dans l'ontologie du processus.
- **HasPrecondition** : elle précise l'une des conditions du service et prend comme valeur les instances de la classe "Precondition" définie dans l'ontologie du processus.
- **HasResult** : elle précise l'un des résultats du service, tel que défini par la classe "Result" dans l'ontologie du processus. Elle précise dans quelles circonstances les résultats sont générés.

Depuis la version 1.2 les sous ontologies suivantes du "ServiceProfile" ont été déplacées vers des fichiers connexes qui peuvent être importés dans le fichier Profile.owl selon les besoins :

- **Actor** (name, title, phone, fax, email, physicalAddress, webURL).
- **ServiceParameter** (serviceParameterName, sParameter)
- **ServiceCategory** (categoryName, taxonomy, value, code)

**ServiceGrounding** : Le "ServiceGrounding" traite le niveau concret des deux spécifications abstraites définies dans le "ServiceProfile" et le "ServiceModel" afin de faciliter l'accès au service web.

**Resource** : Les services web ont besoin de ressources pour pouvoir s'exécuter. On en distingue deux types de ressources : celles qui sont consommables "ConsumableAllocation", et celles qui restent réutilisables "ReusableAllocation".

### **3 La description ontologique de l'approche de découverte de services web**

#### **3.1 Ontologie associée à la pédagogie liée aux ressources pédagogiques**

L'ontologie associée à la pédagogie liée aux ressources pédagogiques est définie comme une sous ontologie de l'ontologie "ServiceProfile". L'ontologie résultante baptisée OWL-S+ utilise donc l'ontologie OWL-S et l'ontologie associée à la pédagogie liée aux ressources pédagogiques. Les classes de cette ontologie donnent une description du contenu pédagogique

et sémantique du service ainsi que les caractéristiques techniques et financières jugées nécessaires pour la description des ressources pédagogiques. Les classes principales de cette ontologie sont donc liées aux aspects pédagogie, technique et financier.

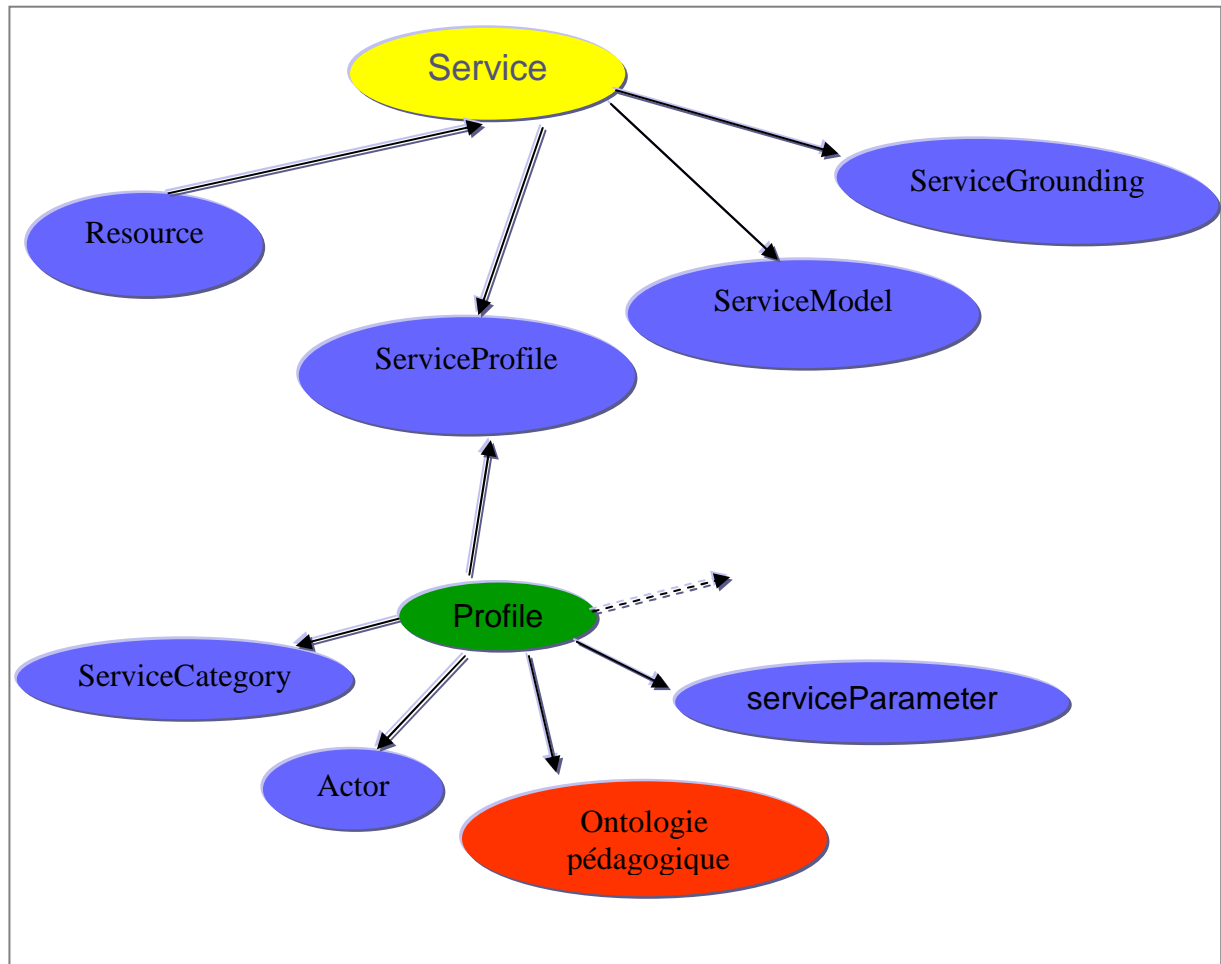


Figure 3.2 : L'ontologie OWL-S+

**Aspect Pédagogie.** Les sous classes de cette classe donnent une description de l'aspect pédagogique de la ressource et de son contenu, elle a comme sous classes :

– **Domaine.** Cette classe, comme son nom l'indique, décrit les informations relatives au domaine d'apprentissage tels que :

**Nom domaine :** désignation du domaine d'apprentissage.

**Description :** description textuelle du domaine d'apprentissage.

– **Titre.** Indique le titre de la ressource

– **Description.** Donne une description textuelle de la ressource.

- **Langue.** Langue dans laquelle la ressource est décrite.
  - **Concept.** Indique le concept enseigné par la ressource.
  - **Type.** Décrit le type pédagogique de la ressource qui peut être une activité d'apprentissage, un scénario, un module d'apprentissage, un exercice, un test d'évaluation, un logiciel spécifique, une plate forme d'e-learning ...
  - **Éléments connaissance concept.** Liste des éléments de connaissance associés au concept.
  - **Poids concept.** Densité sémantique du concept dans la ressource pédagogique.
  - **Concept pré-requis.** Définit l'ensemble des concepts pré-requis pour l'utilisation de la ressource pédagogique.
  - **Élément pré-requis.** Définit l'ensemble des éléments de connaissance pré-requis nécessaires pour l'acquisition du concept enseigné par la ressource.
  - **Acteur.** Décrit l'utilisateur destinataire de la ressource. C'est la superclasse des classes suivantes :
    - **Rôle** : Le rôle de l'utilisateur qui peut être selon les cas un apprenant, un formateur, un auteur-concepteur ou un administrateur.
    - **Profil initial** : Définit l'ensemble des concepts et éléments de connaissance qui doivent être déjà acquis par l'acteur apprenant.
    - **Profil final** : Définit l'ensemble des concepts et éléments de connaissance qui seront acquis par l'apprenant après l'utilisation de la ressource.
- Aspect Technique.** Englobe toutes les informations concernant le matériel, la sécurité, la collaboration, la taille, le format et la durée d'apprentissage.
- Aspect Financier.** Elle a comme sous-classes les classes suivantes :
- **Coût** : Le coût de la ressource.
  - **Droit** : Les droits d'utilisation de la ressource

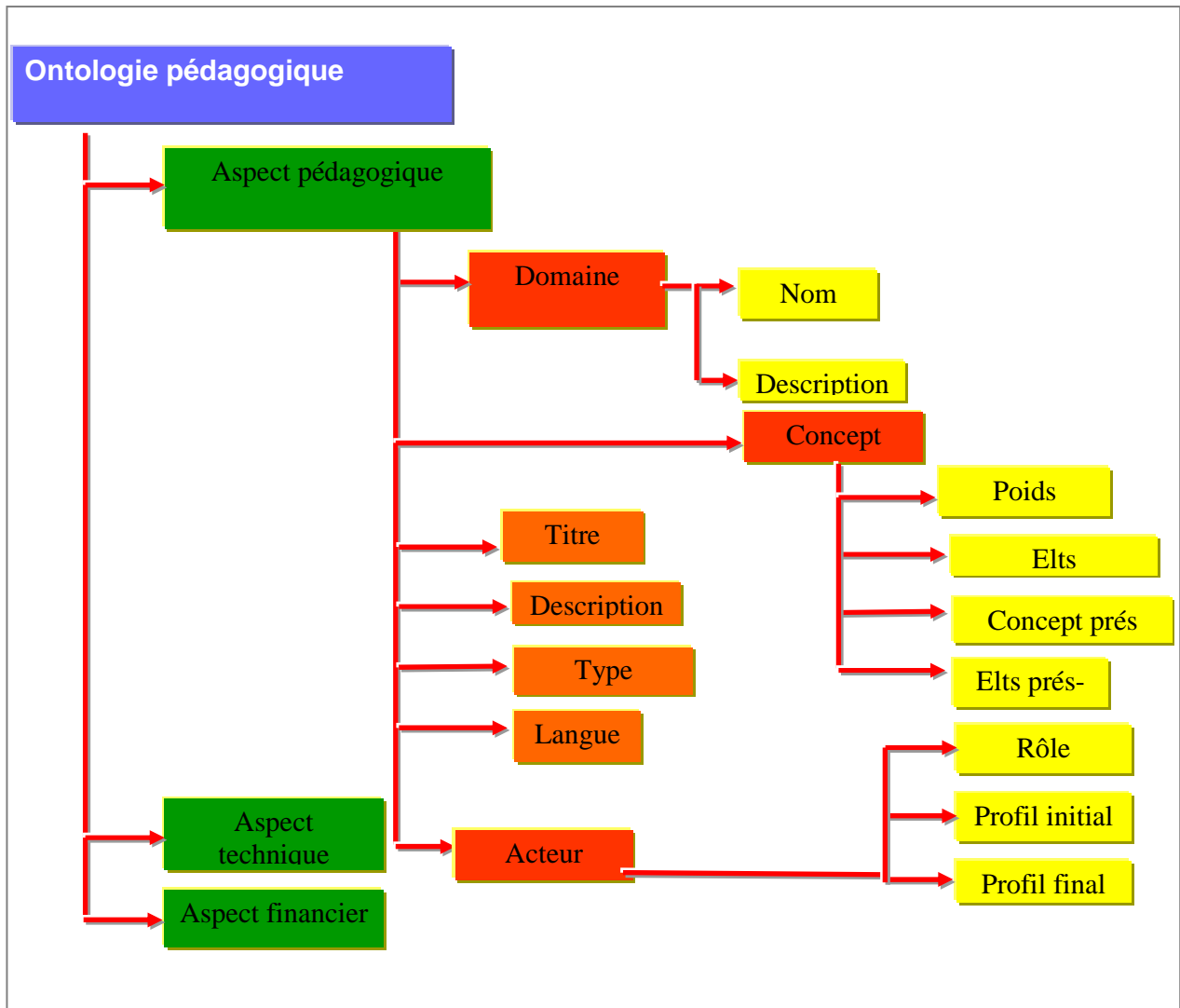


Figure 3.3 : Ontologie associée à la pédagogie

### 3.2 Ontologie du Domaine à Enseigner

Une ontologie de domaine définit un vocabulaire commun pour les différents acteurs intervenants dans un domaine donné, et qui ont besoin de partager l'information dans ce domaine. C'est une méta-description d'une représentation des connaissances du monde ciblé. Elle inclut la définition des concepts et des relations entre concepts du domaine.

