

UNIVERSITE MOHAMED KHIDER DE BISKRA

**Faculté des sciences exactes et
sciences de la nature et de la vie**



Département d'informatique

N°d'ordre :.....

N°de Série :.....

Thèse

En vue de l'obtention du diplôme de

Doctorat en Informatique

(Option: Intelligence artificielle et systèmes distribués)

Présentée par:

BOUREKKACHE Samir

THEME

*Un environnement sémantique à base d'agents
pour la formation à distance (E-Learning)*

Soutenue devant le jury

Président :	Mahmoud Boufaida	Professeur	Université de Constantine 2.
Rapporteur :	Kazar Okba	Professeur	Université de Biskra.
Examineur :	Mohamed Ahmed Nacer	Professeur	Université USTHB.
	Badache Nadjib	Professeur	Université USTHB.
	Terissa Labib Sadok	MCA	Université de Biskra.
	Bennoui Hammadi	MCA	Université de Biskra.
Invitée :	BenHarkat Nabila	MCA	INSA De Lyon

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

au nom du dieu le méséricordieu le clément

In the Name of Allah the Merciful, the Compassionate

Remerciements

Avant, toute personne, je tiens à remercier notre Dieu Tout Puissant pour m'avoir éclairci le chemin de ce travail.

*Mes vifs remerciements vont également à Monsieur **Okba KAZAR** professeur à l'université de Biskra d'avoir assuré l'encadrement de cette thèse ainsi que pour ses précieux conseils et la confiance qu'il m'a accordée qui ont fortement contribué à mener à bien ce travail.*

*Je tiens à remercier vivement Monsieur **Mahmoud Boufaïda**, Professeur à l'université de Constantine 2, pour m'avoir fait l'honneur de présider le jury de ma soutenance.*

*Je tiens à exprimer toute ma gratitude à Monsieur **Mohamed Ahmed Nacer**, Professeur à l'université USTHB, pour avoir bien voulu juger le travail.*

*Je tiens aussi à exprimer toute ma gratitude à Monsieur **Badache Nadjib**, Professeur à l'université USTHB pour avoir bien voulu juger le travail et faisant partie de jury de soutenance.*

*Je tiens à exprimer toute ma gratitude à Monsieur **Terissa Labib Sadok**, Maître de conférences A à l'université de Biskra, d'avoir accepté d'examiner mon travail.*

*Je tiens à remercier vivement Monsieur **Bennoui Hammadi**, Maître de conférences A à l'université de Biskra, pour avoir bien voulu juger le travail et faisant partie de jury de soutenance.*

*Je dois aussi remercier vivement les membres du laboratoire **LIRIS de L'INSA de Lyon** pour l'accueil et l'aide précieuses, je cite en particulier docteur Mme **Nabila Benharkat** et le professeur **Youcef Amghar** Chef de département d'informatique.*

Enfin, Un grand merci s'adresse à ceux qui m'ont aidé de près ou de loin.

Dédicaces

Je dédie ce modeste travail:

À ceux que j'ai de plus cher au monde : mes parents;

À ma femme et ma petite fille Anfel

À mes frères et mes sœurs et mes neveux.

À tous mes amis et collègues.

À tous ceux qui m'ont aimé et me souhaitent le bonheur et la réussite.

Résumé: Aujourd'hui, les établissements d'enseignement, tels que les universités, de plus en plus offrent des contenus d'E-Learning. Certains de ces cours sont utilisés avec l'enseignement traditionnel (face à face ou présentiel), tandis que d'autres sont utilisés entièrement en ligne. La création de contenu d'apprentissage est une tâche principale dans tous les environnements d'apprentissage en ligne. Les contraintes de réduire au minimum le temps nécessaire pour développer un contenu d'apprentissage, d'augmenter sa qualité scientifique et de l'adapter à de nombreuses situations (contenu adaptatif), ont été un principal objectif et donc plusieurs approches et méthodes ont été proposées. En outre, les caractéristiques intellectuelles et sociales, ainsi que les styles d'apprentissage des individus, peuvent être très différents. Ces différences conduisent les personnes à adapter le contenu d'apprentissage en tenant compte des profils des apprenants et de leurs objectifs et caractéristiques. Cette recherche ouvre des portes pour les systèmes d'apprentissage avancés, qui fournissent aux apprenants immédiatement, des contenus d'apprentissage adaptés selon plusieurs critères de chaque apprenant. Alors que, il ne peut pas être pratique si nous n'avons pas plus d'informations sur l'apprenant et le contenu d'apprentissage (objectifs d'apprentissage, les prérequis, préférences, niveaux ...etc). Par conséquent, nous développons un système collaboratif, où plusieurs auteurs travaillent en collaboration, pour créer et annoter le contenu éducatif en utilisant le système multi-agents. La contribution de notre système est l'hybridation des techniques d'adaptation avec celles de la collaboration et du Web sémantique (ontologie, annotation). Nous représentons les profils des apprenants et le contenu d'apprentissage en utilisant des ontologies et des annotations pour répondre à la diversité et aux besoins individuelles des apprenants. Nous utilisons le paradigme agent, dans notre système, pour bénéficier des points forts de ce paradigme tels que la modularité, autonomie, flexibilité... etc.

Mots-clés: E-learning, Système collaboratif; système multi-agent, Web sémantique, Métadonnées, Ontologie, Contenu éducatif, Annotation, Profils des apprenants, Styles d'apprentissage, système adaptatif.

Abstract: Nowadays, educational institutions, such as universities, more and more offer E-Learning contents. Some of these courses are blended with traditional education, while others are conducted completely online. The creation of learning content is a main task in every E-learning environment. The constraints of minimizing the time required for developing a learning content, for increasing its scientific quality and to adapt it in many situations (adaptive content), have been a principal aim and so several approaches and methods were proposed. Moreover, the intellectual and social characteristics, as well as the learning styles of individuals, can be very different. These differences lead persons to adapt the learning content by taking into account the learners profiles and their objectives. This research opens ways for advanced learning systems, which are able to learn the needs and characteristics of learners, respond to them immediately, and provide learners with learning content where adaptation is frequently improved and updated to the learners' needs. So that, it may not be convenient if we don't have additional information about the learner and the learning content (learning objective, prerequisites, learner background, levels ... etc.). Therefore, we develop a collaborative system, where several authors work in a collaborative manner, to create and annotate educational materials using multi-agents system. The contribution of our system is the hybridization of adaptation techniques with those of collaboration and Semantic Web (ontology, annotation). We represent the learners' profiles and the learning content using ontologies and annotations to meet the diversity and individual needs of the learners. We use the paradigm Agent in our system to benefit from the strong points of this paradigm such as modularity, autonomy, flexibility ... etc.

Key words: E-learning; Collaborative system, Multi-agent system, Semantic Web, Metadata, Ontology, Learning Content, Annotation, learners' Profiles, learning styles, Adaptive system.

ملخص: في الوقت الحاضر، المؤسسات التعليمية مثل الجامعات تزودنا بالكثير من محتويات التعليم الإلكتروني. يتم مزج بعضها مع التعليم التقليدي، في حين تستعمل محتويات تعليمية أخرى كلياً على الإنترنت. إن إنشاء محتوى التعلم هو المهمة الرئيسية في كل وسط خاص بالتعلم الإلكتروني. تعتبر قيود التقليل من الوقت اللازم لتطوير محتوى التعلم، وزيادة الجودة العلمية و تكيفها مع الكثير من الحالات الشخصية للمتعلمين (المحتوى التكيفي) أهم الأهداف الرئيسية. وقد اقترحت عدة أنظمة وأساليب من أجل ذلك. وعلاوة على ذلك، الخصائص الفكرية والاجتماعية، فضلاً عن أساليب التعلم للأفراد، يمكن أن تكون مختلفة جداً. هذه الاختلافات تؤدي إلى تكيف محتوى التعلم مع مختلف المتعلمين من خلال الأخذ بعين الاعتبار خصائص المتعلمين وأهدافهم. يفتح هذا البحث السبل لنظم التعلم المتقدمة، والتي هي قادرة على تحديد احتياجات و خصائص المتعلمين، والاستجابة لها على الفور، وتوفير المحتوى التعليمي المكيف لكل متعلم وذلك يتم بتحسين نظام التكيف وتحديثها مع احتياجات المتعلمين. لكن، قد لا يكون سهلاً ومتاحاً إذا لم يكن لدينا معلومات إضافية حول المتعلم والمحتوى التعليمي (الهدف من التعلم، المتطلبات الأساسية، خلفية المتعلم، المستويات ... الخ). وبالتالي، فإننا نعمل على إنجاز نظام تعاوني، حيث يعمل العديد من المعلمين بطريقة تعاونية، لإنشاء وتعليم المواد التعليمية باستخدام نظام وكلاء متعددة. مساهمة نظامنا هو توجيه تقنيات التكيف مع تقنيات التعاون والويب الدلالي (الأنطولوجيا، التعليم). تمثل معلومات وخصائص المتعلمين ومحتوى التعلم باستخدام الأنطولوجيا لتلبية التنوع والاحتياجات الفردية للمتعلمين. نستخدم نموذج العميل في نظامنا للاستفادة من نقاط القوة في هذا النموذج مثل: النمطية، والاستقلال الذاتي، والمرونة ... الخ

. الكلمات المفتاحية: التعليم عن بعد، نظام تعاوني، نظام تكيف، الوب الدلالي، المحتوى التعليمي، أساليب وأنماط التعلم.

Table des matières

Introduction générale	1
Chapitre I : La formation à distance : E-Learning	6
I.1. Introduction	6
I.2. Un peu d’historique de E-Learning	6
I.3. Définition de E-Learning	8
I.4. Les éléments de E-Learning	9
I.5. D’une formation traditionnelle à une formation à distance	11
I.6. Topologie de E-Learning	12
I.7. Les avantages de E-Learning	13
I.8. Outils de E-Learning	14
I.8.1. Curriculum	15
I.8.2. Bibliothèque numérique	15
I.8.3. Outil de représentation des connaissances	15
I.9. Les limites de la formation à distance	16
I.9.1. Le sentiment d’isolement	16
I.9.2. Les difficultés à maîtriser une méthode de travail inconnue.....	16
I.9.3. La difficulté pour certains publics à travailler seuls	16
I.9.4. La conception des ressources	16
I.10. Les enjeux de E-Learning	16
I.10.1. L’efficacité et l’adaptabilité des processus d’apprentissage	17
I.10.2. L’accès à la connaissance.....	17
I.10.3. L’autonomie de l’apprenant	17
I.10.4. L’accompagnement de l’apprenant	17
I.10.5 Les nouveaux rôles de l’enseignant	18
I.10.6. Le développement des systèmes du E-Learning	18
I.11. Document pédagogique et hypermédia	19

I.11.1. Document pédagogique.....	19
I.11.2. Hypermédia	19
I.11.3. Entre hypertexte et hypermédia.....	19
I.11.4. Hypermédia adaptatif	20
I.11.5. Techniques d'adaptation	20
I.11.5.1. Adaptation du contenu	20
I.11.5.2. Adaptation des liens (navigateur)	21
I.11.6. Systèmes hypermédias adaptatifs dynamiques	21
I.12. Plate-forme de Formation à Distance.....	22
I.12.1. Quelques standard.....	22
I.12.2. Quelques Plateformes.....	24
I.12.2.1. Claroline	24
I.12.2.2. Ganesha	25
I.12.2.3. Moodle.....	26
I.12.2.4. Sakai	27
I.12.2.5. Spiral	28
I.12.2.6. WebCT	28
I.12.3. Synthèse sur les plates-formes présentées.....	29
I.13. Styles d'apprentissage.....	31
I.13.1. Modèle de Dunn et Dunn	31
I.13.2. Modèle de Felder et Silverman	32
I.13.3. Modèle de Grasha - Riechmann	32
I.14. Conclusion	33
Chapitre II : Le web sémantique et les ontologies.....	34
II.1. Introduction	34
II.2. Le Web sémantique	35
II.2.1. Définition du mot "sémantique"	35
II.2.2. définition du Web sémantique	35
II.2.3. Les métadonnées.....	37
II.2.3.1. Métadonnées syntaxique	37
II.2.3.2. Métadonnées structurelles.....	37
II.2.3.3. Métadonnées sémantiques.....	38

II.3. Les ontologies	38
II.3.1. Qu'est-ce qu'une Ontologie ?	38
II.3.2. Classification des ontologies	40
II.3.3. Les composants d'une ontologie.....	42
II.3.3.1. Concepts.....	42
II.3.3.2. Relations.....	43
II.3.3.3. Les fonctions	44
II.3.3.4. Axiomes	44
II.3.3.5. Instances	44
II.3.4. Méthodologies pour la construction d'ontologies	44
II.3.4.1. Les principes de construction d'une ontologie.....	44
II.3.4.2. La méthode On-To-Knowledge.....	45
II.3.4.3. La méthode Methontology	46
II.3.5. Les langages d'ontologie	47
II.3.5.1. XML	48
II.3.5.2. RDF	49
II.3.5.3. DAML+OIL	50
II.3.5.4. OWL.....	51
II.3.6. Comparaison des langages de représentation.....	52
II.3.7. L'utilisation des ontologies	53
II.4. Web sémantique et E-Learning	53
II.5. Conclusion	56

Chapitre III : Etat de l'art sur le E-Learning sémantique à base d'agent.....

III.1. Introduction	57
III.2. E-Learning à la base de la sémantique	57
III.3. E-learning et systèmes multi-agents	59
III.3.1. Le concept d'agent.....	59
III.3.2. Système multi-agents.....	61
III.3.2.1. Définition d'un système multi-agents.....	62
III.3.2.2. Les plateformes de développement des SMA.....	63
III.3.2.2.1. JADE.....	63

III.3.2.2.2. Madkit	65
III.3.2.2.3. Comparaison des plateformes	66
III.3.3. Quelques travaux de E-learning basés agent	67
III.3.3.1. Baghera	67
III.3.3.2. SIGFAD	69
III.3.3.3. iPédagogique.....	70
III.3.3.4. MASCE.....	70
III.3.3.5. I-MINDS	71
III.3.3.6. Etude comparative des plateformes présentées	71
III.3.4. Les raisons du choix du paradigme d'agent	72
III.4. Conclusion	73
Chapitre IV : Modélisation de l'approche.....	75
IV.1. Introduction	75
IV.2. Fondements théoriques.....	76
IV.2.1. Web sémantique et E-Learning	76
IV.2.2. Apprentissage personnalisé	76
IV.2.2.1. Adaptation de contenu éducatif	77
IV.2.2.2. Styles d'apprentissage	77
IV.2.2.2.1. Perception.....	77
IV.2.2.2.2. Participation	78
IV.2.2.2.3. Pratique	78
IV.2.2.2.4. Psychologique	78
IV.2.2.2.5. Sociologique.....	78
IV.2.2.2.6. Responsabilité	79
IV.2.3. Les questions	79
IV.2.4. Les tests	79
IV.2.5. Le profil	79
IV.3. Architecture générale du système	80
IV.3.1. Identification des acteurs du système	82
IV.3.1.1. Enseignant	82
IV.3.1.2. Apprenant	82

IV.3.1.2.1. Modèle d'apprenant	82
IV.3.1.3. Administrateur	86
IV.3.1.4. Contenu éducatif.....	86
IV.3.1.4.1. La couche1: Préparation de domaine d'apprentissage.....	88
IV.3.1.4.2. La couche2: Préparation du MainDoc et le vocabulaire	88
IV.3.1.4.3. La couche3: Création et annotation de RefDoc	92
IV.4. Fonctionnement du système de création et d'annotation	95
IV.5. Architecture détaillée du système	98
IV.5.1. Agent interface	99
IV.5.2. Agent enseignant	99
IV.5.3. Agent gestion profil enseignant	100
IV.5.4. Agent gestion cours (AGC)	101
IV.5.5. Agent création cours (ACC)	102
IV.5.6. Agent d'annotation (AA)	102
IV.5.7. Agent collaborative (ACol)	102
IV.5.8. Agent administrateur (AAdm)	102
IV.5.9. Scenario de fonctionnement du système	103
IV.5.9.1. Ouvrir une session	103
IV.5.9.2. Création et annotation d'un fragment de cours	105
IV.6. Architecture de système d'apprentissage et d'adaptation.....	108
IV.6.1. Agent Interface Apprenant (AIA)	108
IV.6.2. Agent Apprenant (AApp)	109
IV.6.3. Agent Pédagogique (AP)	109
IV.6.4. Agent Gestion Cours (AGC)	109
IV.6.5. Agent Adaptation (AAdp)	110
IV.6.6. Fonctionnement du système d'adaptation	111
IV.7. Conclusion.....	112
 Chapitre V : Implémentation.....	 113
 V.1. Introduction	 113
V.2. Environnement de développement.....	113
V.3. Choix du langage de programmation	114
V.4. La plateforme JADE	115

V.4.1. La création d'agent	116
V.4.2. Identificateur	116
V.4.3. Comportement d'agent	116
V.4.4. Le langage ACL.....	119
V.4.4.1. L'envoi d'un message.....	120
V.4.4.2. La reception du message	120
V.4.5. La création d'un agent avec une interface graphique	120
V.5. L'éditeur d'Ontologie Protégé.....	122
V.6. Jena	123
V.7. JSP	124
V.8. Architecture d'application Web	124
V.9. Serveur TOMCAT	124
V.10.principe de l'application	125
V.11. Quelques ontologies de notre système.....	126
V.12. Présentation de quelques fenêtre de notre système	129
V.13. Conclusion	134
Conclusion Générale.....	135
Bibliographie	137

Table des figures

Figure I.4 : les acteurs et les éléments de E-Learning.....	10
Figure II.2.2 : Architecture du Web sémantique proposée par Tim Berners-Lee.....	36
Figure II.2.3 : Types de métadonnées et des annotations sémantiques selon Sheth.....	37
Figure II.3.1 : Un exemple d'ontologie.....	39
Figure II.3.2.a : Classification des ontologies selon Van Heijst	40
Figure II.3.2.b : Classification des ontologies selon Guarino	41
Figure II.3.4.2 : Les cinq etapes du processus On-To-Knowledge	46
Figure II.3.4.3 : Les différentes phases de Methontology	46
Figure II.3.5 : Généalogie des langages	47
Figure II.3.5.4.a : Origine du langage OWL	51
Figure II.3.5.4.b : Les types du langage OWL	52
Figure III.4 : Agent agit sur l'environnement.....	60
Figure III.5.6.1 : Architecture de la plateforme JADE	64
Figure III.5.6.2 : Architecture de la plateforme Madkit	65
Figure III.6.1 : Architecture générale de Baghera.....	69
Figure III.6.2 : Vue d'ensemble de SIGFAD.....	70
Figure IV.4 : Architecture générale du système	81
Figure IV.4.1.2.a : Ontologie de profil d'apprenant	84
Figure IV.4.1.2.b : Ontologie de styles d'apprentissage	85
Figure IV.4.1.4.a : le contenu éducatif	87
Figure IV.4.1.4.b : les étapes de construction de contenu éducatif	88
Figure IV.4.1.4.2.a : l'ontologie de domaine (vocabulaire).....	89
Figure IV.4.1.4.2.b : Organisation de MainDoc (Module).....	90
Figure IV.4.1.4.2.c : Ontologie du MainDoc	91
Figure IV.4.1.4.3.a : Ontologie de RefDoc	93
Figure IV.4.1.4.3.b : Une ontologie de MainDoc et RefDoc et leurs relations	94
Figure IV.4.a : Processus de création du MainDoc	95
Figure IV.4.b : Processus Processus d'annotation du RefDoc	96

Figure IV.5 : Architecture détaillée du système (Création et annotation de contenu éducatif).....	98
Figure IV.5.2 : Architecture structurelle de l'agent Enseignant.....	99
Figure IV.5.4 : Architecture structurelle de l'Agent Gestion Cours	101
Figure IV.5.9.1. : Diagramme de séquence « ouvrir Session »	104
Figure IV.5.9.2.a : Diagramme de séquence « Création de MainDoc »	106
Figure IV.5.9.2.b : Diagramme de séquence « Création de RefDoc »	107
Figure IV.6 : Architecture de système (adaptation)	108
Figure IV.6.6 : Fonctionnement du système d'adaptation	111
Figure V.2 : la fenêtre principale de l'Eclipse	114
Figure V.4 : la fenêtre principale de la plateforme JADE	115
Figure V.4.3 : Les chemins d'exécution des comportements d'un agent	117
Figure V.4.4 : Le paradigme du passage d'un message asynchrone	119
Figure V.5 : La fenêtre principale de Protégé 4.3	122
Figure V.8 : Architecture application Web.	124
Figure V.9 : Serveur Tomcat	125
Figure V.10 : principe de l'application	125
Figure V.11.a : ontologie Profil apprenant	126
Figure V.11.b : l'ontologie de domaine (vocabulaire)	127
Figure V.11.c : Organisation de MainDoc (Module)	128
Figure V.11.d : Ontologie de RefDoc	128
Figure V.11.e : Ontologie RefDoc et MainDoc et leurs relations	129
Figure V.11.f : exemple de propriété de classe d'une ontologie	139
Figure V.12.a : fenêtre principale de notre application	130
Figure V.12.b : les agents de notre système	131
Figure V.12.c : fenêtre d'inscription de l'enseignant	131
Figure V.12.d : fenêtre de connexion de l'enseignant	131
Figure V.12.e : fenêtre de session de l'enseignant	132
Figure V.12.f : fenêtre de création de RefDoc	132
Figure V.12.g : le diagramme de séquence (création RefDoc) en utilisant l'agent Sniffer	133
Figure V.12.h : le contenu de message envoyé entre agents, en utilisant l'agent Sniffer	134

Introduction Générale

Contexte et problématique de recherche

- **Contexte**

L'internet a donné l'opportunité d'avoir une autre méthode d'apprentissage, où les apprenants peuvent apprendre n'importe où et n'importe quand. Ce nouveau mode est appelé E-Learning ou la formation à distance. Ce nouvel mode de formation a importé plusieurs concepts et pas mal des avantages qui améliorent la qualité d'apprentissage chez l'apprenant.

La FAD a facilité l'engagement dans une formation, et donne une chance d'apprentissage aux personnes qui n'ont pas le temps suffisant à faire une formation classique. Elle permet notamment de s'affranchir des contraintes géographiques. L'utilisation du Web comme outil d'enseignement, aide les apprenants à échanger entre eux des cours et des documents, aussi le travail en collaboration sur un sujet de recherche devient possible avec des encadreurs.

L'accès quotidien à l'information sur le cours et plus libre que la formation classique qui exige le temps de formation. Dans ce type de formation l'apprenant est considéré l'acteur principal de son apprentissage c'est à dire l'apprenant devient plus actif et peut interagir avec le contenu du cours, le formateur et les autres apprenants. Avec la diffusion des technologies de l'information dans les activités éducatives, les barrières de temps et de lieu sont supprimés, ainsi que la gestion et l'exécution de ces activités peuvent être exécutées plus efficacement. Des progrès significatifs dans les études de E-Learning ont été effectuées à la fois sur les plates-formes d'enseignement ouvert et de la formation professionnelle.

Les systèmes de gestion d'apprentissage (LMS :Learning Management Systèmes) comme WebCT, Blackboard, et Moodle sont couramment utilisés avec succès dans l'E-éducation. Tandis qu'ils se concentrent sur le soutien des enseignants dans la création et l'organisation de cours en ligne, ils ne considèrent pas généralement les différences individuelles des apprenants. Cependant, les apprenants ont des besoins différents et des caractéristiques telles que les backgrounds, la motivation, les préférences et les styles d'apprentissage. Récemment, une attention croissante est accordée à des caractéristiques telles que les styles d'apprentissage, leur impact sur l'apprentissage, et comment ces caractéristiques individuelles peuvent être prises en charge par les systèmes d'apprentissage. Ces enquêtes sont motivées par les théories pédagogiques, qui affirment que donner des cours qui répondent aux caractéristiques individuelles des apprenants rend l'apprentissage plus facile pour eux et donc, augmente la progression de leur apprentissage.

Dans le domaine de E-Learning, le Web Sémantique s'avère une technologie prometteuse pour implémenter les systèmes du E-Learning [6][7][8]. L'objectif du Web sémantique est de rendre le profil d'apprenant et contenu éducatif accessibles et interprétables

par des agents logiciels [6]. L'utilisation de la technologie de Web sémantique dans un environnement du E-Learning, permet la communication selon une base sémantique entre les agents humains et machine. L'ontologie permet l'organisation des profils d'apprenants et des documents d'apprentissage autour de petites pièces d'objets d'apprentissage sémantiquement annotés. Cette annotation des contenus éducatifs facilite l'organisation et la compréhension de ces contenus par la machine, et accélère la livraison de document pertinent juste à temps. Aussi, elle offre l'opportunité de fournir les contenus éducatifs à l'utilisateur selon son profil (représenté sémantiquement) et les besoins (pertinence) [122].

• **Problématique**

Dans un environnement E-learning, les apprenants sont, généralement, hétérogènes c'est à dire ont différents backgrounds, différentes capacités intellectuelles et préférences et différents objectifs et besoins [121] [122]. Ainsi, les apprenant n'apprennent pas si nous utilisons les mêmes cours et mêmes styles d'apprentissage dans toutes les situations; donc la qualité de compréhension va être faible pour la plupart des apprenants. Pour affranchir ce problème, nous devons utiliser des techniques d'adaptation de contenu et de stratégies selon chaque individu pour satisfaire les besoins et les préférences des apprenants hétérogènes et fournir un apprentissage personnalisé pour chaque apprenant. Donc, l'adaptation selon plusieurs critères et situations des apprenants est une technique inévitable.

Le support principal pour l'apprentissage dans un environnement E-Learning est le contenu éducatif. Mais, ces apprenants utilisent dans le processus d'apprentissage ces documents pédagogiques qui ne sont pas créés (développés) pour cet objectif d'adaptation. Dans les systèmes existants, les apprenants : 1) lisent les cours de façon séquentiel ; 2) sont libres de choisir le cours de façon individuel. Dans le premier cas, il est évident que les apprenants ne peuvent pas apprendre par la même séquence de cours et le même style d'apprentissage. Dans le deuxième cas, les apprenants vont être désorientés dans un environnement qui contient d'énorme quantité de cours (manque de suivi et d'orientation : comme Moodle).

Nous développons le contenu éducatif qui est de haute qualité scientifique et qui facilite l'apprentissage adaptatif et personnalisé. Mais, la création de ces types de contenu est un processus très important, difficile et complexe. Ce processus de création doit offrir un document spécifique, simple à exploiter et à adapter selon plusieurs situations. Le document pédagogique doit aussi transmettre le savoir de l'enseignant à l'apprenant de façon pertinente et efficace dans le processus d'apprentissage.

Le Web sémantique est considéré comme une solution prometteuse pour faciliter la manipulation et la compréhension des informations par la machine [8]. Donc la représentation ontologique des profils des utilisateurs et de contenu éducatif devient une exigence dans les systèmes d'E-Learning adaptatif. Cependant, dans les systèmes d'apprentissage adaptatif « E-learning basé sur l'adaptation », plusieurs problèmes techniques sont posés. Tout d'abord, c'est la question de savoir comment créer des profils d'utilisateurs précis et complets pour les apprenants et comment ils peuvent être utilisés pour reconnaître les apprenants et décrire ses

comportements. Aussi, il y a la question de la façon de construire les relations hiérarchiques et de navigation entre les différentes parties du matériel d'apprentissage et comment celles-ci peuvent être déterminées, en fonction du profil de l'utilisateur, par la reconnaissance de la prochaine étape de l'apprentissage dans le processus d'apprentissage.

Le nombre de sources d'informations disponibles dans le web est de plus en plus énorme. Donc, la description syntaxique de ces informations devient insuffisante. Par conséquence, il est nécessaire d'ajouter une couche sémantique pour la description et la représentation des connaissances. Dans un environnement E-Learning, les objets éducatifs sont traités par la machine (ordinateur) ; ainsi, les contenus éducatifs doivent être exprimés par l'aspect structurel et également l'aspect sémantique. Cette sémantique peut être exprimée à l'aide de Web sémantique et d'ontologies. L'utilisation des ontologies pour la représentation des connaissances s'avère avantageuse afin de faciliter l'utilisation de ces connaissances par la machine, comme dans [7][122].

Le Web sémantique (Semantic Web) se présente comme une extension du Web classique. C'est un moyen d'extraction et d'exploitation de l'intelligence collective du Web, en rendant le contenu plus facile à échanger et à partager entre les différents utilisateurs. Ce nouveau Web vise à faciliter l'exploitation des ressources par les machines en accédant aux ressources grâce à leur représentation sémantique. Selon Goble [42], le Web sémantique vise à favoriser la découverte, l'échange d'information et l'intégration de l'information par l'étiquetage (tagging) ou le balisage (marking up), et à développer des technologies et infrastructures pour annoter le contenu du Web avec des informations additionnelles appelées "métadonnées" pouvant être traitées par des agents logiciels.

Contributions

Généralement l'apprenant interagit avec le système éducatif à travers le contenu d'apprentissage (les cours, les objets éducatifs). Pour cela, on s'intéresse à la création et l'annotation sémantique de manière collaborative de contenus d'apprentissage avec le souci de minimiser le temps de développement de ces contenus et d'augmenter leur qualité, interopérabilité et flexibilité. Notre approche permet aux auteurs (enseignants) de créer et d'annoter les documents pédagogiques de façon collaborative. On s'intéresse aussi à l'adaptation de contenus et de stratégies (styles d'apprentissage : style active, réflexif, auditif, visuel, collaboratif, compétitif et intuitif ...etc.), pour répondre à la diversité et aux besoins individuels des apprenants.

Dans notre travail, nous avons développé une approche éducatif qui assure l'intervention de plusieurs auteurs pour la création et l'annotation des documents éducatif. Le Web sémantique est utilisé pour ajouter l'aspect sémantique aux profils d'utilisateurs et aux documents pour faciliter leur manipulation par le système. L'adaptation de contenu et de stratégies, est une activité fondamentale dans notre système en utilisant les documents annotés sémantiquement. Nous exploitons le profil de l'apprenant ainsi que les tests et les questions posées par l'apprenant pour lancer une procédure d'adaptation adéquate et au moment

opportun. L'objectif du système est de garder l'apprenant toujours motivé à continuer la formation.

Le Web sémantique repose sur l'association de métadonnées explicites aux informations et leur mise en relation avec des ontologies [7][8]. Les métadonnées explicites permettent d'ajouter des informations additionnelles afin de décrire les pages Web ou des ressources et les ontologies pour décrire un domaine avec ses concepts et les relations qui existent entre ses concepts. Par conséquent, la description sémantique des profils et des ressources est un concept fondamental du Web sémantique. Pour toutes ces raisons, nous nous intéressons dans notre travail au processus de création et d'annotation des contenus éducatif et l'exploitation de ces contenus pour fournir un apprentissage personnalisé en se basant sur l'utilisation de la sémantique des informations et les ontologies (ontologie de profil, ontologie de vocabulaire de domaine, ontologie des styles d'apprentissage, et ontologie de contenu éducatif).

La phase de la conception d'un système de formation à distance exige multiples disciplines qui doivent converger vers un seul objectif qui est la transmission de connaissances à un individu (un apprenant). La conception d'un système d'apprentissage fait appel à un ensemble de processus complexes qu'il est parfois difficile d'associer dans un système cohérent et évolutif. Les systèmes multi-agents s'avèrent pertinents pour résoudre ces problèmes parce qu'ils possèdent des caractéristiques qui permettent de mieux structurer les systèmes de transmission de connaissances et ouvrent de nouvelles perspectives d'assistance à la formation à distance. On utilise le paradigme "Agent" dans notre système afin de profiter des points forts de ce paradigme tel que la modularité, autonomie, flexibilité ...etc. Les systèmes multi-agents (SMA) sont considérés comme une solution pour pallier les limites posées dans l'IA. Les systèmes multi-agents sont très utilisés dans les systèmes d'apprentissage à distance [121], ils donnent plusieurs avantages et facilitent les fonctions de base de la FAD avec satisfaction des besoins des apprenants.

Organisation de la thèse

Cette thèse est organisée en cinq chapitres. Après, la présentation de la problématique et l'introduction générale, nous avons choisi d'introduire notre thèse par un chapitre qui présente les concepts liés à la formation à distance ou E-learning. Ce premier chapitre présente l'avantage de E-Learning par rapport la formation traditionnelle en classe, et les plateformes de E-Learning. Aussi, nous avons décrit les différents styles d'apprentissage existant dans la littérature.

Le deuxième chapitre est consacré à la présentation de la technologie de Web sémantique et les ontologies. Ce chapitre présente les différents éléments et langages d'ontologie et l'objectif de l'utilisation de cette technologie dans le domaine de E-Learning.

Le troisième chapitre est un survol sur les travaux du E-Learning sémantique à base d'agent. Ce chapitre a l'objectif de se situer notre travail par rapport les travaux existant dans la littérature.

Une vue conceptuelle de notre système est décrite dans le quatrième chapitre, ce chapitre présente notre contribution qui est la création et l'annotation des contenus éducatif et l'adaptation de contenu et de stratégies selon plusieurs situations d'apprenants. Le dernier chapitre présente les outils utilisés dans la phase de l'implémentation. Nous terminons notre thèse par une conclusion générale, d'autre part nous n'oublions pas de signaler les perspectives possibles à ce travail.

Chapitre I : La formation à distance : E-Learning

I.1. Introduction

Le changement des circonstances et de technologie a conduit au changement considérable de l'économie, la société, les besoins, la culture et la vie. Les entreprises et les organisations doivent élaborer des systèmes et des réponses rapides et précises à ces changements qui augmentent chaque jour. En particulier, on doit redéfinir notre méthode classique d'apprentissage dite "apprentissage traditionnelle ou présentielle" où les étudiants sont obligés d'être présent dans des temps prédéfinis. Donc, beaucoup des travaux ont commencé une grande compétition pour évoluer et améliorer notre méthode d'apprentissage en utilisant les nouvelles technologies, notamment la technologie de l'information et de la communication. Par cette intégration de TIC dans les activités éducatives, les barrières de temps et de lieu sont supprimées, ainsi que la gestion et l'exécution de ces activités peuvent être effectués plus efficacement. Donc, ce mariage de TIC et l'éducation a conduit à l'ère de la naissance de la E-Learning. Dans un environnement E-Learning.

La formation classique est considérée assez lente, coûteuse et ne convient pas tous les étudiants car leurs occupations. La demande de rapidité, exigée par nos jours, dans le processus d'apprentissage agrandit jour par jour, a fait appel au nouveau style de formation. Alors, Ce nouveau style d'apprentissage dû à l'exigence de la vie actuelle, des progrès significatifs dans les études de E-Learning ont été réalisées à la fois sur les plates-formes d'enseignement ouvert et de la formation professionnelle non seulement du point de vue éducation mais aussi du point de vue de la nouvelle économie.

Dans ce chapitre, après une brève introduction nous allons présenter quelques définitions de E-Learning pour bien expliquer l'environnement de notre sujet de recherche. La section suivante est consacré à justifier l'utilisation des systèmes E-Learning et leurs avantages par apport la formation traditionnelle. Les topologies, les avantages et les limites de E-Learning sont présenté dans la section suivante. Ensuite, on présente une vue panoramique des différentes plateformes de E-Learning et un survol sur les documents pédagogique et les techniques d'adaptation. Finalement, on conclut le chapitre.

I.2. Historique de E-Learning

Au milieu du dix-neuvième siècle, les études par correspondance ont apparu sous le nom de la formation à distance. Le concept de la formation à distance a connu une évolution considérable au fil des années. En premier lieu les cours ont été envoyés sous la forme de papier par poste ou par fax, puis les cassettes audio et vidéo, ensuite la diffusion hertzienne via la radio et les émissions de télévision. Dans les années 1970, le développement des technologies de l'informatique (l'ordinateur, CDROM ...etc) a conduit à l'arrivée des premiers systèmes d'Enseignement Assisté par Ordinateur (EAO). Le transfert des connaissances était l'objectif de ces systèmes. Plusieurs programmes et systèmes éducatifs ont été développés mais leur contenu était limité et leur utilisation rigide. L'aspect cognitif a été totalement ignoré avec peu de travaux de recherche, de diagnostic et d'adaptation de

stratégies. Mais, malgré leur application limitée, ces systèmes ont eu des retombées significatives dans le domaine de l'éducation. L'intelligence Artificielle et les systèmes experts ont donné l'application des techniques de raisonnement. L'intégration de ces techniques dans le domaine d'apprentissage a permis la naissance des systèmes d'Enseignement Intelligemment Assisté par Ordinateur (EIAO) qui pallient les nombreux inconvénients des systèmes précédents. Les recherches ont développé des Systèmes Tutoriels Intelligents (STI) qui ont offert un degré très élevé d'interaction entre l'apprenant et la machine afin d'adapter l'apprentissage au niveau de connaissances de l'apprenant. Ces systèmes ont pour but de reproduire le comportement d'un tuteur intelligent afin de dispenser un enseignement personnalisé à l'utilisateur. Ces systèmes offrent une possibilité de génération dynamique d'exercice, des adaptations au niveau de difficultés selon les performances de l'étudiant ainsi que l'analyse de l'interprétation du comportement de l'étudiant. En effet, les Systèmes Tutoriels Intelligents sont capables de réaliser des inférences sur des connaissances de l'étudiant, et peuvent interagir intelligemment avec lui en adaptant dynamiquement les sujets à présenter en fonction des résultats acquis et du mode d'apprentissage qui lui convient le mieux [12].

Selon Nipper [52] la formation à distance a connu trois générations. La première a été celle de l'imprimerie qui a permis l'offre de cours par correspondance et la deuxième celle du multimédia (imprimé, radio, télévision, vidéo) mieux connue au Québec sous "enseignement télévisé". La troisième génération est fondée par micro-informatique, entre autres, Internet [52].

- **Première génération : l'enseignement par correspondance**

L'imprimé marque le début de la formation à distance et constitue le principal vecteur d'enseignement et de tutorat [62].

- **Deuxième génération : l'enseignement télévisé**

Dès les années 60, s'ouvre l'ère des multi-médias caractérisée par l'usage de différents médias (imprimé, radio, télévision, vidéo) [62].

- **Troisième génération : le multimédia et les télécommunications**

La naissance de la micro informatique puis de la télématique dans les années 80, marque l'époque contemporaine d'Internet, des hypermédias et du multimédia multiutilisateur.

Selon le type de technologies utilisées, Taylor et Swannel [60] décompose de manière plus fine la dernière étape, en Telelearning puis Flexible Learning Model. Le télé-enseignement (Telelearning) se base essentiellement sur les technologies de la téléprésence, l'audio et la vidéoconférence. Le dernier modèle (Flexible Learning Model) est celui du multimédia interactif, de la Communication Médiatisée par Ordinateur (CMO) et des cours basés sur les ressources d'Internet. Ces classifications n'ont qu'une valeur indicative et demandent sans doute à être nuancées et complétées au fil du développement des nouvelles technologies. Néanmoins, l'évolution de la place des TIC (Technologies de l'Information et de la Communication) dans la société et dans l'éducation amène à se poser la question de la FAD à mettre en œuvre [62].

I.3. Définition du E-learning

Dans la littérature, il existe différentes définitions du e-learning. Cependant, lorsque ces définitions sont considérées du point de vue plus large, on constate que généralement ils mettent l'accent sur les mêmes caractéristiques et fonctions. Dans le Plan d'action européen de E-Learning, le concept est défini comme suit : " l'utilisation des nouvelles technologies multimédias et de l'Internet pour améliorer la qualité de l'apprentissage en facilitant l'accès aux ressources et aux services, ainsi que les échanges à distance et la collaboration " [33]. Dans cette définition, les technologies électroniques sont considérées comme des véhicules pour les services et les ressources de l'éducation, et des conduits de la coopération et de la communication. Pour éclaircir le domaine de la formation à distance et le terme E-learning on présente d'autres définitions :

- E-Learning est l'apprentissage à distance qui utilise la technologie de l'ordinateur (généralement Internet) [24].
- E-Learning peut être défini comme suit: l'accès en ligne à des ressources d'apprentissage, n'importe où et n'importe quand [33].
- E-Learning est essentiellement le transfert par le réseau des compétences et des connaissances. Donc E-learning se réfère à l'utilisation des applications et des processus électroniques à apprendre. Les applications des processus d'apprentissage en ligne incluent: l'apprentissage sur le Web, l'apprentissage assisté par ordinateur, des classes virtuelles et la collaboration numérique [24].
- La formation à distance est défini comme "fournir un contenu pédagogique qui est individuel, compréhensible et dynamique en temps réel, aider au développement des communautés de la connaissance, relier les apprenants et les praticiens avec des experts " [27].

Dans le contexte d'apprentissage et de la performance, le concept de E-Learning peut être défini d'un point de vue plus large et complet de la manière suivante: E-Learning est l'utilisation des technologies de l'Internet afin de créer un environnement d'apprentissage riche qui comprend une grande variété de ressources d'information et d'instructions et de solutions. Et le but de l'apprentissage est d'améliorer la performance individuelle et organisationnelle [34]. Dans cette définition, les solutions, les instructions et les informations sont incluses. Donc, dans un environnement éducatif, les contenus de cours et les applications sont utilisés dans des systèmes pertinents, même les devoirs et les examens peuvent être effectués par voie électronique.

Le terme E-Learning signifie l'éducation électronique et il est essentiellement la livraison en ligne de l'information, la communication et l'apprentissage. E-learning semble avoir une multiplicité de définitions de chacun de ses utilisateurs et le terme semble signifier quelque chose de différent. Une définition très complète a été proposée par le système Cisco, qui définit le E-éducation est l'apprentissage sur l'Internet. Les composants de E-learning peuvent inclure la livraison de contenu dans des formats multiples, la gestion de l'expérience d'apprentissage, une communauté des apprenants, les développeurs de contenu et des experts. E-learning offre un apprentissage plus rapide à des coûts réduits, un meilleur accès à l'apprentissage, et une responsabilisation claire pour tous les participants au processus d'apprentissage. Dans la culture trépidant d'aujourd'hui, les organisations qui mettent en

œuvre E-learning offrent leur force de travail avec la possibilité de transformer ce changement en avantage [28].

L'auteur de l'article [31] a présenté deux définitions plus technologiques:

- "An individual or group use of electronic mediums that provide access to online learning tools and resources."
- "These dynamic mediums offer shared community spaces, support digital communication and collaboration, and link to information sources, such as streamed video, podcasts, webcasts, digital libraries, web pages, and videoconferencing."

Une définition plus large a été présentée dans [32] : "The use of the internet to access learning materials; to interact with the content, instructor and other learners; and to obtain support during the learning process, in order to acquire knowledge, to construct personal meaning, and grow from the learning experience. "

En effet, la venue de l'informatique dans le monde des communications a permis d'assister l'encadrement pédagogique des étudiants (apprenants) dans un contexte d'enseignement à distance qui est un enseignement suivi sans présence d'enseignant et qui est défini comme: "Une famille de méthodes éducatives où l'enseignement se réalise séparément de l'apprentissage et couvre les différentes formes d'étude sans supervision continue et immédiate des étudiants" [40]. Cette définition ajoute l'idée que le suivi de l'apprenant n'est pas immédiat dû au fait que l'étudiant et l'enseignant ne doivent pas être présents les deux en même temps ou en même place.

Rheume insiste dans leur définition que l'apprenant est plus libre, il peut apprendre ne pas seulement à l'école mais à leur travail, leur maison...etc. : "Les enseignements à distance sont au moins ouverts par le lieu, puisqu'ils sont organisés pour permettre aux apprenants d'apprendre là où ils se trouvent, à domicile ou à leur lieu de travail" [51].

Finalement, on présente cette définition : " C'est un système de formation conçu pour permettre à des individus de se former sans se déplacer sur le lieu de formation et sans la présence physique d'un formateur. La formation à distance recouvre plusieurs modalités (cours par correspondance, e-learning ...) et est incluse dans le concept plus général de Formation Ouverte et à Distance " [50].

I.4. Les éléments du E-Learning

Après une période d'enthousiasme où l'E-Learning a été perçu comme un moyen de résoudre les problèmes de formation (logistiques et budgétaires) en remplaçant les formations traditionnelles ou présentiel, nous pouvons dire qu'aujourd'hui l'E-learning atteint l'âge de raison.

Le schéma ci-dessous synthétise l'ensemble des éléments d'un dispositif de formation à distance:

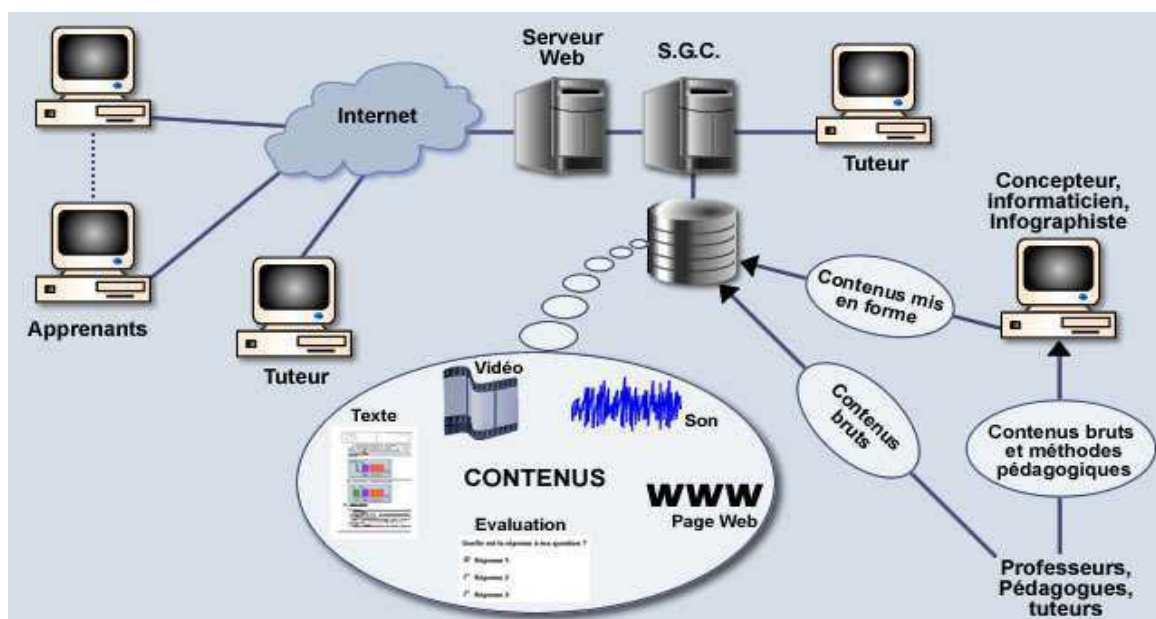


Figure I.4 : les acteurs et les éléments de E-Learning [39]

Les acteurs principaux dans un environnement E-Learning sont : apprenants, tuteurs, concepteurs.

- **Apprenant :** suit depuis son poste de travail une formation dont le contenu est organisé selon un parcours pédagogique défini par le formateur en fonction des compétences possédées par l'apprenant avant la formation, en fonction des objectifs pédagogiques visés par la formation et en fonction des résultats obtenus lors des évaluations intermédiaires : le parcours est donc individualisé.
- **Tuteur :** le rôle du tuteur apparaît comme fondamental, il ne s'agit plus de transmettre des connaissances, mais d'accompagner l'apprenant dans ses acquisitions, de lui faire acquérir un maximum d'autonomie. Il est là pour combattre la sensation d'isolement que pourrait ressentir l'apprenant dans une classe virtuelle et pour maintenir sa motivation par des encouragements, des questionnements, des suggestions d'approfondissement. Il est de plus en plus fréquent que les formations soient associées à la mise en place d'une communauté d'apprentissage que le tuteur sera chargé d'animer. Le tuteur humain crée les éléments de la formation (modules de cours, évaluations, tests, simulations, ...etc.) et suit, guide et oriente l'apprenant durant le processus d'apprentissage.
- **Le concepteur :** c'est l'administrateur qui gère le système, il travaille à partir du scénario pédagogique fourni par l'enseignant. Il a en charge la médiatisation des contenus, c'est-à-dire la mise en forme des ressources pédagogiques et leur intégration sur la plateforme.
- **Les contenus :** depuis l'amélioration des vitesses de transmission, les documents ressources ne sont plus limités à des documents textuels, ils peuvent s'enrichir de modules sonores, d'images en haute résolution ou de vidéo, rendant les contenus d'autant plus attractifs et permettant de limiter la sensation d'isolement ressentie par l'apprenant. Les standards sont une garantie de pérennité des dispositifs et des contenus et d'interopérabilité des applications. Les échanges entre concepteurs ou entre organismes sont facilités, minimisant les coûts de développement. L'intégration de contenus respectant les standards

peut se faire sur toute plateforme qui les respecte, ce qui induit une diminution des coûts lors d'une migration de plateforme.

- **Méthodes et objectifs pédagogiques** : Toutes les méthodes pédagogiques peuvent être envisagées dans un dispositif de formation à distance : transmissive, démonstrative, interrogative, pédagogie de projet, méthode coopérative. Elles peuvent être utilisées alternativement selon les modules, les types de formation, les contextes. L'insistance est mise sur la qualité de l'accompagnement, c'est-à-dire, l'apprenant est l'acteur de la construction de son savoir, le tuteur est un facilitateur. Il n'y a pas de dispositif idéal, valable pour tous. La réussite d'une formation à distance dépend de la capacité d'autonomie de l'apprenant, de la qualité du dispositif et du tutorat, du contexte et de la disponibilité, de la synergie émergente dans la classe virtuelle. Elle peut aussi être utilisée dans un mode hybride (blended learning) alternant formation en présentiel et formation à distance [37].

I.5. D'une formation traditionnelle à une formation à distance

La formation présentielle ou traditionnelle est caractérisée par la présence physique de l'apprenant dans des temps prédéterminés, l'enseignant explique le contenu pédagogique aux étudiants, par conséquent il y a pas des explications et des contenu personnalisés (le contenu pédagogiques doit satisfaire les besoins de chaque étudiant). Ce style classique est considéré comme statique en termes de contenu pédagogique, le temps et le lieu. Aussi l'apprentissage présentielle est très lent, ne répond pas aux besoins de plusieurs étudiants. Pour cela un nouveau style (E-learning) rend la formation rapide, distribuée, just-in-time, dynamique et personnalisée (répond aux besoins des apprenants selon leurs préférences et objectifs). Le tableau suivant exprime une comparaison entre la formation traditionnelle et E-learning [27].

Dimension	Formation traditionnelle	E-learning
Livraison	L'instructeur qui détermine les séquences des connaissances	L'apprenant qui détermine son propre agenda
Réactivité	Anticipation: Suppose de connaître le problème	Réactionnaire: Répond à problème à la main
Accès	Linéaire : séquence de connaissances	Non-linéaire: Permet d'accéder directement à la connaissance dans un ordre quelconque selon chaque situation
Symétrie	Asymétrique: la formation se produit comme une activité distincte	Symétrique: L'apprentissage se fait comme une activité intégrée
Modalité	Discrète : La formation se déroule en morceaux dédiés avec des départs et des arrêts prédéfinis	Continue: l'apprentissage fonctionne dans les boucles parallèles et ne s'arrête jamais
Autorité	Centralisée: le contenu est sélectionné à partir d'une bibliothèque de documents élaborés par l'éducateur	Distribué: contenu provient de l'interaction des participants et les éducateurs

Personnalisation	Produit de masse: Le contenu doit répondre aux besoins d'un grand nombre	Personnalisé: le contenu est déterminé par les besoins de l'utilisateur individuel et vise à satisfaire les besoins de chaque utilisateur
Adaptabilité	Statique: contenu et l'organisation restent sous leur forme originale sans égard aux changements environnementaux	Dynamique : Modifications de contenu sans cesse à travers l'entrée d'utilisateur, les expériences, les nouveaux apprenants, les règles de gestion et l'analyse heuristique

Tableau I.5 : Les différences entre la formation traditionnelle et E-learning [26].

I.6. Topologie de E-Learning

Il existe plusieurs topologies de E-learning, de plus basiques aux plus avancées:

- **les bases de données de connaissances :** Ces bases de données sont la forme la plus élémentaire de E-learning. Ces bases de données sont visibles sur les sites de logiciels offrant des explications indexées et des orientations pour les questions de logiciels, ainsi que des instructions étape par étape pour effectuer des tâches spécifiques. Ce sont généralement modérément interactives [25].
- **support en ligne :** support en ligne se présente sous la forme de forums, chat, babillards en ligne, e-mail ou le soutien direct de messagerie instantanée. Il est légèrement plus interactif que les bases de connaissances [25].
- **Formation multimédia interactive :** Interactif consiste à ce nouvel outil offre plusieurs choix et scénarios ; le logiciel interagit avec l'étudiant et répond à ses propositions, comme dans [54]. Multimédia consiste à l'intégration de graphiques, de musiques, d'effets sonores, de voix, de vidéos et autres animations aux processus de formation, comme dans [53].
- **la formation asynchrone :** Dans ce type de formation, l'échange avec les autres apprenants ou avec les tuteurs s'effectue via des modes de communication ne nécessitant pas une connexion simultanée. Il peut s'agir de forums de discussion ou bien encore de l'échange d'e-mails. Par ailleurs, ce mode de formation repose souvent sur un apprentissage dit "autodirigé", avec des cours, des exercices et des évaluations automatisées, impliquant une certaine autonomie de l'apprenant. Elle est appelée aussi l'autoformation [55].
- **la formation synchrone :** Il se fait en temps réel avec un instructeur de faciliter la formation. Tout le monde se connecte en même temps et peuvent communiquer directement avec l'enseignant et entre eux. Ce type de formation implique la connexion simultanée des participants à une session de formation. Ils peuvent communiquer en temps réel, soit par web-conférence ou visioconférence. Ils peuvent également partager des applications et interagir sur celles-ci au moment où le tuteur leur donne la main sur le document. Le tuteur analyse avec plus d'efficacité les capacités et les connaissances acquises par les apprenants. La classe virtuelle est l'un des exemples de la formation synchrone. Elle permet un échange à distance et en temps réel avec le formateur. Parmi les

autres outils disponibles, on peut aussi citer : Les campus ou universités virtuels d'entreprise, la télévision interactive, les bornes interactives, les simulations, les jeu-formation, les agents intelligents [55].

I.7. Avantages du E-learning

E-learning présente des avantages par rapport la formation traditionnelle en classe. Alors que les plus évidents sont la flexibilité, l'économie de coûts et l'apprenant n'est pas obligé de se déplacer. Ce nouveau monde d'apprentissage a facilité l'engagement dans une formation, et donne une chance d'apprentissage aux gens qui n'ont pas le temps suffisant à faire une formation classique. Elle permet notamment de s'affranchir des contraintes géographiques, mais il y a aussi d'autres avantages [25] :

Le rapport WR Hambrecht¹ présente quelques avantages généraux de E-learning comme vu du côté des entreprises de E-Learning.

- **La technologie a révolutionné les affaires, maintenant il doit révolutionner l'apprentissage** : La mission des entreprises de E-Learning est de fournir la main-d'œuvre avec un programme mis à jour et rentable qui donne des travailleurs motivés et qualifiés.
- **N'importe où, n'importe quand, n'importe qui**: La croissance de la World Wide Web, ordinateurs de bureau, internet haut débit, les réseaux d'entreprise de grande capacité ont permis aux systèmes de E-learning de distribuer la formation et l'information critique à plusieurs endroits facilement et commodément.
- **Des économies substantielles en raison de l'élimination des frais de voyage** : des programmes synchrones ont continué coût associé à la gestion du tuteur de la classe, mais ce coût est toujours inférieure au coût de cours traditionnels.
- **Accès just-in-time à l'information en temps opportun** : produits basés sur le Web permettent aux enseignants de mettre à jour les leçons et les matériaux à travers l'ensemble du réseau instantanément. Cela permet de maintenir le contenu frais et cohérent et donne aux étudiants un accès immédiat aux données les plus récentes. Une revue de formation a rapporté que la formation basée sur la technologie a 50-60 % une meilleure cohérence de l'apprentissage par rapport l'apprentissage traditionnelle.
- **Améliorer la collaboration et l'interactivité entre les apprenants** : Il permet l'interaction entre les apprenants des petits groupes. Dans un environnement E-learning les apprenants ont plus de contact pour passer plus de temps sur le travail en classe, donc ils comprennent mieux les cours.
- **La formation en ligne est moins intimidante** : Les apprenants qui suivent un cours en ligne entrent dans un environnement sans embarras, dans lequel ils peuvent essayer de nouvelles choses et faire des erreurs sans s'exposer. Ce type d'expérience d'apprentissage élimine l'embarras de l'échec en face d'un groupe.
- **L'uniformité de contenu** : L'information délivrée peut être conforme à tous les apprenants.
- **Contenu personnalisé** : L'information peut être développé à des apprenants individuels. Cours et programmes peuvent être créés pour traiter les points forts de chacun et les faiblesses.

- **Contenu mis à jour rapidement** : les mises à jour et les changements de contenu peuvent être effectués et livrés en temps réel. Don, il augmente la vitesse à laquelle la connaissance est acquise.

On résume les avantages de E-learning comme suit [30]:

- E-learning propose une large gamme d'outils pour permettre aux enseignants et aux apprenants d'être innovants, créatifs et ingénieux dans toutes les activités d'apprentissage. Les enseignants et les apprenants peuvent facilement personnaliser les ressources d'apprentissage numériques pour adapter le rythme et le niveau, appropriés à n'importe quel style d'apprentissage et les degrés d'intelligences.
- E-learning offre une formation dans un temps assez court par rapport à la formation classique et on accélère le délai d'apprentissage. Cette formation donne la possibilité de déterminer le rythme et le temps d'apprentissage ainsi les cours à étudier.
- E-learning crée en ligne des communautés de pratique. L'Internet peut apporter les apprenants, les enseignants, les communautés de spécialistes, des experts, des groupes d'intérêt pour partager des idées, des informations et de bonnes pratiques.
- E-learning peut fournir une expérience d'apprentissage individualisé pour tous les apprenants, y compris ceux qui sont défavorisés, handicapés, ont programme spécial ou des besoins personnelles.
- E-learning peut faciliter une plus grande participation et un accès plus équitable à un enseignement supérieur, en créant la possibilité de commencer à apprendre et à choisir des cours et du soutien en fonction des besoins des apprenants.
- E-learning offre un environnement d'apprentissage personnalisé grâce à des informations, des conseils et des services d'orientation. Il peut aider les apprenants à trouver le cours en fonction de leurs besoins et préférences.
- E-learning fournit des mondes virtuels d'apprentissage où les apprenants peuvent apprendre à travers des simulations, les jeux, le contrôle à distance d'outils et de dispositifs du monde réel, ou la collaboration.
- E-Learning n'est jamais en grève, et plus vous l'utilisez, plus son coût relativement bas. Dans une formation à distance on consomme moins des moyens utilisés dans la procédure de formation. La FAD élimine certains coûts (transport, papier, location de salle, gain de temps, ...etc.).

I.8. Outils de E-Learning

Ici, nous discutons trois types d'outils E-Learning : (i) des outils de curriculum, (ii) des outils de la bibliothèque numérique et (iii) des outils de représentation des connaissances [23]. Nous pouvons généralement dire que chaque type d'outil met l'accent sur les différentes parties du processus d'apprentissage. Outils de curriculum offrent un environnement systématique et standard pour soutenir et simuler l'apprentissage en classe (traditionnelle). Outils de bibliothèque numérique facilite l'accès efficace et effective aux ressources et aide à la collection et l'exploitation des ressources tandis que les outils de représentation des connaissances mettent l'accent sur la formulation et la représentation.

I.8.1. Curriculum

Les outils curriculum sont largement utilisés à l'école de l'éducation : école secondaire et l'université. Les outils pédagogiques sont bien organisés pour faciliter les activités d'apprentissage. D'autres outils, tels que des forums de discussion et des quizzes, sont intégrés pour fournir la collaboration et l'évaluation. Un outil typique de curriculum commercial comprend trois parties principales: outils pédagogiques, outils d'administration et outils des apprenants. Outils pédagogiques comprennent la conception de curriculum et des quizzes avec évaluation et classement automatisés. Les outils d'administration comprennent l'authentification, gestion de fichiers, et l'autorisation. Les fonctions de l'outil des apprenants sont [23]:

- Navigation du contenu: des lectures, des tests et examens, des projets, d'autres ressources.
- Collaboration et partage : babillards asynchrones et synchrones et forums de discussion.
- Programmes d'apprentissage et de suivi : rappels d'affectation et de la soumission, agendas personnels, et les listes d'activité.
- Test et évaluation automatiques : les tests sont conçus par des tuteurs pour évaluer le rendement des apprenants.

Exemple de Curriculums : WebCT et Blackboard sont des outils les plus populaires curriculums commerciaux. Une étude de comparaison des deux outils a présenté que Blackboard est plus flexible souple gestion de contenu et supporte le travail de groupe, donc Blackboard fournir un apprentissage autonome et collaborative. WebCT a une structure étroite et contient plusieurs outils de soutien et d'orientation qui aident à bien guider les apprenant, mais l'apprentissage est moins indépendante (moins autonome). En général, ces outils sont adaptés plus pour soutenir les activités de classe que des recherches indépendantes et apprentissage indépendant [22].

I.8.2. Bibliothèque numérique

Bien que les outils de curriculums supportent les activités de classe, les outils de bibliothèque numérique se concentrent sur la recherche de ressources et de documents. Ces fonctions prennent en charge les phases d'exploration et de collection de la recherche d'information. Les outils de bibliothèque numérique aident les utilisateurs à trouver la bonne information au milieu d'une énorme quantité de documents numérique. Les fonctions de bibliothèque numérique incluent habituellement la recherche, la navigation et la découverte des collections ou des répertoires spéciaux. La recherche et la navigation sont utilisées pour localiser des ressources. Les collections spéciales contiennent des contenus organisés qui représentent un trésor unique pour les utilisateurs intéressés [23].

I.8.3. Outil de représentation des connaissances

L'outil de représentation des connaissances aide les apprenants à la révision visuelle, la capture, ou l'acquérir des connaissances. Outils de curriculum reposent principalement sur une approche de mode textuel pour décrire le contenu des cours. Cette approche échoue souvent de décrire la relation entre les concepts et les compétences visées dans un cours. Elle n'exprime pas la base de connaissances que l'apprenant va acquérir à la fin de son programme

d'études. Un outil de visualisation peut engager les apprenants et les enseignants dans un processus d'apprentissage actif, ils construisent une sémantique spatiale de la connaissance, des concepts et des compétences que l'apprenant va apprendre et acquérir [21].

I.9. Les limites de la formation à distance

Cependant les limites de la FAD sont clairement répertoriées et sont autant de défis dans l'élaboration d'une offre de formation de qualité :

I.9.1. Le sentiment d'isolement

On peut voir qu'une des problématiques essentielles concerne les relations entre l'apprenant et le formateur. Dans ce type d'enseignement l'apprenant est isolé. Cet isolement peut démotiver les apprenants. Plusieurs tentatives de résoudre cette limite comme la recherche des nouvelles formes d'accompagnement que suppose la mise en place de tutorat de proximité. Une autre solution qui a été proposée par Pierre Moeglin, pour lui "la médiation pédagogique" est au centre des préoccupations actuelles [63].

I.9.2. Les difficultés à maîtriser une méthode de travail inconnue

Pour plusieurs étudiants, il est difficile d'être familiers Avec ce nouveau style d'apprentissage donc on ne doit pas " lâcher " quelqu'un dans un cycle à distance sans encadrement. Dans chaque cours on peut ajouter des guides pour aider l'apprenant à suivre son cours.

I.9.3. La difficulté pour certains publics à travailler seuls

Les caractéristiques des personnes diffèrent d'un à autre, Notamment l'intelligence, l'apprenant qui est considéré moins intelligent ou moins organisé peut être incapable à comprendre ses cours et incapable à organiser son temps.

I.9.4. La conception des ressources

Le temps de conception des ressources est extrêmement long lors d'approches réellement interactives. La phase de la création des cours et un processus difficile et généralement exige plusieurs compétences [63].

1.10. Les enjeux du E-Learning

Dans un Environnement de E-Learning, les développeurs s'intéressent, généralement, à l'objet de plusieurs enjeux. Ici, nous nous intéressons aux enjeux liés aux apprenants, enseignants et le contenu éducatif, et nous ne parlons pas aux enjeux financiers et économiques [37] :

- Les enjeux liés à l'efficacité et à l'adaptabilité des processus d'apprentissage ;
- Les enjeux liés à l'accès à la connaissance ;
- Les enjeux liés à l'autonomie de l'apprenant ;

- Les enjeux liés à l'accompagnement de l'apprenant ;
- Les enjeux liés aux nouveaux rôles de l'enseignant ;
- Les enjeux liés au développement des technologies éducatives.

1.10.1. L'efficacité et l'adaptabilité des processus d'apprentissage

L'un des défis posé par les nouveaux styles d'apprentissage est l'adaptabilité. La volatilité incroyable des marchés de nos jours demande des méthodes pour assister le besoin de savoir des apprenants de E-Learning. Il est aussi clair que ce nouveau style d'apprentissage est guidé par les exigences de la nouvelle économie : Rapide, juste à temps et pertinent. Ainsi, les processus d'apprentissage doivent être efficaces, rapides et appliqués juste à temps. Les apprenants qui suivent un processus de formation sont, généralement, hétérogènes au point de vue intelligence, capacité, background, personnalité, préférences ...etc. Donc, il faut fournir des mécanismes puissants pour organiser une telle formation. Aussi, l'apprentissage doit être un service en ligne adaptable, initié par les profils des utilisateurs.

1.10.2. L'accès à la connaissance

Une organisation de E-learning doit fournir un accès à la connaissance plus efficace, plus solide, plus adapté aux rapides évolutions. Une conception harmonieuse d'un dispositif de E-formation doit permettre de bénéficier tant des avantages des technologies éducatives, comme l'interactivité ou la simulation, que des avantages de la formation à distance avec ses possibilités d'une plus grande autonomie et l'élimination de certaines contraintes (temps, lieux) liées à la formation, et ceci en éliminant les principaux inconvénients comme les risques de déshumanisation de la formation [56].

1.10.3. L'autonomie de l'apprenant

L'apprenant doit avoir un certain degré d'autonomie pour organiser des tâches, à les mener à bien, seul ou en équipe, et à les évaluer, identifier et gérer des ressources, travailler avec des pairs, maîtriser les techniques et outils d'apprentissage, ...etc.

Par ailleurs, la question d'autonomie ne dépend pas uniquement des apprenants car il faut d'abord créer le cadre où peut s'exercer cette autonomie. Dans une plate-forme de E-Learning, l'autonomie s'évalue en termes de prise de contrôle de l'apprenant sur les ressources offertes par la plate-forme. Le besoin d'autonomie est causé par l'isolement et l'absence de relation (face-à-face) avec les autres acteurs. L'autonomie s'exprime aussi par la possibilité de choisir son temps de formation et entraîne une interaction différée avec les autres acteurs (synchronie/asynchronie), elle se traduit par une diversification des interactions pédagogiques accessibles et des modes d'apprentissage (formation tuteurée, formation coopérative, autoformation accompagnée, formation autodirigée).

1.10.4. L'accompagnement de l'apprenant

Pour faire face au problème de l'isolement de l'apprenant, un tel système éducatif doit avoir des méthodes d'accompagnement humain et technique de l'apprenant. Mais, l'accompagnement s'avère particulièrement complexe en raison de l'interpénétration de

multiples espaces-temps sociaux de l'apprenant. Ces espaces-temps sociaux sont ceux liés aux situations :

- Individuelles et collectives de formation proprement dites ;
- De relations avec l'institution de formation ;
- Professionnelles et personnelles.

Donc pour fournir un bon accompagnement, une plate-forme de E-Learning doit intégrer des dimensions non seulement pédagogiques mais aussi organisationnelles, professionnelles et personnelles [37] :

- Information sur le processus, les ressources, les modalités d'apprentissage ;
- Communication et coordination entre les acteurs (forum, tchat, téléconférence, ...) ;
- Ajustement du planning des activités au cours de la formation ;
- Suivi, guidage et régulation (carnets de bord, aide à l'articulation entre les apprentissages, aide à l'allocation des ressources, ...) ;
- Construction et gestion de parcours (outils d'autodiagnostic, logiciels de gestion de parcours, ...) ;
- Adaptation de processus d'apprentissage et de contenu éducatif selon les préférences de l'apprenant et les caractéristiques personnelles.

1.10.5 Les nouveaux rôles de l'enseignant

L'accompagnement est assuré par l'enseignant humain ou tuteur logiciel. Les fonctions de l'accompagnement nécessitent une évolution des compétences des formateurs, une coordination et un pilotage pour intégrer et réguler les accompagnements. Cela exige la modification du rôle de l'enseignant; E-Learning n'est pas simplement une révolution technique, elle se rapporte plutôt davantage à la reconceptualisation de l'enseignement, de la formation. Alors, Le nouvel enseignant devra donc devenir un modérateur capable d'aider les apprenants dans leurs nouveaux environnements de formation. Pour cette raison, les enseignants doivent acquérir de nouvelles compétences, qui ne sont pas uniquement techniques et technologiques, mais encore pédagogiques aussi bien qu'éthiques. Les enseignants ne sont plus appelés désormais à travailler isolément mais au contraire à partager leurs expériences et à s'enrichir mutuellement de leurs propres acquis, à prendre conscience des potentialités qu'offre la diversité [57].

1.10.6. Le développement des systèmes du E-Learning

L'un des enjeux majeurs est le développement des systèmes pour E-Learning. Maintenant, il existe plusieurs plates-formes de E-learning chacune essaye répondre aux besoins spécifiques sans chercher à adapter l'une ou l'autre plate-forme existante. Il est néanmoins clair que l'on épuise ainsi une quantité très importante de ressources pour arriver à un résultat qui est très souvent en dessous de ce qui existe déjà.

I.11. Document pédagogique et hypermédia

Dans nos jours, les documents de web ne sont plus seulement textuels, plusieurs documents multimédia (texte, image, son, vidéo ...) de base sont utilisés. L'apport et les bénéfices du multimédia s'avèrent très évident, notamment dans les systèmes de E-Learning. Souvent, l'apprenant comprend et mémorise plus facile une information multimédia que une information textuelle. D'autre côté, les composants hypertexte sont très importants dans un système éducatif puisqu'ils aident beaucoup l'apprenant à mieux représenter la connaissance et favorisent l'initiative de l'apprenant. Dans un processus d'apprentissage, nous pouvons utiliser des documents multimédias et textuels.

I.11.1. Document pédagogique

Une définition du document pédagogique plus utile est présentée dans [58]: "Un document pédagogique est un regroupement d'objets reliés entre eux par des relations de différentes nature (logique, spatiale, hypertexte et temporelle)". Une autre définition présentée par IEEE (Institute of Electrical and Electronics Engineers): "objet pédagogique comme une entité numérique ou non, qui peut être utilisée, réutilisée ou référencée dans toute activité liée à l'enseignement ou à l'apprentissage" [59]. Donc le document pédagogique est un ensemble de nœuds (objets, pages) reliés entre eux par des liens pour faciliter la navigation des objets, on peut utiliser et réutiliser ces objets pédagogiques dans des cas différents.

I.11.2. Hypermédia

La fusion des techniques hypertextes avec celles de Multimédia donne l'hypermédia. Dans le cadre éducatif, l'hypermédia offre des avantages considérables. En effet, le composant multimédia peut améliorer l'aspect visuel et ludique de l'apprenant et renforcer son intérêt. Aussi, l'intégration des composants hypermédia peut améliorer la qualité de l'apprentissage. Maintenant l'hypertexte et l'hypermédia sont considérés comme synonymes. Le segment de données, ne comporte plus que du texte, mais aussi des images fixes, des images animées, des séquences vidéo et du son.

I.11.3. Entre hypertexte et hypermédia

Un hypertexte est caractérisé par sa structure non linéaire et il est constitué de nœuds et des liens. Un nœud représente la page (informations textuelle) de l'hypertexte et le lien est considéré comme un pont entre les nœuds (pages). L'utilisateur peut afficher le contenu de la page en activant ce lien. La différence majeure entre l'hypermédia est le contenu des nœuds [64]. Dans un système hypermédia les nœuds contiennent des données textuelles et des média divers, tels que des images, des séquences animées, des vidéos, ...etc. Cependant, pour montrer l'intérêt de la structure globale de hyperespace et n'est pas du contenu, généralement, les auteurs utilisent indifféremment les termes hypertextes ou hypermédia.

I.11.4. Hypermédia adaptatif

Pour améliorer la qualité de l'apprentissage, des recherches ont alors essayé de créer des hypermédias adaptatifs. L'objectif de ces systèmes est d'adapter la présentation de la connaissance et d'aider l'apprenant à se diriger dans l'hyperespace. De ce fait, dans un hypermédia nous devons pouvoir modifier aussi bien le contenu des pages que les liens entre les différentes pages [65]. Mais c'est surtout sur l'adaptation des liens que le plus grand nombre de techniques ont été développées

Edmonds [66] distingue les trois types de systèmes intégrant l'adaptation :

- Les systèmes **adaptés** dans lesquels l'adaptation est l'œuvre du concepteur lui-même qu'il implante après une phase de tests. Dans ce cas, l'adaptation ne peut pas alors être propre à chaque individu.
- Un système **adaptable** est un système qui peut être modifié sur demande explicite de l'utilisateur. Celui-ci peut choisir l'aspect physique des éléments de l'interface, à l'aide par exemple de tableaux de bord. Il peut également demander la création de macro-commandes
- Un système **adaptatif** est un système qui s'adapte de lui-même aux besoins de l'utilisateur. Des mécanismes de suivi (tracking) des comportements de l'utilisateur permettent de connaître les besoins de ce dernier en fonction de l'environnement, de son état psychologique, de ses connaissances ...etc.

I.11.5. Techniques d'adaptation

Puisque les besoins et les préférences des apprenants sont hétérogènes et les capacités d'intelligence n'ont pas les mêmes pour tous les apprenants, l'adaptation devient indispensable dans un système éducatif. Un hypermédia adaptatif permet de mieux guider l'apprenant dans son processus d'apprentissage. Un document hypermédia est composé de pages et de liens. De ce fait, nous pouvons adapter le contenu des pages ou les liens entre ces dernières. Selon Brusilovsky [65], les techniques d'adaptation sont de deux sortes, ceux qui adaptent le contenu présenté à l'apprenant (adaptation de contenu et présentation) et ceux qui adaptent les liens présentés à l'apprenant (adaptation de navigation ou de lien).

I.11.5.1. Adaptation du contenu

Cette technique propose l'adaptation du contenu des pages de l'hypermédia en fonction des caractéristiques, des volontés, des préférences et des buts de l'apprenant. Ainsi, les apprenants qui accèdent à une même page, mais en ayant des profils différents, doivent visualiser en fait des pages différentes. Parmi les systèmes qui adaptent le contenu présenté se distingue trois types [65] :

- Adaptation du texte par une adaptation des fragments : par sélection, filtrage, l'ordonnement des fragments.
- Ceux qui choisissent le média le plus approprié ;
- Adaptation du mode de présentation.