Le découpage des connaissances du domaine d'enseignement (figure suivante) permet de classifier les connaissances d'un domaine à enseigner spécifique suivant un cadre générique réutilisable organisé autour des concepts suivants [3] [4] :

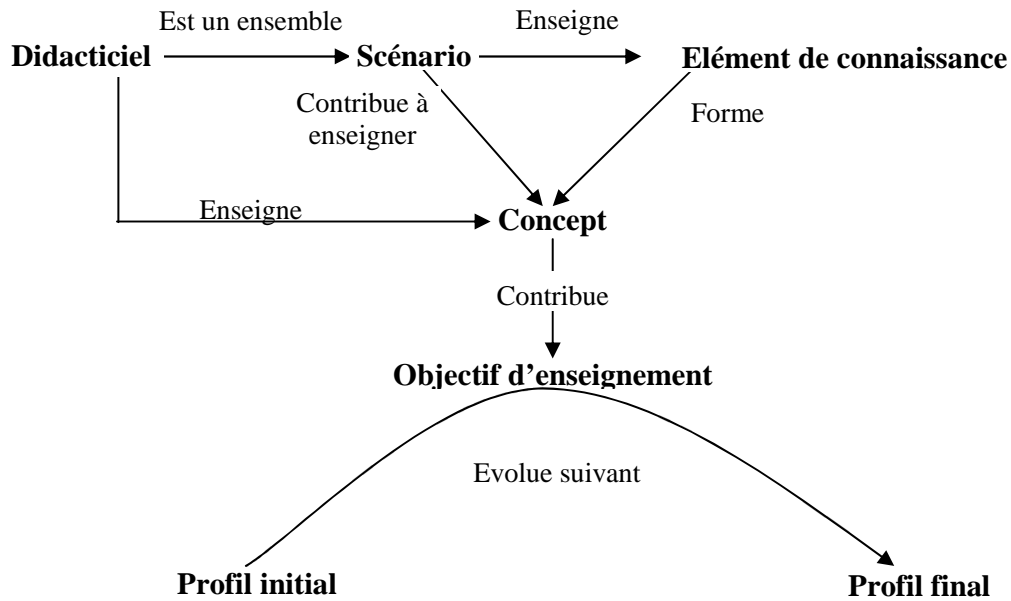


Figure3.4 : Découpage des connaissances du domaine d'enseignement [5]

**Didacticiel** : Un didacticiel est un logiciel pédagogique dédié, d'aide à l'enseignement et/ou à la formation personnalisée. Il est constitué d'une collection de scénarios et enseigne des concepts.

**Concept** : Un concept est constitué d'un ensemble d'éléments de connaissance valués, il peut être lié à d'autres concepts par diverses relations.

**Élément de connaissance** : C'est le « granule » de la matière à enseigner. Il est présenté seul ou combiné avec d'autres éléments de connaissance.

**Scénario** : C'est un ensemble d'exposés, d'exercices d'assimilation et de contrôle de connaissances.

**Profil initial** : Le profil initial est décrit par la liste des concepts que l'élève est supposé a priori posséder.

**Profil final (objectif d'enseignement)** : L'objectif d'enseignement est un ensemble de concepts à faire acquérir à l'apprenant.

L'ontologie du domaine à enseigner est alors décrite par la figure.

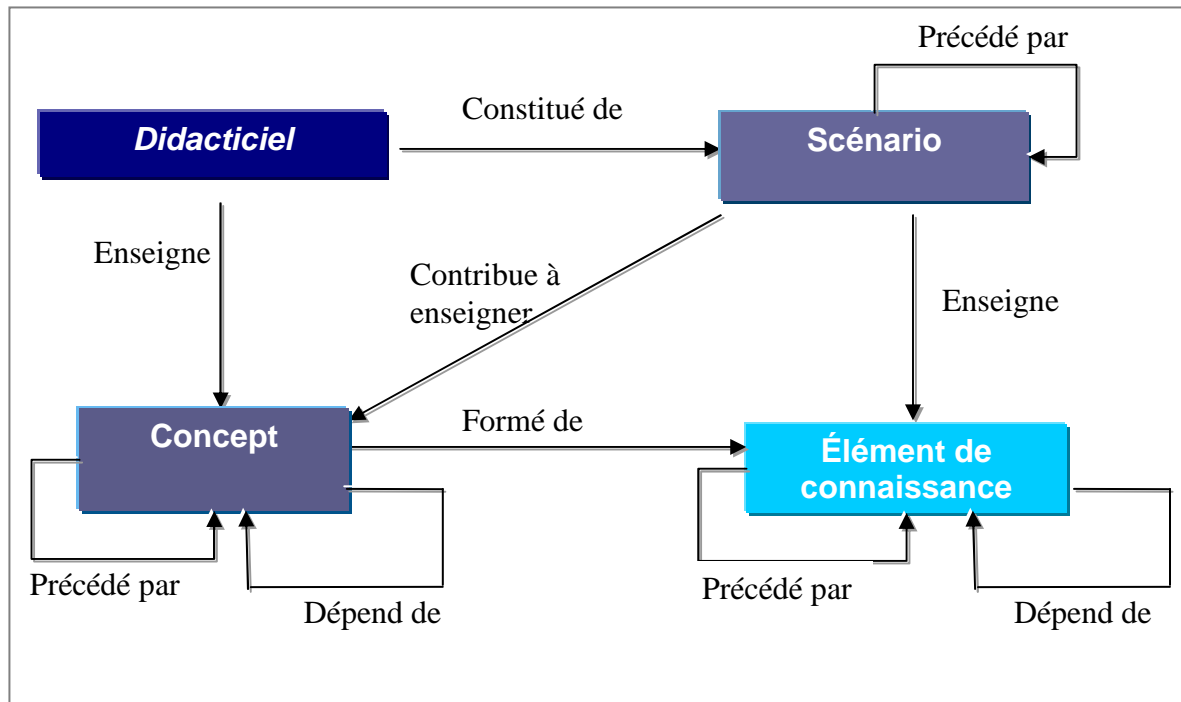


Figure 3.5: L'ontologie du domaine à enseigner

Ces concepts sont reliés entre eux par les relations suivantes :

- Relation entre didacticiel et scénario : la relation d'agrégation « constitué de ».
- Relation entre didacticiel et concept : la relation sémantique « enseigne ».
- Relation entre scénarios eux-mêmes : la relation de structuration « précédé par ».
- Relation entre scénario et concept : la relation sémantique « contribue à enseigner »
- Relation entre scénario et éléments de connaissance : la relation sémantique « enseigne »
- Relations entre concepts eux-mêmes : les relations de structuration « précédé par » et « dépend de »
- Relation entre concept et éléments de connaissance: la relation d'agrégation «formé de »
- Relations entre éléments de connaissance eux-mêmes : les relations de structuration « précédé par » et « dépend de »

### 3.2.1 Diagramme de l'ontologie du domaine à enseigner

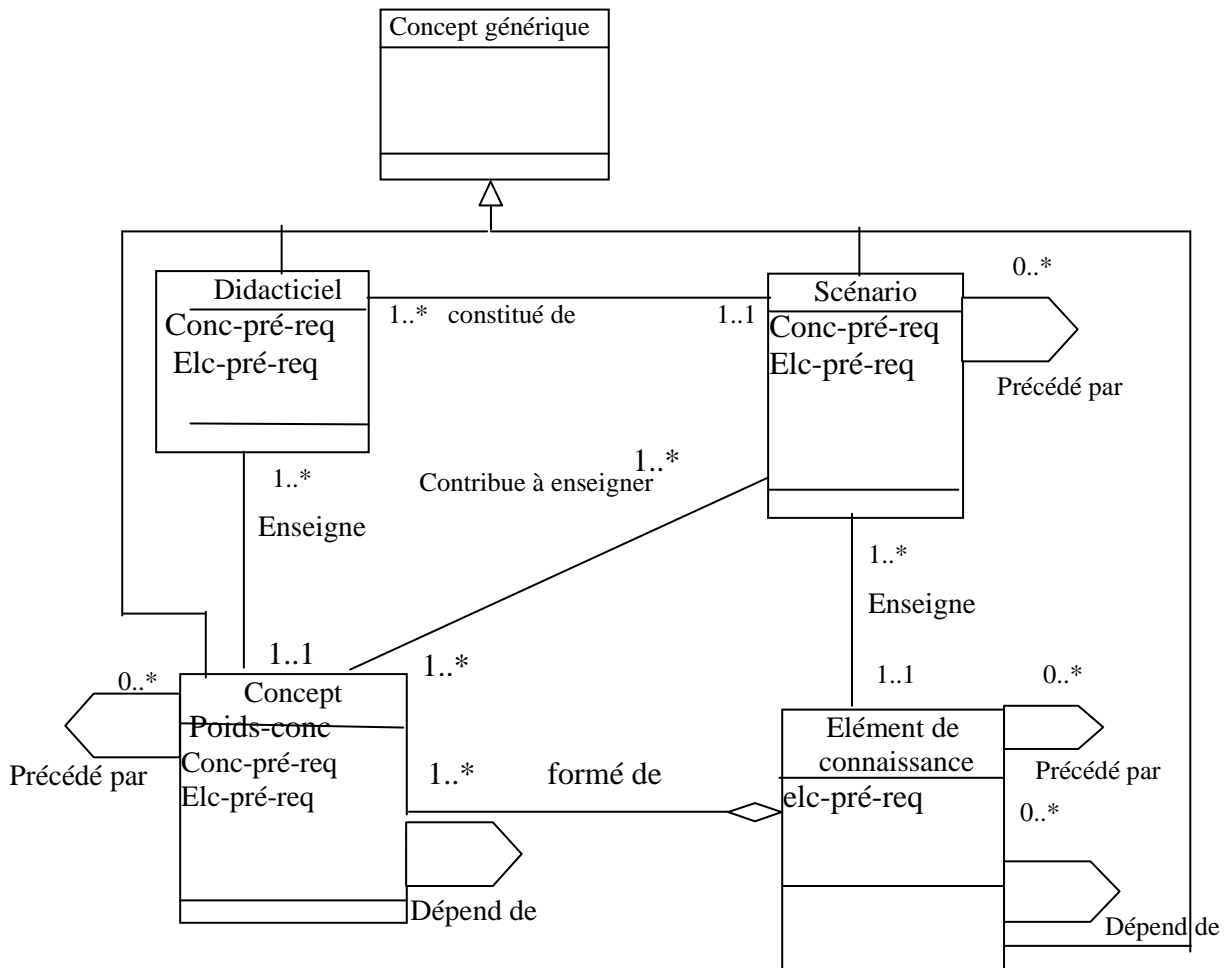


Figure 3.6 : Diagramme de l'ontologie du domaine à enseigner

## 4 Les acteurs intervenants du système

Les acteurs de notre système sont classés et organisés en trois catégories. La figure suivante présente le diagramme de contexte général de notre application :

**Les apprenants :** Un apprenant est une personne qui cherche à acquérir de la connaissance et d'appréhender un savoir en utilisant les services et les ressources offerts.

Un apprenant peut chercher une ressources qui lui convient et enseigne un concept dans l'ensembles de services publiés.

Dans notre cas, le profil de l'apprenant peut avoir l'une des valeurs suivantes : débutant, intermédiaire ou avancé.