I.11.5.2. Adaptation des liens (navigateur)

L'utilisation de cette adaptation aide l'utilisateur à se repérer dans l'hypermédia ou à l'obliger à utiliser certains liens plutôt que d'autres. Dans ce cadre d'adaptation de lien, il y a plusieurs techniques : le guidage direct, l'ordonnement des liens, le masquage des liens, l'annotation des liens ou encore les cartes adaptatives [65] et [67] :

- **Guidage direct** : Elle base sur l'ajout d'un lien hypertexte, nommé souvent « suivant » (ou next en anglais), qui permet d'accéder à la meilleure page, c'est-à-dire celle qui est en adéquation avec les objectifs et/ou capacités de compréhension de l'utilisateur. On peut utiliser cette technique en laissant les autres hyper liens existants au préalable ou en les supprimant.
- **Ordonnement des liens** : Cette technique propose d'afficher les liens hypertextes suivant un ordre définissant l'intérêt ou l'importance des pages cibles. Cette technique ne peut pas être utilisée dans tous les cas. Certaines études ont en effet montré que l'instabilité d'une liste de liens pour une page donnée pouvait désorienter l'apprenant. Et d'autres études ont montré qu'elle pouvait réduire les temps de navigation des hypermédias surtout ceux axés sur la recherche d'informations.
- **Masquage des liens** : Dans cette technique on supprime les liens dont les pages cibles sont soit en inadéquation avec le modèle de l'utilisateur (trop simples ou trop compliquées), soit en inadéquation avec les objectifs de l'utilisateur. Elle est facile à mettre en œuvre, puisqu'il suffit, avant d'envoyer la page à l'utilisateur, d'enlever les liens non désirés. Mais elle peut poser un problème chez l'apprenant à une mauvaise représentation mentale, donc on peut supprimer les liens adéquates.
- **Annotation des liens** : Ici pour chaque lien on adjoint des annotations (explications) sur la page cible ou alors définir une syntaxe ou un codage particulier (par exemple telle icône pour dire que c'est une aide, telle couleur pour dire qu'il s'agit d'un exemple, etc.). Cette technique, assez simple à mettre en œuvre, peut être utilisée pour tous les types de liens, et ne semble pas avoir d'effets de bord néfastes.
- **Cartes adaptatives** : Pour réaliser cette technique on doit représenter l'organisation de l'hypermédia dans une carte (map en anglais), à l'aide de liens, soit sous forme textuelle (dans ce cas nous avons souvent une présentation hiérarchique de l'hypermédia), soit sous forme graphique. Il est possible de présenter à l'utilisateur une organisation plus ou moins simplifiée en fonction de son profil. Cette technique aide les apprenants d'être bien guidés et les enseignants de mieux structurer leurs documents pédagogiques.

I.11.6. Systèmes hypermédias adaptatifs dynamiques

L'objectif principal de ces systèmes est d'améliorer la qualité d'adaptation. Ils sont particulièrement caractérisés par le fait d'offrir un hypermédia virtuel. Le système dans ce cas n'est pas constitué de pages et de liens prédéfinis. Ces derniers sont construits dynamiquement et doivent adapter leur offre de formation de manière dynamique, en fonction des règles pédagogiques et des réactions des apprenants. Ces systèmes donnent un moyen d'accès intuitif et non-linéaire à l'information et facilitent la navigation [64]. Ils sont

généralement constitués de trois composants essentiels: modèle du domaine, modèle apprenant, et générateur de cours.

I.12. Plate-forme de Formation à Distance

On peut définir Les plates-formes de formation à distance comme étant des solutions technologiques informatiques particulières créées par des entreprises privées ou publiques ou issues de travaux de recherche [68]. L'apparition de ces plates-formes a été dans les années 90 pour ambition d'aider les concepteurs et formateurs à mener à bien l'essentiel des fonctions pédagogiques impliquées par la formation à distance [68] :

- Production et intégration des ressources pédagogiques (création de cours).
- Présentation de l'offre et des programmes de formation (bibliothèque de formation).
- Diffusion et accès aux ressources.
- Positionnement, construction et gestion des parcours de formation individualisés (gestion des compétences de l'apprenant).
- Animation des personnes et des groupes (accompagnement de l'apprenant en synchrone ou asynchrone).
- Administration financière, technique.

Les travaux de mise en place d'une plateforme de la formation à distance sont incontournables. Donc le nombre de plateformes de la formation à distance ne cesse de croître, nous ne pouvons pas toutes les recenser. A titre d'exemple, le site Thot [70] recense plusieurs centaines de logiciels de ce type [69]. En 2007, l'organisme Thot a recensé 233 plateformes, 47 plates-formes open-source, 11 plates-formes publiques et 175 plates-formes commerciales [39].

I.12.1. Quelques standard

Maintenant il n'y a pas une seule norme ou un seul standard pour la formation à distance mais il y a des groupes et des comités qui travaillent sur la définition d'une spécification pour des applications particulières ou pour toutes les applications de FAD. Le problème de la standardisation est l'interopérabilité de plateforme c'est à dire l'échange des informations en plus haut degré de transparence.

I.12.1.1. IMS global learning consortium (Instructional Management System global learning consortium)

IMS Global Learning Consortium (1997) est une organisation de standards qui est concernée avec l'établissement de l'interopérabilité pour les systèmes de formation. IMS a spécifié des métadonnées pour l'étiquetage des ressources d'enseignement et d'apprentissage et élaboré diverses autres spécifications relatives notamment aux contenus, au classement, au design pédagogique, aux profils des apprenants et aux besoins des entreprises (*IMS Content Packaging, IMS Questions and Tests, IMS Meta-data, IMS Learning Design ...etc*) [37]. Leur principale activité est le développement des spécifications techniques et les standards pour les applications des domaines suivants:

- Description de matériels pédagogiques (catalogage, métadonnée, granularité);

- Interopérabilité de matériels et plates-formes.
- Gestion et contrôle de l'apprenant.

Des techniques standard ont été développées pour chacun de ces points. Ils offrent la DTD et des schémas XML. IMS travaille en collaboration avec AICC et a adopté des spécifications AICC s'appliquant à l'interopérabilité des applications de FAD.

I.12.1.2. SCORM (Shareable Course Object Reference Model)

SCORM a été fondé sur l'initiative du Ministère de la Défense nationale d'US pour le but de définition et la standardisation d'un modèle de contenu de FAD. Les objectifs de SCORM sont comme suit :

- garantir le transfert de contenu d'une plateforme à une autre, spécialement, toute qui concerne l'apprenant.
- Promouvoir la création de matériels réutilisables et granulaires.
- Faciliter la recherche automatique de matériels pédagogiques pour la formation sur les bases de données et sur le Web.

I.12.1.3. IEEE (Institute of Electrical and Electronics Engineers)

Les groupes AICC, ADL, IMS, ARIADNE, PROMOTEUS, aussi bien que DCMI, LRN 2.0... etc, sont concernés par tous les domaines impliquant E-formation. Leur objectif complet est d'atteindre l'accord et créer des standards communs. C'est pourquoi les spécifications attachées avec tous les secteurs de formation en ligne (métadonnée, des technologies, le contenu... etc.) sont par la suite présentées à l'IEEE. Par le LTSC (Learning technology Standards Committee), c'est-à-dire le comité spécialisé en FAD.

Le Learning Object Metadata (LOM) est un schéma de description de ressources d'enseignement et d'apprentissage. Le LOM peut être utilisé pour décrire des ressources tant numériques que non numériques. Techniquement, son nom est IEEE 1484.12.1-2002 (LOM) [35]. Ce standard a été conçu, en 2002, par le comité IEEE-LTSC-LOM, Institute of Electrical and Electronics Engineers, Inc. - Learning Technology Standards Committee - Learning Objects Metadata Working Group, organisme international regroupant des représentants du Canada, des États-Unis, de plusieurs pays d'Europe, d'Afrique, d'Amérique latine, d'Asie et de certaines régions du Pacifique. En France, le LOMFR2 correspond à la norme française NF Z76-040 publiée par l'AFNOR en décembre 2006.

Le LOM a été, dès ses débuts, proposés pour suivre un processus de normalisation au niveau ISO. Le projet de normalisation s'est en définitive développé en optant pour des choix de modélisation et des options techniques différentes de ceux proposés par le LOM. Ce processus a abouti en 2011 à la publication de la norme ISO/IEC 19788 - Metadata for Learning Resources et conduit certains pays à établir des correspondances entre le LOM et cette nouvelle norme [52].

I.12.1.4. ISO

La standardisation est organisée au niveau international par l'ISO, l'Organisation Internationale de Standardisation, qui comprend de près sur 130 pays. L'Europe est représentée par le Comité Européen de la Normalisation (CEN, établie à Bruxelles en 1961). L'autorité appropriée en France est l'Association Française de la normalisation (AFNOR). Pendant une réunion de l'ISO en 1999, il a été décidé pour créer un sous-domaine " Learning Technology ", le Sous-comité 36 (SC36) dont le titre exact est : la normalisation des technologies de l'information destinées à l'apprentissage, à l'enseignement et à la formation. Son fonctionnement est sur la standardisation de thème dans le domaine de technologies de l'information pour étude, éducation et formation pour soutenir individus, groupes, ou organisations et permettre interopérabilité et réutilisabilité de ressources et outils. Le but de SC36 est de travailler sur la standardisation ouverte de méthodes de médiation et pas en enseignement de ressources, et il se compose de plusieurs groupes qui œuvrent, entre autres, sur le vocabulaire, les technologies collaboratives, l'architecture, les métadonnées, ainsi que les adaptations culturelles, linguistiques et fonctionnelles [37].

I.12.2. Quelques Plateformes

Dans cette section, nous allons décrire en détail des systèmes ou des plates-formes dédiés au E-Learning:

I.12.2.1. Claroline

Claroline [71] est une plateforme libre et gratuite. Elle est développée en 2002 par l'université de Louvain en Belgique pour la formation à distance et de travail collaboratif. Elle permet aux formateurs de créer des espaces de cours en ligne et de gérer des activités de formation sur Internet. Traduite en 35 langues, Claroline [71] bénéficie de l'appui d'une communauté mondiale d'utilisateurs et de développeurs.

La plateforme Claroline est utilisée par des institutions issues de 84 pays, elle permet de créer sans coût de licence des espaces de travail et des cours en ligne. Pour chaque cours, le formateur dispose d'une série d'outils lui permettant de [39]:

- ❖ Rédiger une description du cours;
- ❖ Publier des documents dans tous les formats nécessaires (texte, PDF, HTML, vidéo...)
- ❖ Administrer des forums de discussion publics ou privés
- ❖ Elaborer des parcours pédagogiques au standard SCORM 1.2 ou composés de documents.
- ❖ Créer des groupes de participants ayant des documents en commun et des forums privés.
- ❖ Composer des exercices (QCM)
- ❖ Structurer un agenda avec des tâches et des échéances
- ❖ Publier des annonces (envoyées aussi par messagerie électronique)
- ❖ Proposer des travaux à rendre en ligne
- ❖ Consulter les statistiques de fréquentation et de réussite aux exercices

Après une période d'incertitude, suite au départ de son fondateur : Thomas de Praetere parti fonder le produit dérivé **Dokeos** (plateforme basée sur Claroline mais évoluant vers un style plus « entreprise », plus « nouvelles technologies ») [69].

Récemment un style plus professionnel et plus moderne a été adopté par la plateforme, elle utilise un site web fonctionnel et très complet, des services payants sont proposés pour des hébergements, des formations, Ce qui concerne la documentation en ligne est traduite en plusieurs langues et fournit des présentations multimédias interactives. Enfin depuis la version 1.8, Claroline structure ses développements en modules afin d'assurer une meilleure gestion des mises à jour et de gagner en flexibilité et en souplesse. Une méthode déjà largement développée par Moodle et aujourd'hui en passe de devenir l'unique méthode d'enrichissement « gérable » d'une plateforme. Les points faibles de cette plateforme peuvent être résumés comme suit [39]:

- La capacité de stockage de dossier documents et liens du groupe est petite (7 Mo).
- L'inscription d'un étudiant est possible, mais l'étudiant ne peut pas se désinscrire.
- « Chat » ou messagerie instantanée : l'outil n'est pas très ergonomique mais on peut maintenant grâce aux modules changer l'outil et mettre celui de son choix.
- On ne peut pas copier les informations qu'il a créées dans un autre espace (planning, annonces, pages html).
- Manque des outils synchrones, il existe seulement l'outil « Discussion », une fonctionnalité comme un tableau blanc ou un outil de visio-conférence serait complémentaire.
- Le parcours des fenêtres exige une amélioration parce que pour atteindre au contenu il faut parcourir trop d'écrans. Si la ressource a beaucoup d'items, il n'y a pas de menu dynamique, cela fait beaucoup de « scrolls » et de clics pour suivre le parcours.

I.12.2.2.Ganesha

Ganesha [72] est une plateforme de téléformation ou LMS (Learning Management System) créée en 2001 et éditée par la société de formation spécialisée en e-learning ANEMA [72]. Elle est open source et gratuite. La mise au point de la plateforme Ganesha a été faite par des spécialistes de l'approche de la formation à distance par Internet. Ganesha a connu un grand succès chez les organismes de formations institutionnels ou privés, l'éducation nationale, des agences multimédia et des organismes consulaires dans le monde entier. L'un de ses points forts est l'intégration de contenus « sur étagère » : des parcours pédagogiques ou des questionnaires qui ont été créés avec des logiciels ou des éditeurs externes à la plateforme [73].

Les parcours pédagogiques sont constitués de séquences par le créateur de cours. Le contenu est caractérisé par une granularisation extrême qui permet une individualisation de l'apprentissage. On fait appel intensivement au processus d'ingénierie pédagogique dans la création de parcours. L'apprenant est intégré dans des groupes ou classes à qui on affecte des parcours pédagogiques. L'écran majeur dans Ganesha est le tableau de bord, en opposition au plan du cours dans d'autres logiciels. Les plate-formes universitaires s'organisent autour d'un corpus de documents et d'une communauté d'apprentissage. Elle est peu orientée vers la gestion des compétences et la gestion de communautés [73].

Puisque les entreprises ont intervenu pour financer le développement de fonctionnalités pour la plateforme, Ganesha débute une nouvelle étape dans son développement. Cette plateforme permet à un formateur ou un service de formation, dans le cadre d'une formation à distance ou pour enrichir le présentiel, de mettre à la disposition d'un ou plusieurs groupes de stagiaires, un ou plusieurs modules de formation avec supports de cours, compléments, quiz et tests d'évaluation ainsi que des outils collaboratifs. Ses autres fonctionnalités sont [39] :

- La possibilité d'avoir une messagerie personnelle dans la plateforme pour envoyer des pièces jointes, cela permet aux apprenants de contacter entre eux.
- Un forum permet aux stagiaires et aux tuteurs de poster des messages qui seront accessibles à l'ensemble des membres du groupe de formation, de répondre aux messages déjà postés et ainsi engager une discussion sur un sujet donné.
- La possibilité d'entrer d'une session de chat (ou messagerie instantanée) qui permet à l'ensemble des membres du groupe de discuter en temps réel.
- La possibilité de poster des documents pédagogiques dans une zone de dépôt qui permet de proposer des documents sous format numérique à l'ensemble du groupe, et de laisser des commentaires sur les documents postés.
- Un quizeur Flash afin de réaliser des questionnaires à choix multiples ou simples QCM/QCU à partir de la Plateforme. Les points faibles de cette plateforme peuvent être résumés comme suit [39]:
- Les cours sont paramétrés par l'administrateur et les formateurs-tuteurs ne peuvent pas les paramétrer, ainsi que l'administrateur de la plateforme peut organiser les formations en ligne contrairement à beaucoup d'autres plateformes. C'est un choix organisationnel à prendre en compte.
- La limitation d'envoi d'un seul fichier à la fois dans la Messagerie interne.
- Les fonctionnalités du forum sont réduites (le forum permet de poster des messages accessibles à l'ensemble des membres).
- Le Chat est peu ergonomique et n'a pas de traces
- Dans cette plateforme Wiki, blog, vidéoconférence ne sont pas implémentés. Planning, agenda, non implémentés aussi.
- La plateforme ne fournit pas le cryptage des mots de passe.
- Connexion LDAP non fonctionnelle.

I.12.2.3. Moodle

Le terme Moodle [74] est l'acronyme de "Modular Object-Oriented Dynamic Learning Environment", mais il veut aussi dire « flâner » en anglais. La conception de la plateforme Moodle a été influencée par les travaux dans le cadre de ses recherches doctorales de Martin Dougiamas (auparavant administrateur de la plateforme WebCT). Martin Dougiamas a étudié les apports du constructivisme social dans la pédagogie en ligne. Moodle est considérée comme une plateforme d'apprentissage en ligne servant à créer des communautés d'apprenants autour de contenus et d'activités pédagogiques. Elle est Dotée d'un système de gestion de contenu - (SGC) performant [39].

Moodle est une plateforme de la formation Ouverte et à Distance (FOAD), afin de favoriser un cadre de formation Socio-constructiviste. Tout ce que vous lisez, voyez, entendez, ressentez et touchez est comparé à vos connaissances antérieures et si cela est viable dans votre monde mental, cela pourra former une nouvelle connaissance qui vous appartiendra. L'utilisation des connaissances dans l'environnement renforce ces connaissances. Cela met l'accent sur le fait qu'il n'y a pas seulement un transfert d'information d'un cerveau à un autre, mais que tout est conditionné par l'interprétation et l'interaction avec un groupe social [39]. Les points faibles de cette plateforme qui peuvent être mentionnés sont [39]:

- Le test de positionnement est automatique, mais gestion manuelle, pas d'affectation automatique de parcours
- Il n'existe pas un agenda personnel.
- Communication en mode synchrone : clavardage mais pas de mode vidéoconférence sauf si l'on rajoute un module.
- Moodle est une plateforme très riche en fonctionnalités, sa prise en main par les apprenants peut nécessiter un temps d'adaptation car les pages peuvent être très chargées en informations.
- La complexité et la diversité et la spécificité de tous les paramètres des outils

I.12.2.4. Sakai

En 2004, quatre universités américaines (Indiana University, Massachusetts Institute of Technology (MIT), Stanford University et University of Michigan) ont lancé le projet Sakai [75] afin de construire une plateforme d'apprentissage complète. Les systèmes de gestion de cours utilisés par ces universités sont différents, souvent développés en interne. Au groupe se sont joints des membres de uPortal [75]. Sakai est un projet, auquel sont rattachés une fondation, un comité de direction et des partenaires institutionnels et commerciaux ; une communauté qui regroupe plusieurs institutions qui coopèrent et maintiennent Sakai. Elle regroupe actuellement plus de 80 universités dans plusieurs pays dans le monde. Sakai est une plateforme utilisable dans un contexte d'éducation et de formation basée sur une structure ouverte et extensible, incluant une suite d'outils pour le support de l'apprentissage, de la collaboration et de la recherche [39].

Sakai essaie de développer une plateforme complète de gestion de cours libre de qualité équivalente ou supérieure aux autres produits du marché. Au premier temps, cette plateforme a été orientée vers institutions universitaires. Ce sont ces dernières qui définissent les orientations, les spécifications et les priorités quant au développement du projet [39].

La documentation de la plateforme Sakai est en Anglais, les utilisateurs qui ne parlent pas cette langue ne peuvent pas facilement l'utiliser. Cependant beaucoup d'institutions au Québec s'intéressent au projet Sakai, ils ont fondé une association à but non lucratif : Sakai Québec [76] et sont en train de développer une traduction en français et en arabe pour les pays maghrébins du projet POLLES. Les points faibles de cette plateforme peuvent être résumés comme suit [39]:

- L'interopérabilité est l'un des points qui n'existent pas dans Sakai. Sakai est une plateforme en pleine mutation mais qui évolue rapidement.

- Le SCORM n'est pas encore implémenté.
- Le problème de la langue de la documentation car Sakaïpedia est aujourd'hui, exclusivement en anglais, cela peut gêner les utilisateurs qui ne parlent pas cette langue.
- Les importations de modules IMS 1.1.2 ne sont pas toujours opérationnelles, cela dépend des logiciels auteurs.
- Les administrations de Sakai doivent maîtriser Java car le langage de développement de cette plateforme est Java, ainsi que l'installation est complexe.
- Les points qui n'existent pas dans cette plateforme sont: vidéoconférence, traduction en français, portfolio, intégration du SCORM, de CanCore1, de NORMETIC2.

I.12.2.5. Spiral

La solution SPIRAL [77] est une solution développée par le service PRACTICE de l'Université Claude Bernard Lyon 1 dont la première version a été délivrée en octobre 2003. SPIRAL est une solution 100% Web permettant la création et la diffusion de modules de formation sur Internet

Après identification, les utilisateurs accèdent à un espace de travail personnalisé suivant leurs profils et leurs droits. Cette plateforme présente d'avantage d'utilisation d'être gratuite.

I.12.2.6. WebCT

Le développement de cette plateforme a été réalisé par l'université de British Columbia au Canada. Elle est considérée parmi l'une des plateformes les plus vendues dans le monde. Plusieurs universités dans différents pays utilisent cette plateforme. A titre d'exemple les universités suivantes en France utilisent cette plateforme: Nancy (vidéoscop), Paris3, le CNAM (Paris), l'ENST de Brest, Paris 6, ...etc. Elle fournit plusieurs versions: Anglais, Espagnol, Finlandais, Français, Hollandais. L'apprenant peut contacter les autres apprenants et ses enseignants grâce aux outils de communication qui sont à sa disposition : tchat, le forum, la messagerie et le tableau blanc.

I.12.3. Synthèse sur les plates-formes présentées

Le tableau suivant présente une grille récapitulative des points clés par plateforme :

	Claroline	Ganesha	Moodle	Sakai
Communauté, dynamisme, international	positive	négative	positive	positive
Documentation	positive	négative	positive	négative
Gestion du nombre d'utilisateurs	positive	négative	positive	positive
Outils collaboratifs	positive	positive	positive	positive
Adaptabilité et modularité	négative	négative	positive	positive
Intégration de spécifications et standards	négative	négative	positive	négative
Installation et gestion	positive	négative	positive	positive
Adaptation de la charte graphique	positive	positive	positive	positive
Systèmes d'exploitation; navigateurs, plug-ins clients	positive	négative	positive	négative
Ergonomie et utilisabilité	positive	positive	Point de vigilance	Positive
Résultats d'évaluation	positive	3	9	7
	vigilance	0	1	0
	négative	2	0	3

Tableau I.10.2.a : Grille récapitulative des points clés par plateforme [39].

	Outils Collaboratifs	Notion de groupe	Méthode de formation de groupe	Paradigme objet ou agent	Architecture	Suivi de l'utilisateur
Sakai	Oui	Oui	Non	Objet	Client/serveur	Suivi minimal
Claroline	Oui	Oui	Non	Objet	Client/serveur	Récolte les traces et les structures
Moodle	Oui	Oui	Formation de groupe de façon aléatoire	Objet	Client/serveur	Récolte les traces des apprenants

Tableau I.10.2.b : Comparaison des plateformes présentées.

	Documentation, aide en ligne	Visibilité des parcours pédagogiques
Dokeos	Documentation en anglais et français de bonne qualité, documentation spécifique pour l'initiation et la prise en main. Forum.	Il existe une fonction Parcours qui permet de proposer un scénario pédagogique à l'apprenant qu'il visualise sous forme d'un sommaire.
Moodle	Documentation en anglais et français, aide en ligne sous forme de forum, FAQ, tutoriaux...	Pas de parcours défini dans la plateforme mais importation possible de scénarios pédagogiques. Plutôt animation de la communauté par groupe et parcours de formation en attribuant des modules aux apprenants. La visibilité dépend de la ressource utilisée.
Spiral	Documentation en anglais et français de bonne qualité, ma, que parfois des aides en ligne, forum.	Il existe plusieurs présentations des cours et des modules qui favorisent la présentation des parcours pédagogiques avec la possibilité de définir des phases avec dates de début et de fin.
Ganesh a	Documentation en français complète et aide en ligne souvent de qualité.	Il existe une fonction Séance qui permet de scénariser les modules qui composent les cours. La visibilité du parcours est faite à travers l'Agenda.

Tableau II.10.2.c: comparaison entre les plateformes (Documentation, aide en ligne, Visibilité des parcours pédagogiques) [69].

Sakai permet un suivi minimal de l'apprenant. Moodle permet le suivi des apprenants en récoltant les traces des activités de chaque apprenant mais engendre un tas d'information difficile à exploiter ce qui rend la tâche plus difficile pour l'enseignant qui se trouve ainsi rapidement débordé. De son côté, Claroline proposent un suivi bien structuré et organisé. Cependant, nous trouvons que ces plateformes en particulier et les plateformes d'E-Learning en général doivent prendre plus en compte l'enseignant. En effet, le suivi des apprenants est une tâche parmi beaucoup d'autres que doit accomplir l'enseignant. Ainsi, pour que le suivi des apprenants soit le plus efficace possible, il faut doter les plateformes des outils capables de traiter les informations récoltées et ainsi renseigner les enseignants à tout moment sur l'état des apprenants (actif, passif...) ainsi que sur la progression de leur travail.

A partir de l'étude comparative de ces plateformes on ne peut pas dire qu'une plateforme est complète. Chaque plateforme couvre des points et manque d'autres selon les besoins pour lesquels elle a été conçue. Mais on peut constater que Moodle et Claroline présentent plus des avantages par rapport aux autres plateformes.

I.13. Styles d'apprentissage

Le domaine de styles d'apprentissage est complexe et affecté par plusieurs aspects, conduisant à différents concepts et points de vue. De nombreux modèles de style d'apprentissage existent dans la littérature, chacun propose différentes descriptions et classifications des types d'apprentissage. On peut définir le style d'apprentissage comme suit : un style d'apprentissage est la meilleure façon de traiter personnellement avec les informations et l'expérience. Il décrit un apprenant en termes de conditions d'enseignement qui sont les plus susceptibles de favoriser son apprentissage, l'apprenant se distingue par son style d'apprentissage ; cela signifie que certaines approches pédagogiques sont plus efficaces pour lui que d'autres [11]. Nous présentons les classifications suivantes : Modèle de Dunn et Dunn, Modèle de Felder et Silverman, et Modèle de Grasha – Riechmann :

I.13.1. Modèle de Dunn et Dunn

L'une des approches des styles d'apprentissage les plus anciennes et les plus largement utilisés est celle proposée par Rita et Kenneth Dunn (1978, 1992a, 1992b, et Dunn, 1986). Grâce à leur travail dans les écoles, ils ont observé des différences marquées dans les façons dont les étudiants ont répondu au matériel didactique. Certains ont aimé apprendre seul, tandis que d'autres préfèrent l'apprentissage en groupe ou d'un l'enseignant. De ce travail préliminaire, ils ont identifié cinq dimensions clés sur les styles d'apprentissage des élèves différent: 1) l'environnement 2) le soutien affectif, 3) la composition sociologique, 4) physiologique, et 5) éléments psychologiques. Le Tableau suivant résume le modèle de Dunn et Dunn [11]:

Dimension	Eléments	Questions clé
Environnement	Son Lumière Température	Les étudiants préfèrent un endroit bien éclairé, environnement bruyant, animé, chaleureux ou non, modéré, environnement calme refroidisseur?
Émotionnel	Motivationnel Persistance Individuel Responsabilité Structure	Les apprenants ont besoin de beaucoup de soutien émotionnel? Vont-ils persister sur des tâches d'apprentissage? Peuvent-ils assumer la responsabilité individuelle? Est-ce qu'ils ont besoin de beaucoup de la structure?
Sociologique	Individuel Paires ou équipes conseils Varié	L'apprenant apprend mieux seul ou de travailler avec quelqu'un? Combien de conseils des matures qu'ils veulent ou ont besoin ?
Physiologique	Perceptuel Compréhension	L'apprenant est auditif, visuel, tactile, ou kinesthésique ?

	Temps Mobilité	Quel est le moment optimal pour l'apprentissage? Est-ce que l'apprenant nécessite la liberté de se déplacer lors de l'apprentissage?
Psychologique	Global Analytique Impulsif Réflectif	Comment l'apprenant réfléchit pour résoudre les problèmes : global ou analytique ? Est-ce que l'apprenant commence directement la résolution des problèmes ou il réfléchit avant démarrer le processus de résolution ?

Figure III.13.1 : Modèle de Dunn et Dunn

I.13.2. Modèle de Felder et Silverman

Felder et Silverman [120] ont développé leur modèle de style d'apprentissage pour deux raisons : pour capturer les plus importantes différences entre les styles d'apprentissage chez les étudiants, et à fournir une bonne base pour les instructeurs d'ingénierie pour concevoir une approche d'enseignement qui permettrait de répondre aux besoins d'apprentissage de tous les étudiants. Le modèle Felder - Silverman classe les préférences d'apprentissage des élèves dans l'une des catégories dans chacune des quatre dimensions de style d'apprentissage suivants : détective ou intuitive, visuelle ou verbale, active ou réflexive, ou séquentielle ou globale:

- Les apprenants **défectifs** (pratique, orientée vers les faits et procédures) ou les apprenants **intuitifs** (conceptuel, innovante, orientée vers des théories et des significations) ;
- Les apprenants **visuels** (préfèrent des représentations visuelles des documents présentés - photos, diagrammes, cartes) ou apprenants **verbaux** (préfèrent des explications écrites et parlées);
- Les apprenants **actifs** (apprendre en collaboration avec d'autres apprenants) ou **réflectifs** (Ce genre des apprenants préfère à travailler seul et réfléchit silencieusement à l'information d'abord);
- Les apprenants **séquentiels** (linéaires, ordonnée, apprendre à petits pas progressifs) ou apprenants **globaux** (holistiques, systèmes penseurs, d'apprendre dans de grands bonds).

I.13.3. Modèle de Grasha - Riechmann

Le modèle de style d'apprentissage Grasha – Riechmann [119] a été défini en fonction de l'interaction sociale des apprenants avec les enseignants ou les autres apprenants. Les modèles de Grasha et Riechmann sont basées sur le comportement des apprenants, ils ont identifié trois dimensions afin de comprendre le comportement des apprenants : le participant / non-participant, de collaboration / compétition, et dépendant / indépendant dimension:

- **Participant / non-participants** : cette dimension exprime combien un étudiant souhaite s'impliquer dans la salle de classe. Les étudiants participants préfèrent apprendre le contenu du cours et profiter d'assister le cours. Ils prennent la responsabilité de leur propre apprentissage et aiment participer aux activités d'apprentissage. Les étudiants non-

participants ne préfèrent pas assister à la classe. Ils évitent les activités d'enseignement et ne prennent pas la responsabilité de leur apprentissage.

- **Collaboratifs / Compétitifs:** mesure la nature de l'interaction de l'apprenant avec les autres. Les apprenants de collaboration sont plus motivés lorsque le travail en mode collaboratif avec les autres. Ils sont caractérisés comme des apprenants qui sont coopérative, aiment travailler avec les autres. D'autre part, les apprenants compétitifs sont plus motivés quand ils voient les autres comme des concurrents. Ils ont la motivation pour faire mieux que les autres, plaisir de la compétition, et de voir le processus d'apprentissage comme une situation gagnant-perdant.
- **Dépendants / indépendants** à l'égard des enseignants et à quel point les étudiants ont le désir de liberté et de contrôle dans l'environnement d'apprentissage. Les étudiants dépendants considèrent l'enseignant comme la source d'information et la structure. Ils veulent avoir les ordres pour faire les activités (par les autorités) et apprendre seulement ce qui est nécessaire. Apprenants indépendants sont caractérisés comme des apprenants confiants et curieux. Ils préfèrent penser pour eux-mêmes et de travailler sur leur propre.

I.14. Conclusion

Aujourd'hui, les systèmes de E-Learning sont très importants car ils deviennent le support de base de l'apprentissage sur le web. Ils offrent plusieurs services aux apprenants : le parcours des cours, la disponibilité, flexibilité, pas de frontières géographiques, haute qualité, et plusieurs techniques pour aider l'apprentissage des connaissances...etc.

Dans ce chapitre, nous avons présenté une vue globale sur le E-Learning. Nous avons commencé par décrire quelques concepts de base dans la littérature, ainsi que les avantages et les limites de E-Learning. Nous avons remarqué que le marché est plein d'outils, des technologies pour satisfaire les besoins d'évolution de e-Learning. C'est pour cela qu'on peut trouver plusieurs plates-formes d'apprentissage, des différents standards et normes. Le support essentiel dans un environnement E-Learning sont les documents pédagogiques et l'hypermédia dédiés à l'éducation, donc on a présenté la définition et les types des documents pédagogiques. Nous avons présenté les techniques d'adaptation qui sont non évitables pour fournir un système éducatif qui répond aux besoins des apprenants hétérogènes.

La technologie Web sémantique et les ontologies seront décrites dans le chapitre suivant puisque notre travail est destiné à la proposition d'une approche sémantique exploitant le paradigme agent pour le E-Learning. Nous proposons donc, dans le chapitre suivant, une étude concernant le web sémantique et présentons les avantages qu'apporte cette nouvelle génération de web, notamment, dans le domaine du E-learning.

Chapitre II : Le web sémantique et les ontologies

II.1. Introduction

Les sources d'informations disponibles dans le web sont de plus en plus nombreuses et complexes. Donc, il est nécessaire de permettre une description de ces informations non seulement en termes de structure (aspect syntaxique) mais également en termes de signification (aspect sémantique). La description de métadonnées sur les sources d'informations prend en compte la description structurelle et sémantique des informations est un problème important. Cette sémantique peut être exprimée à l'aide d'ontologies. L'utilisation d'ontologies en informatique vise à intégrer une couche de connaissances aux systèmes afin de permettre des traitements élaborés de l'information qu'ils manipulent.

A partir des années 90, les ontologies sont apparues dans le domaine de l'ingénierie de connaissance, et plus largement dans le domaine de l'intelligence artificielle, comme une approche de modélisation et de représentation des connaissances. L'utilisation des ontologies pour la représentation des connaissances s'avère avantageuse afin de faciliter l'utilisation de ces connaissances par la machine.

Le web sémantique est une nouvelle technologie pour augmenter l'aspect représentatif des données dans le web. Il facilite l'exploitation, la recherche et l'échange de contenu en représentant ce contenu sémantiquement. Dans le web sémantique, la machine peut comprendre l'aspect sémantique des données qui facilite leur manipulation par la machine. Selon Goble [104], le Web sémantique vise à favoriser la découverte, l'échange d'information et l'intégration de l'information par l'étiquetage (tagging) ou le balisage (marking up), et à développer des technologies et infrastructures pour annoter le contenu du Web avec des informations additionnelles appelées "métadonnées" pouvant être traitées par des agents logiciels [9].

Dans ce chapitre nous étudierons le web sémantique et ses éléments. Ensuite nous présentons les métadonnées et leurs catégories. La section suivante est consacrée à une présentation des ontologies comme un outil de représentation sémantiques des données. Les classes et les composants d'une ontologie fait l'objet de la section suivantes. Dans la littérature, il existe plusieurs langages pour le développement des ontologies, pour cela nous présentons ces langages et effectué une comparaison entre eux.

II.2. Le Web sémantique

Le Web sémantique (Semantic Web) se présente comme une extension du Web classique. C'est un moyen d'extraction et d'exploitation de l'intelligence collective du Web, en rendant le contenu plus facile à échanger et à partager entre les différents utilisateurs. Ce nouveau Web vise à faciliter l'exploitation des ressources par les machines en accédant aux ressources grâce à leur représentation sémantique.

II.2.1. Définition du mot "sémantique"

La sémantique est une branche de la linguistique. Ce mot a été inventé à la fin du XIX ème siècle par la linguistique française Michael Breal [38], (l'auteur du premier traité de sémantique). La sémantique touche plusieurs domaines : la signification des mots, et les rapports entre les mots. Dans des quelques domaines, le terme sémantique est utilisé en opposition à de celui de syntaxe. Il y a entre la sémantique et la syntaxe la même relation qu'il y a entre le fond et la forme. En d'autres termes, l'analyse lexicale s'intéresse au mot dans son ensemble, par rapport à un énoncé. Par contre la sémantique s'intéresse à un mot pour le mot.

II.2.2. Définition du Web sémantique

Les pages Web dans le Web classique, se basent essentiellement sur la structure du document et les liens entre ces pages [105]. Donc l'exploitation de l'information est quasiment impossible par les machines car le manque de sémantique. A la différence de cela, le Web sémantique offre une représentation sémantique du contenu, ce qui permet aux agents logiciels l'extraction de l'information. Selon Tim Berners, appelé "le père de Web", : " le Web sémantique n'est pas un Web séparé, mais une extension du Web actuel dans lequel l'information est munie précisément d'une signification bien définie permettant aux ordinateurs et aux personnes de mieux travailler en coopération." [9]. Nagarajan [106] présente le Web sémantique comme " ...Une vision avec l'idée de disposer de données sur le Web définies et liées de telle manière qu'il puisse être utilisé par les machines non seulement à des fins d'affichage, mais pour l'automatisation, l'intégration et la réutilisation des données entre diverses applications".

A partir de ces définitions, nous pouvons dire que le Web sémantique ajoute une extension du Web actuel. Il associe à l'information un sens bien défini pour améliorer le traitement de ces informations par les machines. Donc, durant le processus de la recherche et de combinaison des ressources, cette amélioration permet d'aider les utilisateurs en facilitant ces processus. Le Web sémantique peut être vu comme une infrastructure complétant le contenu informel du web actuel avec la connaissance formalisée. Cette connaissance doit s'appuyer essentiellement sur des ontologies afin de pouvoir être partagée et munie d'interprétations opérationnelles [9].

Le Web sémantique comprend un certain nombre de technologies organisées en couches interdépendantes. Dans ces différentes couches, nous pouvons citer les métadonnées, les ontologies, ainsi que la logique et l'inférence [9]. La figure suivante illustre ces différentes couches :

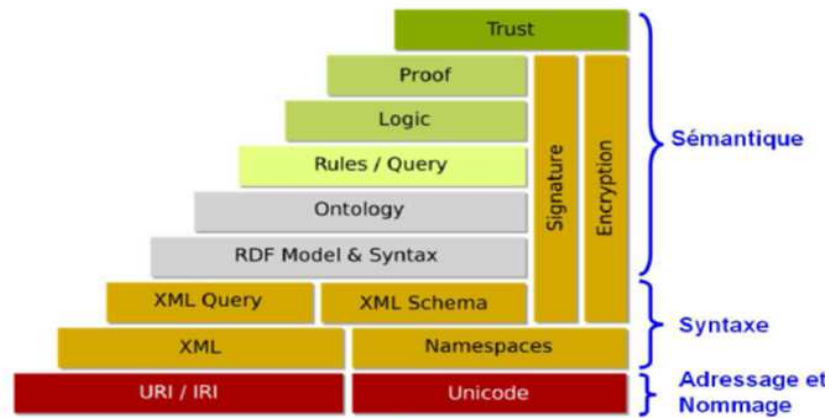


Figure II.2.2: Architecture du Web sémantique proposée par Tim Berners-Lee

Les différentes couches sont les suivantes [9]:

- **URI (Uniform Resource Identifier)** : permet d'identifier les ressources sur les réseaux.
- **XML (Extensible Markup Language)** : le langage extensible de balisage est une simplification du langage normalisé de balisage généralisé SGML (Standard Generalized Markup Language). Ses deux principes essentiels: 1) la structure d'un document XML est définissable et valable par un schéma et 2) un document XML est entièrement transformable dans un autre document XML. Enfin, son objectif initial est l'interopérabilité, de faciliter l'échange automatisé de contenus entre systèmes d'information hétérogènes et particulièrement adapté pour l'envoi de documents sur le Web.
- **XML-Schéma ou XMLS** : Schéma XML est un langage de définition à syntaxe XML, il permet de construire des documents XML suivant un vocabulaire et une structure hiérarchique spécifique. Appliqué pour la description des structures de données des documents XML.
- **RDF (Resource Description Framework)** : est un modèle de données doté d'une syntaxe à balises qui permet de représenter des objets ou des ressources ainsi que des relations entre ces objets sous forme de triplets. Chaque triplet RDF est une association de (sujet, prédicat, objet) ou le sujet représente la ressource à décrire, le prédicat représente un type de propriété applicable à cette ressource et l'objet représente une donnée ou une autre ressource.
- **RDF-Schéma ou RDFS** : est un vocabulaire qui permet de décrire les classes et propriétés pour des ressources RDF. Ce langage extensible de représentation des connaissances appartient à la famille des langages du Web sémantique et fournit des éléments de base pour la définition d'ontologies ou vocabulaires destinés à structurer des ressources RDF.
- **Ontologies et Logique** : Ces couches concernent la définition de langages ontologiques comme OWL (Ontology Web Language) qui s'appuie sur la logique de description. Le langage OWL est préconisé comme standard par le W3C consortium pour modéliser des ontologies. Il permet de modéliser les informations d'un domaine afin de faciliter l'échange d'informations par des processus automatisés.
- **Preuve** : Les langages logiques permettent la mise en œuvre d'outils de raisonnement. Ces outils sont disponibles pour des langages comme OWL et permettent par exemple de tester

la cohérence des informations, les classiers, ...etc. Les modèles à base de règles comportent également des moteurs d'inférence qui permettent d'inférer des informations à partir des informations décrites. Des efforts sont en cours, pour uniformiser ces différentes approches. La spécification du langage SWRL (Semantic Web Rule Language) s'inscrit dans ces travaux émergents.

- **Confiance** : La couche confiance est située au haut de la pyramide. Elle concerne l'utilisation de signatures numériques et d'autres types de connaissances afin de garantir la fiabilité et l'origine des informations, par des recommandations d'agents de confiance, de la notation, des organismes de certification et des organismes de consommateurs.

II.2.3. Les métadonnées

Le Web sémantique est un moyen d'échange et de partage des ressources grâce à la représentation sémantique du contenu. Ces ressources sont décrites par des informations structurées additionnelles "les métadonnées". Les métadonnées sont des "données sur les données" [107]. L'utilisation des métadonnées permet de décrire le contenu de la ressource afin de rendre son contenu exploitable. Les métadonnées peuvent exister sur différents niveaux distincts. Les informations décrivant la syntaxe, la structure et le contexte sémantique. La figure suivante illustre la représentation faite par Sheth [108], des différents types de métadonnées :

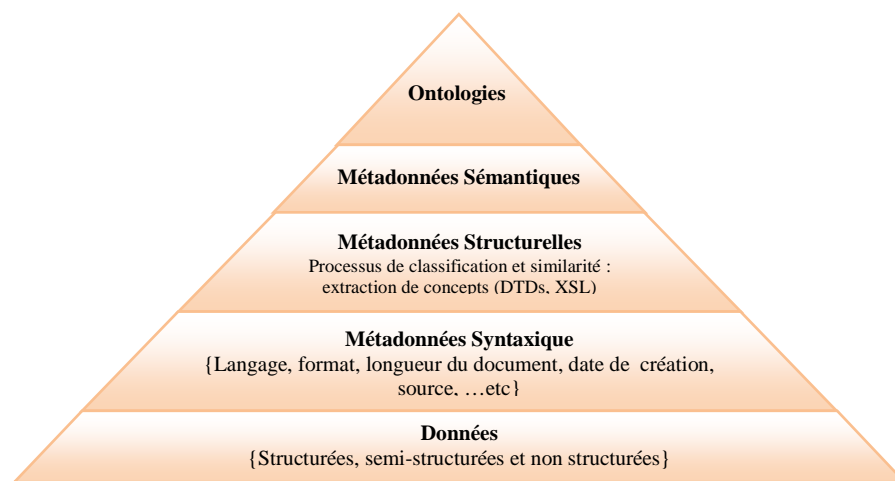


Figure II.2.3 : Types de métadonnées et des annotations sémantiques selon Sheth [108]

II.2.3.1. Métadonnées syntaxique

Les métadonnées syntaxiques sont la forme la plus simple de métadonnées et décrivent l'information non contextuelle sur le contenu en fournissant des informations très générales, en associant des étiquettes. Certaines métadonnées syntaxiques peuvent être la taille du document, l'emplacement ou la date de création.

II.2.3.2. Métadonnées structurelles

Les métadonnées structurelles fournissent des informations concernant la structure du contenu et décrivent comment les éléments sont assemblés ou arrangés.

II.2.3.3. Métadonnées sémantiques

Les métadonnées sémantiques ajoutent les relations, les règles et les contraintes aux métadonnées syntaxiques et structurelles. Elles fournissent un contexte pour l'interprétation des informations basées généralement sur un modèle de métadonnées d'un domaine spécifique ou une ontologie. Ces métadonnées permettent aux applications de comprendre la signification réelle des données.

II.3. Les ontologies

Nous ne pouvons pas parler du Web sémantique sans présenter l'ontologie qui est généralement considérée comme une base de connaissances et est au centre des développements émergents du Web sémantique. Une ontologie permet la modélisation d'un domaine de connaissances et peut être vue comme un modèle conceptuel d'un domaine particulier, qui décrit les concepts de ce domaine et les relations entre ces concepts. Les ontologies permettent la modélisation d'informations agréées par une communauté de personnes et accessibles par une machine pour développer des services automatisés et par conséquent, jouent un rôle de référence pour d'écrire la sémantique des informations à partager.

II.3.1. Qu'est-ce qu'une Ontologie ?

Historiquement, le terme Ontologie a été défini en philosophie comme une branche de la métaphysique qui s'applique à l'être en tant qu'être. Ontologie est un terme grec qui est composé de «ontos» participe présent de verbe être et « logos » qui veut dire discours.

Il est extrêmement difficile de parler du Web Sémantique sans utiliser le mot ontologie. Ce concept est d'ailleurs un élément fondamental dans notre présent travail de recherche. Les ontologies sont utilisées dans des domaines très variés tels que l'intelligence artificielle, le Web sémantique, le génie logiciel, l'informatique biomédical, l'architecture de l'information.

Au début des années 90, des chercheurs en intelligence artificielle se sont intéressés à cette notion pour la formalisation des connaissances. Du point de vue de l'ingénierie des connaissances, différentes définitions de l'ontologie ont été données. Ces définitions offrent des points de vue à la fois différents et complémentaires sur un même concept. Une première définition est donnée par Neches et ses collègues [109]: "*An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary*". Donc, on peut dire que l'on utilise une ontologie pour définir les termes d'un thème (le vocabulaire) et les relations entre ces termes; aussi on définit les règles afin de combiner les termes et les relations pour définir les extensions pour le vocabulaire.

SOWA [110] définit l'ontologie comme "la méthode pour extraire un catalogue des choses ou des entités (C) qui existent dans un domaine (D) selon la perception d'une personne qui se sert d'un certain langage (L) pour le décrire". La plus définition référencée et aussi la plus synthétique est sans doute celle formulée par GRUBER [112] "une spécification explicite d'une conceptualisation". Borst et son équipe [111] ont amélioré la définition de Gruber en

affirmant que " Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée", Dans cette définition on a [41] :

- **Formelle** : Se réfère au fait que l'ontologie doit être lisible et compréhensible par les machines, cela est nécessaire pour qu'elle puisse être munie de capacités de raisonnement.
- **Explicite** : Signifie que les concepts utilisés ainsi que les contraintes sur leur utilisation sont définis d'une manière claire et précise.
- **La conceptualisation** : Signifie un modèle abstrait d'un certain aspect du monde.
- **Partagée** : Renvoie à l'idée qu'une ontologie rend compte d'un savoir consensuel.

Basées sur la définition de Gruber, plusieurs autres définitions ont été proposées. Guarino et Garetta [9] ont fourni une autre clarification terminologique entre les différentes utilisations de l'ontologie comme un "système conceptuel informel", un "état sémantique formel", une "spécification d'une conceptualisation", une "représentation d'un système conceptuel par l'intermédiaire d'une théorie logique", le "vocabulaire utilisé par une théorie logique", et une "spécification d'une théorie logique". L'équipe de Swartout [9] déclare que "une ontologie est un ensemble de termes hiérarchiquement structuré pour décrire un domaine qui peut être utilisé comme base squelettique pour une base de connaissances".

Une ontologie est donc la manifestation d'une compréhension partagée d'un domaine entre un certain nombre d'agents, et un tel accord facilite des communications précises et efficaces qui mènent, à terme, à d'autres avantages tels que l'interopérabilité, la réutilisation et le partage [41].

Le but principal de l'application d'ontologie en intelligence artificielle n'est pas de servir d'argument déductif pour définir la réalité, mais de permettre une explication des termes et des significations pour définir une base consensuelle pour l'interopérabilité dans le domaine. Par opposition à la perception philosophique, il n'y a pas de "vérité absolue" dans la perception de l'IA ; la spécification dans l'ontologie dépend de l'utilisateur et est développée comme un mécanisme pour atteindre un accord (consensus) sur les différentes interprétations de la réalité. Les deux perceptions sur l'ontologie fournissent différents principes et approches pour la conception et le développement d'ontologie [41].

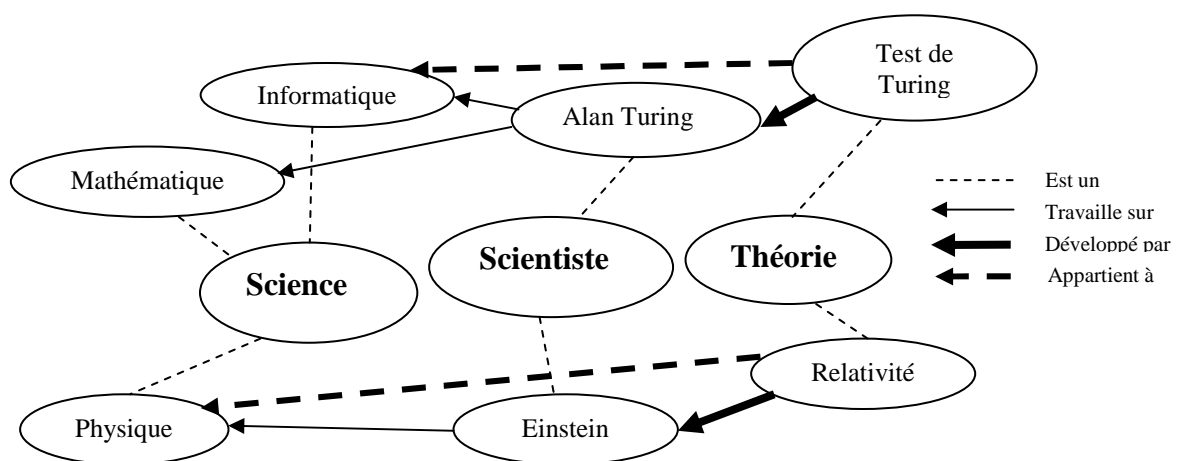


Figure II.3.1 : Un exemple d'ontologie

II.3.2. Classification des ontologies

Dans la littérature, il existe plusieurs classifications qui ont été proposées. Chaque classification se base sur des critères particuliers. Dans ce qui suit, on va présenter les classifications : Van Heijst [113], Guarino [114].

Selon Van Heijst [113], l'ontologie est classifiée par deux types de critères. La première classification se base sur les types et la richesse des structures utilisées dans l'ontologie. Donc, on peut distinguer trois catégories d'ontologies:

- **Les ontologies terminologiques** qui sont utilisées pour spécifier les termes du vocabulaire d'un domaine de connaissances.
- **Les ontologies d'information** qui spécifient la structure/le schéma d'une base de données pour permettre le stockage d'informations.
- **Les ontologies qui modélisent de la connaissance** qui proposent des structures internes plus riches et qui sont davantage définies en fonction de leurs utilisations comme par exemple le partage d'informations.

Le deuxième critère de classification est également la prise en compte des "objectifs" de la modélisation. Alors, on a quatre catégories selon ce critère:

- **Les ontologies d'applications** qui spécifient les informations nécessaires à une ou plusieurs applications particulières. Elles contiennent toutes les définitions nécessaires pour modéliser la connaissance requise pour une application particulière. Les ontologies d'application étendent et spécialisent souvent le vocabulaire des ontologies de domaine et de tâche pour une application donnée.
- **Les ontologies de domaine** qui expriment la conceptualisation des connaissances d'un domaine particulier.
- **Les ontologies génériques** qui sont utilisées pour représenter la connaissance commune (consensuelle) réutilisable dans les domaines. Ces ontologies incluent le vocabulaire lié aux choses, aux événements, au temps, à l'espace, à la causalité, au comportement, à la fonction, à la mereologie, etc.
- **Les ontologies de représentation** qui visent à expliciter les conceptualisations sous-jacentes aux formalismes de représentation des connaissances. Elles représentent les entités du monde réel sans a priori, de façon "neutre". Ces concepts des ontologies de représentation peuvent être utilisés dans les ontologies génériques ou les ontologies de domaine. Le schéma suivant illustre cette classification :

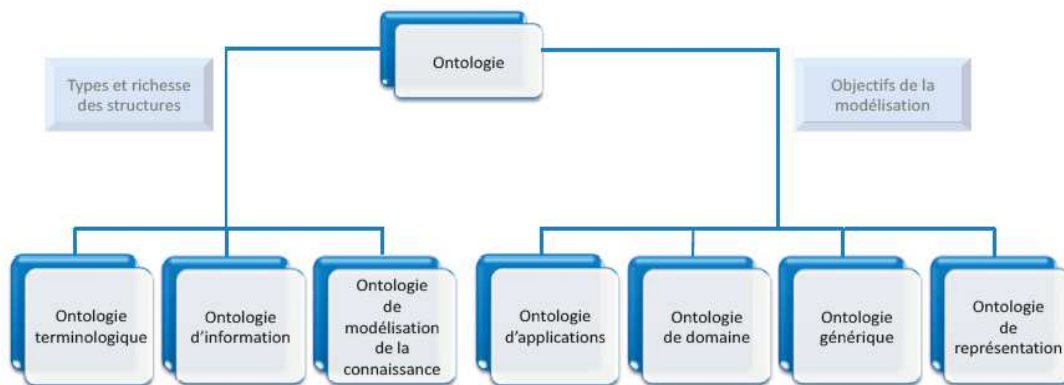


Figure II.3.2.a : Classification des ontologies selon Van Heijst [113]

Selon Guarino [114], les ontologies sont classifiées en quatre catégories. Cette classification se base sur le degré de généralité ou du niveau de dépendance d'une tâche:

- **Les ontologies de haut niveau** qui décrivent des concepts très généraux et fournissent des notions générales sous lesquelles tous les termes racines dans les ontologies existantes devraient être liés. Le principal problème est qu'il y a plusieurs ontologies de haut niveau et elles diffèrent sur les critères suivis pour classifier les concepts les plus généraux de la taxonomie.
- **Les ontologies de domaine** qui sont réutilisables dans un domaine spécifique donné. Ces ontologies fournissent les vocabulaires sur les concepts et leurs relations dans un domaine, sur les activités qui ont lieu dans ce domaine, et sur les théories et les principes élémentaires régissant ce domaine. Il y a une frontière claire entre les ontologies de domaine et les ontologies de haut niveau. Les concepts dans les ontologies de domaine sont habituellement des spécialisations des concepts déjà définis dans les ontologies de haut niveau, et le même principe pourrait se produire avec les relations. Exemple : e-commerce, médecine, science de l'ingénieur (modèles mathématiques), entreprise, chimie, ...etc.
- **Les ontologies des tâches ou d'activités** qui décrivent le vocabulaire relié à une tâche générique ou à une activité (comme diagnostiquer, programmer, vendre, etc.) en spécialisant les termes dans les ontologies de haut niveau. Les ontologies de tâche fournissent un vocabulaire systématique des termes utilisés pour résoudre les problèmes liés aux tâches qui peuvent ou ne peuvent appartenir au même domaine. Ce sont des ontologies dépendantes des applications. Les ontologies de tâche de domaine ("Domain-Task ontologies") sont des ontologies de tâche réutilisables dans un domaine donné, mais pas à travers des domaines. Elles se veulent indépendantes des applications.
- **Les ontologies d'applications** où les concepts dépendent à la fois d'un domaine et d'une activité en particulier. Elles donnent les définitions des concepts et des relations appropriés appliquées pour spécifier un processus de raisonnement afin de réaliser une tâche particulière. Une ontologie sur une activité par les moyens de la décomposition de tâche appartiendrait à cette catégorie.



Figure II.3.2.b : Classification des ontologies selon Guarino [114]

La catégorisation selon le nombre de points de vue de concepteurs donne deux types d'ontologies [41]:

- **Les ontologies inspirationnelles** : dans le cas où l'ontologie est conçue selon le point de vue individuel d'un concepteur sur le domaine,
- **Les ontologies collaboratives** : c'est le cas de la conception avec de multiples points de vue de différents concepteurs et acteurs sur le domaine.

II.3.3. Les composants d'une ontologie

Maedche et Staab [47][41] déclarent qu'une ontologie comporte un lexique des concepts, des relations et des axiomes. A partir de différentes études, les composants primaires d'une ontologie comme suit :

- **Les "classes" ou "catégories"** est un ensemble d'objets et la base de la représentation de la connaissance dans une ontologie. Les catégories incluent les concepts, et tandis que les concepts sont liés aux sémantiques cognitives, les catégories sont universelles dans le monde réel.
- **Les "relations"** représentent des types d'interactions entre les classes (ou concepts) dans une ontologie. Les relations peuvent être taxonomiques, associatives, nominales, locatives, fonctionnelles ou causatives. Encore d'autres spécifications axiomatiques pour des relations spécifient des relations formelles de mereologie, d'identité, d'unité, d'essence, de rigidité et d'autres critères de dépendance.
- **Les "axiomes"** sont utilisés pour modeler les conditions qui sont toujours vraies pour le domaine. Une spécification axiomatique riche est critique en distinguant une ontologie à partir d'une taxonomie terminologique.

Aussi, une ontologie peut être vue comme un ensemble structurée de concepts et de relations entre ces concepts, destinés à représenter les objets du monde sous une forme compréhensible. Les connaissances intégrées dans les ontologies sont formalisées en mettant en jeu cinq types de composants : les concepts, les relations, les fonctions, les axiomes et les instances.

II.3.3.1. Concepts

Un concept peut représenter un objet matériel, une notion ou encore une idée [44]. Les concepts sont aussi appelés termes ou classe de l'ontologie. Les concepts manipulés dans un domaine de connaissances sont organisés au sein d'un réseau de concepts. L'ensemble des concepts est structuré hiérarchiquement et les concepts sont liés par des propriétés conceptuelles. La propriété utilisée pour la structuration de la hiérarchie des concepts est la subsomption. Un concept $C1$ subsume un concept $C2$ si toute propriété sémantique de $C1$ est aussi une propriété sémantique de $C2$, c'est-à-dire $C2$ est plus spécifique que $C1$. L'extension d'un concept est forcément plus réduite que celle d'un concept qui le subsume tandis que son intention est par contre plus riche en instances [43].

Propriétés portant sur un seul concept [43]

- **La généricité** : un concept est générique s'il n'admet pas d'extension. Par exemple : la vérité est un concept générique.

- **L'identité** : un concept porte une propriété d'identité si cette propriété permet de conclure quant à l'identité de deux instances de ce concept. Cette propriété peut porter sur des attributs du concept ou sur d'autres concepts. Par exemple : le concept d'étudiant porte une propriété d'identité liée au numéro de l'étudiant, deux étudiants étant identiques s'ils ont le même numéro.
- **La rigidité** : un concept est rigide si toute instance de ce concept en reste instance dans tous les mondes possibles. Par exemple : humain est un concept rigide, étudiant est un concept non rigide.
- **L'anti-rigidité** : un concept est anti-rigide si toute instance de ce concept est essentiellement définie par son appartenance à l'extension d'un autre concept. Par exemple : étudiant est un concept anti-rigide car l'étudiant est avant tout un humain.
- **L'unité** : un concept est un concept unité si, pour chacune de ses instances, les différentes parties de l'instance sont liées par une relation qui ne lie pas d'autres instances de concepts.

Propriétés portant sur deux concepts [43]:

- **L'équivalence** : deux concepts sont équivalents s'ils ont même extension. Par exemple : étoile du matin et étoile du soir.
- **La disjonction** : deux concepts sont disjoints si leurs extensions sont disjointes. Par exemple : homme et femme.
- **La dépendance** : un concept C1 est dépendant d'un concept C2 si pour toute instance de C1 il existe une instance de C2 qui ne soit ni partie ni constituant de l'instance de C1 (par exemple le concept "parent" est un concept dépendant du concept "enfant" et vice-versa).

II.3.3.2. Relations

Elles traduisent les interactions existant entre les concepts présents dans le domaine analysé. Ces relations nous permettent de capturer la structuration ainsi que l'interaction entre les concepts, ce qui permet de représenter une grande partie de la sémantique de l'ontologie. Une relation permet de lier des instances de concepts, ou des concepts génériques. Une relation est caractérisée par un terme (voir plusieurs) et une signature. Les termes sont les labels ou les étiquettes de la relation. La signature précise le nombre d'instances de concepts, que la relation lie, leurs types et l'ordre des concepts (c'est à dire la façon dont la relation doit être lue). Par exemple, la relation "écrit" lie une instance du concept "personne" et une instance du concept "texte", dans cet ordre.

Propriétés liant deux relations sont :

- **Les propriétés algébriques** : comme la symétrie, la réflexivité et la transitivité ;
- **La cardinalité** : indique le nombre possible de relation de ce type entre les mêmes concepts ou instances du concept.
- **L'incompatibilité** : deux relations sont incompatibles si elles ne peuvent lier les mêmes instances de concepts (par exemple les relations "être rouge" et "être vert" sont incompatibles).
- **L'inverse** : deux relations binaires sont inverses l'une de l'autre si, l'une lie deux instances $I1$ et $I2$, l'autre lie $I2$ et $I1$ (par exemple les relations "a pour père" et "a pour enfant" sont inverses l'une de l'autre).

- **L'exclusivité** : deux relations sont exclusives si, l'une lie des instances de concepts, l'autre ne lie pas ces instances, et vice-versa (par exemple "l'appartenance" et "la non appartenance" sont deux relations exclusives). L'exclusivité entraîne l'incompatibilité.

II.3.3.3. Les fonctions

Elles sont des cas particuliers de relations dans lesquelles le nième élément de la relation est défini de manière unique à partir des n-1 premiers. Formellement, les fonctions sont définies ainsi : $F : C_1 * C_2 * \dots * C_{n-1} \rightarrow C_n$. Comme exemple de relation binaire, nous avons la fonction mère de.

II.3.3.4. Axiomes

Les axiomes permettent de modéliser des assertions acceptées comme vraies, à propos des abstractions du domaine traduites par l'ontologie. Leur inclusion dans une ontologie peut avoir plusieurs objectifs : interviennent dans la définition des significations des composants d'ontologie, les contraintes sur les valeurs des attributs, les arguments de relations et dans l'inférence de nouvelles informations, etc. Les axiomes formels sont utilisés pour vérifier la consistance de l'ontologie [43].

II.3.3.5. Instances

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

II.3.4. Méthodologies pour la construction d'ontologies

Les ontologies étant destinées à être utilisées dans un système d'information, leur développement doit s'appuyer sur des principes similaires à ceux utilisés en génie logiciel. Afin qu'une ontologie soit pérenne et réponde aux objectifs attendus, il est donc nécessaire de la construire en respectant certaines règles. De nombreuses méthodologies de construction d'ontologie existent sans qu'il y ait toutefois de consensus autour des meilleures pratiques à adopter [115][46]. Néanmoins, quelle que soit la méthodologie retenue, les auteurs s'accordent sur un processus de développement découpé en trois principales phases : (1) **la spécification sémantique**, (2) **la conceptualisation**, (3) **la formalisation**. Cette dernière peut être complétée par une phase **d'ontologisation** et une phase **d'opérationnalisation**. Enfin, tous soulignent également que la création d'une ontologie n'est pas linéaire mais que de nombreux allers-retours entre les phases de conceptualisation et de formalisation sont nécessaires. Le cycle de développement est donc par nature itératif et incrémental.

II.3.4.1. Les principes de construction d'une ontologie

Afin de construire une ontologie susceptible à répondre aux objectifs d'une ontologie, on doit respecter un ensemble de principes. Gruber [112] a proposé un ensemble des critères pour la construction des ontologies :

- **La clarté et l'objectivité** : l'ontologie doit fournir le sens des termes définis en offrant des définitions objectives ainsi que de la documentation associée en langage naturel.
- **L'exhaustivité** : une définition exprimée par une condition nécessaire et suffisante est préférable à une définition exprimée seulement par une condition nécessaire ou par une condition suffisante.
- **La cohérence** : l'objectif de ce critère est de formuler des inférences cohérentes avec les définitions.
- **L'extensibilité monotone maximale** : les nouveaux termes, qu'ils relèvent de la langue générale ou d'une langue de spécialité, devraient être inclus dans l'ontologie sans entraîner de modifications des définitions existantes.
- **L'intervention ontologique minimale** : intervenir le moins possible sur le monde en phase de modélisation. L'ontologie devrait spécifier le moins possible le sens de ces termes, de façon à ce que les parties impliquées dans l'ontologie aient les mains libres pour spécialiser et instancier l'ontologie à leur guise.

II.3.4.2. La méthode On-To-Knowledge

Développée dans le cadre d'un projet européen³³, la méthode On-To-Knowledge se décompose en cinq étapes dont seules les quatre dernières portent réellement sur le développement d'ontologies [47]. La première étape (Feasibility Study), bien que nécessaire, ne peut être considérée comme faisant partie du cycle de développement de l'ontologie. Il s'agit en fait d'identifier le problème et de réaliser une étude de faisabilité par rapport à celui-ci. Cela permettra alors de sélectionner les outils nécessaires et les personnes ressources. Les quatre phases suivantes sont quant à elles directement rattachées au cycle de vie de l'ontologie :

- **Phase de lancement** (*kickoff*) : durant laquelle les objectifs sont définis, les conditions requises sont spécifiées et les sources d'information identifiées. Cela permet d'aboutir à une première ontologie qualifiée de semi-formelle.
- **Phase d'amélioration** (*refinement*) : l'ontologie précédemment créée est améliorée en fonction des retours des experts du domaine. Les relations et les axiomes sont également précisés. Cela permet d'aboutir à une ontologie opérationnelle (Target ontology) qui doit être conforme aux objectifs identifiés durant la phase de *kickoff*.
- **Phase d'évaluation** : cette phase passe par trois niveaux (technologique, utilisateur, ontologique) et permet d'aboutir à une ontologie utilisable en production.
- **Phase de production et d'évolution** (*application & evolution*) : une fois dans le système, l'ontologie va continuer à évoluer. Il est donc nécessaire que soient identifiées la ou les personnes qui auront en charge sa maintenance et son évolution.

Comme cela est illustré par les boucles de rétro-action partant des phases d'évaluation et de production vers la phase d'amélioration, le processus de création de l'ontologie n'est pas linéaire. Ces itérations successives permettent de vérifier à chaque étape si l'ontologie créée est en adéquation avec les besoins précédemment exprimés. Comparativement à certaines autres méthodes plus parcellaires, l'un des points forts de On-To-Knowledge est de couvrir la totalité du cycle de vie de l'ontologie.

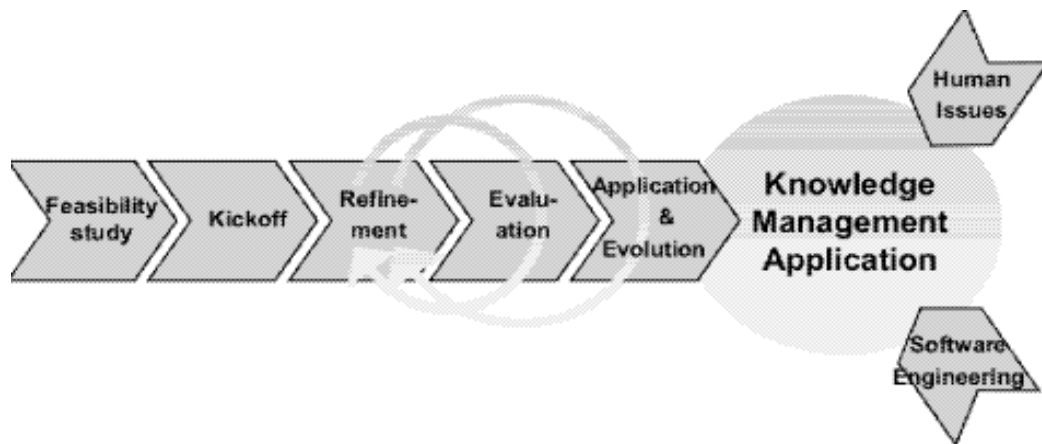


Figure II.3.4.2: Les cinq étapes du processus On-To-Knowledge [47].

II.3.4.3. La méthode Methontology

Développée par le Ontological Engineering Group de l'Université Polytechnique de Madrid, Methontology s'inspire des travaux déjà effectués en ingénierie des connaissances et en génie logiciel. Basée sur un prototypage évolutif, elle couvre l'ensemble du cycle de vie d'une ontologie [115]. L'étude comparative menée par Corcho [48] montre que Methontology est la méthode la plus avancée, ce qui lui vaut d'être recommandée par la FIPA34 en matière de construction d'ontologies.

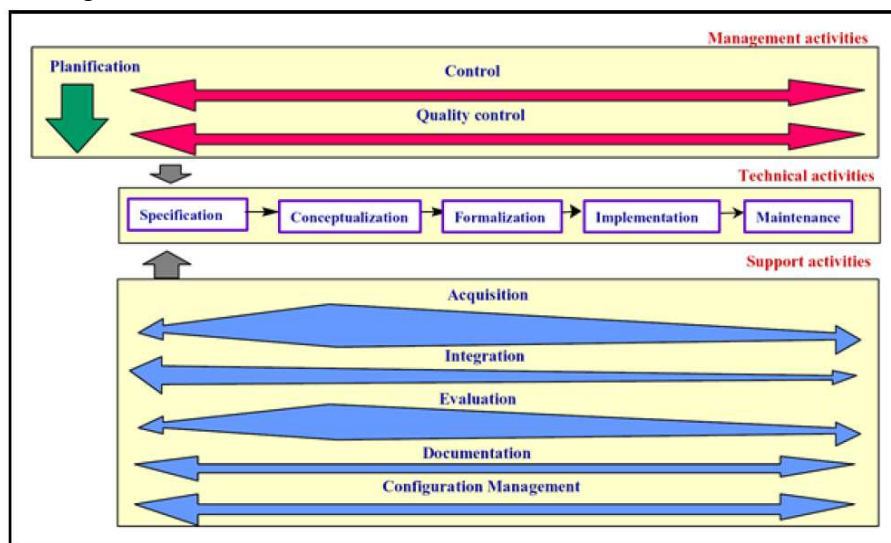


Figure II.3.4.3: Les différentes phases de Methontology [48].

Bien que les noms des activités de développement diffèrent, une logique similaire aux cinq phases de développement définies dans On-To-Knowledge peut être observée :

- **Spécification** : définit l'objectif de l'ontologie, les utilisateurs finaux ainsi que son dimensionnement.
- **Conceptualisation** : il s'agit ici d'organiser la connaissance. Pour cela, onze activités principales ont été définies allant de la définition des termes jusqu'à la description des

instances et des règles. C'est l'une des phases les plus importantes car l'essentiel de la connaissance est acquis au début de la construction.

- **Formalisation** : c'est dans cette phase que les connaissances sont traduites sous la forme d'une ontologie. Il s'agit de passer d'un modèle purement conceptuel à une implémentation informatique.
- **Implémentation** : l'ontologie est alors traduite dans un langage ontologique comme le OWL ou le RDF.
- **Maintenance** : la phase de maintenance corrige ou améliore l'ontologie créée.

Methontology est l'une des méthodologies les plus complètes disponibles actuellement. Elle décrit en détails les différentes phases, mais aussi l'ordre des tâches à effectuer dans chacune d'entre elles. Cette complétude fait qu'elle a été utilisée dans des projets et domaines aussi divers que l'industrie chimique, la minéralogie ou encore la détection de polluants.

II.3.5. Les langages d'ontologie

Les ontologies jouent un rôle très important dans le Web Sémantique en permettant de spécifier de manière formelle des vocabulaires pour la description du contenu du Web [43]. Ainsi, les informations du Web peuvent être accessibles et compréhensibles à la fois par les humains et par les machines. Dans la littérature plusieurs langages ont été utilisés pour la description des ontologies. Ces langages regroupent le langage de représentation des hiérarchies XML (eXtensible Markup Language), le langage RDF (Resource Description Framework), le langage DAML+OIL (Darpa Modeling Language of Ontology + Ontology Inference Layer) et le langage OWL (Ontology Web Language). Ces langages offrent différents niveaux d'expressivité [43].

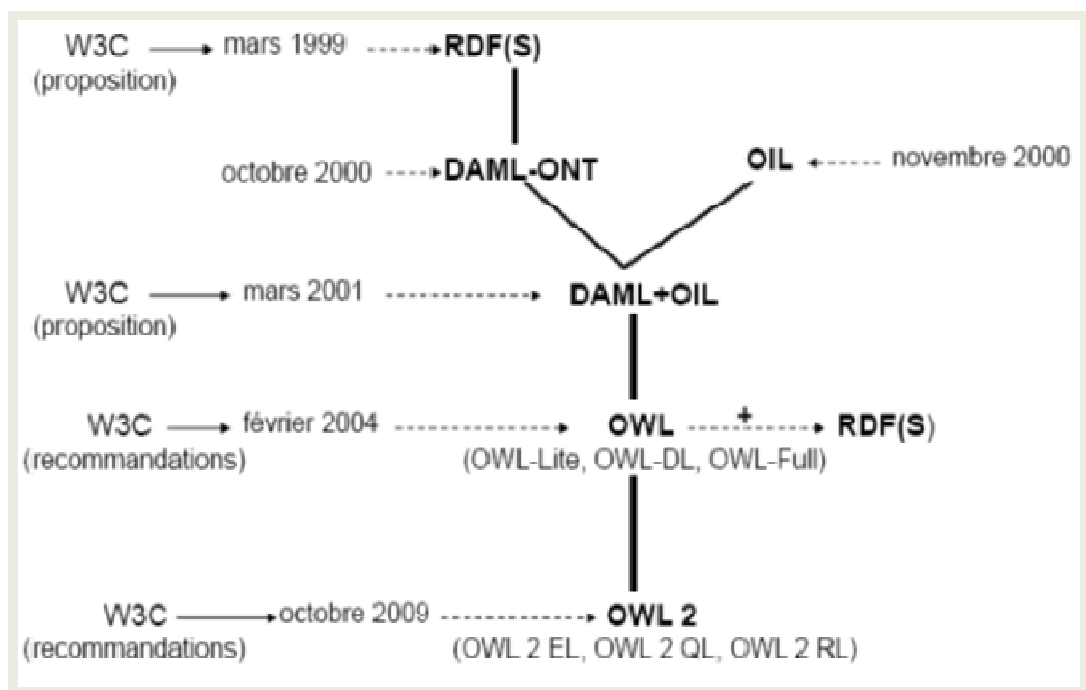


Figure II.3.5 : Généalogie des langages

II.3.5.1. XML

Le langage *eXtensible Markup Language* (XML) soutenu par le W3C depuis février 1998 pour le balisage des documents, conçu et optimisé pour la transmission des données via le Web, XML fournit un mécanisme simple mais robuste pour l'encodage des informations sur le Web sémantique [41]. On peut citer quelques règles syntaxiques du XML:

- Un prologue qui indique la version de XML employée.
- Chaque élément doit commencer par une balise ouvrante et se termine par une balise fermante.
- l'imbrication des éléments du document se fait sans chevauchement. Concrètement, cela signifie que si la balise ouvrante d'un élément se trouve à l'intérieur d'un élément parent, la balise fermante se trouvera dans le même élément.