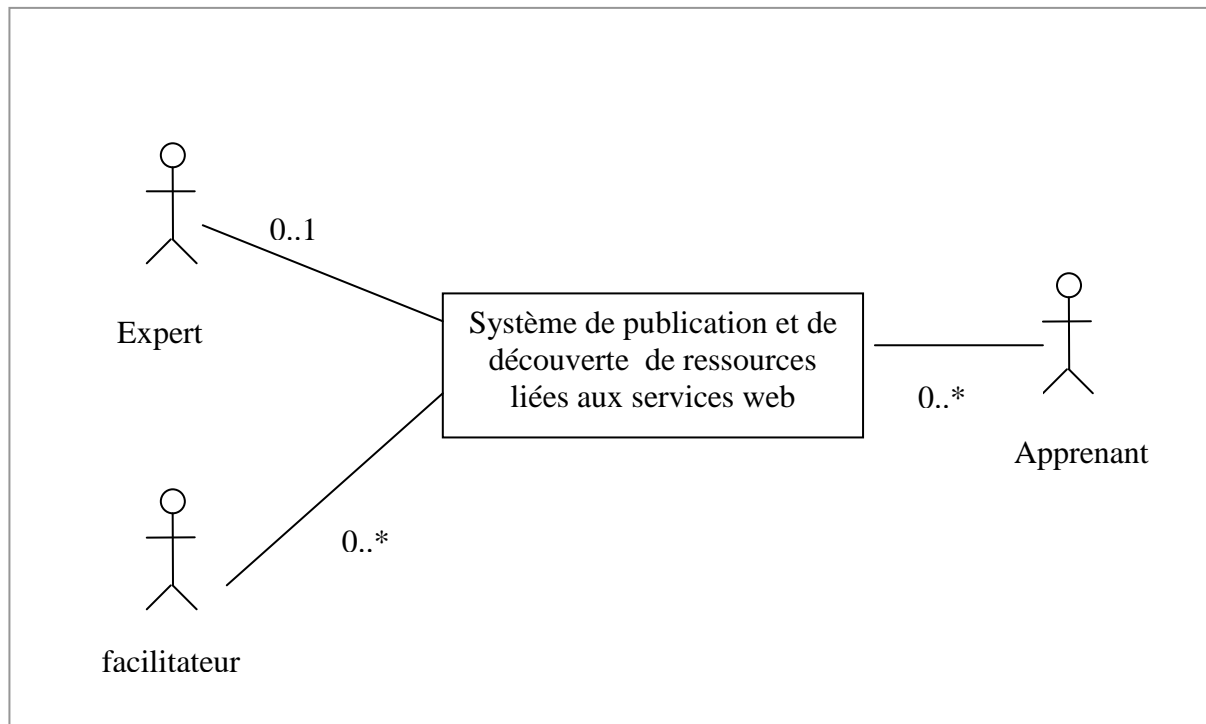


Figure 3.7 : Diagramme de contexte.

**Les experts :** ils sont experts dans le domaine à enseigner, leur rôle est de concevoir l'ontologie du domaine, la définition des concepts et les liens sémantiques, ainsi que sa mise à jours.

**Autres intervenants (facilitateur) :**

Ces intervenants, permettent et facilitent à l'apprenant de suivre une formation, ce sont : les formateurs, les gestionnaires, les concepteurs et les présentateurs. Ils ont comme tâches :

- la description ontologique et la publication des ressources pédagogiques (services web)
- la mise à jours des services web
- recherche d'un service web utile pour la conception d'un autre service.
- Identification des apprenants.



## 5 Diagramme de cas d'utilisation principal

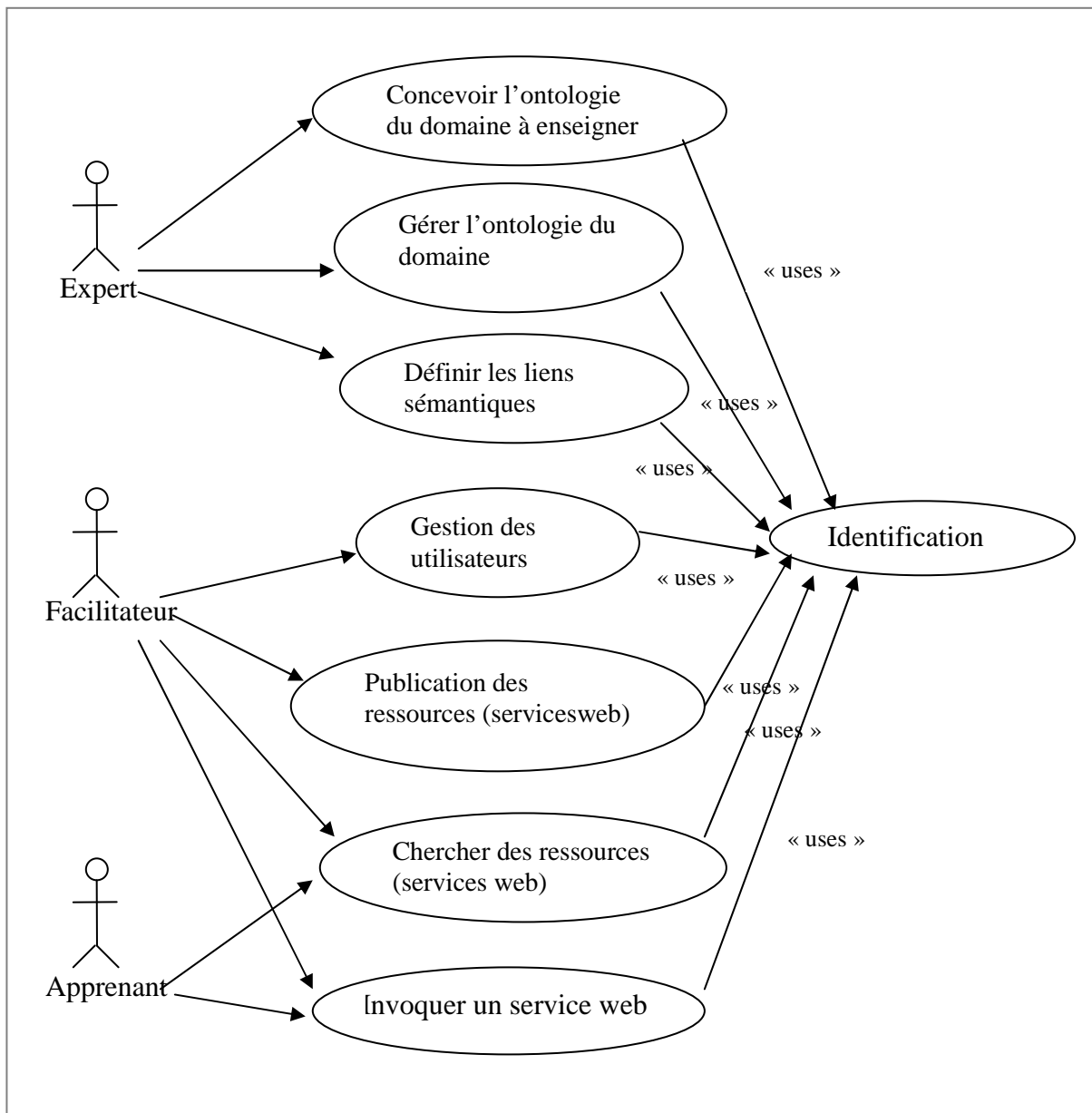


Figure 3.8 : Diagramme de cas d'utilisation principal

## 6 Description des Fonctionnalités du Système

Le système basé sur cette approche proposée offre aux divers acteurs des plates formes d'e-learning les deux fonctionnalités suivantes :

- La publication des ressources pédagogiques sous forme de services web dotés d'une description pédagogique et sémantique.
- La découverte et l'invocation des services web liés aux ressources pédagogiques qui répondent le mieux aux besoins et spécificités des différents usagers.

Ces fonctionnalités permettent aux plates formes d'e-learning en général une ouverture directe sur le web et aux apprenants en particulier un accès personnalisé aux ressources pédagogiques disponibles.

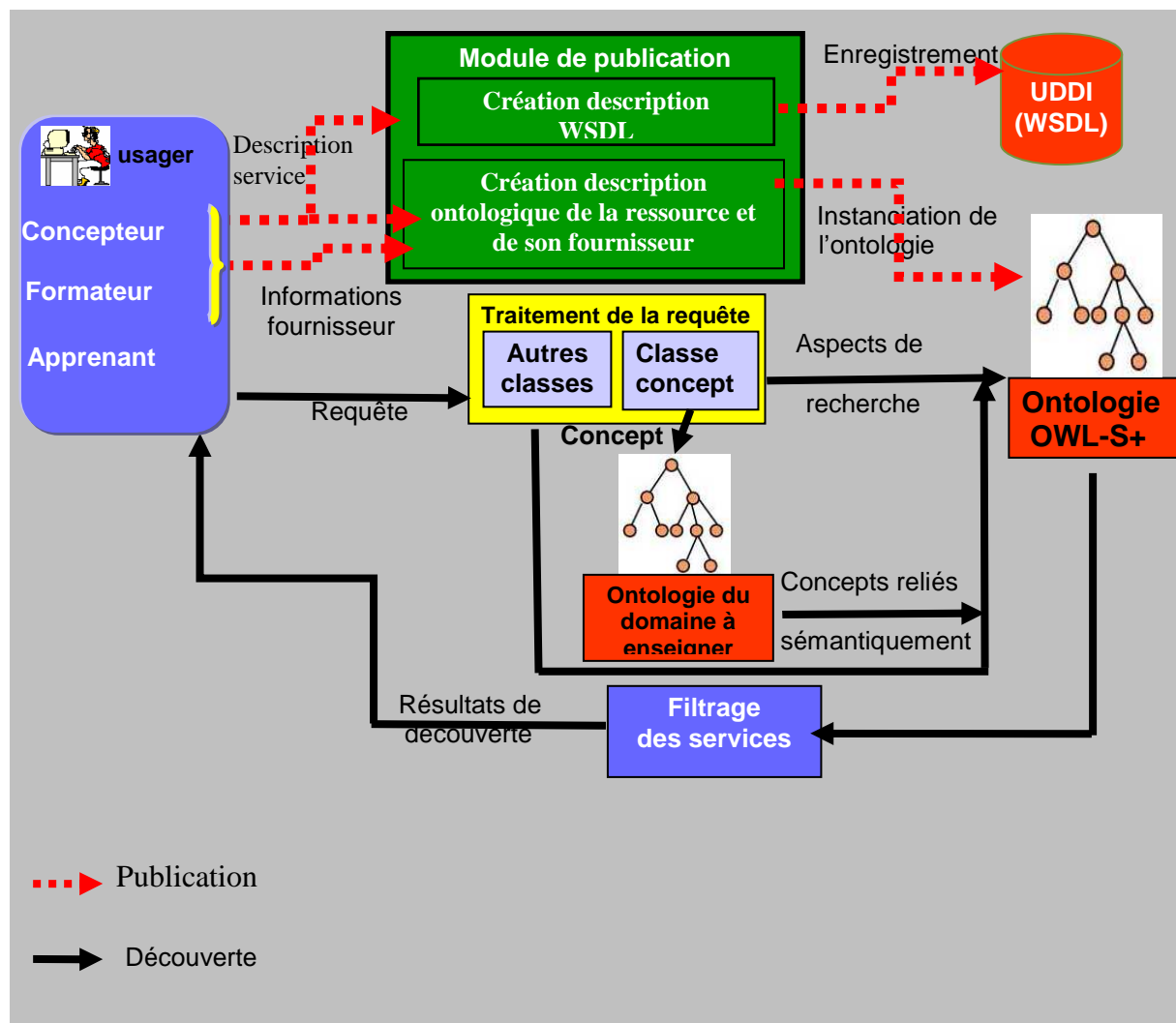


Figure 3.9: Architecture du système proposé

## 6.1 Publication de Services Web

Cette fonctionnalité permet à l'enseignant formateur de publier son cours sous forme de services web, et à l'administrateur-gestionnaire de publier les contenus des formations en ligne sous forme de services web internes (i.e. accessibles par les apprenants et les usagers inscrits) ou externes (i.e. utilisables par d'autres plates formes d'e-learning).

En publiant un service, le fournisseur de services fournit les informations qui présentent le service ainsi qu'une description conforme au format défini dans l'ontologie associée à la pédagogie liée aux ressources pédagogiques.