Le langage XML possède une syntaxe pour la description des documents structurés. Il ne permet pas d'imposer des contraintes sémantiques à la signification des documents décrits. Le langage XML est un langage à balises.

Un schéma XML est une description du type d'un document XML. Le schéma contient un ensemble de règles. Ces règles sont des contraintes sur la structure et le contenu du document XML. Un document XML doit respecter le XML schéma qui lui est dédié afin d'assurer la validité du document selon son schéma. Le DTD (Document Type Définition) permet de définir une grammaire permettant de vérifier la conformité du document XML. Le DTD et le schéma XML sont développés pour exprimer des schémas XML.

Un document XML contient l'espace de noms (namespace) qui est un contexte ou un conteneur abstrait. L'espace des noms contient des noms, des termes, des mots qui représentent des objets ou des concepts dans le monde réel. Un nom défini dans un espace de noms correspond à un et un seul objet. Deux concepts ou objets différents sont référencés par deux noms différents dans un même espace de noms. Le langage XML ne permet de présenter que la structure des documents. Il ne prend pas en charge la sémantique de l'information. La sous-section suivante décrit le langage de description de ressources RDF

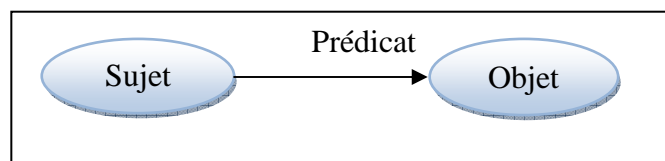
Un exemple de document XML permet de décrire une personne :

```
<?xml version="1.0" encoding="UTF-8"?>
  <personne>
    <nom>Bourekache</nom>
    <prenom>Samir</prenom>
    <naissance>
      <lieu>
        <ville>biskra</ville>
        <pays>Algerie</pays>
      </lieu>
      <date>
        <jour>28</jour>
        <mois>7</mois>
        <annee>1983</annee>
      </date>
    </naissance>
  </personne>
```

II.3.5.2. RDF

Le langage RDF est un modèle de méta-données pour référencer des objets (ressources) et les liens entre ces objets. Dans ce langage, les ressources sont identifiées par les URIs (Uniform Resource Identifier). Il s'agit d'un langage qui permet de décrire la sémantique des données, de manière compréhensible par les machines. Le modèle de données RDF est constitué de trois types de composantes :

- **Sujet** : il représente la ressource à décrire. Il est nécessairement identifié par une URI.
- **Prédicat ou Propriété** : il s'agit d'une propriété utilisée pour caractériser et décrire une ressource. Un prédicat est nécessairement identifié par une URI. Il exprime un lien ou une relation entre le sujet et l'objet
- **Objet** : représente une donnée ou une autre ressource (identifiée par une URI).



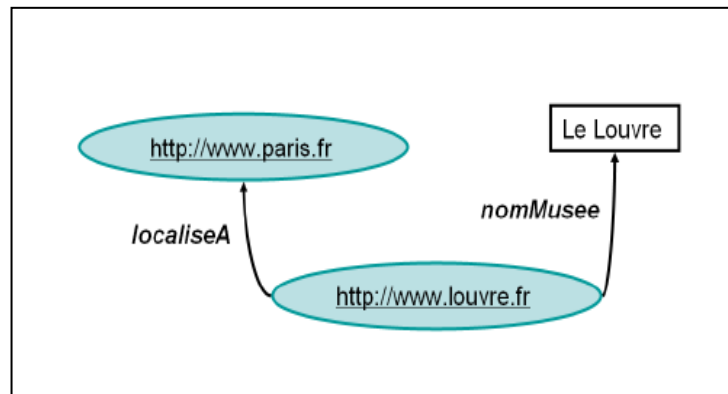
Les triplets RDF peuvent être représentés sous une forme graphique à travers un graphe ou décrits en XML.

Le schéma RDF est un langage qui permet de décrire des vocabulaires, des propriétés et des classes de ressources dans le modèle RDF. Le schéma RDF est une extension sémantique du langage RDF. Il assure des mécanismes pour la description des groupes de ressources similaires (classes) et des relations entre ces ressources (propriétés). Les descriptions du vocabulaire du schéma RDF sont écrites en RDF. Les deux langages RDF et schéma RDF sont référencés par RDF(S). Le RDF(S) possède des mécanismes pour la description des ressources ainsi que leurs caractéristiques.

Voici par exemple une déclaration RDF qui décrit une ressource (un musée) par la propriété *localiseA* et par la propriété *NomMusee* :

```
<rdf:Description rdf:about="http://www.louvre.fr">
  <rdf:type rdf:resource="http://www.parisart.com/LieuxCulturels#Musee"/>
  <localiseA rdf:resource="http://www.paris.fr"/>
  <nomMusee> Le Louvre</nomMusee>
</rdf:Description>
```

Les mêmes déclarations peuvent être représentées dans un graphe étiqueté, où les nœuds ovales représentent les ressources identifiées par des URIs et les nœuds rectangulaires représentent les valeurs de type de base. Les arcs représentent les propriétés des ressources dont la source correspond au sujet et la cible correspond à l'objet d'une déclaration RDF.



Une des caractéristiques les plus importantes du schéma RDF est la possibilité de définir les liens de "subsumption" entre des classes et des relations. Ces liens sont assurés par les primitives `rdfs:subClassOf` et `rdfs:subPropertyOf`. Ces primitives sont des liens de "spécialisation" ou "is_a". Elles permettent aux classes et aux relations d'hériter des caractéristiques définies dans les classes (ou des relations) parentes. Ce mécanisme assure le raisonnement dans RDF(S). La sous-section suivante présentera le langage de description des ontologies DAML+OIL.

II.3.5.3. DAML+OIL

DAML+OIL [42] est la fusion entre deux langages DAML et OIL. Le langage DAML a pour objectif de fournir les fondations pour la génération du Web sémantique. Le langage a adopté d'abord RDF(S) comme langage ontologique pour garantir l'interopérabilité sémantique. Comme RDF(S) n'est pas suffisamment expressif, relativement aux exigences du futur Web, un nouveau langage a été adopté nommé DAMLONT (Darpa Modeling Language of Ontology). Ce dernier représente une extension du RDF(S) avec les capacités d'un langage de représentation des connaissances [42]. Dans la même période, un groupe de chercheurs européens ont développé un langage d'ontologies appelé OIL. Ce langage a une syntaxe basée sur RDF. Il a été construit de façon à ce que sa sémantique puisse être spécifiée à travers une description logique très expressive.

Ainsi, DAML+OIL est la combinaison de ces deux langages. Il doit ses primitives de modélisation intuitives aux "frames". Il utilise la syntaxe des standards XML et RDF. Il adopte la logique de description pour exprimer la sémantique formelle et les mécanismes de raisonnement. D'un point de vue formel, DAML+OIL est basé sur une logique de description très expressive SHIQ [49]. En outre, DAML+OIL permet de définir un ensemble de constructeurs et d'axiomes, qui permettent respectivement de créer une ontologie et de détecter automatiquement les incohérences, et classer automatiquement les concepts d'une ontologie. Les principaux axiomes du langage DAML+OIL sont `daml:Class` (définition de classes), `daml:objectProperty` (création de propriétés), `daml:subClassOf` (définition de sous-classes), `daml:subPropertyOf` (définition de souspropriétés), `daml:disjointWith` (disjonction entre classes) et `daml:inverseOf` (définition de deux propriétés inverses).

Cependant, DAML+OIL présente aussi les limites suivantes. En effet, DAML+OIL ne permet pas la composition et la fermeture transitive. Il prend en compte uniquement des

relations unaires et binaires. Le langage DAML+OIL ne permet pas la déclaration des variables. Le langage DAML+OIL n'est pas un langage normalisé. Néanmoins, DAML+OIL est considéré comme le premier langage à fournir des mécanismes d'inférence sur les concepts d'ontologies.

L'apparition de ces nombreux langages, poursuivant pour la plupart des objectifs communs, a encouragé le World Wide Web Consortium (W3C) à recommander un langage standardisé plus puissant au niveau de l'expressivité. Ce langage est spécialement conçu pour la représentation des ontologies dans le cadre du Web sémantique. Le langage d'ontologies recommandé est le langage OWL. Il est dérivé du langage DAML+OIL. Le langage OWL couvre la plupart des caractéristiques du langage DAML+OIL et renomme la majorité de ses primitives. La sous-section suivante décrit le langage OWL de représentation des ontologies pour le Web.

II.3.5.4. OWL

OWL (Ontology Web Language) est un langage basé sur RDF. Il enrichit le modèle du schéma RDF en définissant un vocabulaire riche pour la description d'ontologies complexes. Le langage OWL est basé sur une sémantique formelle définie par une syntaxe rigoureuse. Le langage OWL est un langage pour la représentation des ontologies dans le cadre du Web sémantique. Il représente une extension du vocabulaire de RDF(S). Le langage OWL est dérivé du langage d'ontologies DAML+OIL. Le langage OWL permet, en plus des primitives RDF(S), des relations entre les classes (disjonction, intersection, union, etc.), la cardinalité, l'égalité, etc. Le langage OWL permet la définition des types de propriétés (propriété d'objet, annotation, etc.), des caractéristiques des propriétés (par exemple le symétrique et la transitivité) et des classes énumérées.

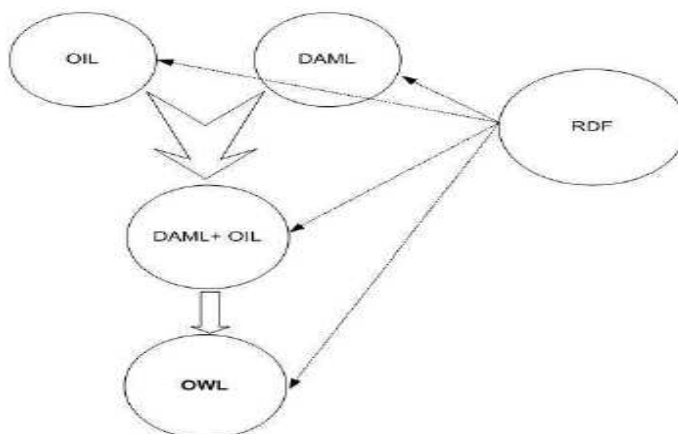


Figure II.3.5.4.a : Origine du langage OWL

Le langage d'ontologies OWL est décliné en trois sous-langages possédant une expressivité ascendante à savoir : OWL-Lite, OWL-DL (Ontology Web Language-Description Logic) et OWL-Full. La raison principale de cette division concerne la complexité, la calculabilité et l'implémentation du langage.

Le sous-langage OWL-Lite possède la complexité formelle la plus basse et une expressivité minimale. Il est recommandé pour la représentation des thésaurus, des

taxonomies ou hiérarchie de classes avec des contraintes simples. Le sous-langage OWL-DL est caractérisé par une expressivité maximale avec une complétude computationnelle (un temps de calcul $_ni$) [42].

Le sous-langage OWL-DL est dédié pour la représentation d'ontologies nécessitant une puissance d'expressivité toute en garantissant la calculabilité. Il est destiné aux utilisateurs qui demandent une expressivité maximale tout en retenant la complétude du calcul (toutes les inférences sont garanties calculables) et la décidabilité (tous les calculs s'achèveront dans un intervalle de temps $_fini$).

Le sous-langage OWL-Full est conçu pour les utilisateurs, qui nécessitent une expressivité maximale. Ce langage possède une syntaxe libre de RDF. Ce sous-langage ne garantit pas la calculabilité. Le OWL-Full est considéré comme une extension du langage RDF, tandis que les deux sous-langages OWL-Lite et OWL-DL sont considérés comme des extensions restreintes de RDF. Cependant, ce sous-langage OWLFull est une extension de OWL-DL, qui est à son tour une extension de OWL-Lite. Tous les documents OWL sont des documents valides en RDF. Un document RDF est un document OWL-Full, par contre seulement quelques documents décrits avec le langage RDF sont des documents valides en OWL-Lite et OWL-DL.



Figure II.3.5.4.b : Les types du langage OWL

L'utilisation du langage de représentation OWL dans le cadre d'apprentissage à distance permet d'une part de faire reposer le système E-learning, et de représenter le contenu pédagogique sémantiquement pour faciliter l'exploitation d'un contenu annoté durant le processus d'apprentissage.

II.3.6. Comparaison des langages de représentation

Dans cette section, on étudie les différentes caractéristiques d'expression de chaque langage. En effet, l'étude comparative permet de mettre en valeur les différentes primitives appartenant aux différents langages [43].

	XML & DTD	XML schema	RDF(S)	DAML+OIL	OWL
Listes			X	X	X
Cardinalité	X	X		X	X
Expressions de classe				X	X
Types de données		X		X	X
Définition de classe				X	X
Énumérations	X	X		X	X
Equivalence				X	X
Extensibilité			X	X	X

Sémantiques formelles			X	X	X
Héritage			X	X	X
Inférence				X	X
Restrictions locales				X	X
Contraintes qualifiées				X	
Réification			X	X	X

Tableau II.3.6 : Comparaison des langages de représentation

II.3.7. L'utilisation des ontologies

Au début, Les ontologies ont été connues pour une large utilisation dans le domaine de l'intelligence artificielle. Aujourd'hui les ontologies sont largement utilisées pour différents buts (traitement de langage naturel, gestion de la connaissance, E-commerce, intégration intelligente d'information, le Web sémantique, E-Learning, ...etc.) dans différentes communautés (c'est-à-dire, ingénierie de connaissance, bases de données et génie logiciel). Notons que la communauté de bases de données aussi bien que la communauté de conception orientée objet construisent également des modèles de domaine en utilisant les concepts, les relations, les propriétés, ...etc., mais la plupart du temps les deux communautés imposent des contraintes moins sémantiques que celles imposées dans les ontologies lourdes.

Le Web sémantique [116] ajoute la sémantique compréhensible par la machine (méta-information) au World Wide Web en utilisant une ontologie pour définir et organiser cet espace de méta-information. Le Web sémantique vise à réaliser l'intégration de toutes les sources d'informations sur le Web, permettant la réutilisation des données à travers les applications et rendant la recherche intelligente sur Internet possible.

Donc, nous pouvons dire que les ontologies possèdent un grand nombre d'applications et d'usages tels que :

- Les systèmes de E-learning et E-commerce
- Les systèmes de Recherche d'informations (RI).
- Les inférences, pour découvrir les incohérences sur les connaissances ou les données.
- Les systèmes à base de connaissances dédiés à la résolution de problèmes.
- L'aide au diagnostic : des maladies, des problèmes techniques etc.
- L'interopérabilité des systèmes d'information.
- La coopération des logiciels.
- Les systèmes de traitement automatique du langage naturel.
- Décrire et traiter des ressources multimédia.
- Permettre l'intégration des ressources hétérogènes d'information.
- Intégration d'information géographique.
- Gestion de ressource humaine.
- Aide à l'analyse en biologie, suivi médicale informatisé ;

II.4. Web sémantique et E-Learning

Dans un environnement de Web sémantique, nous pouvons exploiter plusieurs propriétés tel que : sens partagé commun et les métadonnées traitables par les machines, offertes par un ensemble adéquat d'agents, apparaît suffisamment puissante pour satisfaire les

exigences du système du E-Learning : rapide, juste à temps et apprentissage pertinent [8][7]. Le Web sémantique permet l'annotation sémantique du contenu éducatif, donc on peut facilement le combiner avec les autres parties de cours ou le retrouver à la demande. Selon ses préférences, un utilisateur peut facilement trouver le contenu d'apprentissage utile. Le processus est basé sur les requêtes Web Sémantique et la navigation à travers le matériel d'apprentissage activée par un background ontologique. De ce fait, le Web Sémantique peut être traité comme une plateforme adéquate pour implémenter un système E-Learning, du moment qu'il fournit tous les moyens pour le développement d'une ontologie (d'apprentissage), l'annotation basée ontologie du contenu éducatif d'apprentissage, leur composition dans des cours et la livraison active des cours à travers des portails d'apprentissage.

Dans le E-Learning, les ontologies sont habituellement utilisées pour représenter les contenus pédagogiques [7][8], le modèle d'apprenant et les services qui sont offerts dans les systèmes du E-Learning et qui sont donnés aux apprenants. Dans le tableau suivant [25], nous présentons les possibles utilisations du Web Sémantique pour la réalisation des exigences du E-Learning.

Dimension	E-learning	Web sémantique
Livraison	L'apprenant qui détermine son propre agenda	Éléments de connaissance (des matériaux d'apprentissage) sont distribués sur le web, mais ils sont liés à une ontologie (s) commune et agréée. Cela permet la construction d'un cours personnalisé, par interrogation sémantique des sujets d'intérêt.
Réactivité	Réactionnaire: Répond à problème à la main	Les agents logiciels sur le Web sémantique peuvent utiliser la langue de service commune, qui permet la coordination entre les agents et prestation proactive de matériaux d'apprentissage dans le contexte des problèmes réels. La vision est que chaque utilisateur a son propre agent personnalisé qui communique avec d'autres agents
Accès	Non-linéaire: Permet d'accéder directement à la connaissance dans un ordre quelconque selon chaque situation	L'utilisateur peut décrire la situation à portée de main (objectif de l'apprentissage, les connaissances préalables ...etc) et effectuer l'interrogation sémantique pour le matériel d'apprentissage adapté. Le profil d'utilisateur est également

		pris en compte. L'accès au savoir peut être étendu par la navigation sémantique définie.
Symétrie	Symétrique: L'apprentissage se fait comme une activité intégrée	Le Web sémantique offre la possibilité de devenir une plateforme d'intégration de tous les processus d'affaires d'une organisation, y compris les activités d'apprentissage.
Modalité	Continue: l'apprentissage fonctionne dans les boucles parallèles et ne s'arrête jamais	Livraison active de l'information (basée sur des agents personnalisés) crée un environnement d'apprentissage dynamique.
Autorité	Distribué: contenu provient de l'interaction des participants et les éducateurs	Le Web sémantique sera aussi décentralisée que possible. Cela permet une gestion de contenu coopératif efficace.
Personnalisation	Personnalisé: le contenu est déterminé par les besoins de l'utilisateur individuel et vise à satisfaire les besoins de chaque utilisateur	Un utilisateur (en utilisant un agent personnalisé) recherche pour le matériel pertinent pour ses besoins d'apprentissage. L'ontologie est le lien entre les besoins et les caractéristiques de l'utilisateur du matériel d'apprentissage.
Adaptabilité	Dynamique : Modifications de contenu sans cesse à travers l'entrée d'utilisateur, les expériences, les nouveaux apprenants, les règles de gestion et l'analyse heuristique	Le Web sémantique permet l'utilisation de la connaissance fournie sous diverses formes, par l'annotation sémantique de contenu.

Tableau II.4 : comparaison entre E-learning et E-learning en utilisant le Web sémantique

II.5. Conclusion

Dans ce chapitre, nous avons présenté les principaux concepts du Web sémantique et ses différents éléments. Les ontologies sont apparues comme la principale représentation de la sémantique d'un domaine donné. Le Web Sémantique peut être défini comme un substrat supportant des fonctions avancées pour la collaboration (homme-homme, homme-machine, machine-machine), qui permet de partager des ressources et de raisonner sur le contenu de ces dernières. Une des caractéristiques principales est la compréhension partagée basée sur un squelette d'ontologie.

L'ontologie est un composant principal dans les applications du web sémantique, ce qui nécessite un grand effort pour la réaliser, une partie importante du temps de la réalisation de ces systèmes est consommé dans la création de l'ontologie.

Le chapitre suivant présente une étude pour les travaux sur E-Learning qui exploitent le paradigme agent comme outil de modélisation et les travaux de formation à distance qui utilisent la technologie du web sémantique.

Chapitre III : Etat de l'art sur le E-Learning sémantique à base d'agent

III.1. Introduction

L'intelligence artificielle distribuée (IAD) est apparue comme une solution pour pallier aux insuffisances et les limites de l'intelligence artificielle dans des nombreux problèmes complexes qui sont physiquement et fonctionnellement distribués [18]. L'IAD s'intéresse à l'étude des connaissances et des raisonnements nécessaires pour la coordination d'actions d'un ensemble des modules de traitements [79] [78].

Les systèmes multi-agents sont apparus comme un sous-domaine de l'IAD pour résoudre les problèmes que l'IA a échoué à résoudre (les limites théoriques et applicatives des techniques classiques de l'IA). L'intérêt qu'ils suscitent est lié à leur capacité d'aborder les problèmes complexes d'une manière distribuée et de proposer des solutions modulaires et robustes. De nos jours, l'utilisation des systèmes multi-agents a pris une place de plus en plus importante en informatique. Ces systèmes ont dominé le domaine de l'intelligence artificielle, de celui des systèmes informatique distribués, de la robotique et du génie logiciel. C'est une discipline qui s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités autonomes et flexibles appelées agents.

Ce chapitre introduit d'abord, les travaux à base de la sémantique ensuite nous présentons une étude des travaux sur E-Learning qui exploitent le paradigme agent comme outil de modélisation.

III.2. E-Learning à la base de la sémantique

Plusieurs travaux ont créé des documents d'apprentissage à l'aide de l'annotation ou les ontologies. Dans cette section, nous examinons certains travaux récents qui sont proches de notre travail. Dans [1], l'auteur a présenté un corpus adaptatif et ouvert de E-learning en utilisant les ontologies. Il a essayé de développer un contenu adapté et bien organisé pour assurer l'apprentissage personnalisé et la réutilisation de ce contenu. L'auteur propose un modèle conceptuel pour trouver le contenu externe, ce dernier sera ajouté aux ressources d'apprentissage et décrit sémantiquement. Dans le cas de la modification du contenu d'apprentissage par n'importe quel utilisateur (élève ou enseignant), le système trouve, manipule, annote, et organiser le nouveau contenu d'apprentissage. En outre, il effectue l'enrichissement de la description du nouveau contenu à l'aide de l'ontologie interactif d'apprentissage. Pour ce but, l'auteur a défini quatre couches: la couche de l'interface utilisateur, la couche de service, la couche de métadonnées et de la connaissance et la couche de ressources d'apprentissage.

Dans un autre travail [2], les auteurs ont proposé la création de contenus d'apprentissage et l'intégration des annotations en utilisant des ontologies dans les différentes couches (trois couches ont été définis: Curriculum ontologie, Syllabus ontologie, et Objet

ontologie). L'objectif principal de ce travail est de développer un système de soutien E-learning basé sur l'ontologie qui permet aux apprenants de construire des parcours de formation d'adaptation à travers la compréhension curriculum, Syllabus les objets de cours. Les fonctionnalités de ces composants du système sont les suivantes:

1. Comprendre la demande de l'utilisateur et l'invocation de questionnaires appropriés.
2. Traduire la requête de l'utilisateur sous forme d'une liste de mots-clés ou modèle dans SPARQL ou TMQL.
3. La création d'instances d'ontologies par la collecte, l'analyse et la classification des ressources, c'est à dire des pages Web du programme d'enseignement, la description cours des pages Web, et ainsi de suite.
4. Identifier et gérer les résultats d'apprentissage en termes de cours individuels et des zones académiques.
5. Création semi-automatique des chemins d'apprentissage adaptatif selon les besoins d'apprentissage de l'utilisateur.

Dans ce travail [2], les auteurs ont défini Curriculum un ensemble de cours et de syllabus, qui fait partie du Curriculum et un squelette d'un cours. Ils représentaient comme une collection de différentes ressources liées à un certain cours. Diverses relations sémantiques sont organisées grâce à l'ontologie Curriculum: hasSubtype, prerequisiteOf, basicOf, advancedOf, combinedOf, et ainsi de suite, entre les cours individuels. Curriculum ontologie conceptualise la connaissance des concepts de programmes, à savoir: ProgramOfStudy, Course, KeyConcept, AttainmentGoal, AttainmentLevel, et établit des connexions directes avec un ou plusieurs ontologies de syllabus. Ontologie de Syllabus représente les structures internes et externes de cours identifiés dans l'ontologie Curriculum. L'ontologie de Syllabus définit le vocabulaire unifié de programme pour aider à compromettre les différents vocabulaires utilisés par les différents instructeurs. Une classe de syllabus, qui est le concept de base de l'ontologie de Syllabus, dispose de 9 propriétés de type de données: titleOfCourse, description, gradingPolicy, goalOfCourse ...etc et 12 propriétés de type d'objet: oldVersionOf, hasInstructor, hasMaterial, hasSchedule, hasLectureRoom, pour décrire les connaissances sémantiques extraites de Syllabus traditionnels.

Les auteurs de [4] ont proposé une ontologie pour faciliter la découverte et la réutilisation des objets d'apprentissage stockés dans des dépôts. Ils ont défini les ontologies pour marquer la structure des objets d'apprentissage afin de permettre aux machines de comprendre ces objets d'apprentissage. Ils reposent sur plusieurs types de relations structurelles: Prev, Next, isPartOf, hasPart, References, isReferencedBy, IsBasedOn, isRequiredBy ... etc.

Un autre travail qui utilise l'indexation des objets d'apprentissage est présenté dans [15]. Dans ce dernier, en plus d'utiliser la norme LOM (Learning Object Metadata), les auteurs ont utilisé un vocabulaire d'indexation spécifiques et un vocabulaire d'indexation structurée pour les objets d'apprentissage plus élémentaires pour soutenir les tâches de récupération de développeurs. Pour surmonter les difficultés de l'indexation sémantique, ils ont utilisé trois questions concernant la cohérence d'indexation des objets d'apprentissage:

- 1) la mesure dans laquelle différents indexeurs annotent de la même manière;
- 2) la mesure dans laquelle la structure dans les listes de valeurs prend en charge l'indexation uniforme;

3) différents degrés de cohérence dans l'annotation des différents types et attributs des médias.

Pour les attributs les plus abstraits et les types de médias, ils ont proposé un ensemble d'ontologies qui peuvent contribuer à une consistance plus élevée pour les indexer et pourraient améliorer la récupération en faisant des concepts plus abstraits disponible.

Le matériel d'apprentissage de haute qualité est coûteux à créer, alors qu'il est très important d'assurer à la réutilisation des objets d'apprentissage. Pour cet objectif, les auteurs de [12] ont proposé une approche qui annote les contenus d'apprentissage sémantiquement par les étiquetant avec des métadonnées standards. Les auteurs ont étudié la faisabilité de marquage automatique des matériaux d'apprentissage avec la spécification IEEE LOM 9.0 de métadonnées. En outre, ils présentent l'approche de la machine d'apprentissage en utilisant k-Nearest Neighbor et le réseau de neurones de propagation pour identifier automatiquement le sujet du matériel d'apprentissage. Ils ont présenté les résultats des tests du classificateur : environ 84 % pour K-NN (K-Nearest Neighborhood) et une précision de 93% pour le réseau de neurones de propagation.

Pour améliorer la pertinence de la recherche d'information, les auteurs ont proposé dans [16] une approche basée sur l'utilisation de l'indexation de la collection de documents en utilisant l'ontologie de domaine. Aussi, ils ont utilisé des liens sémantiques entre les documents ou fragments de documents de la collection afin de permettre la recherche de tous les documents pertinents. Le test de cette approche est sur le domaine du E-learning de l'informatique dans le contexte du web sémantique.

III.3. E-learning et systèmes multi-agents

L'opération de conception d'un système d'apprentissage à distance fait appel à un ensemble de processus complexes qu'il est parfois difficile d'associer dans un système cohérent et évolutif. Les systèmes multi-agents s'avèrent pertinents pour résoudre ces problèmes parce qu'ils possèdent des caractéristiques qui permettent de mieux structurer les systèmes de transmission de connaissances et ouvrent de nouvelles perspectives d'assistance à la formation à distance. La technologie de systèmes multi-agent a été utilisée par nombreux systèmes des applications d'enseignement à distance comme par exemple dans [122][56][101]. Dans cette section on va présenter quelques notions de l'agent et systèmes multi-agents ensuite les travaux de E-learning basés agent :

III.3.1. Le concept d'agent

Dans la littérature, il existe plusieurs définitions parce que le concept d'agent est d'une généralité qui le rend difficile à définir et peut être vue par plusieurs angles. Les chercheurs ont pris l'habitude de définir un agent plutôt par ses propriétés. Dans ce qui suit, nous présentons quelques-unes :

- Une définition très générale de l'agent donnée par Russel et Norvig: « *Un agent est une entité qui perçoit son environnement et agit sur celui-ci* » [19].
- Jennings et Wooldridge [20] définissent l'agent comme un système informatique situé dans un certain environnement, capable d'exercer de façon autonome des actions sur cet environnement en vue d'atteindre ses objectifs.

- Deloach et al [80] pour leur part définissent un agent comme un ensemble de processus qui communiquent entre eux pour atteindre un objectif donné.
- Un agent est un système informatique, **situé** dans un environnement, et qui agit d'une façon **autonome** et **flexible** pour atteindre les objectifs pour lesquels il a été conçu [20].
- Selon Chaib-Draa [81][86], un agent est un système informatique, situé dans un environnement, qui agit d'une façon autonome et flexible pour atteindre les objectifs pour lesquels il a été conçu.
- M. Minsky donne différentes définitions du terme agent comme suite: une personne qui travaille et gère une agence, une substance qui provoque une réaction chimique, un organisme qui provoque une maladie,... etc.
- Selon Ferber [82] l'agent est une entité physique ou virtuelle qui est capable d'agir dans son environnement, pouvant communiquer avec d'autres agents, il est doté d'autonomie et possède des ressources propres, son comportement est la conséquence de ses objectifs, de sa perception, de ses représentations, de ses compétences et des communications qu'elle peut avoir avec les autres agents, il possède des compétences et offre des services.

On remarquera aisément que la définition d'un agent est celle d'un organisme vivant dont le comportement qui se résume à communiquer, à agir et éventuellement à se reproduire vise à la satisfaction de ses besoins et de ses objectifs à partir de tous les éléments dont ils disposent [81].

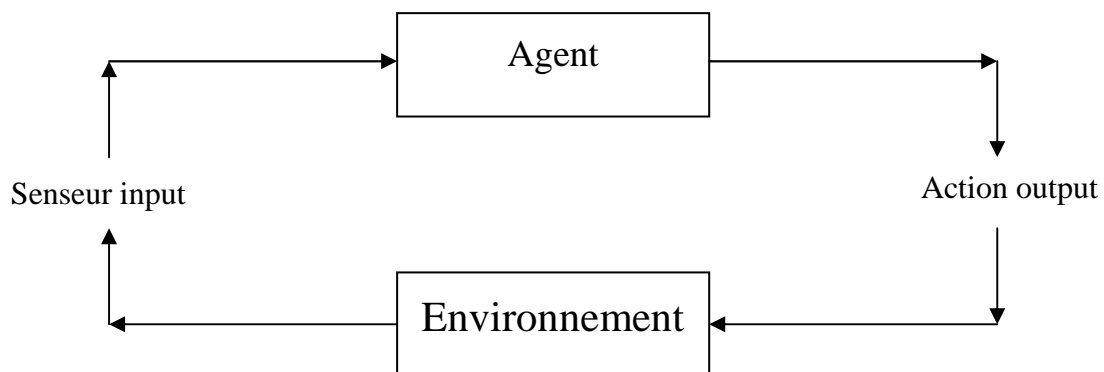


Figure III.3.1 : Agent agit sur l'environnement

L'objet est une technologie déjà maîtrisée par les chercheurs. Afin de maîtriser la nouvelle technologie ou d'un nouveau concept d'agent, plusieurs notions d'agent ont été comparées par rapport des notions voisines de l'objet [83]. Un agent, comme un objet, est défini par un état (Attributs) et un comportement (Méthodes) mais il existe des différences majeures [84] :

- Un agent contrôle son comportement, c'est-à-dire qu'il décide que faire des informations qu'il reçoit ou perçoit. Tandis que, un objet ne contrôle que son état par des méthodes d'accès. Dans un SMA, il y a plusieurs flux de contrôle comme pour des objets concurrents mais plusieurs types de contrôles sont néanmoins possibles (Réactif, délibéré, social, ...etc).
- L'encapsulation de l'état interne est une faculté des agents comme les objets et peuvent agir sur cet état par le biais de leurs méthodes. Dans le cas d'objet, l'invocation d'une méthode par un autre objet (par l'envoi de message) doit être explicite. Cependant, l'agent

recevant une information décidera par lui-même de l'action subséquente [84][85]. Le modèle standard d'un objet ne dit rien à propos de ces types de comportements [84][85], tandis que le comportement de l'agent est très flexible.

- Une autre différence entre les objets et les agents se présente dans le fil d'exécution. Chaque agent est censé d'avoir son fil d'exécution propre (thread) [84][85]. Mais dans un système développé par l'objet standard, il existe un fil d'exécution unique pour tout le système, bien que l'on voie apparaître du parallélisme dans la programmation orientée objet (POO) avec des langages comme Java offrant la possibilité de lancer plusieurs fils d'exécution en même temps. Les objets actifs sont généralement autonomes, c'est-à-dire qu'ils peuvent exhiber un comportement sans avoir été sollicités par un autre objet [56].
- Odell et al. [83] présentent les agents comme une extension d'objets actifs possédant une prise d'initiative (la capacité à initier des actions sans intervention externe) et un libre arbitre (la capacité à refuser ou à modifier une requête externe). Odell et ses collègues définissent un agent comme un objet qui peut prendre des initiatives et refuser de répondre à une sollicitation.
- Nwana [87] propose sept catégories d'agents : agents collaboratifs, agents d'interface, agents mobiles, agents d'information ou d'Internet, agents réactifs, agents hybrides, agents intelligents [56].

III.3.2. Système multi-agents

Le passage du comportement individuel aux comportements collectifs est considéré, non seulement comme une extension, mais aussi un enrichissement de l'intelligence artificielle, d'où émergent de nouvelles propriétés et de nouveaux comportements. Les SMA permettent d'introduire dans un système, un ensemble d'agents qui sont capables d'interagir et sont dotés de connaissances, d'intentions et de capacité d'évolution différente. Donc l'intérêt de l'agent apparaît au sein d'un univers d'agents dans lequel les agents communiquent pour résoudre ces problèmes complexes, on parle des systèmes multi-agents (ou SMA). Dans un système multi-agents, l'agent a un comportement de temps en temps individuel ou collectif et il a ses objectifs à atteindre et les ressources et les compétences qu'il dispose [18].

La puissance d'un SMA ne provient pas du comportement individuel des agents mais des interactions entre ses agents. Le comportement collectif des agents se traduit à partir de leurs interactions : coopération, négociation et coordination. Les SMA sont à l'intersection de plusieurs domaines scientifiques: les systèmes distribués, les interfaces homme-machine, les bases de données et les bases de connaissances distribuées, les systèmes pour la compréhension du langage naturel, les protocoles de communication et les réseaux de télécommunication, la programmation orientée agents et le génie logiciel, la robotique cognitive et la coopération entre robots,... etc [18].

III.3.2.1. Définition d'un système multi-agents

Selon [81], un système multi-agents est un système distribué composé d'un ensemble d'agents. Contrairement aux systèmes d'IA, qui simulent dans une certaine mesure les capacités du raisonnement humain, les SMA sont conçus et implantés idéalement comme un ensemble d'agents interagissants, le plus souvent, selon des modes de coopération, de concurrence ou de coexistence [85].

Selon Guessoum, un système multi-agents est un ensemble d'entités (physiques ou virtuelles) appelées agents, partageant un environnement commun (physique ou virtuel), qu'elles sont capables de percevoir et sur lequel elles peuvent agir. Les perceptions permettent aux agents d'acquérir des informations sur l'évolution de leur environnement, et leurs actions leur permettent entre autres de modifier l'environnement [88]. Les agents interagissent entre eux directement ou indirectement, et exhibent des comportements corrélés créant ainsi une synergie permettant à l'ensemble d'agents de former un collectif organisé.

Plus précisément, selon Ferber [18] un SMA est un système composé des éléments suivants:

- E : un environnement, c'est-à-dire un espace disposant généralement d'une métrique.
- O : un ensemble d'objets : on peut associer une position E pour chaque objet à un moment donné. Donc ces objets sont situés. Ces objets peuvent être perçus, créés, détruits et modifiés par les agents.
- A : un ensemble d'agents : qui sont des objets particuliers ($A \subseteq O$), ce sont les entités actives de système.
- R : un ensemble de relations : qui unissent des objets et des agents entre eux.
- Op : un ensemble d'opérations : elles permettent aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O.
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette technique de modification, que l'on appellera les lois de l'univers [18].