Le système de publication de services web crée alors une description WSDL contenant les informations nécessaires pour l'invocation de ce service et fait son enregistrement dans l'annuaire UDDI. Les informations relatives au service ainsi que celles en rapport avec le fournisseur de service sont enregistrées comme instances de l'ontologie OWL-S+. Chaque ressource pédagogique aura ainsi une représentation dans l'ontologie OWL-S+ comme une instance des classes de cette ontologie

## 6.2 Diagramme de séquence du module de publication de services web

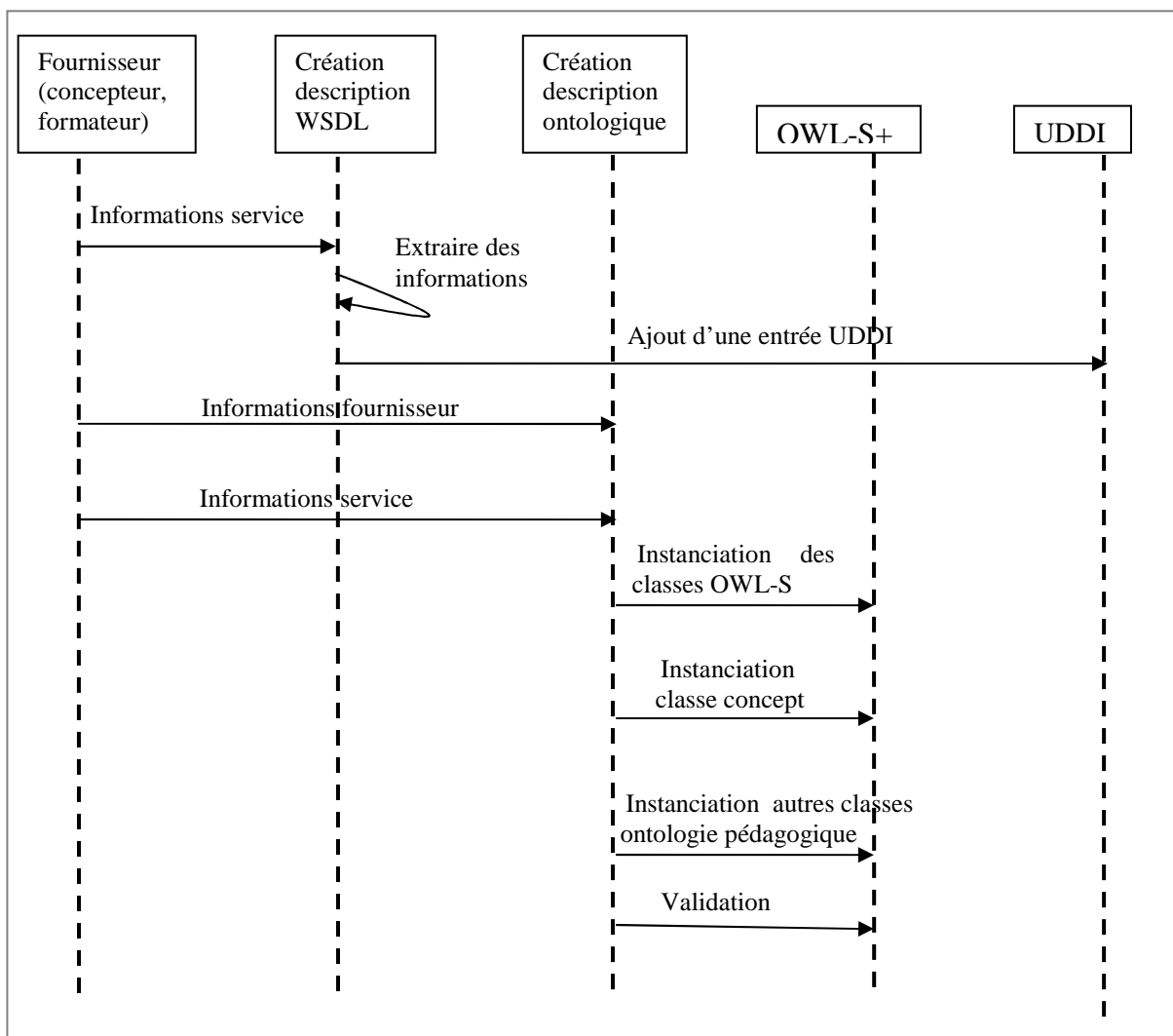


Figure 3.10 : Diagramme de séquence de publication des services web

### 6.3 Découverte de Services Web

Cette fonctionnalité permet aux enseignants formateurs de consulter ou d'utiliser des ressources pédagogiques, et aux apprenants de trouver des ressources internes ou externes en rapport avec leurs activités d'apprentissage.

Dans notre approche, la découverte des services web (figure 3.9) est basée sur deux ontologies : l'ontologie OWL-S+ et l'ontologie du domaine à enseigner permettant d'exploiter la couverture sémantique des concepts du domaine d'apprentissage.

Le système de découverte de services web détermine les classes à utiliser lors de la recherche en identifiant dans la requête introduite par l'utilisateur les concepts utilisés ainsi que les autres aspects de la recherche.

La découverte de services consiste à trouver dans l'ontologie résultante OWL-S+ les instances qui répondent directement aux aspects de la recherche et l'utilisation de l'ontologie du domaine à enseigner pour trouver les concepts liés sémantiquement à la classe concept considéré dans le but de les utiliser un à un comme instance de la classe concept (en gardant toujours les mêmes autres aspects de la recherche introduits dans la requête) pour découvrir éventuellement d'autres services web.

Un filtrage est ensuite effectué sur les résultats de la découverte selon la densité sémantique des concepts associés aux ressources trouvées. Ce mécanisme permet ainsi de sélectionner les services qui répondent le mieux aux besoins et spécificités des usagers.

### 6.4 Diagramme de séquence du module de découverte de services web

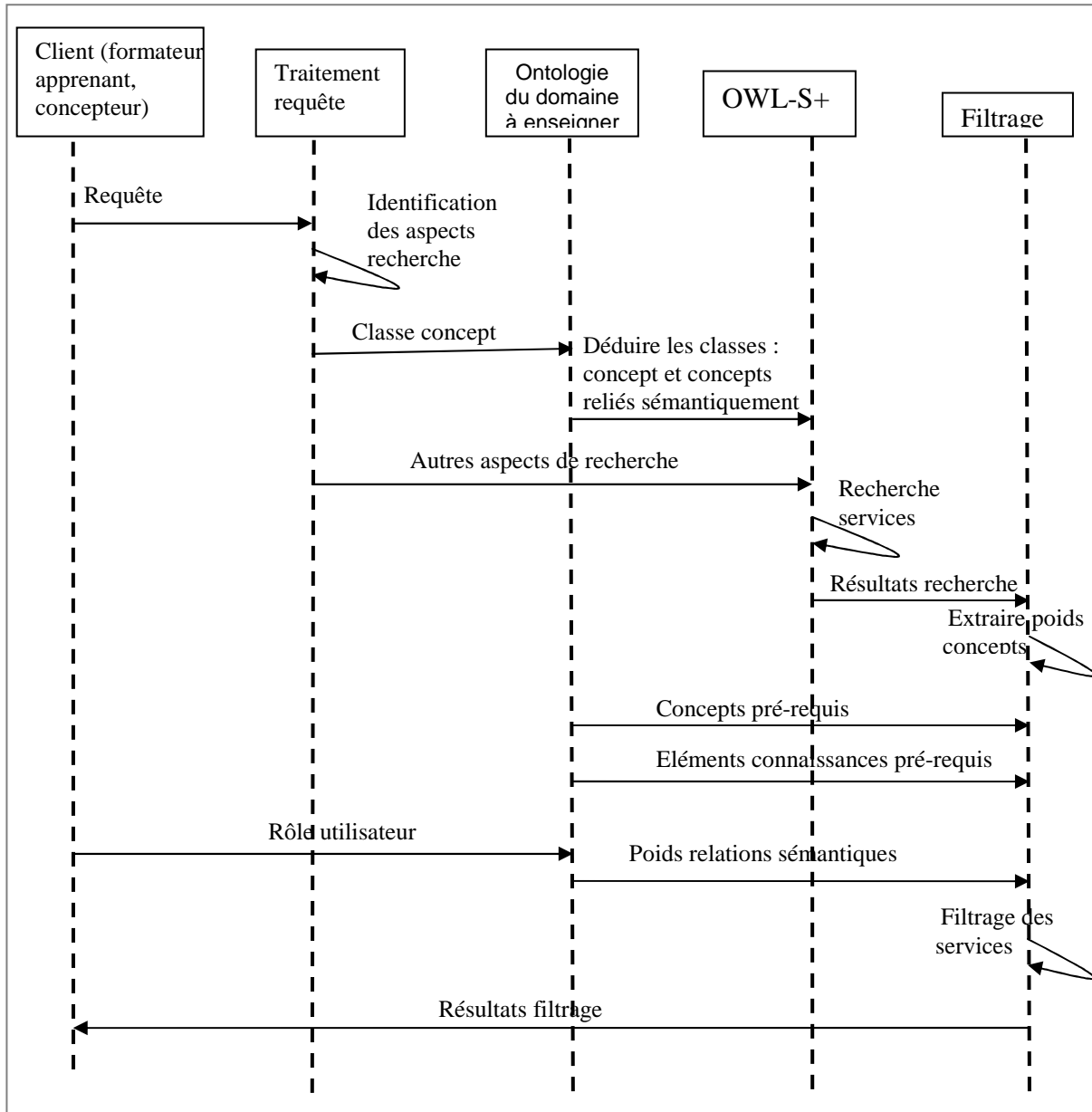


Figure 3.11: Diagramme de séquence de découverte de services web

## 7 La procédure de matching

La découverte de services consiste à trouver dans l'ontologie résultante OWL-S+ les instances qui répondent directement aux aspects de la recherche et l'utilisation de l'ontologie du domaine à enseigner pour trouver les concepts liés sémantiquement à la classe concept considéré dans le but de les utiliser un à un comme instance de la classe concept (en gardant toujours les mêmes autres aspects de la recherche introduits dans la requête) pour découvrir éventuellement d'autres services web.

Un filtrage est ensuite effectué sur les résultats de la découverte selon la densité sémantique des concepts associés aux ressources trouvées. Ce mécanisme permet ainsi de sélectionner les services qui répondent le mieux aux besoins et spécificités des usagers.

Une fois le concept est introduit par l'utilisateur avec l'ensemble des aspects de la recherche souhaités, le système effectue les opérations suivantes :

- utilise le concept comme entrée pour l'ontologie du domaine pour avoir en sortie :
  - \* tous les concepts reliés sémantiquement au concept introduit et les relations sémantiques qui les relient avec le concept requête.
  - \* les éléments de connaissance pré-requis et les concepts pré-requis du concept introduit, ces éléments de connaissance et concepts sont prédéfinis par les experts concepteurs de l'ontologie du domaine à enseigner dans le but de les utiliser comme un critère pour le filtrage.
- Puis le système cherche dans l'ontologie OWL-S+ tous les services qui sont reliés aux concepts trouvés dans l'ontologie du domaine et le concept introduit par l'utilisateur pour avoir un ensemble de services web reliés sémantiquement entre eux.

Un filtrage va être effectué sur cet ensemble de services web selon :

- le poids du concept dans la ressource donné par le fournisseur du service dans la classe poids concept pour les services reliés directement au concept requête, ou le poids du concept déduit à partir de l'ontologie du domaine dans le service web et la priorité de la relation qui le relie avec le concept de la requête.
- Dans les deux cas, le poids va être incrémenté selon la satisfaction du service web aux autres aspects de recherche sélectionnés par l'utilisateur y compris les éléments de connaissance et les concepts pré-requis.

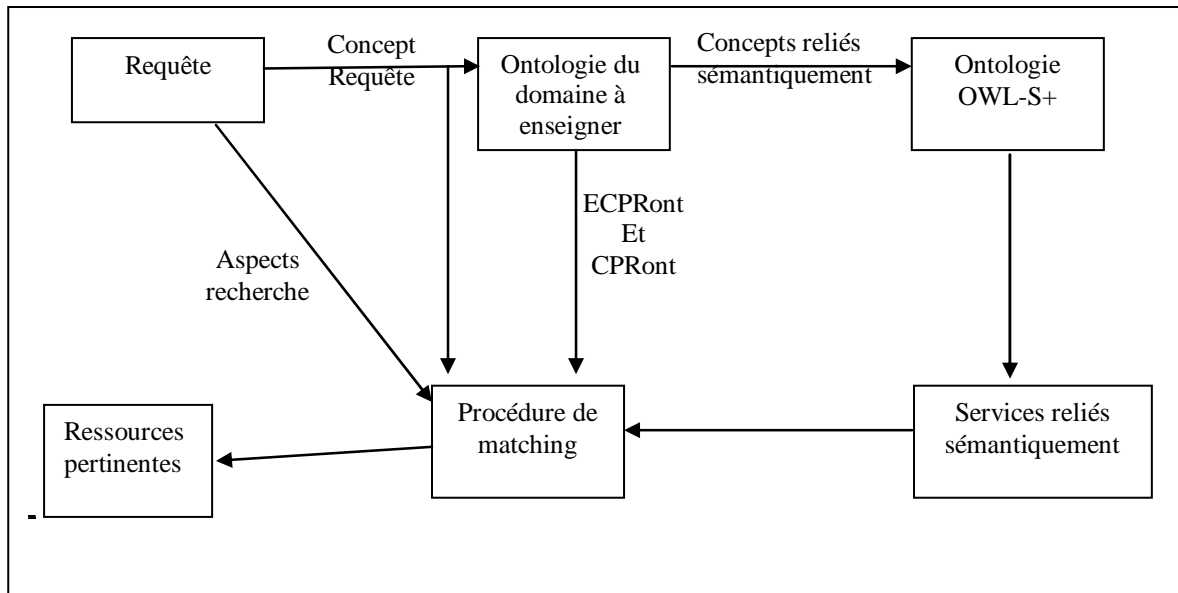


Figure 3.12: Les étapes du filtrage

- les éléments de connaissance et les concepts pré-requis du concept requête prédéfinis par les concepteurs de l'ontologie de e-learning (ECPont et CPont), sont utilisés comme des critères de matching des résultats de découverte, par exemple si une ressource qui enseigne un concept donné demande des éléments de connaissance ou des concepts pré-requis en plus de ceux jugés et définis par les experts donc cette ressource n'est pas utile pour l'apprenant. Ces deux critères sont utilisés pour le contrôle indirect de l'accès de l'apprenant aux ressources pédagogiques en lui donnant l'impression d'avoir un accès libre aux ressources tel que :

$$\text{Poids éléments connaissance pré-requis} = P_{elt-pr} = \begin{cases} 1 & \text{si } \{ECP_{service}\} \underline{C} \{ECP_{ont}\} \\ 0 & \text{si non} \end{cases}$$

$$\text{Poids concepts pré-requis} = P_{cpt-pr} = \begin{cases} 1 & \text{si } \{CP_{service}\} \underline{C} \{CP_{ont}\} \\ 0 & \text{si non} \end{cases}$$

- avec  $ECP_{service}$  est l'ensemble d'éléments de connaissance pré-requis définis par la description du service dans l'ontologie OWL-S+

- et  $CP_{service}$  est l'ensemble des concepts pré-requis définis par la description du service dans l'ontologie OWL-S+

On a remarqué que le poids et le degré de pertinence d'un service relié à un concept déduit à partir de l'ontologie du domaine dépend de :1) poids concept dans le service et 2) le poids de la relation sémantique qui le relie au concept introduit dans la requête, Sachant que les poids des relations sémantiques changent selon le profil de l'utilisateur, par exemple Pour un apprenant débutant on cherche à simplifier le concept puis à assurer un enchaînement d'acquisition adéquat, pour cela on définit l'ordre suivant :

Formé de  $\geq$  constitué de  $\geq$  enseigne  $\geq$  composé de  $\geq$  précédé de  $\geq$  dépend de  $\geq$  contribue à enseigner

Et pour un apprenant avancé on suppose qu'il maîtrise les concepts de base et on définit l'ordre des poids suivant :

Composé de  $\geq$  précédé par  $\geq$  dépend de  $\geq$  formé de  $\geq$  constitué de  $\geq$  enseigne  $\geq$  contribue à enseigner.

Et pour les formateurs et les concepteurs auteurs toutes les relations ont le même poids qui égal une constante  $K \in ]0, 1[$

on peut avoir les cas suivant :

- 1- un poids concept service considérable (important) et le poids de la relation faible
- 2- un poids concept service faible et poids relation important

Dans le but de donner la même priorité pour le poids concept service et poids relation dans le calcul du poids concept on utilise le produit des deux valeurs, et comme la valeur de poids relation  $\in ]0, 1[$  alors le poids concept service diminue

$$\text{Poids concept} = P_{cpt} = \begin{cases} P_{cpt \text{ service}} & \text{si concept} = \text{concept requête} \\ P_{cpt \text{ service}} * P_{relation} & \text{si concept est relié sémantiquement au concept} \end{cases}$$



Avec  $P_{cpt}$  service est le poids concept donné par la présentation ontologique de la ressource pédagogique.

Maintenant, il reste à définir le poids de la satisfaction de la ressource pédagogique des autres aspects de recherche comme : le type, l'acteur, les informations technique, les information financière et autre....

Donc le poids d'un service va être incrémenté à chaque fois que le service répond à un aspect recherché, d'où les définitions suivante :

$$\text{Poids d'un aspect de recherche} = P_{aspt} = \begin{cases} 1 & \text{si le service satisfait l'aspect} \\ & \text{recherché} \\ 0 & \text{si non} \end{cases}$$

Et :

$$\text{Poids aspects de recherche} = P_{aspt-rech} = \sum_{i=1}^n P_{aspt}$$

Avec  $n$  est le nombre des aspects de recherche sélectionnés par l'utilisateur.

D'après les définitions données en haut, on définit le poids d'un service comme suit :

$$\text{Poids Service} = P_{cpt} ( P_{elt-pr} + P_{cpt-pr} + P_{aspt-rech} )$$

Et la procédure du matching est la suivante :

**{services sélectionnés}** : ensemble des services pédagogiques reliés sémantiquement

**{Aspects}** : ensemble des aspects de recherche sélectionnés par l'utilisateur

**{Services pertinents}** : ensemble des services pédagogiques pertinents

**DEBUT**

**{Services pertinents} ← ∅ ;**

**Pour tous service<sub>i</sub> ∈ {services sélectionnés}**

**Paspt ← 0 ;**

**Pour tous aspect<sub>j</sub> ∈ {Aspects}**

**Si service satisfait l'aspect<sub>j</sub>**

**Paspt ← Paspt + 1**

**J ++ ;**

**Fin pour**

**ECPRont ← les éléments de connaissance pré-requis du concept définis dans l'ontologie du e-learning ;**

**Si ECPRont  $\subseteq$  ECPRservice alors**

**Pelt-pr = 1**

**Sinon Pelt-pr = 0 ;**

**Finsi**

**CPRont ← les concepts pré-requis du concept définis dans l'ontologie du e-learning ;**

**Si CPRont  $\subseteq$  CPRservice alors**

**Pcpt-pr = 1**

**Sinon Pcpt-pr = 0 ;**

**Finsi**

**Concept ← la valeur de la classe concept de la description ontologique du service sélectionné ;**

**Si concept  $\approx$  concept requête alors**

**Pcpt ← la valeur de la classe Poids concept**

**PoidsService<sub>i</sub> = Pcpt ( Pelt-pr + Pcpt-pr + Paspt )**

**Sinon Pcpt service ← la valeur de la classe Poids concept**

**Prelation ← le poids de la relation qui relie le concept du service avec le concept de la requête ;**

**Pcpt = Pcpt \* Prelation ;**

**PoidsService<sub>i</sub> = Pcpt ( Pelt-pr + Pcpt-pr + Paspt ) ;**

**Si PoidsService<sub>i</sub> ≠ 0 alors {Services pertinents} ← {Services pertinents} U {service<sub>i</sub>}**

**Finsi**

**i + + ;**

**Fin pour**

**Affichage des ressources pédagogiques, l'ensemble {Services pertinents} ; selon l'ordre décroissant de leurs poids**

**FIN.**

## **8 Conclusion**

Nous avons présenté dans ce chapitre la description et la modélisation de notre proposition qui consiste à développer une approche de découverte de services adaptés aux besoins et spécificités des usagers des utilisateurs des des plates formes d'apprentissage. Celle-ci est basée sur deux ontologies : une ontologie associée à la pédagogie liée aux ressources pédagogiques qui donne une description sémantique et pédagogique des services web associés aux ressources pédagogiques, et une ontologie du domaine à enseigner relative au domaine d'apprentissage, et un algorithme de matching pour le filtrage des résultats.

Le chapitre suivant comprend l'implémentation du système.

# *Chapitre 4 :*

## *La réalisation*

### **Sommaire**

1	Introduction .....	74
2	Technologies d'implémentation.....	74
2.1	Java .....	74
2.2	JSP et Servlet.....	75
2.3	Serveur Apache Tomcat.....	75
2.4	L'éditeur Protégé.....	76
2.5	API Jena.....	76
2.6	Registre UDDI.....	77
3	Architecture fonctionnelle du système.....	77
4	Développement des ontologies.....	79
4.1	Ontologie du domaine à enseigner.....	79
4.2	Ontologie des services web .....	79
5	Les interfaces du système.....	80
5.1	Interface de publication .....	81
5.2	Interface de découverte .....	83
6	Conclusion.....	84

## *Liste des Figures*

Figure 4.1 : Architecture fonctionnelle du Système.....	78
Figure 4.2 : Ontologie du domaine à enseigner.....	79
Figure 4.3 : Ontologie des services web.....	80
Figure 4.4 : Interface principale du système.....	81
Figure 4.5 : Interface de publication des services web.....	82
Figure 4.6 : Interface de découverte de service web.....	84

## 1 Introduction :

L'objectif de ce chapitre est de présenter les techniques, les langages et les outils utilisés pour la mise en œuvre de notre prototype.

Le développement de l'architecture proposée est fait en cinq étapes principales :

- Développement des deux ontologies proposées.
- Implémentation de la procédure de matching.
- Développement de l'interface de publication des descriptions sémantiques.
- Développement de l'interface de découverte de services Web.
- et enfin le test de vérification afin de s'assurer du fonctionnement cohérent du système.

## 2 Technologies d'implémentation

Pour l'implémentation et le dé déploiement de notre architecture nous avons opté pour les technologies et les outils suivants :

### 2.1 Java

Java est un langage de programmation orienté objet mis au point en 1991 par la firme SUN. Il est caractérisé par les points suivants :

- **Java est indépendant de toute plate-forme** : Une application développée en Java fonctionne dans n'importe quel environnement disposant d'une MVJ (Machine Virtuelle Java).
- **Java est un langage orienté objet** : Dans ce type de programmation, on ne manipule pas des fonctions et des procédures, mais des objets qui s'échangent des messages. Le principal avantage, outre le fait que l'on peut créer des objets de toutes natures représentant les véritables objets du problème à traiter, est que chaque objet peut être mis au point séparément.
- **Java est extensible à l'infini** : Java est écrit en Java. Idéalement, toutes les catégories d'objets (appelées classes) existant en Java sont définies par extension d'autres classes, en partant de la classe de base la plus générale : la classe Object. Pour étendre le langage, il suffit donc de développer de nouvelles classes. Ainsi, tous les composants écrits pour traiter un problème particulier peuvent être ajoutés au langage et utilisés pour résoudre de nouveaux problèmes comme s'il s'agissait d'objets standards.

- **Java est un langage à haute sécurité** : Java a été développé dans un souci de sécurité maximale. L'idée maîtresse est qu'un programme comportant des erreurs ne doit pas pouvoir être compilé. Ainsi, les erreurs ne risquent pas d'échapper au programmeur et de passer les procédures de tests. En détectant les erreurs à la source, on évite qu'elles se propagent en s'amplifiant.
- **Java est un langage compilé** : Java est un langage compilé, c'est-à-dire qu'avant d'être exécuté, il doit être traduit dans le langage de la machine sur laquelle il doit fonctionner.