Dans un système, il n'est pas nécessaire de décrire tous ces éléments, selon Yves Demazeau [89], un SMA est un système composé d'agents qui évoluent dans un environnement et interagissent de manière organisée. Ainsi, un SMA est défini selon l'équation ci-dessous :

SMA = Agents + Environnement + Interactions + Organisations (AEIO)

Contrairement d'une collection d'agents indépendants, les agents ont des relations d'interaction entre eux en vue de réaliser conjointement une tâche ou d'atteindre conjointement un but particulier [86]. Par ailleurs, cette notion est une caractéristique principale d'un SMA en plus de l'organisation qui est produite par l'interaction entre les agents. Un système multi-agents est caractérisé par son organisation sociale [82] c'est à dire la manière dont le groupe est constitué, à un instant donné, pour pouvoir fonctionner. Selon Ferber [92], toute organisation est le résultat d'une interaction entre agents et le comportement des agents est contraint par l'ensemble de structures organisatrices. Dans un système ouvert, une organisation n'a pas nécessairement un objectif global et les agents peuvent être en concurrence, interaction, négociation, coopération, coordination ou en communication :

- **L'Interaction** : considérée comme la caractéristique principale d'un système multi-agents. Ferber [18] définit une interaction comme étant la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques. Grâce à l'interaction entre les entités informatiques autonomes (agents entre eux) et leur environnement, ces agents peuvent combiner les efforts et s'entraider ou bien entrer en compétition et négocier [93].
- **La coopération** : est la forme générale d'interaction. Une activité de coopération prend place si un ensemble particulier d'agents converge vers un but global par l'achèvement de leurs propres buts locaux.
- **La négociation** : elle joue un rôle fondamental dans les activités de coopération en permettant aux personnes de résoudre des conflits qui pourraient mettre en péril des comportements coopératifs. Donc, la négociation [94] est le processus d'améliorer les accords en réduisant les inconsistances et l'incertitude sur des points de vue communs ou des plans d'action grâce à l'échange structuré d'informations pertinentes.
- **La coordination** : Les agents doivent partager le même environnement pour pouvoir coordonner entre eux. La coordination peut être définie comme l'articulation des actions individuelles accomplies par chacun des agents de manière à ce que l'ensemble aboutisse à un tout cohérent et performant [92][95]. Selon Chaib Draa [81][97] : la coordination est une question centrale pour les SMA et la résolution de systèmes distribués. En effet, sans coordination un groupe d'agents peut dégénérer rapidement en une collection chaotique d'individus [96]. Selon Nwana, la coordination améliore l'action du groupe en diminuant les conflits et en augmentant les performances [87].
- **La communication** : La communication dans les systèmes multi-agents est à la base des interactions et de l'organisation des agents mais aussi de la résolution coopérative des problèmes. En communiquant [97], les agents sont capables de coopérer, de négocier, de coordonner leurs actions ou de réaliser des tâches en commun. Afin de faciliter la communication et l'interopérabilité entre les agents, plusieurs langages de communication ont été développés à titre d'exemple : KQML (pour Knowledge Query and Manipulation Language) et FIPA-ACL (FIPA Agent Communication Language) qui est une extension du langage KQML [10].

III.3.2.2. Les plateformes de développement des SMA

Il existe plusieurs environnements de développement permettant de construire des applications SMA. On présente les deux plateformes les plus utilisées JADE et Madkit.

III.3.2.2.1. JADE

JADE "Java Agent Development Environment" facilite le développement des systèmes multi-agents. Elle est une plateforme implémentée en Java qui permet l'implémentation des systèmes multi-agents et des applications conformes aux standards FIPA, notamment pour permettre l'interopérabilité entre agents [98]. JADE permet la création des agents et l'envoi des messages entre eux et autorise l'utilisation de n'importe quelle plate-forme et des outils associés, en particulier le fonctionnement dans un environnement ouvert et dynamique,

fournit une classe de base Agent qui traite toutes les tâches liées aux fonctionnement des agents, telles que l'enregistrement, la configuration, la gestion à distance, etc., et un ensemble de méthodes qui peuvent être appelées pour implémenter les tâches spécifiques à l'agent, comme l'envoi des messages, utilisation des protocoles d'interaction standards, ...etc., et gérant tous les aspects de contrôle des comportements. JADE propose également des classes de bases pour des types de comportements spécifiques (CyclicBehaviour, ParallelBehaviour, ReceiverBehaviour, SenderBehaviour, SequentialBehaviour...). La méthode 'action()', qui est exécutée en premier et qui correspond au code du comportement défini pour l'agent, et la méthode 'done()' qui est exécutée après la méthode action(), et qui vérifie si le comportement doit être remis dans la file d'attente de l'ordonnanceur ou non. Un comportement peut donc être exécuté plusieurs fois. Un agent n'exécute qu'un seul comportement à la fois [99].

La plateforme Jade se compose principalement de deux composants (figure III.5.1). Une plateforme agents compatible FIPA qui est fournie avec une License open source LGPL et des packages offrant aux développeurs un ensemble de fonctionnalités prédéveloppées et prenant en charge la gestion des threads agents, le transport des messages, l'architecture distribuée, ...etc. JADE utilise le langage de communication FIPA-ACL (Agent Communication Language) et a implémenté le protocole Contract-Net qui a été normalisé par le FIPA.

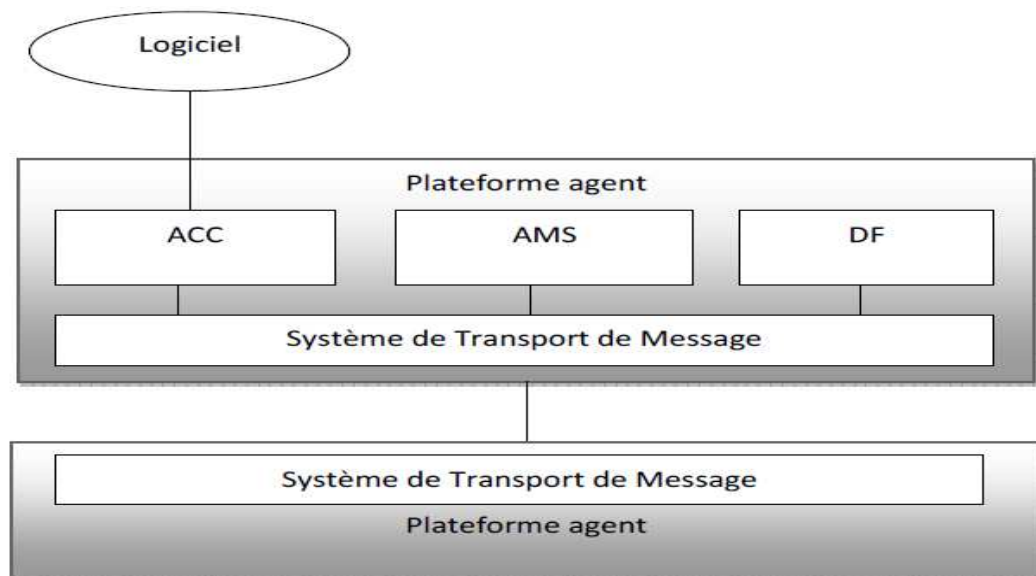


Figure III.3.2.2.1 : Architecture de la plateforme JADE [10]

A chaque démarrage de la plateforme, trois modules sont actives :

- le DF (pour Director Facilitator) qui fournit à la plateforme un service de pages jaunes.
- l'ACC (pour Agent Communication Channel) qui gère la communication entre les agents.
- l'AMS (pour Agent Management System) qui sert de superviseur pour la plateforme.

Tout agent créé doit s'enregistrer auprès de l'AMS pour obtenir son identifiant (AID), indispensable à la communication. En effet, chaque agent peut obtenir la liste de tous les AID des agents de la plateforme et ainsi les agents peuvent communiquer en s'envoyant des messages à partir des AID.

III.3.2.2.2. Madkit

Madkit (Multi-Agent Développement Kit) est une plateforme multi-agents modulaire et scalable écrite en java (figure). Elle permet la création des SMA en se basant sur le modèle organisationnel AGR. Ce modèle sert à la réalisation des applications implémentées sur Madkit, mais aussi pour le fonctionnement interne de la plateforme. Madkit se base sur trois principes architecturaux : un micro noyau, une agentification des services et un modèle graphique componentiel [100].

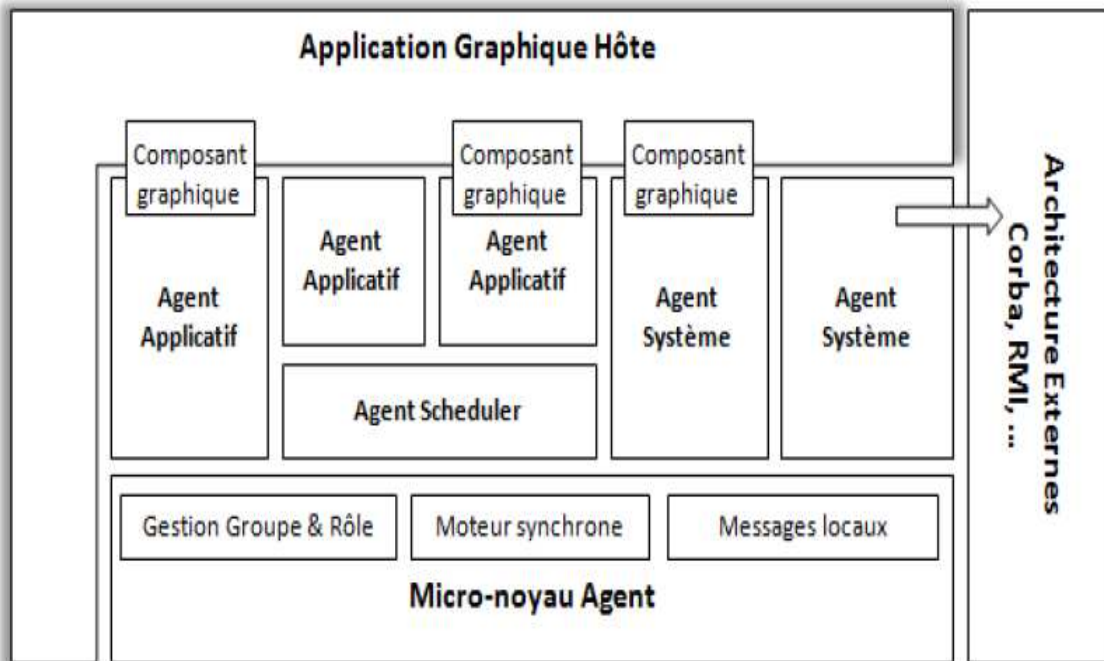


Figure III.3.2.2.2: Architecture de la plateforme Madkit [10]

a) Le micro-noyau : Le Micro-noyau de MadKit est très petit (moins de 100Ko de code). En effet, il contient un nombre réduit d'outils qui facilitent le déploiement des services du système et se charge de la gestion des mécanismes permettant l'utilisation des notions de groupes et de rôles. En effet, il a la responsabilité de maintenir une information correcte sur les membres des groupes et les rôles tenus. Il définit un ensemble de méthodes qui permettent à un agent de rejoindre un groupe, puis d'y demander un rôle particulier. Le noyau lui-même est en fait encapsulé dans un agent particulier, le KernelAgent qui est créé automatiquement à l'amorçage de la plate-forme [100].

b) L'agentification des services : Le Micro-noyau de Madkit maintient un ensemble minimal de fonctionnalités permettant le déploiement d'autres systèmes. Tous les autres services ne font pas partie du micro noyau et en particulier la communication en distribue. Ces services sont assurés par des agents encapsulés par un seul agent initial : c'est l'agentification des services. Par exemple, l'agent SiteAgent assure la communication inter-noyaux en utilisant d'autres agents qui offrent différents modes de communication.

c) Le modèle graphique componentiel : L'interface graphique de Madkit est componentielle et découplée du noyau et des agents ce qui permet de supporter différentes modes d'utilisation

et d'exploitation de la plateforme. En effet, Madkit est dotée d'un modelé graphique base sur des composants graphiques indépendants, utilisant la spécification JAVA Beans dans la version standard.

d) Structure et fonctions d'un agent : Tout agent qui tourne sous la plateforme Madkit est issu d'une classe d'agent abstraite nommée AbstractAgent. Cette classe d'agent offre un certain nombre des fonctionnalités de base, nécessaires pour le bon fonctionnement d'un agent quel que soit son modèle conceptuel [100].

III.3.2.2.3. Comparaison des plateformes

Notre travail se base sur l'approche sémantique pour E-learning à base d'agents. Pour cela nous devons choisir une plateforme de développement de systèmes multi-agents capable de supporter les caractéristiques de notre approche. Nous avons présenté une comparaison des deux plateformes multi-agents à savoir : Jade et Madkit. Nous nous sommes basés sur les critères définis par [88] pour choisir ceux qui correspondent à nos besoins :

Critère	Jade	Madkit
Langages utilisés	Développée en Java	Développée en Java
Systèmes d'exploitation compatibles	Tous	Tous
Logiciel libre	Oui	Oui
Utilisation du concept du groupe	Oui	Oui
Caractéristiques	Un thread qui a une liste des comportements (objets de type Behaviour)	Aucune architecture n'est imposée Chaque agent est issu de la Classe « AbstractAgent » et dispose de 4 états (création, activation, exécution et destruction)
Langage de communication	FIPA ACL	Pas de langage prédéfini de communication
Communication distribuée	Oui	Oui
Méthodologie d'analyse	construite selon les normes FIPA	Basée sur le modèle AGR
Nécessite la connaissance du langage de développement	Oui	Possibilité d'interprétation du code avec d'autres langages (Scheme, Python, BeanShell et Jess)
Standard	Conforme aux spécifications FIPA	Non
Bibliothèque	Disponible	Disponible
Documentation	Riche	Riche

Tutoriel	Disponible	Disponible
Aide en ligne	Oui	Oui
Manuels d'installation	Disponible	Disponible
Manuels d'utilisation	Disponible	Disponible

Tableau III.3.2.2.3 : Comparaison des plateformes JADE et Madkit [10].

Nous avons choisi JADE pour l'implémentation de notre système. Ce choix est motivé par le fait que cette plateforme est particulièrement destinée à des SMA fondés sur des critères organisationnels et qu'elle fonctionne en mode distribuée de manière transparente. JADE offre plusieurs qualités : simplicité, performance, création des agents avec comportement cyclique, communication en FIPA-ACL ...etc.

III.3.3. Quelques travaux de E-learning basés agent

Plusieurs travaux s'intéressent à l'utilisation des SMA dans le domaine E-learning. En effet, les systèmes multi-agents proposent une approche originale de conception de systèmes intelligents et coopératifs et apportent une solution intéressante aussi bien aux problèmes de structuration et d'exploitation des connaissances qu'aux problèmes liés aux mécanismes de coordination et de communication. Donc, les SMA offrent beaucoup des appropriés pour la conception d'un environnement d'apprentissage ou chaque membre doit gérer et échanger ses connaissances et collaborer avec les autres afin de réaliser ses buts. Dans cette section on va présenter quelques travaux de E-learning basés agent :

III.3.3.1. Baghera

Le projet Baghera [101] est un projet principal du Laboratoire *Leibniz*, son l'objectif est la formation et l'expérimentation de manière radicalement nouvelles et des perspectives sur la conception d'environnements de formation à distance. C'est une plate-forme d'enseignement et d'apprentissage à distance, elle propose les bases d'environnements informatiques pour l'apprentissage humain (EIAH) vu comme une communauté éducative composée d'agents humains et non-humains qui coopèrent et travaillent ensemble pour l'éducation des élèves.

La plateforme BAGHERA repose sur la coopération de trois domaines de recherche, la démonstration automatique, les systèmes multi-agents et la didactique des mathématiques. Cette plateforme est pour la production d'un environnement informatique pour l'apprentissage de la résolution de problèmes de la géométrie. Elle contient plusieurs agents logiciels qui coopèrent avec un ensemble d'acteurs humains (élèves et enseignements) pour fournir des services à des agents humains. En plus, cette plateforme a été implémentée avec l'environnement de développement d'applications multi-agents JATLite.

Dans la plateforme BAGHERA, les agents qui sont définis sont des agents spécialisés à chaque type d'utilisateurs, c'est à dire des agents sont spécialisés à l'élève et autres agents sont spécialisés à l'enseignant. Plus précisément, un élève interagit avec trois agents artificiels: Compagnon, Tuteur et Médiateur; un enseignant interagit avec deux agents artificiels : Compagnon et Assistant. Donc, lorsque un élève se connecte, trois agents sont activés et deux agents pour chaque enseignant connecté. Concernant l'espace de stockage,

chaque utilisateur dispose d'un espace de stockage, pour un enseignant l'espace est " casier électronique ", où l'espace de stockage d'un élève est appelé " cartable électronique ". Les agents présentent sur cette plateforme sont :

- a. **Agent Compagnon-Elève** : cet agent est destiné à accompagner l'élève lors de son travail. L'un des rôles de cet agent est de réaliser le lien avec le reste du système et les élèves. Aussi à travers l'agent Compagnon-Elève, l'élève peut accéder à son cartable électronique et à une interface graphique spécifique qui va lui permettre de travailler sur un de ses exercices, sauvegarder son travail et demander une vérification de la preuve qu'il élabore.
- b. **Agent Médiateur** : le rôle de cet agent est de destiner les solutions de l'élève pour la vérification vers l'agent approprié. Il permet l'analyse de la production d'un élève. Dans le cas de l'application développée autour de la géométrie, cet agent transforme la preuve élaborée par l'élève et exprimée en langage pseudo naturel, grâce à l'interface d'expression des preuves en une syntaxe compatible avec les outils de vérification de preuve de l'atelier d'inférences ATINF.
- c. **Agent Tuteur** : il est considéré comme le concepteur artificiel. Cet agent est obligatoire pour chaque élève enregistré dans le système. Lorsque l'enseignant est absent, l'agent Tuteur prend la charge du suivi pédagogique pour un apprentissage en autonomie au regard de son élève et de la situation d'apprentissage. Il gère le cartable de l'élève et lui donne l'analyse et l'évaluation de ses travaux en appuyant sur l'agent médiateur.
- d. **Agent Compagnon-Enseignant** : l'accompagnement de l'enseignant dans le système est son rôle essentiel. Il permet de réaliser l'interface entre le système et l'enseignant. C'est lui qui assure l'accès de l'enseignant à son casier électronique. L'enseignant s'appuie sur l'agent Compagnon-Enseignant pour connaître les élèves et les autres enseignants qui sont télé-présents, et communiquer avec les autres utilisateurs.
- e. **Agent Assistant** : chaque enseignant enregistré auprès du système dispose d'un agent assistant qui a une fonction essentielle de gérer le casier de l'enseignant. Il permet de fournir les exercices du casier de l'enseignant aux agents tuteurs qui ensuite les transmettent dans cartables électroniques gérés par les tuteurs.

Le schéma ci-dessous présente l'architecture générale du système. Les flèches en pointillé représentent les liens avec les autres agents associés à d'autres utilisateurs.

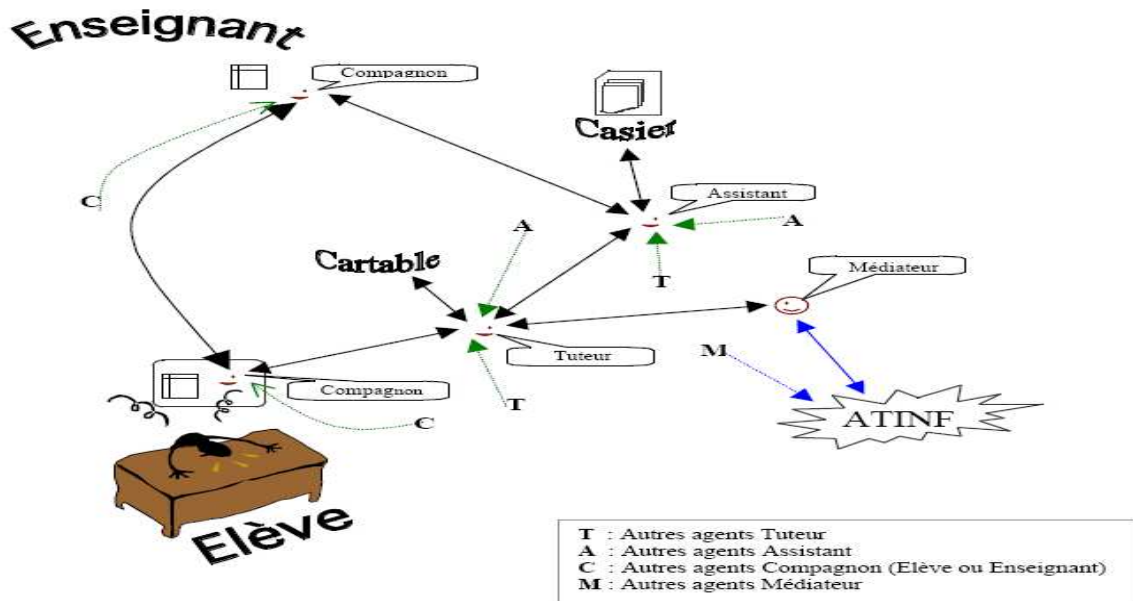


Figure III.3.3.1: Architecture générale de Baghera

III.3.3.2. SIGFAD

SIGFAD [56] est un système multi-agents d'aide à l'enseignement à distance. La réalisation de ce système a suivi la méthodologie MaSe couplée à agentTool [Deloach, 2001] qui est un environnement de développement permettant d'implémenter MaSE et est développé en Java. Les agents SIGFAD sont caractérisés par une architecture BDI. SIGFAD fournit des appréciations au niveau du groupe (l'état du groupe par rapport à la réalisation des activités, connexion des membres), au niveau d'un individu (sa productivité et sa sociabilité) ainsi qu'au niveau de l'activité (niveau de réalisation d'une activité par tous les participants) afin de permettre au tuteur de maintenir les groupes, les dynamiser et de bien conduire la séance de FAD.

SIGFAD distingue trois types d'utilisateurs : les apprenants, les tuteurs, le coordonnateur. Chaque utilisateur a un agent, localisé sur son poste de travail. L'agent devra avoir été installé auparavant. Cet agent est programmé comme une application Java sous forme d'*applets*. Ce choix a été dicté par les restrictions imposées aux *applets* (interdiction d'accéder au système de fichiers local, interdiction d'établir des connexions réseau avec des autres ordinateurs que celui à partir duquel l'*applet* a été chargé).

Un agent superviseur des interactions est défini, il réside sur le serveur et est chargé des transactions avec la base de données des interactions. L'agent constructeur de la base de données réside lui aussi sur le serveur et est chargé de construire la base de données des interactions au cours de la formation. Cet agent est celui qui assure l'interfaçage entre la plate-forme de téléformation et le système SIGFAD multi-agents.

Les agents communiquent entre eux en utilisant des requêtes KQML (*Knowledge Query and Manipulation Language*). KQML est un langage d'interrogation et de manipulation de connaissances. C'est un langage basé sur les actes du langage naturel qui est utilisé surtout pour la communication entre agents. Les utilisateurs communiquent avec leurs agents à travers des interfaces graphiques (GUI pour *Graphic User Interface*). La

communication avec la base de données est effectuée par des requêtes SQL (*Structured Query Language*).

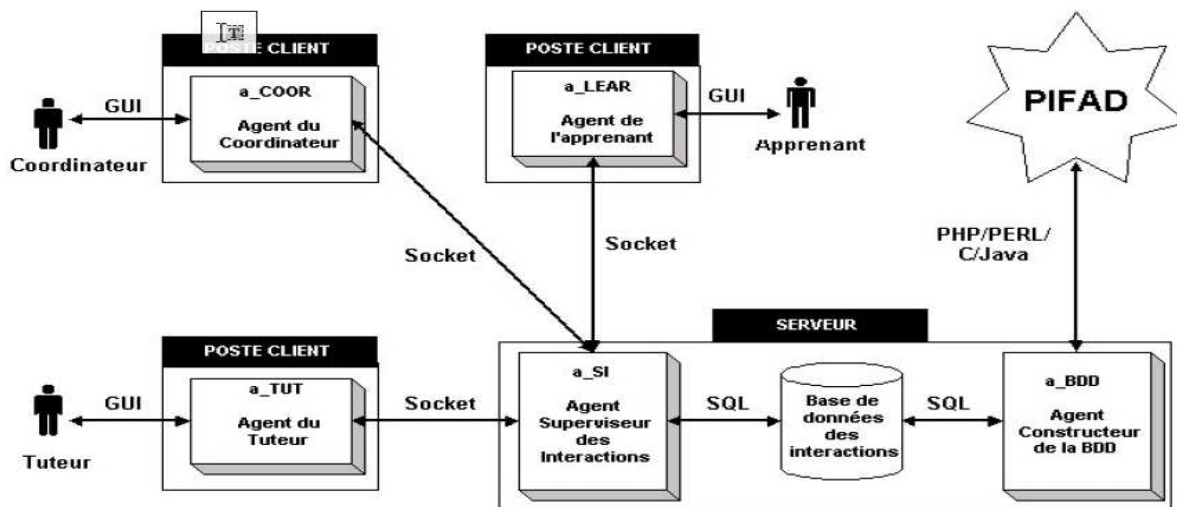


Figure III.3.3.2: Vue d'ensemble de SIGFAD

III.3.3.3. iPédagogique

C'est une plate-forme [102] auteur pour l'enseignement en présentiel et à distance d'unités de valeurs scientifiques et techniques dont la pédagogie est orientée projet. Les auteurs définissent deux principaux objectifs à cette plateforme : structurer et instrumenter les activités coopératives de suivi, de gestion et d'évaluation de projets d'étudiants ; et offrir une assistance aux utilisateurs étudiants et enseignants engagés dans cette coopération. Ces utilisateurs ont quatre possibilités d'utilisation d'iPédagogique qui sont l'enseignement, l'apprentissage, la réalisation et l'interaction.

La plateforme iPédagogique dispose aussi d'un système de médiation qui a pour but de faciliter son utilisation et de le rendre moins complexe pour les utilisateurs. Il est composé d'une interface homme-machine, d'une base de connaissance et d'un système de multi-assistance qui est la composante centrale du système de médiation et est conçu selon une approche multi-agents.

III.3.3.4. MASCE

Ce système [5] est destiné à l'apprentissage collaboratif à distance en utilisant les systèmes multi-agents. MASCE est un système qui complète l'apprentissage traditionnel en face-à-face. En effet, il offre plusieurs fonctionnalités aux apprenants : de revoir les documents de cours, de demander de l'aide et d'évaluer l'aide fourni afin de permettre au système d'avoir une liste de meilleurs aideurs, d'interagir avec leurs tuteurs ou les autres apprenants à l'aide des outils de travail collaboratif (chat, email).

Donc MASCE peut seulement continuer l'apprentissage entamé déjà en classe et de permettre au tuteur d'interagir avec les apprenants afin de les aider dans leur apprentissage, de les évaluer et de suivre leurs activités.

III.3.3.5. I-MINDS

Ce système multi-agents [103] supporte l'apprentissage collaboratif par ordinateur (Computer-Supported Collaborative Learning CSCL). Il fournit une communauté d'agents intelligents qui interagissent entre eux afin de servir les tuteurs et les apprenants. Aussi, il offre la possibilité d'interaction entre le tuteur et les apprenants. Le tuteur peut évaluer, former des groupes et contrôler les performances de chaque apprenant ainsi que celles du groupe. Ainsi, ce système permet à l'apprenant d'apprendre, d'interagir avec les autres apprenants ainsi qu'avec le tuteur et de former un groupe. I-MINDS offre aussi aux utilisateurs des outils de travail collaboratif (chat, tableau de bord). Ces plateformes ont été conçues selon le modèle client-serveur.

III.3.3.6. Etude comparative des plateformes présentées

La présentation des travaux dans le domaine de E-learning nous permet de situer notre travail par rapport à ces travaux. Les deux systèmes iPédagogique et Baghera présentent un environnement pour l'apprentissage mais le suivi des activités des apprenants est minimal. On note aussi que, iPédagogique offre l'opportunité de travail sur les projets. MASCE et I-MINDS mettent l'accent sur l'apprentissage collaboratif et l'évaluation des apprenants. Cependant, I-MINDS offre en plus aux utilisateurs la possibilité de former un groupe. MASCE permet à un apprenant d'évaluer l'aide fournit par un autre apprenant et I-MINDS permet au tuteur de suivre les activités des apprenants et des groupes.

SIGFAD est un outil destiné à être couplé à des plateformes d'apprentissage. Ce système récolte les traces des apprenants et celles des groupes pour fournir à l'enseignant des appréciations sur le groupe, l'apprenant ainsi que l'activité. La plateforme SIGFAD ne traite pas l'apprentissage des agents et la perception et l'analyse des interactions dans les SMA pour arriver à des conclusions concernant des échecs pédagogiques. Aussi, elle n'utilise pas un langage de communication entre agents comme FIPA-ACL.

	Outils Collaboratifs	Notion de groupe	Méthode de formation de groupe	Paradigme objet ou agent	Architecture	Suivi de l'utilisateur
Baghera	Oui	Oui	Non	Agent	Client/serveur	Suivi minimal
SIGFAD	Oui	Oui	Non	Agent	Client/serveur	Récolte les traces des apprenants et fournit a l'enseignant des appréciations sur le groupe, l'apprenant ainsi que les activités
iPédagogique	Oui	Oui	Non	Agent	Client/serveur	Suivi minimal + assistance aux utilisateurs
MASCE	Oui	Oui	Non	Agent	Client/serveur	Pas de suivi mais propose une liste des meilleurs aidesurs
I-MINDS	Oui	Oui	Oui	Agent	Client/serveur	Oui

Formid	Oui	Oui	Non	Objet	Client/serveur	Environnement Formid-Suivi qui permet de suivre les activités des apprenants selon un scénario propose par l'enseignant
Refllet	Oui	Oui	Non	Objet	Client/serveur	Pas de collecte automatique de traces : les étudiants déclarent l'achèvement des activités et l'enseignant suit ainsi l'avancement de réalisation des taches
Croisières	Oui	Oui	Non	Objet	Client/serveur	Oui

Tableau III.3.3.6. : Comparaison des plateformes présentées

III.3.4. Les raisons du choix du paradigme d'agent

L'importance de la technologie agent est très évidente et pertinente en matière de développement logiciel lorsque des critères comme la distance, la coopération entre entités distinctes ou l'intégration de logiciels existants sont à prendre en compte. Plusieurs Domaines comme le domaine de l'enseignement à distance ne peuvent pas échapper à la technologie agent. En effet, ces domaines profitent des caractéristiques des systèmes Multi-Agents (SMA) qu'ils présentent, d'une part de permettre le partage ou la distribution de la connaissance , et d'autre part, la coopération d'un ensemble d'agents et la coordination de leurs actions dans un environnement pour effectuer un but commun. Donc dans l'environnement de formation à distance on cherche la résolution de problèmes et coopération des différents acteurs.

De façon plus spécifique, dans un environnement de formation à distance on cherche la mise en œuvre de système capable de susciter et de soutenir les interactions dans des groupes d'utilisateurs engagés dans une session de FAD. Des systèmes informatiques exigent plusieurs fonctionnalités, tel que les systèmes FAD, ces systèmes sont ouverts, complexes, distribués.

La caractéristique d'ouverture des systèmes suscite le changement dynamique de structure de ces systèmes. Il est difficile de connaître ses composants à l'avance, mais leurs changement est fait au cours de temps, et sont essentiellement hétérogènes (différentes personnes sont mis au point à différentes périodes et utilisant des outils et techniques différentes). Internet est un système ouvert le plus connu. La FAD s'appuie essentiellement sur Internet et exige l'utilisation de technologies appropriées ; donc pour un concepteur d'un système de formation à distance, la mise en place de systèmes ouverts est une grande exigence. En effet, Internet se présente comme une grande source d'informations distribuée; tout système informatique s'appuyant sur Internet doit tenir compte de la diversité des organisations et des outils de conception et de développement sans être constamment guidé par les utilisateurs [56]. Les systèmes sur Internet exigent des techniques de négociation et de

coopération qui sont des caractéristique relèvent du domaine des agents et des systèmes multi-agents.

Puisque la formation à distance se trouve au carrefour de plusieurs disciplines, d'une part, des sciences humaines (psychologie, sciences de l'éducation, sociologie, etc.) et d'autre part, des sciences de l'ingénieur (informatique, télécommunications); elle pose des problèmes d'une grande complexité, diversifiés, interdépendants et difficilement prévisibles.

La modularité et l'abstraction sont de bonnes solutions pour surmonter la *complexité* dans le développement logiciel. Le paradigme agent est considéré comme un bon outil de modularisation, il modélise le système de la FAD par des modules spécialisés. Ces modules spécialisés seront vus comme des agents logiciels.

" L'autonomie présente dans les agents fournit une abstraction comparable à celle offerte par les procédures (des programmes informatiques), les types de données abstraites et les objets. Cette autonomie permet au concepteur de voir son système informatique (support de la FAD) comme une collection d'entités-solutions autonomes et coopérantes. Cette vision de haut niveau est plus appropriée que toute autre alternative" [56].

Dans un environnement de formation à distance, l'apprenant va faire des interactions avec les ordinateurs. Malgré les progrès des IHM (interfaces homme-machine) au cours des deux dernières décennies et de la généralisation des interfaces multi-fenêtres, ces utilisateurs trouvent encore l'usage des logiciels difficile. L'ordinateur doit coopérer avec l'utilisateur pour atteindre ses objectifs. Dans la FAD les utilisateurs sont non seulement des informaticiens mais peuvent être non informaticiens (apprenants, formateurs, concepteurs de cours, tuteurs, administrateurs). Les objectifs de la FAD ne doivent pas se trouver contrariés par la nécessité de maîtriser les techniques informatiques ou la manipulation des ordinateurs et des logiciels.

III.4. Conclusion

Nous signalons, d'abord, l'environnement FAD ne veut dire non seulement pas la technologie, mais l'ensemble entier constitué par la machinerie, ses utilisateurs et son environnement. Deuxièmement, c'est la croyance de projet de base que l'on peut faire face à la complexité de l'apprenant seulement si la conception d'environnements E-Learning prend la collaboration entre des agents artificiels et humains comme un principe fondamental. Cela exige une approche de pluridisciplinaire forte à chaque étape de la conception et de la mise en œuvre. L'approche SMA est un outil fort pour pallier aux problèmes posés dans un environnement de formation à distance. Les caractéristiques des SMA rendent leur utilisation dans des systèmes complexes, ouverts, distribués, de large échelle et dynamiques très appropriée

Un système multi-agents est constitué d'un certain nombre d'agents autonomes travaillant ensemble pour résoudre un problème. Dans cette société d'agents, on peut avoir des conflits car les agents peuvent poursuivre ses propres buts sans communiquer avec les autres. Cependant, un agent tout seul ne peut pas résoudre un problème. Donc, les agents doivent interagir, coopérer, communiquer mais aussi coordonner ses actions avec les autres. En effet, la coordination est une notion clé pour les SMA. Elle permet aux agents de

bénéficier des capacités des autres agents et à la communauté d'éviter de dégénérer en une collection chaotique d'individus.

Dans ce chapitre, d'une part, nous avons mis l'accent sur l'usage de la sémantique dans l'environnement E-Learning, d'autre part nous avons montré l'efficacité de l'agent comme paradigme de modélisation des systèmes du E-Learning.

Dans le chapitre suivant nous allons présenter une conception d'un environnement éducatif sémantique à base d'agent. Le rôle de notre framework est de fournir l'opportunité aux enseignants de créer et d'annoter les documents pédagogiques, ces documents sont représentés sémantiquement pour faciliter la communication entre les différents acteurs de notre approche. Notre benchmark exploite ensuite ces documents annotés pour l'apprentissage personnalisé des apprenants hétérogènes.

Chapitre IV : Modélisation de l'approche

IV.1. Introduction

Aujourd'hui, le E-Learning doit être plus adaptable et flexible pour aider les apprenants à acquérir la connaissance. Typiquement, les systèmes d'apprentissage traditionnels ignorent les fonctions de personnalisation tels que la différence dans les styles d'apprentissage, les capacités, les niveaux de connaissance et les prérequis. Le fait qu'il existe un manque de connaissances sur chaque individu (Apprenant), le processus d'apprentissage n'est pas adapté aux besoins spécifiques des apprenants. Ainsi, ces systèmes utilisent le même processus, les mêmes styles et les mêmes documents pédagogiques pour tous les apprenants. Afin de concevoir un système éducatif pour l'apprentissage adaptatif, nous devons permettre la livraison de contenu d'apprentissage selon plusieurs critères tel que : le niveau d'intelligence, les préférences, les besoins de l'apprenant, et les styles préférés. En outre, les développements récents des technologies du web sémantique ont montré une tendance à l'utilisation d'ontologie de promouvoir le processus d'apprentissage adaptatif qui nous permet de créer des profils d'utilisateurs spécifiques et un modèle du contenu pédagogique.

Plusieurs travaux qui utilisent la technologie de web sémantique, l'annotation des contenus pédagogiques et les ontologies [1][2][3][13][7], ont indiqué que l'utilisation de ces technologies ont beaucoup de succès dans la conception pédagogique et le développement du contenu des cours. Elles fournissent un support d'aide pour les enseignants à créer le contenu pédagogique, à accéder facilement à ce contenu et à livrer les contenus de façon personnalisée et guidée (apprentissage personnalisé).

Dans notre travail, nous proposons une nouvelle approche pour le développement de systèmes d'E-Learning personnalisé. **La première partie** de notre système a pour objectif de développer les contenus pédagogiques en utilisant les ontologies et l'annotation. **La deuxième partie** exploite les documents pédagogiques annotés pour offrir un apprentissage personnalisé. Un apprentissage personnalisé tient en compte les capacités, les préférences et les besoins des apprenants et les différents styles d'apprentissage, et qui représente les différentes activités qui devraient être faites par un apprenant particulier. La force de cette étude est de créer le modèle de l'utilisateur ontologique, modèle de contenu et le modèle pédagogique séparément pour accroître la flexibilité et la réutilisabilité du système.

Notre système est une plateforme de formation à distance qui est modélisée en utilisant un nombre d'agents afin de profiter des points fort de ce paradigme tel que modularité, autonomie ... etc [121].

Ce chapitre est dédié à présenter une telle approche (système multiagent éducatif). Il commence par les fondements théoriques utilisés, ainsi qu'une description de haut niveau de l'architecture multiagents pour la formation à distance. Le point suivant est consacré à la description détaillée de notre système en expliquant les structures et les fonctionnalités des agents utilisés dans l'architecture.