## 2.2 JSP et Servlet

Nous avons choisi les technologies JSP et Servlets pour les raisons suivantes :

→ Une servlet (Server-Side applet) est un programme Java utilisé pour étendre des fonctionnalités d'un serveur web et pour accéder à des applications déjà existantes. C'est une application côté serveur utilisée pour générer du contenu dynamique. Elle est chargée dynamiquement quand elle est demandée et elle est exécutée dans le cadre d'un conteneur qui en pilote son cycle de vie et son contexte.

→ JSP (Java Server Pages) est une technologie pour le développement de pages Web incluant du contenu dynamique. Contrairement à une page HTML qui ne contient que du contenu statique qui reste par définition toujours le même, JSP peut changer selon l'identité du visiteur, de son navigateur Internet, de l'heure, de la configuration du système, des actions du visiteur, etc.

Une page JSP contient des balises standards, comme du HTML (ou du WML, XML...), comme toute page web normale. Pourtant, une page JSP contient aussi des éléments JSP spécifiques (scriptlets), permettant au serveur l'insertion dynamique de contenu (contenu de BDD, préférences du visiteur...)

Lorsqu'un utilisateur accède à une page JSP, le serveur exécute les éléments JSP, fusionne les résultats avec les parties statiques de la page, et envoie le tout au navigateur.

## 2.3 Serveur Apache Tomcat

Apache Tomcat est un conteneur libre de Servlet Java. Issu du projet Jakarta, Tomcat est désormais un projet principal de la fondation Apache. Tomcat implémente les spécifications des Servlets et des JSP de Sun Microsystems, il est aussi considéré comme un serveur HTTP (web).

Nous avons choisi Tomcat pour ces avantages qui sont les suivants :

- Apache Tomcat est un logiciel open source.
- Apache Tomcat est stable et sécurisé.
- Tomcat est simple, beaucoup plus que les serveurs d'application Open Source « complets »
- Il est donc plus simple d'administrer une instance Tomcat qu'un serveur d'applications complet.
- Il n'occupe que 2 ports sur la machine (8080 et 8009), alors que les autres en prennent une dizaine
- 8080 : port propre de Tomcat
- 8009 : port de communication entre Apache et Tomcat (protocole AJP13)

## **2.4 L'éditeur Protégé**

Plusieurs outils sont dédiés à la création d'ontologies, par exemple Ontoedit, OntoTerm, Oiled, WebOnto, OntoSaurus ou encore Ontolingua. Cependant, ces logiciels utilisent un formalisme spécifique et ne sont pas couplés à un système d'indexation et de recherche. Et aussi pour d'autres contraintes techniques nous avons trouvé que l'éditeur protégé développé par l'université de Stanford est le mieux adapté pour développer nos ontologies.

Protégé est un outil software intégré utilisé par les développeurs de système et les experts de domaine pour développer des systèmes à base de connaissances. Les applications développées par Protégé sont utilisées pour résoudre des problèmes et prendre des décisions dans un domaine bien précis.

Protégé est actuellement utilisé dans la médecine clinique et les sciences biomédicales. Il peut être utilisé dans tout champ où les concepts peuvent être modélisés comme une hiérarchie de classes.

Le grand avantage de Protégé est son extensibilité, de fait qu'il fournit au programmeur d'applications un ensemble d'API Java pour étendre son système en fonction des besoins d'applications des utilisateurs. Cet API fournit des packages et des classes Java qui réalisent des opérations complexes telles que créer de nouveaux Widgets.

## **2.5 API Jena**

Pour permettre à notre système de manipuler les ontologies, il faut utiliser une bibliothèque qui fournit la capacité de traitement des ontologies. Les ontologies de notre application sont réalisées en langage OWL. Les APIs pour les manipuler dépendent du langage de programmation utilisé. Pour notre cas, le langage de programmation utilisé est Java. Parmi les



bibliothèques pour le traitement des ontologies Jena. On a choisi le package Jena qui est un projet de code source libre développé chez HP pour le Web sémantique. Ce package nous offre beaucoup d'avantages :

- Il supporte divers langages : DAML+OIL, OWL, etc.
- Il permet de charger un document d'ontologie d'un URL donné ou d'un fichier local.
- Il est capable de détecter et charger optionnellement les ontologies qui sont incluses dans d'autres ontologies.
- Il permet de créer ou mettre à jour des ontologies.
- Comme toutes les informations sont stockées dans un modèle Jena, il nous permet d'avoir un accès uniforme pour divers langages d'ontologies (DAML+OIL, OWL, etc.).
- Les classes associées aux composants du vocabulaire de l'ontologie (classe, propriété, liste, etc.) permettent aux programmes d'accéder directement aux divers éléments de l'ontologie.
- Il permet de naviguer récursivement sur la hiérarchie des classes de l'ontologie.
- Il implémente RDQL, un langage de requête qui est semblable au langage SQL.
- Il supporte le processus d'inférence en fournissant les APIs d'inférence, les raisonneurs RDF, OWL, etc. et nous permet de définir nos propres règles d'inférence.

## 2.6 Registre UDDI

Pour sauvegarder les descriptions des services web dans un registre de publication nous avons utilisé JUDDI et l'API java UDDI4J pour exploiter les données de ce dernier.

JUDDI est un registre open source implémenté en Java ce qui le donne une indépendance aux plates formes et il peut être utilisé avec n'importe quelle base de donnée relationnelle supportant le standard ANSI SQL comme par exemple : MySQL, JDataStore, etc.

## 3 Architecture fonctionnelle du système

Nous avons réalisé notre prototype en utilisant JDK1.6, l'IDE Eclipse 2.3, le serveur Apache Tomcat5.5 comme conteneur des JSP et Servlets, l'annuaire JUDDI 0.9rc4 pour le stockage des descriptions WSDL des services web et l'API Jena pour permettre à notre application de manipuler les fichiers OWL implémentés à l'aide de l'éditeur Protégé 3.0.

Le schéma ci-dessous montre l'architecture du système

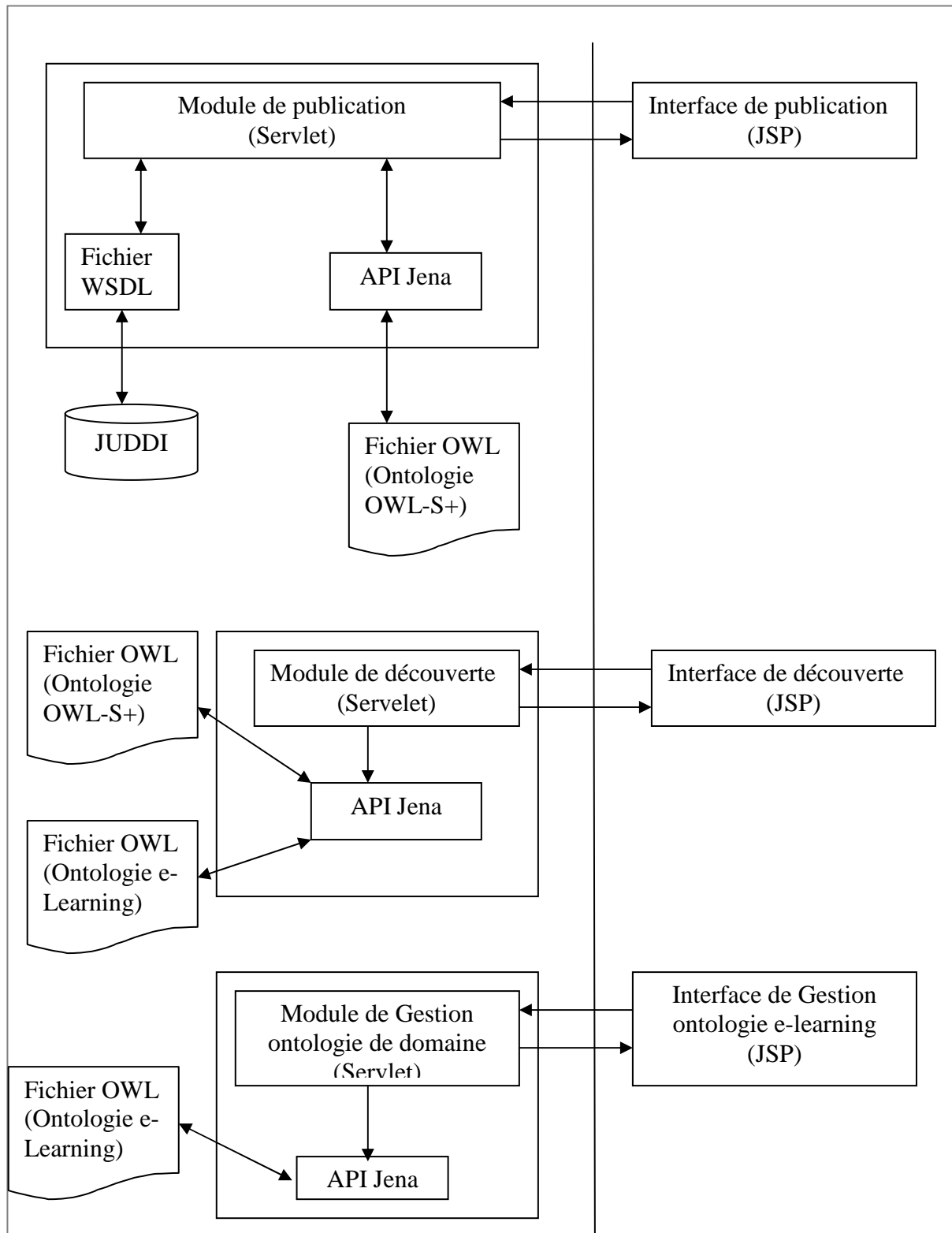


Figure 4.1 : Architecture fonctionnelle du Système

## 4 Développement des ontologies

Les deux ontologies proposées sont basées sur le langage OWL.

### 4.1 Ontologie du domaine à enseigner :

Cette ontologie permet d'organiser les concepts d'un domaine à enseigner spécifique selon une décomposition générique et réutilisable.

La classe principale de cette ontologie est la classe concept générique, c'est une classe abstraite qui permet d'organiser les autres classes définies.

La figure suivante montre cette ontologie :

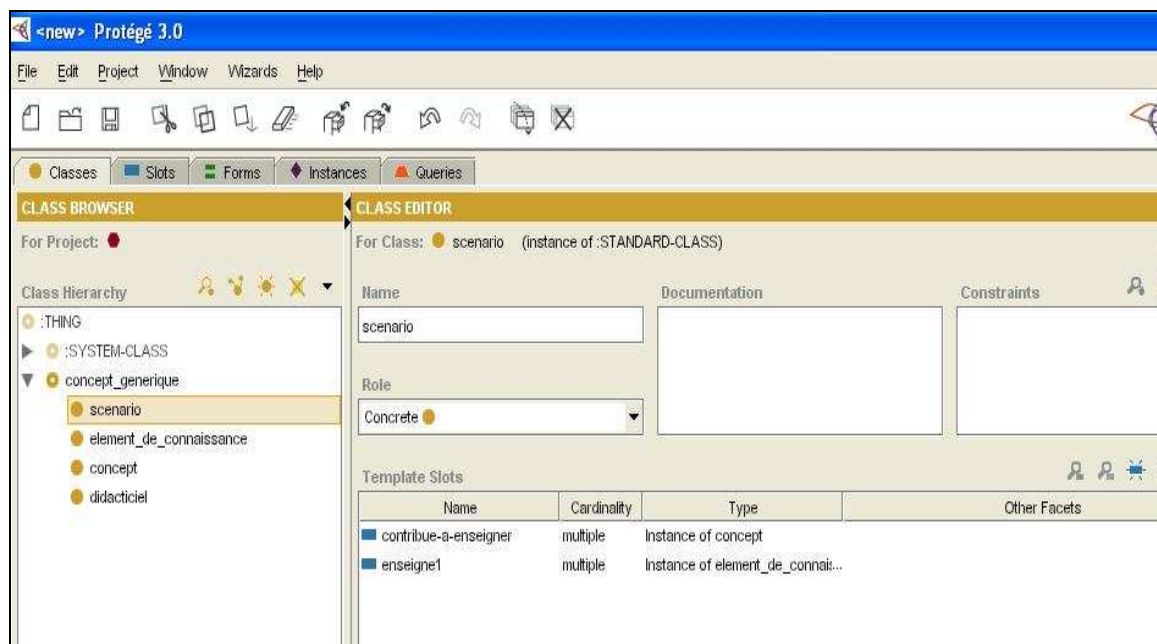


Figure 4.2 : Ontologie du domaine à enseigner

### 4.2 Ontologie des services web :

Pour la conception de l'ontologie des services web nous avons démarré à partir d'une ontologie qui existe déjà OWL-S. Cette ontologie englobe les sous ontologies : ServiceProfil, serviceGrounding et serviceModel.

Notre travail consistait à l'extension du ServiceProfil par une sous ontologie pour la description pédagogique, technique et financière du service web.

La figure suivante est une représentation de cette ontologie.

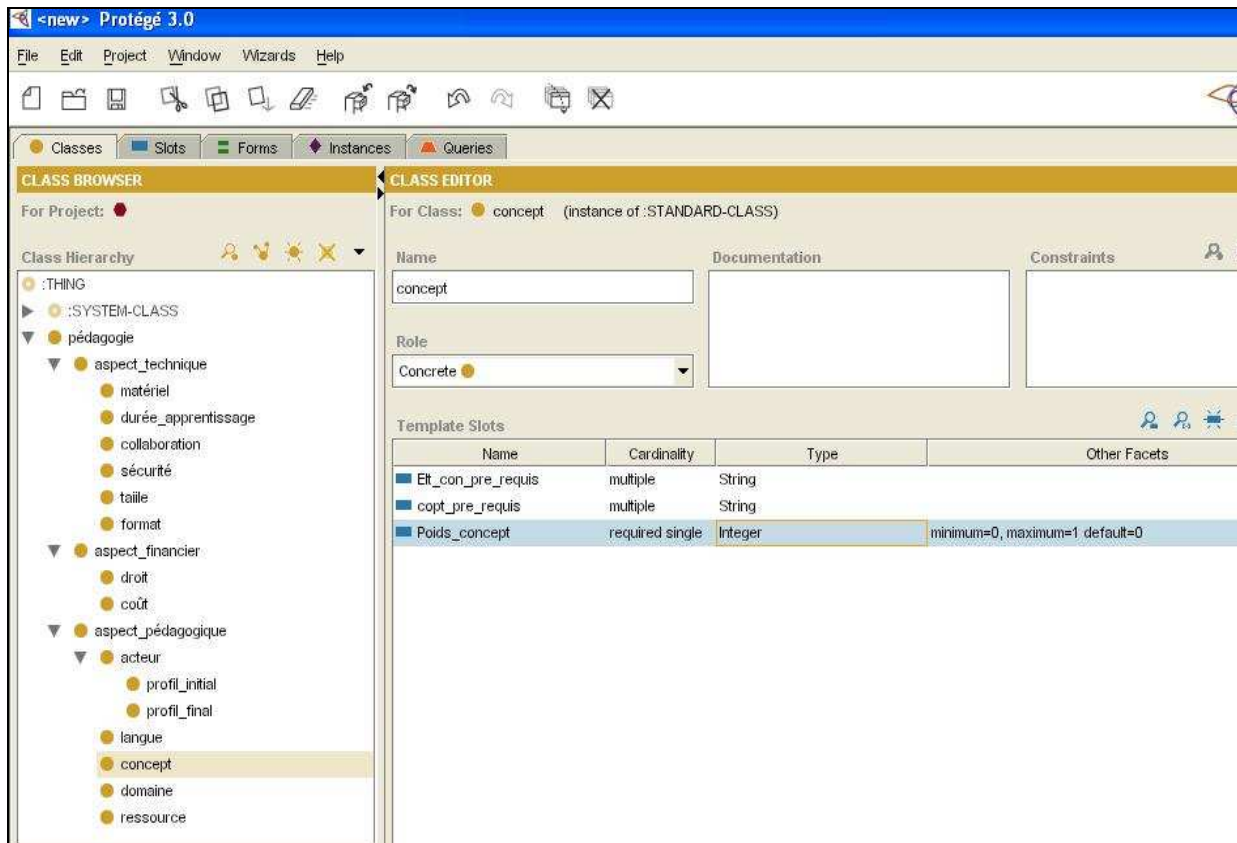


Figure 4.3 : Ontologie des services web

Le code source des deux ontologies développées est en annexe (A) et (B) de ce mémoire.

## 5 Les interfaces du système

L'interface principale de notre application permet aux utilisateurs de s'authentifier afin d'avoir un accès aux différentes fonctionnalités du système comme la publication des services, la découverte et la gestion de l'ontologie du domaine à enseigner.



Figure 4.4 : Interface principale du système.

### 5.1 Interface de publication :

La publication de services web dans notre prototype permet aux fournisseurs du service d'introduire des informations concernant le service. Deux interfaces sont offertes pour cela :

→ La première interface permet au système d'importer et de récupérer les deux descriptions (fichiers) WSDL et OWL-S classique du service à publier à partir d'une URL ou d'un emplacement local.

→ Après validation une deuxième interface apparaîtra.

Cette interface permet d'introduire les informations et les données concernant la pédagogie du service et autres informations jugées importantes pour la publication d'une ressource pédagogique. Après validation, ces informations sont utilisées pour l'instanciation des classes de l'ontologie OWL-S+ et le système ajoute une entrée tModel pour sauvegarder le fichier WSDL et ajoute une entrée serviceEntity pour le service dans l'annuaire des services.

Accueil | A propos de nous | Aide

Outil pour la publication et la découverte des ressources pédagogiques

**Publication d'une ressource pédagogique**

Description WSDL:

Description OWL-S:

**Description de services**  
 à l'enseignant  
 de publier son  
 s forme de services  
 à l'administrateur  
 re de publier les  
 des formations en  
 e forme de services

**re de services**  
 aux enseignants  
 de consulter ou  
 des ressources  
 es; et aux  
 de trouver des  
 es ressources

**Aspect pédagogique:**

Nom domaine:

Description domaine:

Titre:

Langue:  
 ▼

Concept:

Poids concept:

Éléments connaissance concept:

Concept pré-requis:

Élément pré-requis:

Type:  
 ▼

Acteur:  
 ▼

Figure 4.5 : Interface de publication des services web.

## 5.2 Interface de découverte :

La recherche des ressources pédagogiques déjà publiées sous forme de services web peut être simple ou avancée selon le choix du client.

→ Dans la recherche simple l'utilisateur introduit le *concept* et lance sa recherche. Le système récupère et il se connecte à l'ontologie OWL-S+ pour avoir les instances qui enseignent ce *concept* puis il effectue un filtrage selon l'algorithme de filtrage détaillé dans la partie conception.

→ Dans la recherche avancée deux cas se présentent :

-recherche avec prise en compte des liens sémantiques donc l'utilisation de l'ontologie du domaine à enseigner.

-recherche avec prise en compte des aspects de recherche.

Selon le choix de l'utilisateur le système récupère les préférences de l'utilisateur et fait une comparaison entre les aspects de recherche introduits et les instances des classes de l'ontologie OWL-S+ puis effectue le filtrage.

Figure 4.6 : Interface de découverte de service web

## **6 Conclusion**

Ce chapitre décrit l'implémentation de notre prototype pour la publication et la découverte des services web. Nous avons présenté à travers l'architecture fonctionnelle du système, les technologies et les outils que nous avons utilisés pour développer notre application, puis les deux ontologies utilisées lors de la découverte. Enfin nous avons donné les interfaces principales de notre système.



## Conclusion générale

La technologie des services web a apporté plusieurs avantages au domaine de l'e-learning comme l'intégration du contenu dans les systèmes d'apprentissage en tant que service ce qui permet une maintenance plus facile puisque il est stocké une fois sur le serveur du fournisseur et les apprenants se connectent exactement au même contenu. Les mises à jour de ce contenu sont effectuées une seule fois, et une distribution physique du contenu à tous les LMS, qui l'ont importé, n'est pas nécessaire, la configuration des plates formes en fonction des besoins utilisateurs et leurs fonctionnalités peuvent être réutilisés dans différentes plates-formes.

Ce qui justifie la croissance continue du nombre des services e-learning, y compris les ressources pédagogiques, sur le web d'où la nécessité de l'automatisation de la recherche.

La découverte de services web se réfère à la possibilité de localiser automatiquement un Web service qui répond à des besoins particuliers des utilisateurs. Différentes approches ont été proposées dans la littérature pour réaliser la découverte dynamique de services.

Les premières approches de découverte de services proposées dans la littérature étaient des approches syntaxiques. Avec l'émergence de la nouvelle génération du web, des approches sémantiques ont vu le jour. Celles-ci présentent une description sémantique de services web interprétables par une application dans le but d'assurer un degré d'automatisation élevé.

Dans ce papier est proposé le cadre conceptuel et architectural d'une approche pour la publication et la découverte de services web liés aux ressources pédagogiques ainsi que la mise en œuvre d'un système prototype. L'approche développée est basée sur deux ontologies : l'ontologie du domaine à enseigner et l'ontologie OWL-S+ constituée de l'ontologie OWL-S et de l'ontologie associée à la pédagogie liée aux ressources pédagogiques. L'usage de ces ontologies permet d'améliorer le processus de découverte de services web liés aux ressources pédagogiques en sélectionnant ceux qui répondent le mieux aux besoins et spécificités des différents usagers grâce à la procédure de matching proposée pour le filtrage de services. Nos travaux futurs consistent en l'implémentation de notre approche dans une plate forme d'e-learning, ainsi que la mise en œuvre de principes permettant de localiser et comparer des services dans la perspective de sélectionner des services entrant dans la composition d'un autre.

## Références

1. Addour D., Ahmed-Ouamer R.: Vers une Plate Forme pour la Découverte de Services Web Liés aux Ressources Pédagogiques. Rencontres sur la Recherche en Informatique R2I, Tizi-Ouzou, 12-14 Juin (2011) 98-105.
2. ADL(Advanced Distributed Learning). SCORM 2004 Overview. ADL, <http://www.adlnet.org/>,(2004).
3. Ahmed-Ouamer R., Hammache A. : Un Système de Recherche d'Information pour l'e-Learning. Revue Document Numérique, Hermès Lavoisier, vol. 11 n° 1-2, (2008) 85–105.
4. Ahmed-Ouamer R., Hammache A.: Ontology-Based Information Retrieval for e-Learning of Computer Science. The First International Conference on Machine and Web Intelligence IEEE ICMWI 2010. IEEE, ISBN: 978-1-4244-8610-6, Algiers, Algeria, October 03-05 (2010) 229-236.
5. Ahmed-Ouamer, R. développement de systèmes d'EIAO dans AGEDI. Séminaire national d'informatique : SNITO 96, Tizi-Ouzou, Novembre (1996).
6. Baget J-F., Canaud E., Euzenat J. , Hacid M.S. : Les Langages du Web Sémantique. Revue I3, Information, interaction, intelligence, Numéro spécial, (2004).
7. Bechhofer S., Van Harmelen F., Hendler J., Horrocks I., McGuinness D.L., Patel-Schneider P.F., Stein L.A. : OWL Web Ontology Language Reference. W3C Recommendation 10 02 (2004) <http://www.w3.org/TR/owl-ref/>
8. Bellwood T., Clément L., Riegen C.: UDDI Spec Technical Committee Specification (2002) [www.uddi.org](http://www.uddi.org)
9. Bouthry A. : Construire son Projet de Formation en Ligne. Editions d'Organisation. (2003)
10. Brusilovsky P.: KnowledgeTree a Distributed Architecture for Adaptive e-Learning. Thirteenth International World Wide Web Conference, New York (2004) 104-113.
11. Charlet J., Bachimont B., Troncy R. : Ontologies pour le Web Sémantique. Revue I3, Information, interaction, intelligence, Numéro spécial, (2004)
12. Crozat S., Majada M., Spinelli S. : SCENARISup : Un projet pour la gestion des contenus pédagogiques numériques dans l'enseignement supérieur. CNUR'2003, Montpellier : France (2003). <http://edutice.archives-ouvertes.fr/edutice-00000389/fr/>
13. CSC (Computer Sciences Corporation):LES LEARNING MANAGEMENT SYSTEMS (LMS) ;Solutions logicielles de gestion de la formation Etude annuelle 2002/2003 [www.csc.fr](http://www.csc.fr)
14. DAML-S: Semantic Markup of Web Services. The DAML Services Coalition. (2003)
15. Ghurburn R. : Introduction aux Web Services. Master Web Intelligence, (2005)