IV.2. Fondements théoriques

Pour atteindre notre principal objectif on doit utiliser plusieurs techniques, approches, idées, et stratégies. Les principaux points dans notre système pour réaliser nos objectifs sont les suivants :

IV.2.1. Web sémantique et E-Learning

L'utilisation du Web sémantique dans une architecture de E-Learning a plusieurs avantages et propriétés. Il offre un sens partagé commun et des métadonnées traitables par les machines. Notre architecture du système E-Learning contient une communauté d'agents qui travaillent en collaboration pour construire le contenu pédagogique et adapter ce contenu et stratégies pour l'apprenant. Le Web sémantique offre aux agents une puissance pour satisfaire les exigences du système du E-Learning : rapide, juste à temps et apprentissage pertinent que nous avons illustrés en introduction. Aussi, le contenu pédagogique est sémantiquement annoté ; cette annotation facilite la recherche de contenu adéquat dans chaque cas et la combinaison des nouveaux cours d'apprentissage. En utilisant les préférences et les caractéristiques de l'apprenant, nous pouvons facilement trouver le contenu d'apprentissage utile. Le processus est basé sur les requêtes Web Sémantique et la navigation à travers le contenu d'apprentissage activé par un background ontologique.

De ce fait, le Web Sémantique peut être traité comme une plateforme adéquate pour implémenter un système E-Learning, du moment qu'il fournit tous les moyens pour le développement d'une ontologie (d'apprentissage), l'annotation basée ontologie du contenu d'apprentissage, leur composition dans des cours et la livraison active des cours à travers des portails d'apprentissage. L'utilisation de Web sémantique dans notre système de E-Learning est de créer le modèle d'apprenant ontologique, modèle des styles d'apprentissage et le modèle de contenu éducatif pour accroître la flexibilité et la réutilisabilité du système.

IV.2.2. Apprentissage personnalisé

Les apprenants qui participent dans un environnement éducatif sont généralement hétérogènes c'est à dire ils ont différentes préférences, différent background, différents objectifs et différent style d'apprentissages...etc. Si on utilise la même séquence des parties de cours pour tous les apprenants, alors la qualité et le niveau de compréhension seront très faibles. Ainsi, ces apprenants ne peuvent jamais apprendre en utilisant le même contenu pédagogique et le même style d'apprentissage. Donc, l'une des exigences des systèmes d'apprentissage à distance est l'apprentissage personnalisé. Le principe de l'apprentissage personnalisé est de livrer pour chaque apprenant le document pertinent en fonction de ses préférences, besoins, objectifs et ses caractéristiques (intellectuelles, personnelles, sociales ...etc). Aussi, nous pouvons personnaliser le style d'apprentissage pour chaque apprenant. Nous détectons le style adéquat pour l'apprenant à travers ses comportements et les tests. Les stratégies (styles) d'apprentissages sont considérées comme un point crucial dans un système du E-Learning. Nous avons choisi l'utilisation de l'adaptation de contenu éducatif et l'adaptation de styles d'apprentissage.

IV.2.2.1. Adaptation de contenu éducatif

Dans une formation présentielle, l'enseignant explique le cours en utilisant plusieurs exemples, détails et manières d'explications selon le niveau des étudiants. Par exemple dans le cas des étudiants de première année, il faut donner plusieurs définitions de nouveaux concepts. Mais dans la deuxième année on suppose que tous les concepts donnés dans la première année sont connus. A titre d'exemple dans une formation d'Anglais, l'enseignant donne plusieurs explications pour chaque mot pour les étudiants de première année; mais pour les étudiants des niveaux évolués l'enseignant suppose qu'ils connaissent les plupart des mots simple d'anglais. Même si les apprenants sont de même niveau, l'enseignant explique le cours d'une manière et change l'explication ou ajoute des informations additionnelles en cas de question ou lorsqu'il connaît que les apprenants n'ont pas bien compris le cours (à partir de tests par exemple).

Dans notre système nous utilisons l'adaptation de cours pour fournir une plateforme qui répond aux besoins de plusieurs niveaux des apprenants (faible, moyen, fort...etc.), le système fournit aussi l'adaptation de cours pour les apprenants de même niveau. Nous adaptons le contenu selon les résultats des tests pour chaque apprenant. L'adaptation de contenu éducatif dans notre système est la recherche des documents de référence (RefDoc) pertinents pour chaque apprenant.

IV.2.2.2. Styles d'apprentissage

Le domaine de styles d'apprentissage est complexe et affecté par plusieurs aspects conduisant à différents concepts et points de vue. Le style d'apprentissage peut être défini comme : une description des attitudes et des comportements qui déterminent la manière d'apprentissage préférée pour une personne. Felder [120] a défini le style d'apprentissage : "les forces de caractéristiques et préférences dans la manière dont les apprenants prennent les informations et le processus d'apprentissage".

Une autre définition [11] est : la manière complexe et les conditions dans lesquelles, les apprenants perçoivent, traitent et stockent le plus efficacement possible, et rappellent ce qu'ils tentent d'apprendre.

Un style d'apprentissage est la meilleure façon de traiter personnellement avec les informations et l'expérience. Il décrit un apprenant en termes de conditions d'enseignement qui sont les plus susceptibles de favoriser son apprentissage, l'apprenant se distingue par son style d'apprentissage ; cela signifie que certaines approches pédagogiques sont plus efficaces pour lui que d'autres.

En se basant sur les modèles d'apprentissage présentés dans le chapitre 1, nous avons proposé d'utiliser un modèle des styles d'apprentissage basé sur cinq dimensions :

IV.2.2.2.1. Perception

A travers quelle modalité sensorielle l'information externe est-elle efficacement perçue. Nous avons prévu un cadre de référence pour exprimer les modalités de perception (auditive, visuelle, tactile). Ce cadre décrit les modalités de présentation et perception préférées par l'apprenant pour recevoir l'information. Selon cette dimension on a trois types d'apprenants :

- a- Auditifs** : les apprenants auditifs apprennent le mieux en écoutant. Ils réagissent bien aux discussions et aux autres situations qui donnent l'occasion de discuter et d'écouter.
- b- Visuels** : comprennent mieux l'information en la visualisant sous forme du texte (image, vidéo, diagramme, ou graphiques...etc).
- c- Tactiles** : ces apprenants apprennent en bougeant et en touchant. Les situations d'apprentissage pratiques leur conviennent le mieux. Ils aiment bouger lorsqu'ils apprennent de nouvelle chose.

IV.2.2.2.2. Participation

Le degré de participation de l'apprenant dans le groupe d'apprentissage. Nous distinguons deux styles d'apprenant :

- a- Actifs** : les apprenants actifs sont tendance à aimer d'avantage le travail d'équipe, pour eux il est nécessaire de travailler avec les autres apprenants. Voici des propositions pour aider les apprenants actifs :
 - Inciter les apprenants à utiliser des outils de communication (courriels, forum de discussion, chats).
 - Inciter les apprenants à former des groupes pour dresser une ébauche de questions et pour répondre aux questions d'examen.
 - Discuter, partager des connaissances, travailler ensemble sur un projet commun.
- b- Réfléctifs** : ce genre des apprenants préfère à travailler seul et réfléchit silencieusement à l'information d'abord.

IV.2.2.2.3. Pratique

Est-ce que l'apprenant apprend mieux en utilisant les activités pratiques ou théoriques :

- a- Détectifs** : ce sont des apprenants pratiques, orientés vers les faits et les procédures.
- b- Intuitifs**: les apprenants conceptuels, innovantes, orientés vers des théories et des significations.

IV.2.2.2.4. Psychologique

Dans cette dimension, nous pouvons connaître : comment l'apprenant réfléchit pour résoudre les problèmes ; Est-ce que l'apprenant commence directement la résolution des problèmes ou il réfléchit avant démarrer le processus de résolution :

- a- Global** : ces apprenants préfèrent d'apprendre le tous puis ils passent vers les détails ; c'est à dire : holistiques, systèmes penseurs, d'apprendre dans de grands bonds.
- b- Séquentiel**: ces apprenants sont linéaires, ordonnés. Ils apprennent à petits pas progressifs.

IV.2.2.2.5. Sociologique

Cette dimension classe les apprenants selon leur comportement sociologique avec les autres membres de groupe d'apprentissage. Donc, elle mesure la nature de l'interaction de l'apprenant avec les autres:

- a- Collaboratifs :** les apprenants de collaboration sont plus motivés lorsque le travail en mode collaboratif avec les autres. Ils sont caractérisés comme des apprenants qui sont coopérative, aiment travailler avec les autres.
- b- Compétitifs:** ces apprenants sont plus motivés quand ils voient les autres comme des concurrents. Ils ont la motivation pour faire mieux que les autres, le plaisir de la compétition, et de voir le processus d'apprentissage comme une situation gagnant-perdant.

IV.2.2.2.6. Responsabilité

Pour déterminer le style de l'apprenant dans cette dimension, nous devons répondre à la question suivante : peuvent-ils assumer la responsabilité individuelle ? Donc, on le définit à l'égard des enseignants et à quel point les étudiants ont le désir de liberté et de contrôle dans l'environnement d'apprentissage :

- a- Dépendants :** les étudiants dépendants considèrent l'enseignant comme la source d'information et la structure. Ils veulent avoir les ordres pour faire les activités (par les autorités) et apprendre seulement ce qui est nécessaire.
- b- Indépendants :** apprenants indépendants sont caractérisés comme des apprenants confiants et curieux. Ils préfèrent penser pour eux-mêmes et de travailler sur leur propre.

IV.2.3. Les questions

Après chaque partie de cours une session des questions va être ouverte pour recevoir les questions des apprenants. Donc les apprenants reçoivent les réponses de ses questions afin d'éclaircir les points vagues pour eux. Aussi, à partir d'une question, l'enseignant peut avoir des idées sur le niveau ou les points non évidents pour un apprenant. Donc l'enseignant essaie de donner une référence à utiliser ou un conseil ou lancer une adaptation appropriée. La base de données question contient toutes les questions posées par les apprenants et leurs réponses pour les réutiliser (on cherche est ce que la question a été posée précédemment pour afficher la réponse donnée par l'enseignant).

IV.2.4. Les tests

Les tests sont considérés comme un outil de classer les étudiants par niveaux et on peut connaître le degré de compréhension de cours c'est à dire on peut tester est ce que les apprenants ont bien compris le cours ou non. Quand, l'apprenant échoue dans le test d'une partie de cours il faut lancer la procédure d'adaptation pour motiver l'apprenant à faire la formation encore et qui va l'aider à mieux comprendre le cours (référence, changer le style d'apprentissage...etc).

IV.2.5. Le profil

Le profil contient les informations de l'apprenant et l'enseignant. Pour les apprenants, on utilise le profil pour satisfaire les besoins et les préférences des apprenants (style préféré, couleur, langue, ...etc). Donc, il donne un vue global sur les préférences de l'apprenant et l'état d'avancement des apprenants. L'apprenant va fournir quelques informations du profil et le système va détecter et changer les autres informations automatiquement.

IV.3. Architecture générale du système

L'objectif de concevoir tout logiciel informatique est pour satisfaire un certain nombre de besoins des utilisateurs. En ce qui nous concerne, nous exprimons ces besoins en termes de services offerts aux trois acteurs potentiels de l'activité d'enseignement/apprentissage à savoir : les enseignants qui sont les auteurs des contenus éducatifs, les apprenants et les administrateurs. En vue de satisfaire les besoins de ces acteurs, nous proposons une architecture de notre plateforme pour répondre au besoin de travail collaboratif à distance.

D'un côté, notre plateforme est conçue pour répondre aux nouveaux besoins d'enseignants en termes d'assistance des travaux collaboratifs et à distance en donnant l'opportunité à un groupe d'auteurs pour la construction et l'annotation d'un contenu éducatif qui reflètent leurs points de vue et qui enrichit la connaissance sur ce contenu éducatif. D'un autre côté, notre système éducatif et adaptatif offre l'opportunité à adapter les cours et les stratégies pour assurer un apprentissage personnalisé pour chaque apprenant. Nous proposons quelques solutions techniques concernant les besoins d'enseignants et d'apprenants.

Donc, nous nous intéressons par le groupe des enseignants qui participent à la création et l'annotation du contenu éducatif et qui assistent le processus d'apprentissages des apprenants. Aussi, on s'intéresse aux apprenants qui sont l'acteur principal concerné par le processus d'apprentissage. Le contenu pédagogique est considéré comme une partie très importante dans notre travail, on représente ce contenu en utilisant l'aspect sémantique (ontologies et leurs relations) pour faciliter l'utilisation du contenu éducatif par la machine.

La figure suivante présente l'architecture de notre système qui répond aux besoins d'enseignants engagés dans une plateforme d'enseignement à distance pour accomplir leurs tâches pédagogiques en terme de création des contenus de la matière à enseigner puis les annotés d'une manière collaborative et à distance.

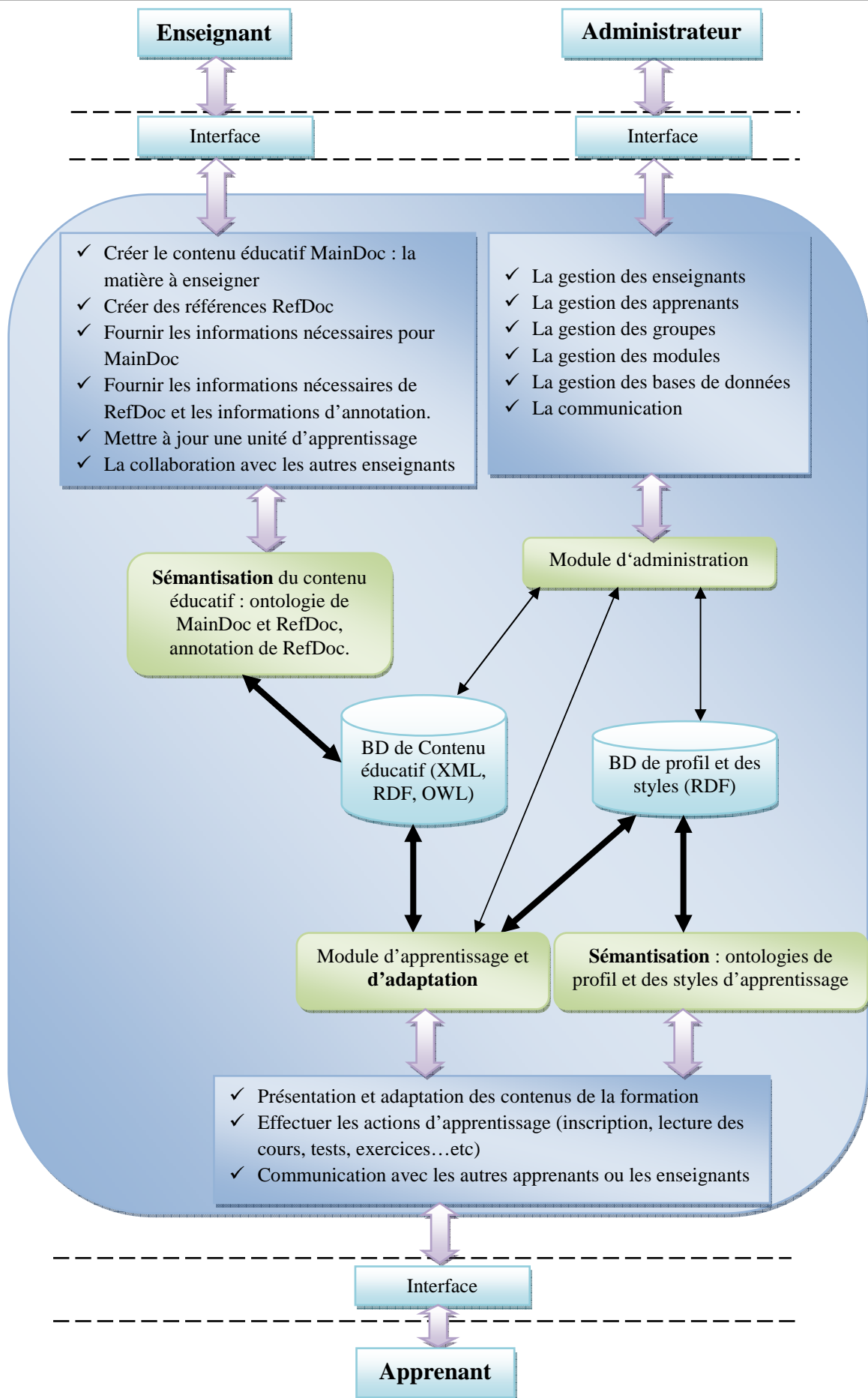


Figure IV.3 : Architecture générale du système

IV.3.1. Identification des acteurs du système

Notre système contient trois acteurs qui agissent dans une plateforme d'enseignement à distance : l'enseignant, l'apprenant et l'administrateur. Afin de concevoir le cadre générale de notre travail, nous déterminons les principaux rôles pour chaque acteur dans le système. Les deux parties essentielles dans notre travail sont :

- 1) la construction collaborative d'un contenu éducatif en utilisant les ontologies de domaine et les annotations (métadonnée).
- 2) guider l'apprenant durant son processus d'apprentissage en utilisant plusieurs techniques : le profil, les besoins, adaptation de contenu, adaptation des stratégies ...etc.

IV.3.1.1. Enseignant

C'est un important point dans notre système qui joue la principale mission dans le processus d'enseignement/apprentissage, ces missions s'articulent principalement sur : l'élaboration et l'indexation du contenu de la matière à enseigner adapter au profil d'apprenant.

L'opération de transmission de savoir et de savoir-faire depuis l'auteur vers un apprenant ne nécessite pas le contenu de la matière à enseigner seulement, mais aussi un ensemble d'information concernant la matière à enseigner (métadonnées) pour assurer le bon déroulement d'une session d'apprentissage d'un côté, et l'adaptabilité de la formation par rapport au profil de l'apprenant d'un autre côté. Pour cela l'enseignant utilise un standard de description du contenu pédagogique qui spécifie la syntaxe et la sémantique des métadonnées décrivant des ressources pédagogiques. L'ensemble des enseignants doivent communiquer entre eux pour accomplir leurs tâches communes.

IV.3.1.2. Apprenant

La mission principale d'un apprenant dans un contexte éducatif est l'acquisition d'un savoir et du savoir-faire devant l'enseignant pour former puis enrichir ces connaissances personnelles. Chaque apprenant doit bénéficier d'une formation personnalisée en fonction de ses caractéristiques particulières. Donc le système fournit une séquence de cours pour chaque apprenant selon plusieurs critères, et change la stratégie d'apprentissage appropriée pour chaque cas.

IV.3.1.2.1. Modèle d'apprenant

Le système utilise ce modèle de l'apprenant pour s'adapter selon les informations sauvegardées dans le profil. A partir de ce modèle le système peut avoir des connaissances sur l'état d'avancement de l'apprenant et tout ce qu'il préfère (caractéristiques et préférences de l'apprenant, style d'apprentissage préféré...etc.). La connaissance du profil est très nécessaire dans notre système parce que nous nous basons sur ses informations. Ce profil contient toutes les informations qui concernent l'apprenant telles que :

- l'état de l'avancement;
- les informations qui identifient l'apprenant (le nom, l'identificateur, le mot de passe)

- niveau de connaissance de l'apprenant sur les cours;
- la formation choisie (s'il existe plusieurs formations);
- les langues maîtrisées et préférées;
- les résultats des tests
- les caractéristiques et préférences de l'apprenant (couleur préférés, style d'apprentissage ... etc.)
- d'autres informations

La modélisation des connaissances de l'apprenant sur le domaine suit le paradigme de représentation le plus fréquent " le modèle overlay"

a. Modèle overlay

Le principe du modèle est d'associer à chaque concept du cours une valeur estimative de la connaissance de l'apprenant sur ce cours. Cette estimation peut être une valeur binaire (0,1) ou mesure qualitative (moyen, bien, très bien, ... etc.). Si la valeur est bien alors on écrit (C1, bien) c'est à dire la connaissance de l'apprenant sur le cours C1 est bien [53].

Dans notre cas, les valeurs utilisées sont des valeurs numériques et variables (1, 2, 3, 4, ...etc.); ces valeurs sont définies par l'administrateur lorsqu'il crée la formation. Dans notre système on utilise les concepts, donc on a (concept de cours, valeur). Le concept est une partie élémentaire de cours qui doit avoir un objectif élémentaire et bien déterminé.

Exemple : (c1, 0): le niveau de l'apprenant est 0 (valeur initiale) sur le premier concept (1) de premier cours (1). Au premier lieu tous les apprenants ont la valeur initiale 0 pour tous les cours des modules

Remarque :

Pour chaque cours, on doit définir un ensemble des "prérequis" qui sont les concepts nécessaires avant de commencer la formation d'un cours. Par exemple : dans un cours sur les sémaphores, l'apprenant doit avoir des connaissances sur le problème d'interblocage (le problème résolu par le sémaphore).

Donc l'apprenant doit avoir une connaissance sur les prérequis puis il commence le cours (les prérequis sont nécessaires pour apprendre le cours) c'est à dire le système éducatif présente aux apprenants les prérequis de cours et il prépare des tests concernant ces prérequis. Si l'apprenant réussit dans le test il passe au cours. En cas où les prérequis sont déjà connus par l'apprenant, alors il passe directement au cours.

b. Ontologie de profil

Dans les systèmes d'apprentissage adaptatif «E-learning qui se base sur l'adaptation », plusieurs problèmes techniques sont posés. Tout d'abord, c'est la question de savoir comment créer des profils d'utilisateurs précis et complets pour les apprenants et comment ils peuvent être utilisés pour reconnaître les apprenants et décrire ses comportements. Aussi, il y a la question de la façon de construire les relations hiérarchiques et de navigation entre les différentes parties du matériel d'apprentissage et comment celles-ci peuvent être déterminées, en fonction du profil de l'utilisateur, par la reconnaissance de la

prochaine étape de l'apprentissage dans le processus d'apprentissage. Un remède possible à ce problème est la conception de profil de l'utilisateur ontologique et du contenu d'apprentissage. La hiérarchie de classe de l'ontologie profil de l'apprenant est représentée sur la figure suivante.

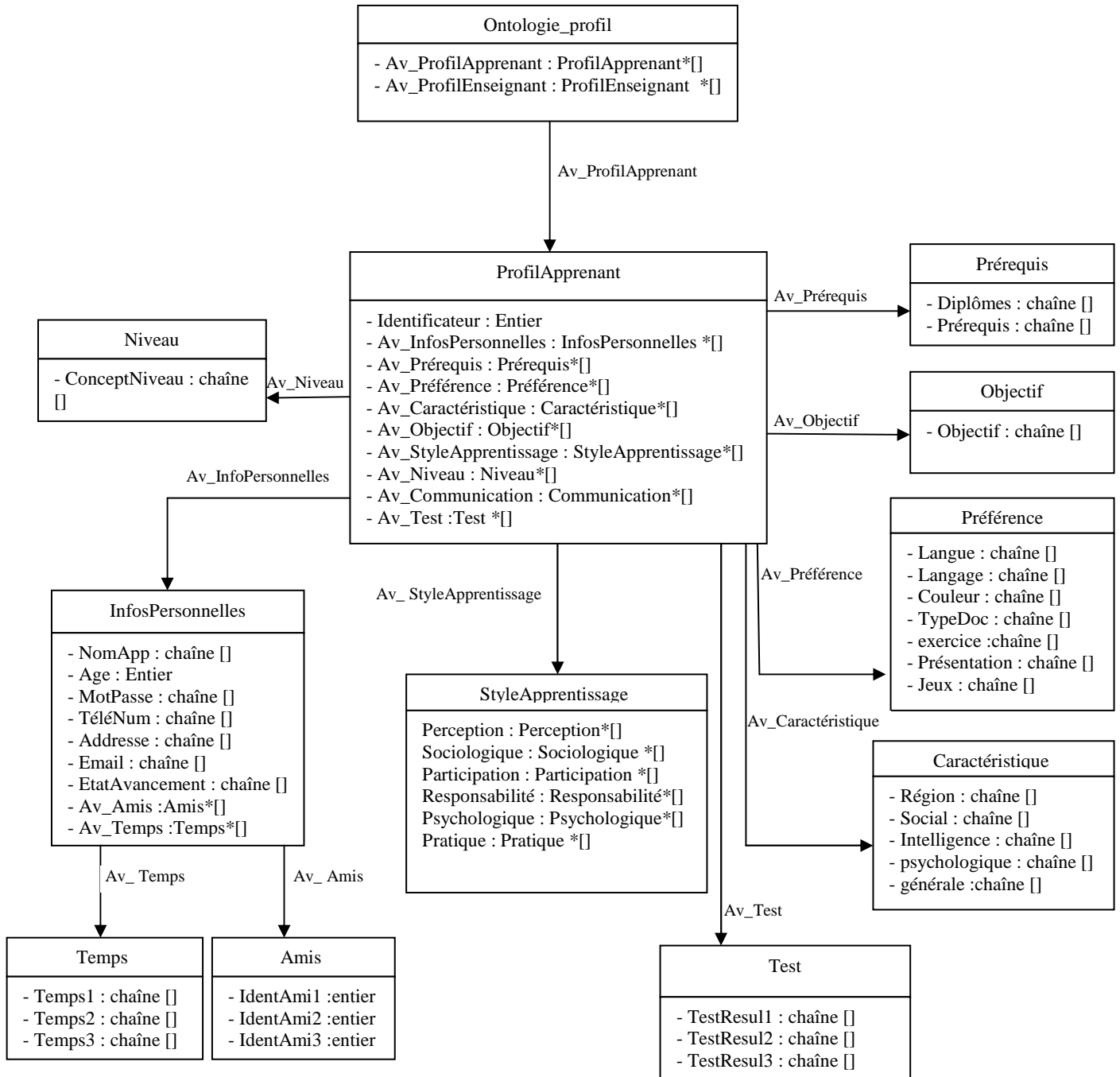


Figure IV.3.1.2.a : Ontologie de profil d'apprenant

L'ontologie de profil contient toutes les informations nécessaires pour avoir un vue précise sur l'apprenant et ses préférences et caractéristiques (quelques information sont fournies par l'apprenant d'autre informations par notre systèmes selon le comportement des apprenants). Cette ontologie nous aide à déterminer le cours le plus approprié pour l'apprenant et à satisfaire ses préférences :

- La propriété *Identificateur* : est un nombre entier considéré comme une clé pour chaque apprenant.
- La classe *Prérequis* : représente le background (connaissances acquises) de l'apprenant et les connaissances à apprendre avant commencer l'apprentissage de cours.
- *Niveau* : est une classe qui représente les résultats de l'apprenant pour chaque concept de cours en utilisant le model Overlay. Cette classe représente le relevé des notes de l'apprenant.
- La classe *InfosPersonnelles* : détermine toutes les informations personnelles de l'apprenant (nom, prénom, age, numéro de téléphone, email, adresse, état d'avancement,...etc). Cette classe contient deux autres sous classes : 1) la classe *Temps* : qui contient les temps de disponibilité de chaque apprenant (de connexion) ; 2) la classe *Amis* : cette classe présente les indicateurs de tous les amis (très proches) de l'apprenant (utilisé dans le cas de collaboration).
- *Préférence* : exprime toutes les préférences de l'apprenant tel que : (couleur, langue, langage, préférer les exercices ou non, préférer les jeux ou non ...etc).
- *Caractéristique* : qui contient les caractéristiques sociales, psychologique, la région ...etc).
- *Test* : contient tous les tests et leurs résultats.
- *StyleApprentissage* : Cette classe contient plusieurs sous classes qui présentent les stratégies ou les styles préférés pour chaque apprenant (on les utilise pour l'adaptation de stratégies).

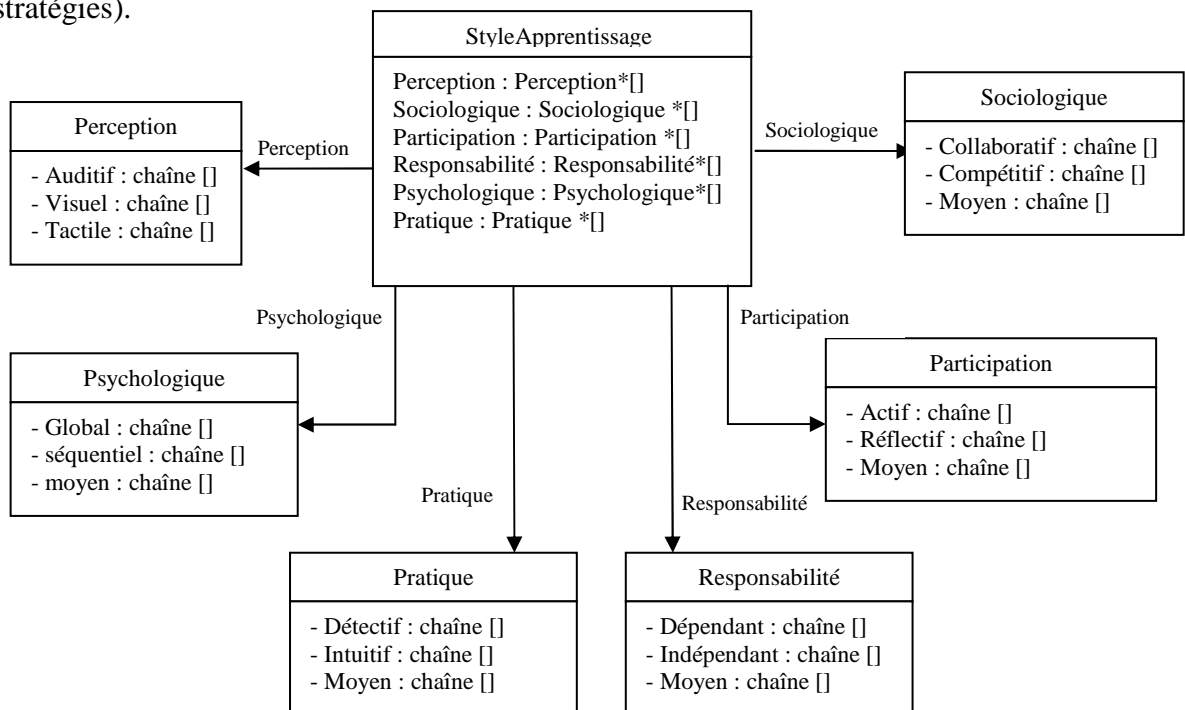


Figure IV.3.1.2.b : Ontologie de styles d'apprentissage

Cette ontologie représente les degrés de styles préférés (Oui, Moyen, Non) pour chaque apprenant. Nous utilisons cette ontologie pour avoir une vue sur les styles de l'apprenant et pour effectuer l'adaptation pertinente dans chaque situation. Ces styles sont présentés et expliqués dans la section IV.2.2.2.

IV.3.1.3. Administrateur

L'administrateur d'une institution de formation assure la gestion de la formation, gestion d'auteurs, gestion d'apprenants, gestion des groupes, ...etc. D'un point de vue administratif et juridique, l'administrateur a besoin d'informer les auteurs et les apprenants engagés dans une formation sur leurs nouveautés administratives.

IV.3.1.4. Contenu éducatif

Cet élément fait l'objet de notre travail, on construit le contenu éducatif on utilisant la technologie du Web sémantique afin d'organiser ce contenu et de faciliter la recherche de document pertinent. Un groupe d'enseignants essayent de rédiger et annoter le contenu pédagogique de façon collaborative, et former les relations sémantiques entre ces documents. La deuxième étape est d'exploiter ce contenu annoté sémantiquement pour guider l'apprenant à mieux comprendre l'objet éducatif en utilisant les techniques d'adaptation (de contenu, et de stratégies).

La quantité énorme de documents éducatifs existe dans un environnement d'enseignement à distance, généralement, désoriente les apprenants et n'aide pas à choisir le document pertinent pour chaque apprenant. Pour résoudre ce problème, nous profitons de la technologie de web sémantique (ontologies, métadonnées...etc) pour ajouter la sémantique aux contenus éducatifs afin de faciliter la manipulation de ces contenus par la machine. Cependant, ce processus exige plusieurs compétence et techniques pour satisfaire les besoins des apprenants qui sont de nature hétérogène et ont des caractéristiques et des capacités intellectuelles différentes.

Les enseignants créent, de manière collaborative, le contenu pédagogique. Nous proposons deux types de documents éducatifs ; chaque type de document a des caractéristiques et des rôles bien déterminés:

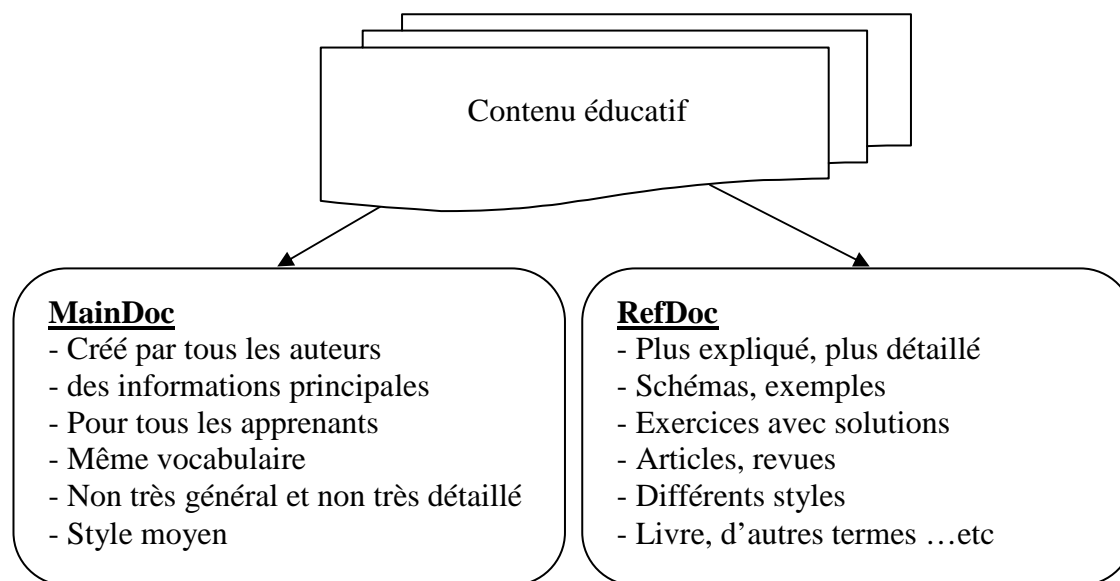


Figure IV.3.1.4.a : le contenu éducatif

Le contenu éducatif est un élément critique dans un environnement de E-Learning puisque les apprenants vont apprendre et manipuler ce contenu éducatif. Aussi, à travers le contenu éducatif, les enseignants peuvent transférer leurs connaissances aux apprenants et le degré de compréhension dépend directement à la qualité du contenu éducatif et les techniques utilisées pour l'apprentissage. Dans notre système, nous créons, organisons et annotons ce contenu pour atteindre le principal objectif qui est un contenu éducatif de haute qualité et adaptatif pour satisfaire les besoins des apprenants hétérogènes.

Pour cela, la première étape est de différencier deux types de contenu éducatif. Chaque type de contenu a un rôle bien déterminé et des caractéristiques particulières :

- **MainDoc** : est le document principal (comme le cours donné par l'enseignant dans l'apprentissage classique ou présentiel). Ce contenu pédagogique est créé par tous les auteurs (enseignants de même cours) de manière collaborative. Il contient les informations nécessaires pour comprendre les concepts de contenu éducatif. Tous les apprenants doivent avoir et lire ce type de contenu.
- **RefDoc** : est le document de référence utilisé dans des cas particuliers (apprenant échouant, excellent, ou une requête de recherche d'une référence). Les enseignants fournissent un ensemble de références (contenu additionnel) d'apprentissage pour plus d'explications, plus de détails, exercices avec solutions, livre, thèse, schémas explicatifs, explication en utilisant d'autres styles et des termes différents ...etc.

Donc, nous avons deux types de documents: MainDoc et Refdoc, ceci est le premier niveau d'annotation. Le processus de la création et de l'annotation du contenu d'apprentissage comporte trois étapes principales:

La couche 1 : Définir le domaine d'apprentissage
La couche 2 : Définir les sous domaines et le vocabulaire, créer MainDoc
La couche 3 : Création et annotation des références RefDoc et l'ajout des relations entre les Docs Pédagogiques

Figure IV.3.1.4.b : les étapes de construction du contenu éducatif

IV.3.1.4.1. La couche 1 : Préparation de domaine d'apprentissage

Dans cette étape, les membres de l'administration définissent le domaine (contexte d'apprentissage) du processus d'apprentissage ex: informatique, apprentissage de l'anglais, du droit, de l'agriculture ... etc. De plus, notre système offre aux enseignants une forme d'inscription qui contient les informations sur les enseignants (nom, âge, diplômes, profession actuelle, l'expérience ... etc.). L'administrateur accepte ou refuse les inscriptions des enseignants. Dans le cas de l'apprentissage académique, le groupe d'auteurs est l'ensemble des enseignants de l'association (université, école, entreprise, société,...etc). Si ce n'est pas la formation académique, l'administrateur invite les enseignants à participer au processus d'apprentissage.

IV.3.1.4.2. La couche 2 : préparation du MainDoc et le vocabulaire

Dans cette étape, les enseignants décomposent le domaine en sous-domaines, par exemple: le «domaine» est l'Informatique, les sous-domaines sont (comme les modules dans l'université): système d'exploitation, l'intelligence artificielle, les langages de programmation ... etc. Ensuite, chaque enseignant choisit un sous-domaine à participer, ici nous formons les groupes des enseignants (pour chaque sous-domaine, nous avons un groupe d'enseignants). Notre système affiche à chaque enseignant un formulaire pour choisir son sous-domaine et compléter les informations importantes (son expérience dans le sous-domaine, le temps de disponibilité,...etc). Chaque enseignant doit choisir, parmi les enseignants du groupe, le chef du groupe. L'administrateur informe le groupe par le chef qui a été choisi par la majorité des enseignants.