16. Godray D., Sterling S., "the elements of cal", Reston Publishing Cy, Reston Va, (1982) p.23-52.
17. Grigori D., Corrales J.C, Bouzeghoub M.: Behavioral matchmaking for service retrieval: Application to conversation protocols. *Inf. Syst.* 33(7-8): 681-698. (2008)
18. Gruber T.:A Translation Approach to Portable Ontology Specifications.*Knowledge Acquisition*, 5(2): pp. 199-220. (1993)
19. Higgs P., Meredith S., Hand T.: Technology for Sharing -A research project to inform VET Australia about Learning Objects and Digital Rights Management including systems and metadata to support them. Flexible Learning Leader 2002 Report, Australian National Training Authority (ANTA). (2003)
20. <http://www.w3.org/TR/owl-guide/>
21. <http://dublincore.org/>
22. Kadima H., Monfort V. : Les Web Services : Techniques, Démarches et Outils XML, WSDL, SOAP, UDDI, RosettaNet, UML. Dunod (2003)
23. Kellert P., Toumani F. : Les Web Services Sémantiques. *Revue I3, Information, interaction, intelligence*, Numéro spécial, (2004)
24. Labat, j-m.quiz : Une Contribution A L'amélioration Des Capacités Pédagogiques Des Tuteurs Intelligents. Thèse d'université. Institut Blaise Pascal, Paris, (1990).
25. Lassila O., Swick R. : Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation 22 February (1999). <http://www.w3.org/TR/REC-rdfsyntax>
26. Li Y., Scott W., Adam C., Lorna M C.: The Future of Interoperability and Standards in Education –System and Process <http://jisc.cetis.ac.uk/publications> (2010)
27. Ministère de la Jeunesse, de l'Éducation : Étude des outils de gestion de ressources. Conseil Business Interactif, 29 Juillet 2003
28. Neeches R., Finin T., Gruber T. R., Senator T., Swartout W. R.: Enabling technology for knowledge sharing. *AI Magazine*, 12, 35-56. (1991).
29. Newcomer E.: Understanding Web Services- XML, WSDL, SOAP and UDDI. chapter 5. Finding Web Services : UDDI Registry. Addison Wesley Professional.
30. Nicaud J.F., Vivet M. : les Tuteurs Intelligents : Réalisation Et Tendances De Recherches. TSI, numéro spécial : Application De L'informatique A La Formation, Dunod-Afcet, 1988, Vol.7, n°1, p.21-45.
31. OWL-S Coalition: OWL-S – Semantic Markup For Web Services. Darpa, (2004). <http://www.daml.org/services/owl-s/>

32. Palathingal P., Chandra.S: Agent approach for service discovery and utilization. In HICSS, 2004.
33. Psyché, V., Mendes, O., Bourdeau, J. : Apport de l'Ingénierie Ontologique aux Environnements de Formation à Distance, Revue des Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation STICEF Volume 10, (2003) 89-126.
34. Rehak, D. (ADL) Standards and SCORM: Maturity, Adoption, Process, online: <http://wiki.cetis.ac.uk/images/6/61/Adl.pdf>, (2010).
35. Sampson D. G., Lytras M. D., Wagner G., Diaz, P.: Ontologies and the Semantic Web for e-Learning. Educational Technology & Society, 7 (4), (2004) 26-28.
36. Shuping Ran: A Model for Web Services Discovery With QoS .CSIRO Mathematical and Information Sciences, Canberra, ACT 2601,Australia 2003.
37. Suwanmanee S., Benslimane D., Thiran P.: OWL-Based Approach for Semantic Interoperability. 19th Int. Conf. Adv. Info. Networking and App.: AINA'05. IEEE (2005)
38. Vossen G., Jaeschke P.: Learning Objects as a Uniform Foundation for e-Learning Platforms. 7th International Conference on Database Engineering and Applications IDEAS, Hong Kong, China (2003) 278 – 287.
39. Vossen G., Westerkamp P.: Why Service-Oriented Architecture Could Make e-Learning Standards Obsolete. International Journal of Technology Enhanced Learning, Volume 1, Number 1-2, (2008) 85 – 97.
40. WWW.W3C.ORG
41. Xiaohong Qiu, Anumit Jooloor : Web Service Architecture for e-Learning. Journal of Systemics, Cybernetics and Informatics, Volume 3, Number 5, (2005) 92-101.
42. Yannick P., Serge G., : Méta-données et annotations dans le Web sémantique

# **Annexes**

## Annexe A :

### Ontologie associée à la pédagogie

Cette ontologie est une extension du ServiceProfile de l'ontologie OWL-S.

```
<owl:Class rdf:ID="pedagogie"/>
<owl:Class rdf:ID="langue">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="aspet_pedagogique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="ressource">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#aspet_pedagogique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="qualite_apprentissage">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#aspet_pedagogique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="domaine">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#aspet_pedagogique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="role">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="acteur"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="concept">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#aspet_pedagogique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="collaboration">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="aspect_technique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="securite">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#aspect_technique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="taille">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#aspect_technique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#aspect_technique">
  <rdfs:subClassOf rdf:resource="#pedagogie"/>
</owl:Class>
<owl:Class rdf:ID="droit">
```

```

    <rdfs:subClassOf>
      <owl:Class rdf:ID="aspect_financier"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#aspet_pedagogique">
    <rdfs:subClassOf rdf:resource="#pedagogie"/>
  </owl:Class>
  <owl:Class rdf:about="#acteur">
    <rdfs:subClassOf rdf:resource="#aspet_pedagogique"/>
  </owl:Class>
  <owl:Class rdf:ID="format">
    <rdfs:subClassOf rdf:resource="#aspect_technique"/>
  </owl:Class>
  <owl:Class rdf:about="#aspect_financier">
    <rdfs:subClassOf rdf:resource="#pedagogie"/>
  </owl:Class>
  <owl:Class rdf:ID="materiel">
    <rdfs:subClassOf rdf:resource="#aspect_technique"/>
  </owl:Class>
  <owl:Class rdf:ID="cout">
    <rdfs:subClassOf rdf:resource="#aspect_financier"/>
  </owl:Class>
  <owl:Class rdf:ID="profil_final">
    <rdfs:subClassOf rdf:resource="#acteur"/>
  </owl:Class>
  <owl:Class rdf:ID="profil_initial">
    <rdfs:subClassOf rdf:resource="#acteur"/>
  </owl:Class>
  <owl:DatatypeProperty rdf:ID="capacite_materiel">
    <rdfs:domain rdf:resource="#materiel"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="titre_ressource">
    <rdfs:domain rdf:resource="#ressource"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="elt_con_pre_requis">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#concept"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="droit_acce_ressource">
    <rdfs:domain rdf:resource="#droit"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="elt_con_copt">
    <rdfs:domain rdf:resource="#concept"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="copt_pre_requis">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#concept"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="description_materiel">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>

```

```

    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="#materiel"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="authentification">
    <rdfs:domain rdf:resource="#securite"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="concept_profil_final">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#profil_final"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="concept-profil_initial">
    <rdfs:domain rdf:resource="#profil_initial"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="taille-ressource">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#format"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="duree_apprentissage">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="#qualite_apprentissage"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="description_domaine">
    <rdfs:domain rdf:resource="#domaine"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="synchronisation">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <rdfs:domain rdf:resource="#collaboration"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="date_creation_ressource">
    <rdfs:domain rdf:resource="#ressource"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="format_ressource">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="#format"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="elt_con_profil_initial">
    <rdfs:domain rdf:resource="#profil_initial"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="elt-con_profil_final">
    <rdfs:domain rdf:resource="#profil_final"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>

```



```

</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="description_ressource">
  <rdfs:domain rdf:resource="#ressource"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="nom_langue">
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#langue"/>
</owl:DatatypeProperty>
<owl:FunctionalProperty rdf:ID="designation_materiel">
  <rdfs:domain rdf:resource="#materiel"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="mise_a_jour">
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:domain rdf:resource="#format"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="difficulte">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  <rdfs:domain rdf:resource="#qualite_apprentissage"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="confidentialite">
  <rdfs:domain rdf:resource="#securite"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="controle_d_acces">
  <rdfs:domain rdf:resource="#securite"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="date_maj_ressource">
  <rdfs:domain rdf:resource="#ressource"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="role_acteur_ressource">
  <rdfs:domain rdf:resource="#role"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="description_outil_collaboration">
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#collaboration"/>
</owl:FunctionalProperty>

```

```
<owl:FunctionalProperty rdf:ID="type_ressource">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain rdf:resource="#ressource"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="nom_domaine">
  <rdfs:domain rdf:resource="#domaine"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="cout_ressource">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
  <rdfs:domain rdf:resource="#cout"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="poids_concept">
  <rdfs:domain rdf:resource="#concept"/>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
</owl:FunctionalProperty>
</rdf:RDF>
```

## Annexe B :

### Ontologie du domaine à enseigner

```
</owl:Ontology>
<owl:Class rdf:ID="concept">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:minCardinality
rdf:datatype="http://www.w3.org/2001/XMLSchema#int "
      >1</owl:minCardinality>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="forme-de"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class rdf:ID="concept_generique"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#concept_generique">
  <protege:abstract>true</protege:abstract>
</owl:Class>
<owl:Class rdf:ID="element_de_connaissance">
  <rdfs:subClassOf rdf:resource="#concept_generique"/>
</owl:Class>
<owl:Class rdf:ID="didacticiel">
  <rdfs:subClassOf rdf:resource="#concept_generique"/>
</owl:Class>
<owl:Class rdf:ID="scenario">
  <rdfs:subClassOf rdf:resource="#concept_generique"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="precede_par2">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="precede2"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#element_de_connaissance"/>
  <rdfs:range rdf:resource="#element_de_connaissance"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#forme-de">
  <rdfs:domain>
  </rdfs:domain>
  <rdfs:range rdf:resource="#element_de_connaissance"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="precede-par">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="precede"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#concept"/>
  <rdfs:range rdf:resource="#concept"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="forme-de2">
  <rdfs:domain rdf:resource="#didacticiel"/>
  <rdfs:range rdf:resource="#scenario"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="enseigner">
  <rdfs:domain rdf:resource="#scenario"/>
  <rdfs:range rdf:resource="#element_de_connaissance"/>
```

```
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#precede">
  <rdfs:domain rdf:resource="#concept"/>
  <rdfs:range rdf:resource="#concept"/>
  <owl:inverseOf rdf:resource="#precede-par"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#precede2">
  <owl:inverseOf rdf:resource="#precede_par2"/>
  <rdfs:range rdf:resource="#element_de_connaissance"/>
  <rdfs:domain rdf:resource="#element_de_connaissance"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="depend_de2">
  <rdfs:range rdf:resource="#concept"/>
  <rdfs:domain rdf:resource="#concept"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="contribue-a-enseigner">
  <rdfs:range rdf:resource="#concept"/>
  <rdfs:domain rdf:resource="#scenario"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="depend_de">
  <rdfs:domain rdf:resource="#element_de_connaissance"/>
  <rdfs:range rdf:resource="#element_de_connaissance"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="enseigne2">
  <rdfs:domain rdf:resource="#didacticiel"/>
  <rdfs:range rdf:resource="#concept"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="precede_par">
  <rdfs:domain rdf:resource="#scenario"/>
  <rdfs:range rdf:resource="#scenario"/>
</owl:ObjectProperty>
</rdf:RDF>
```