Dans ce niveau, les enseignants du groupe de sous-domaine qui créent le contenu principal de leur sous-domaine (dite MainDoc). Ce MainDoc doit être créé par tous les enseignants du groupe et ils doivent utiliser le même vocabulaire de sous-domaine. Le vocabulaire de sous-domaine est créé par les enseignants avec l'aide de notre système. Ce vocabulaire est une ontologie de domaine qui contient tous les concepts à apprendre dans le sous domaine et les relations entre eux.

Nous représentons le vocabulaire de sous-domaine en utilisant XML et OWL (protégé 2000) pour simplifier le traitement du contenu (vérifier le vocabulaire utilisé).

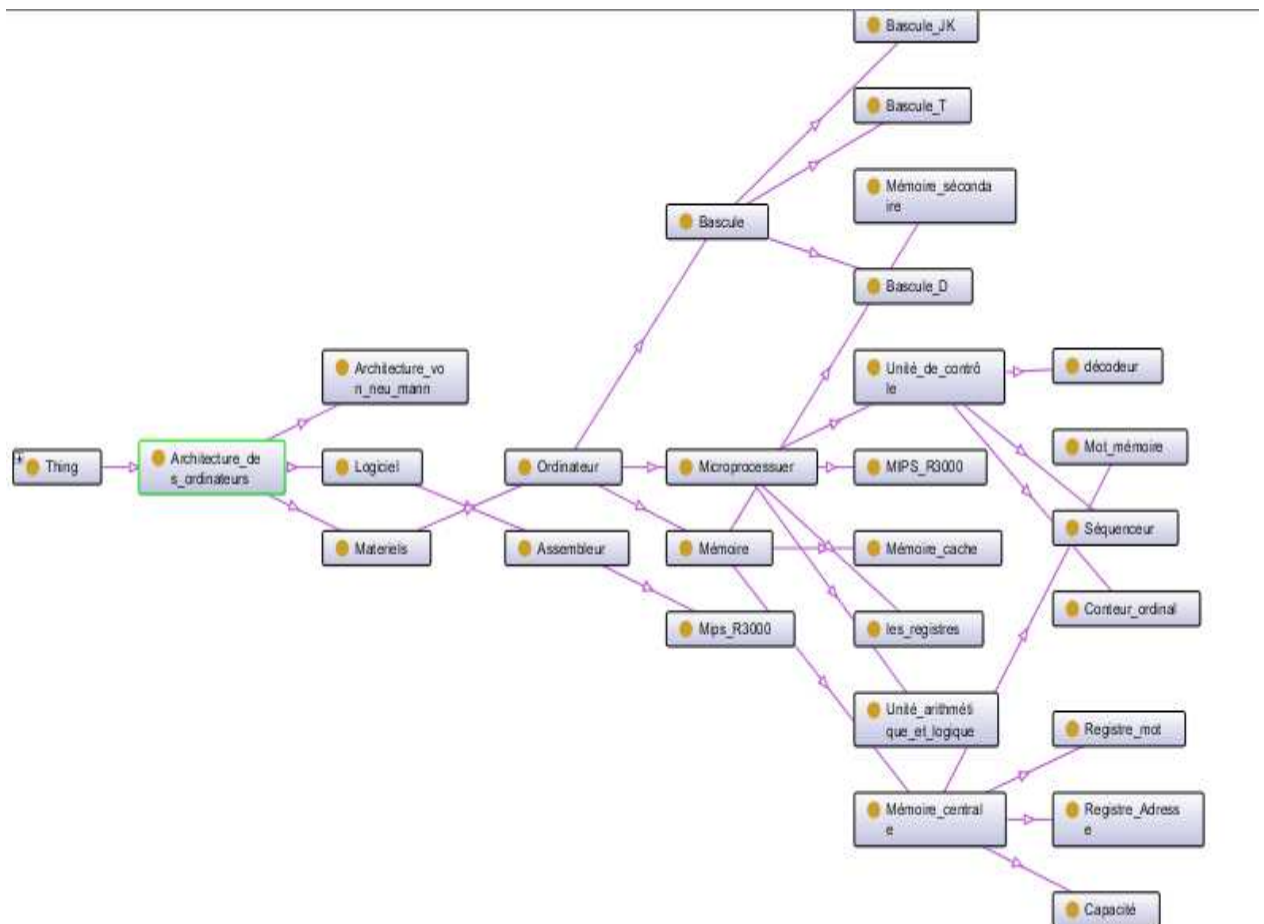


Figure IV.3.1.4.2.a : l'ontologie de domaine (vocabulaire)

```
<?xml version = "1.0" encoding="UTF-8" standalone="yes" ?>
  <content>
    <Computer_architechture>
      <part1 type= "Hardware">
        <course1>microprocessor</course1>
        <course2>memory</course2>
        <course3>bascule</course3>
      </part1>
      <part2 type= "software">
        <course1> assembly</cours1>
        <course2> Misp_r3000</cours2>
        <course3>assembly_mips_r3000</cours3>
      </part2>
    </Computer_architechture>
  </content>
```

MainDoc utilise le vocabulaire de sous-domaine afin d'unifier ses termes, chaque auteur doit suivre le même vocabulaire. Ce sera assuré par la vérification du MainDoc avec l'annotation du Refdoc. Le chef de groupe met le premier contenu, puis les enseignants peuvent modifier et mettre à jour ce contenu de manière collaborative pour avoir un contenu pertinent pour la majorité des apprenants. Le MainDoc ne doit pas être très générale et ne doit

pas contenir beaucoup de détails, donc le MainDoc contient des informations nécessaires pour comprendre le sous-domaine.

Le MainDoc est un ensemble des concepts doit être lit par l'apprenant pour considérer qu'un certain objectif d'apprentissage est atteint. La structure générale du MainDoc est composée de multiples couches reliées entre elles:

- **Module (sous-domaine)** : le domaine est un ensemble des modules (sous-domaine)
- **Chapitre** : l'ensemble des chapitres forme un module
- **Cours** : un chapitre est un ensemble des cours bien définis
- **Objet pédagogique**: l'ensemble des objets pédagogiques comportent un cours. Ces OP peuvent être : prérequis, familiarisation ou renforcement.
- **Concept** : chaque objet pédagogique contient un nombre bien déterminé des concepts.
- **Fragment (unité de base)** : on va partitionner le concept à des unités de base qui sont des parties élémentaires comportent le cours (ne sont pas vastes ou généraux). Chaque fragment peut être : une définition, un test, un exercice, un exemple, schéma, quiz ...etc.

La figure suivante présente l'organisation d'un document pédagogique traité dans notre système adaptatif.

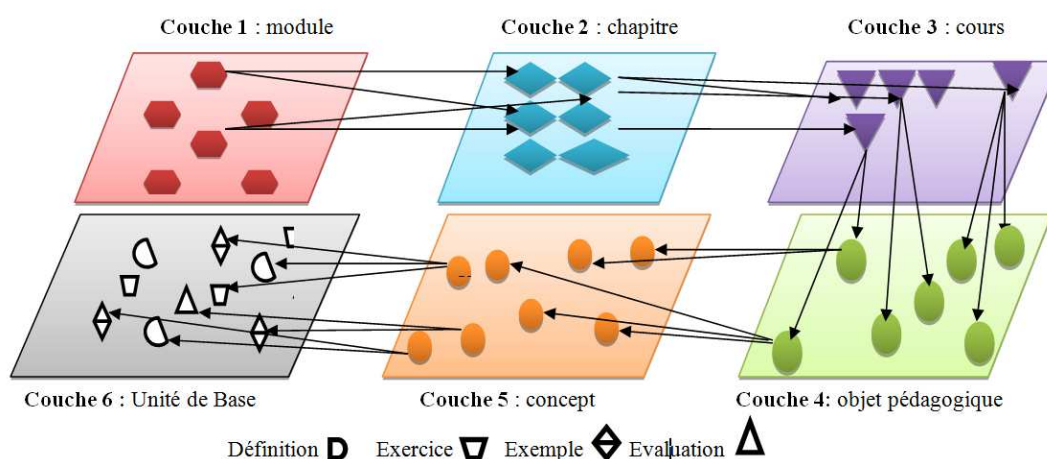


Figure IV.3.1.4.2.b : Organisation de MainDoc (Module)

Le MainDoc est le principal cours dans notre processus d'apprentissage, il est constitué d'un ensemble de concepts. Pour mieux représenter les parties de MainDoc et ces concepts, nous formons l'ontologie du contenu éducatif et les relations entre ces éléments (on commence par l'ontologie de MainDoc). Cette ontologie de MainDoc est définie afin de déterminer une description générale de chaque cours et ses concepts. Donc, nous n'utilisons pas cette ontologie pour la recherche ou l'adaptation (contrairement au RefDoc). Les apprenants doivent avoir le même MainDoc.

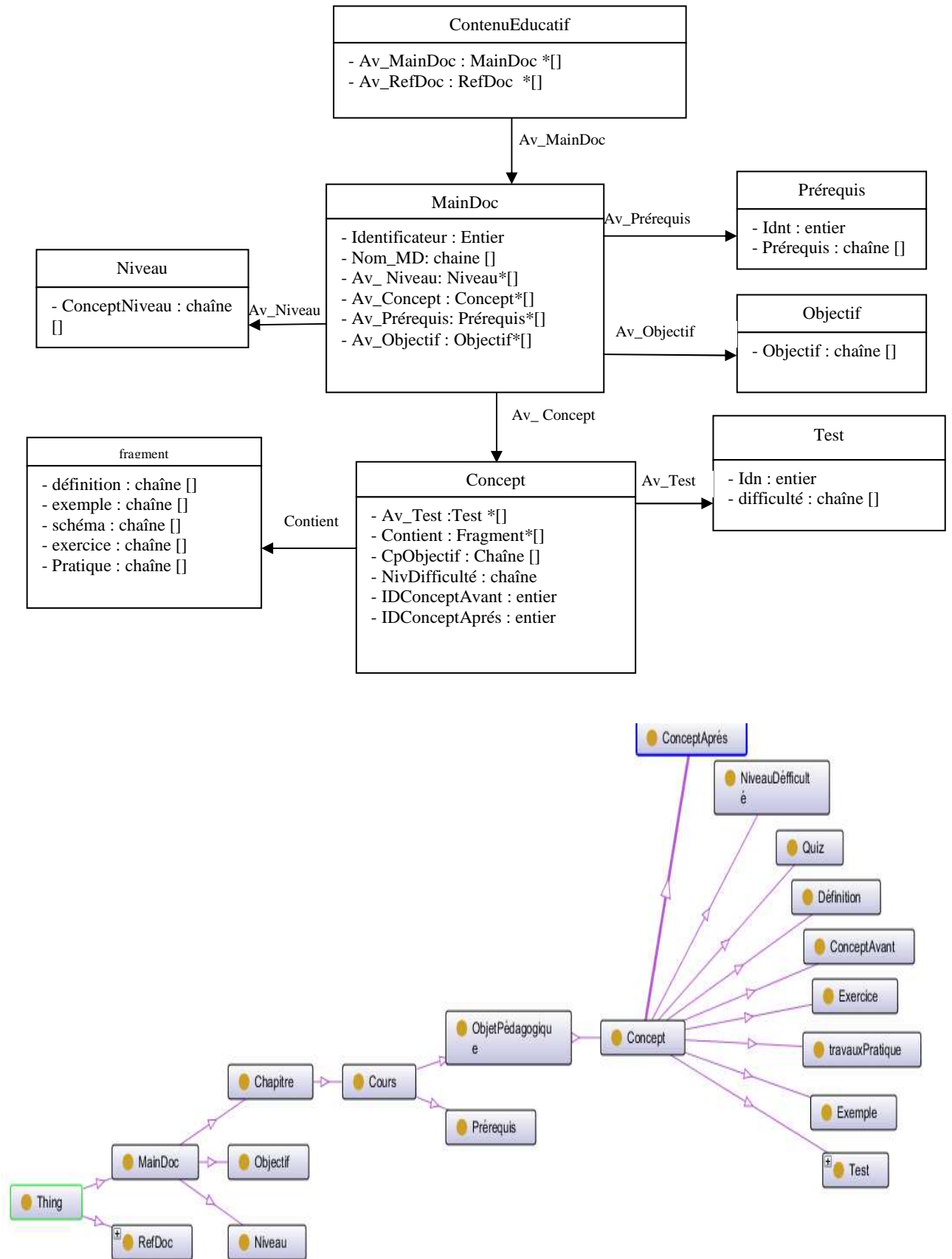


Figure IV.3.1.4.2.c : Ontologie du MainDoc

Le contenu éducatif est composé de deux type de document *MainDoc* et *RefDoc*, chaque *MainDoc* a un ensemble des *prérequis* et orienté vers un *niveau* particulier. Aussi le *MainDoc* Contient ensemble des *concepts* qui forment le module à apprendre. Le *Concept* a plusieurs propriétés : objectif, niveau de difficulté, concept précédent et avant. Le *concept* peut contenir : définitions, exemples, schémas, exercices ...etc. Finalement, pour chaque *concept*, un ensemble des tests sont associés.

IV.3.1.4.3. La couche 3 : Création et annotation des RefDoc

Dans les deux couches précédentes, nous avons formé les groupes de formation et créé le *MainDoc* avec l'ontologie de domaine associée. La troisième couche est consacrée à la création et l'annotation de deuxième type de document éducatif qui est *RefDoc*.

RefDoc est le deuxième type de contenu d'apprentissage, il peut être des exemples explicatifs, des schémas explicatifs, documents contiennent des explications détaillées, des résumés, style différent d'explication, des livres, des articles ... etc. Nous allons utiliser la principale annotation pour le *RefDoc* (en utilisant LOM), nous avons choisi ces critères: Le titre (marqué comme l'index principal), mots clés, les mots les plus répétés dans le document et le degré de pertinence (choisi par l'apprenant et l'enseignant). Aussi, l'enseignant doit fournir des informations supplémentaires sur le *Refdoc* pour compléter l'annotation de *Refdoc*, par exemple:

- *av_type* : type de *RefDoc* (exemple, schéma, livre, ...etc),
- *av_format* : le format de *RefDoc*.
- *Av_Style* : style d'apprentissage (textuel, auditif, ou visuel, Collaboratif, compétitif, global, séquentiel, détectif, intuitif, dépendant, indépendant ...etc). Pour chaque style, on a un pourcentage est affecté.
- *exigé_par* : important pour la compréhension de quel *MainDoc* ou *RefDoc*.
- *patie_de* : partie de quel *RefDoc*.
- *Référence_de* : référence pour quelle partie de *MainDoc*,
- *Av_version* : la version de *RefDoc*.
- *Av_prérequis* : des explications des prérequis,
- *Est_exercice* : Des exercices avec solutions,
- *exige* : Exige la lecture de quel *MainDoc* ou *RefDoc*
- *niv_détail* : détaillée ou non, contient des informations supplémentaires ou non,
- *av_niveau* : Orientés vers les apprenants excellents ou à ceux qui ont échoué, quiz ... etc.

Ces points seront également utilisés comme des annotations du *RefDoc* pour nous aider dans le processus de recherche (de l'extraction des documents de référence appropriés). Ces annotations sont réalisées par Protégé 4.3 en utilisant *SubClass* pour la hiérarchie des classes et *AnnotationProperties* pour les propriétés.

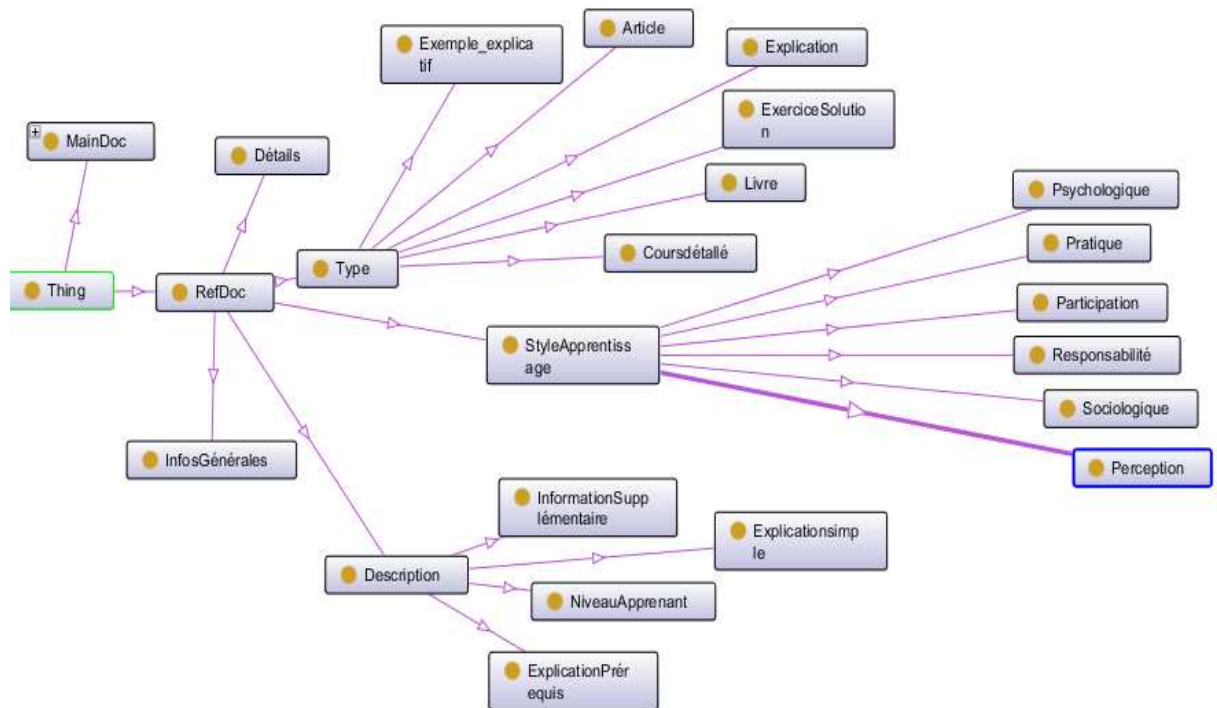
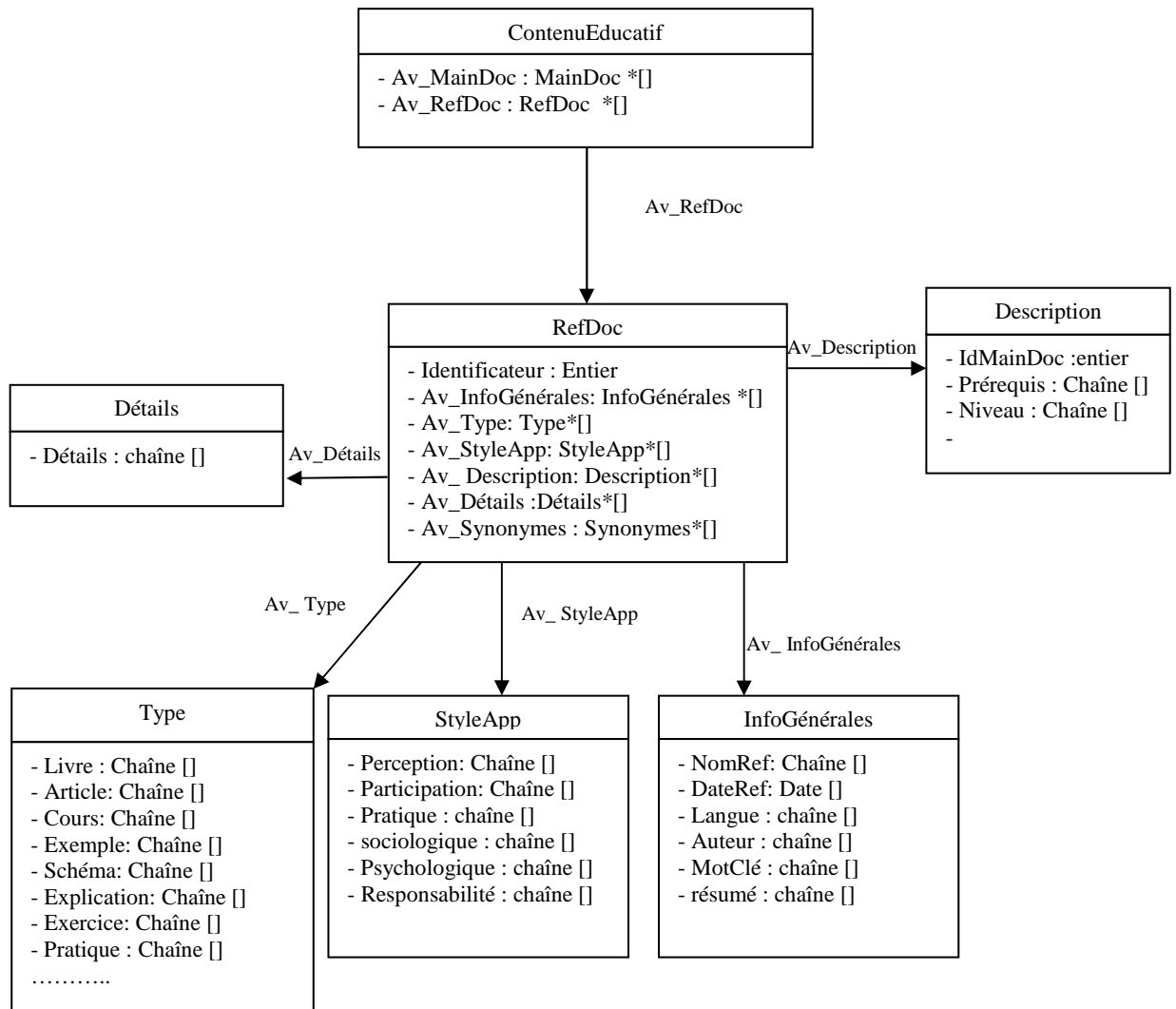


Figure IV.3.1.4.3.a : Ontologie de RefDoc

Pour augmenter l'aspect sémantique de RefDoc on crée des relations sémantiques entre les différents éléments de RefDoc et MainDoc. Ces relations vont aider le système à comprendre la structure et la sémantique de chaque élément, aussi elles facilitent la recherche des documents éducatifs dans le cas d'adaptation de contenu ou de stratégies. On utilisant Protégé 4.3, nous pouvons ajouter les relations (ObjectProperties, DataProperties). Leurs objectif est d'enrichir la sémantique de notre ontologie pour mieux comprendre les différentes relations et propriétés dans l'ontologie de RefDoc et MainDoc. Dans la figure suivante, les relations (propriétés) sont représentées par une ligne discontinuée. Nous avons choisi quelques propriétés pour expliquer notre approche. Les propriétés quantitatives n'apparaissent pas dans cette figure (seulement les propriétés entre les objets).

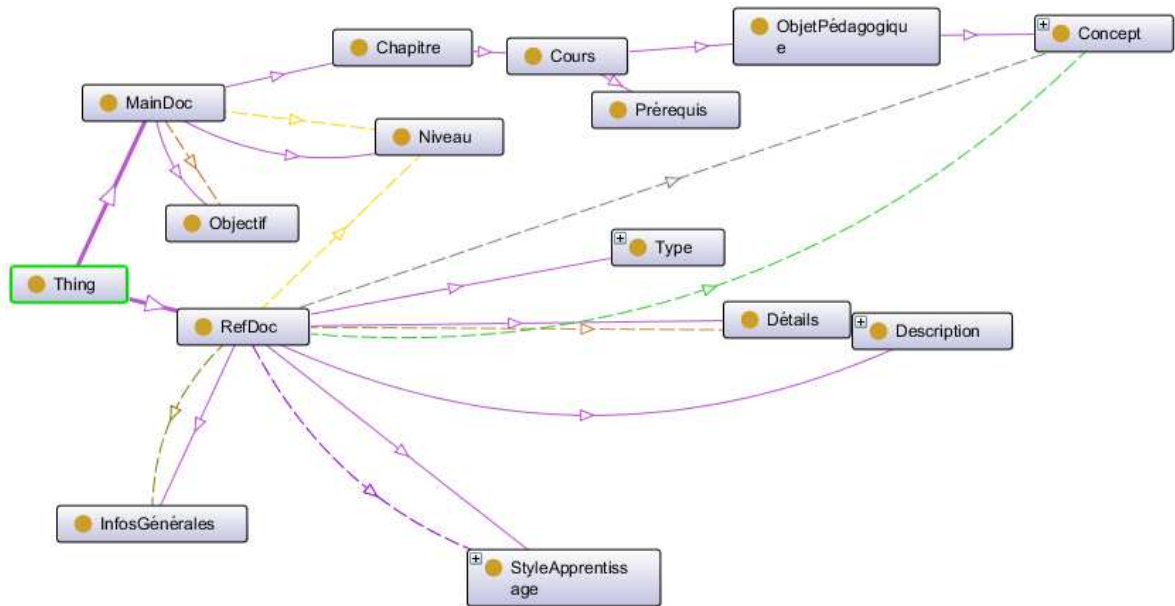


Figure IV.3.1.4.3.b : Ontologie RefDoc et MainDoc et leurs relations

Nom de la relation	Concept source	Cardinalité source	Concept cible	Cardinalité cible	Relation inverse
RéférenceDe	RefDoc	(1, n)	Cours	(1, n)	RéférencéPar
RéférenceDe	RefDoc		Concept	(1, n)	RéférencéPar
avoirInfoGnl	RefDoc	(1, n)	InfoGénérales	(1, n)	
avoirDescription	RefDoc	(1, n)	Description	(1, n)	
Avoir_style	RefDoc	(1, 1)	Style_apprentissage	(1, 1)	
équivalent	RefDoc	(1, 1)	RefDoc	(0, n)	équivalent
AvoirType	RefDoc	(1,1)	Type	(1, n)	
.....					

Dans le RefDoc, le vocabulaire utilisé peut être différent du vocabulaire utilisé dans le MainDoc. Ainsi, nous allons créer troisième annotation pour le Refdoc sur tous les synonymes utilisés dans le RefDoc, par exemple:

- architecture des ordinateurs = organisation des ordinateurs, structure des ordinateurs.

- Unité de Contrôle = unité de commande, unité de synchronisation, unité d'ordonnement.

Ces synonymes seront affichés aux apprenants quand ils lisent le RefDoc. En outre, ces synonymes seront utilisés pour vérifier l'existence de ces synonymes dans le MainDoc, donc on les change par les termes du vocabulaire sous-domaine. Ainsi, ces synonymes facilitent la recherche du RefDoc pertinent en utilisant le mot et ses synonymes dans le processus de recherche.

IV.4. Fonctionnement du système de création et d'annotation

Dans cette section, nous présentons le fonctionnement de notre approche pour atteindre l'objectif de créer et d'annoter les documents éducatif. Premièrement, on explique le processus de développement du MainDoc.

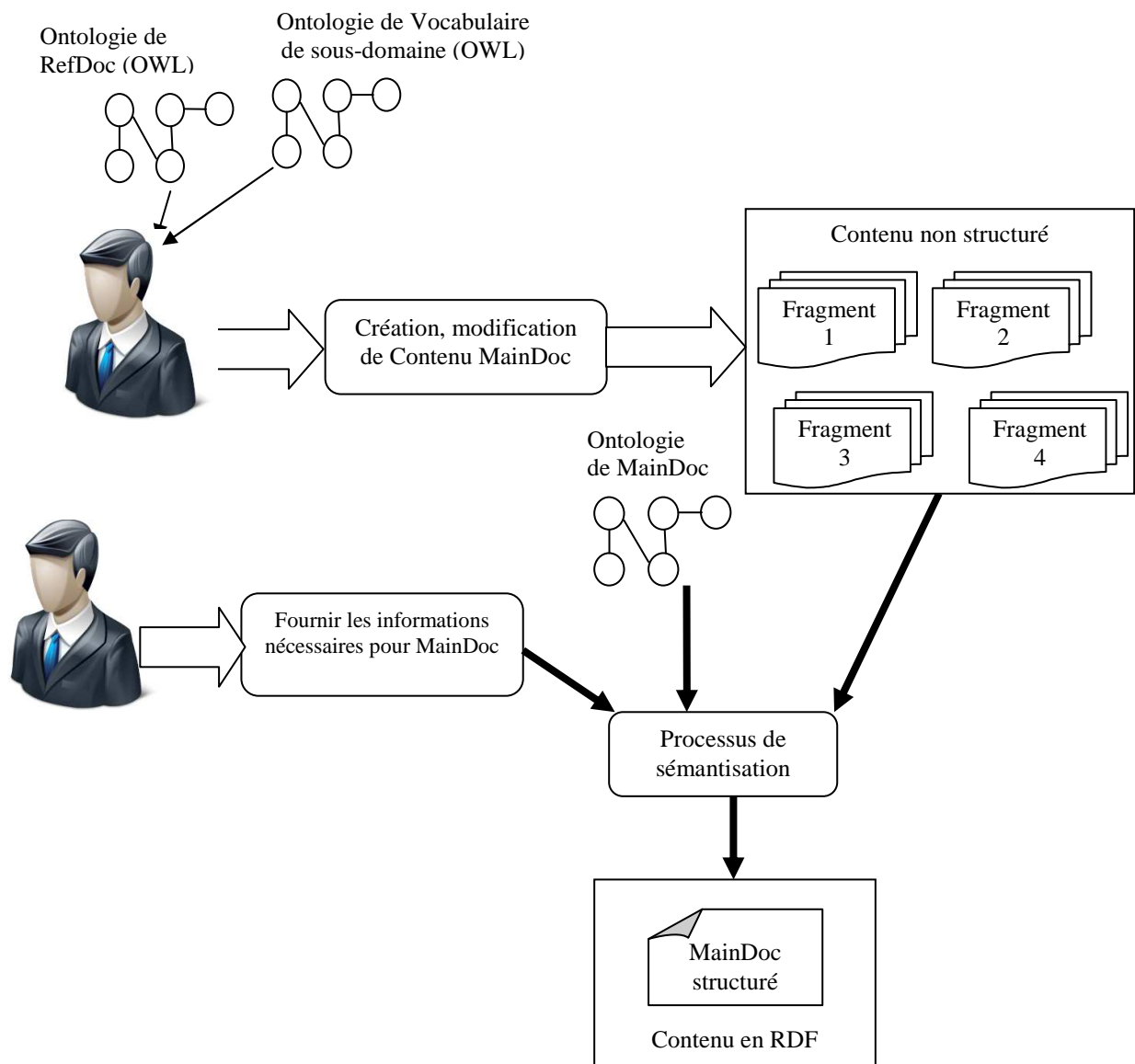


Figure IV.4.a : Processus de création du MainDoc

Le processus de création du MainDoc Commence par la rédaction des fragments (concepts) de contenu MainDoc par l'enseignant en utilisant plusieurs outils qu'il maîtrise : MS Word, Open Office ...etc. Le vocabulaire utilisé dans le concept créé doit suivre l'ontologie de vocabulaire de sous-domaine. Ensuite, l'enseignant utilise l'interface de notre système pour donner les concepts et les informations de chaque concept selon l'ontologie de MainDoc. Le processus de sémantisation (structuration de MainDoc et ses informations générales) est démarré par l'agent Gestion de cours et l'agent Création de cours. Comme résultat, nous avons un document structuré (Chapitre, cours, objet pédagogique, concept ...etc) en utilisant RDF pour la description du MainDoc.

La figure suivante explique la démarche pour créer et annoter le RefDoc en utilisant notre approche.

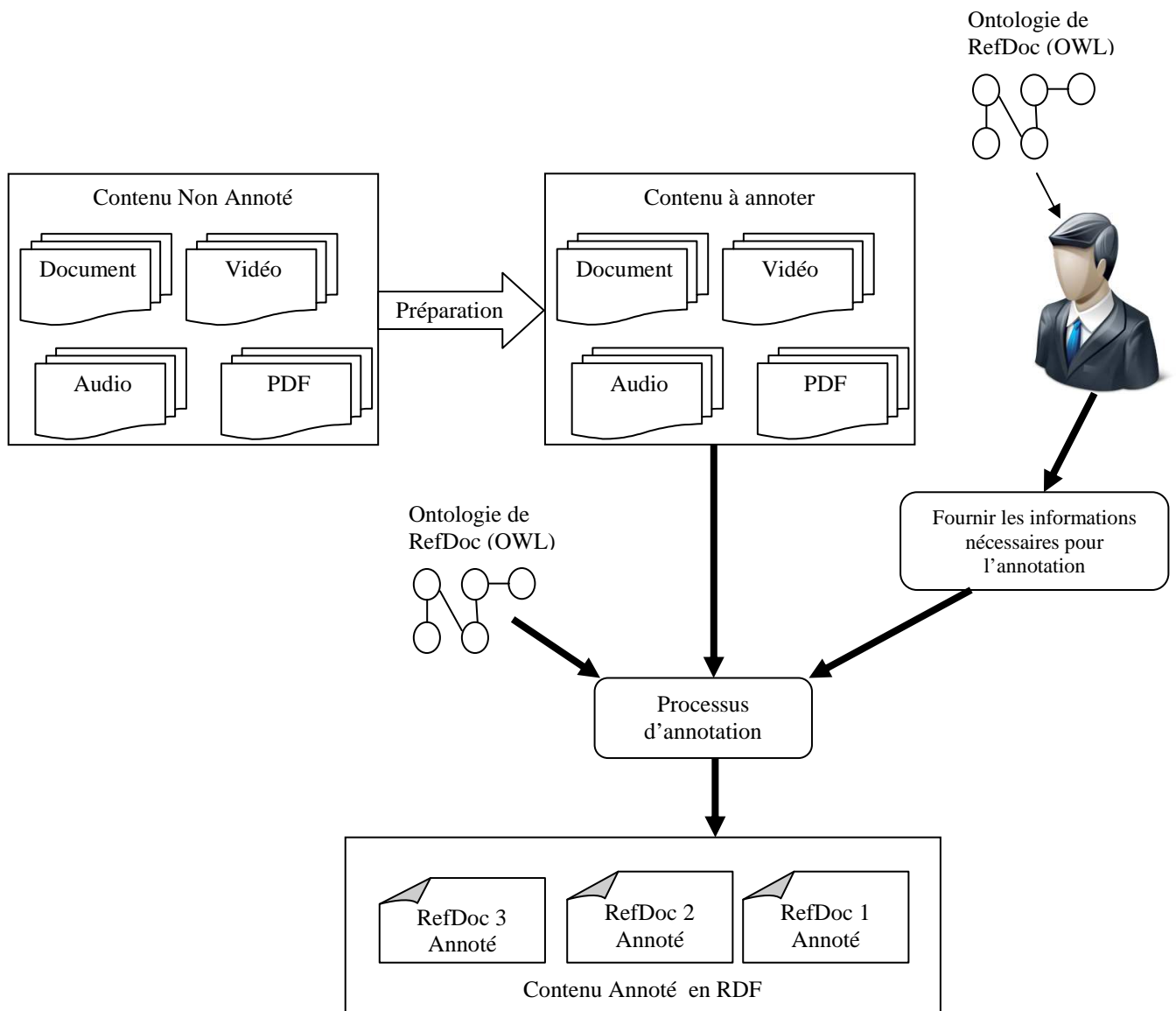


Figure IV.4.b : Processus d'annotation du RefDoc

La figure IV.4.b explique le processus d'annotation des ressources d'apprentissage. Ce processus représente la partie de sémantisation du contenu éducatif. Pour cela, nous avons trois étapes : 1) la préparation de RefDoc à annoter c'est à dire la préparation des descriptions et des informations utilisées pour l'annotation. L'enseignant peut utiliser tous les types de documents existants : PDF, DOC, AVI, MP3, ...etc, aussi il peut prendre une partie de document (partie de livre, article, thèse ...etc) comme il peut créer (écrire) un RefDoc (explication, exercices, ...etc). L'enseignant doit avoir une vue détaillée sur le document (titre, sous titres, niveau, contenu, l'auteur, version, date ...etc). 2) En se basant sur l'ontologie de RefDoc et les annotations utilisées, l'enseignant fournit, à travers l'agent interface, toutes les informations et les descriptions nécessaires pour l'annotation de RefDoc. 3) Notre système exploite ces informations fournies par l'enseignant et l'ontologie de RefDoc pour le stockage de RefDoc avec ses annotations en RDF qui référence (décrit) le RefDoc.

Voici par exemple une déclaration RDF qui décrit une ressource (*schedule.pdf*) par les propriétés : *NameRefDoc*, *TitleRefDoc*, et *StylePerception* ; et la relation *PartOf* .

```
<rdf:Description rdf:about="&RefDoc;schedule.pdf">
  <NameRefDoc> schedule </NameRefDoc>
  <TitleRefDoc> schedule _definition</TitleRefDoc>
  <StylePerception> visual</StylePerception>
  <PartOf rdf:resource="&RefDoc;OperatingSystem.pdf">/>
</rdf:Description>
```

IV.5. Architecture détaillée du système

En raison du succès de la technologie multi-agents et les caractéristiques des agents (autonomie, flexibilité, communication sémantique ...etc) et les avantages d'utilisation de cette technologie dans le Web, ainsi que les systèmes d'enseignement à distance [122][121]. Nous proposons une architecture à base d'agent pour un système éducatif et adaptif pour E-learning [122], cette architecture concerne la première partie de notre système (création du contenu pédagogique). La figure suivante représente notre architecture détaillée qui utilise un ensemble d'agents qui communiquent et coopèrent entre eux afin d'assurer l'intervention de plusieurs enseignants d'une manière collaborative pour créer et annoter le contenu d'apprentissage.

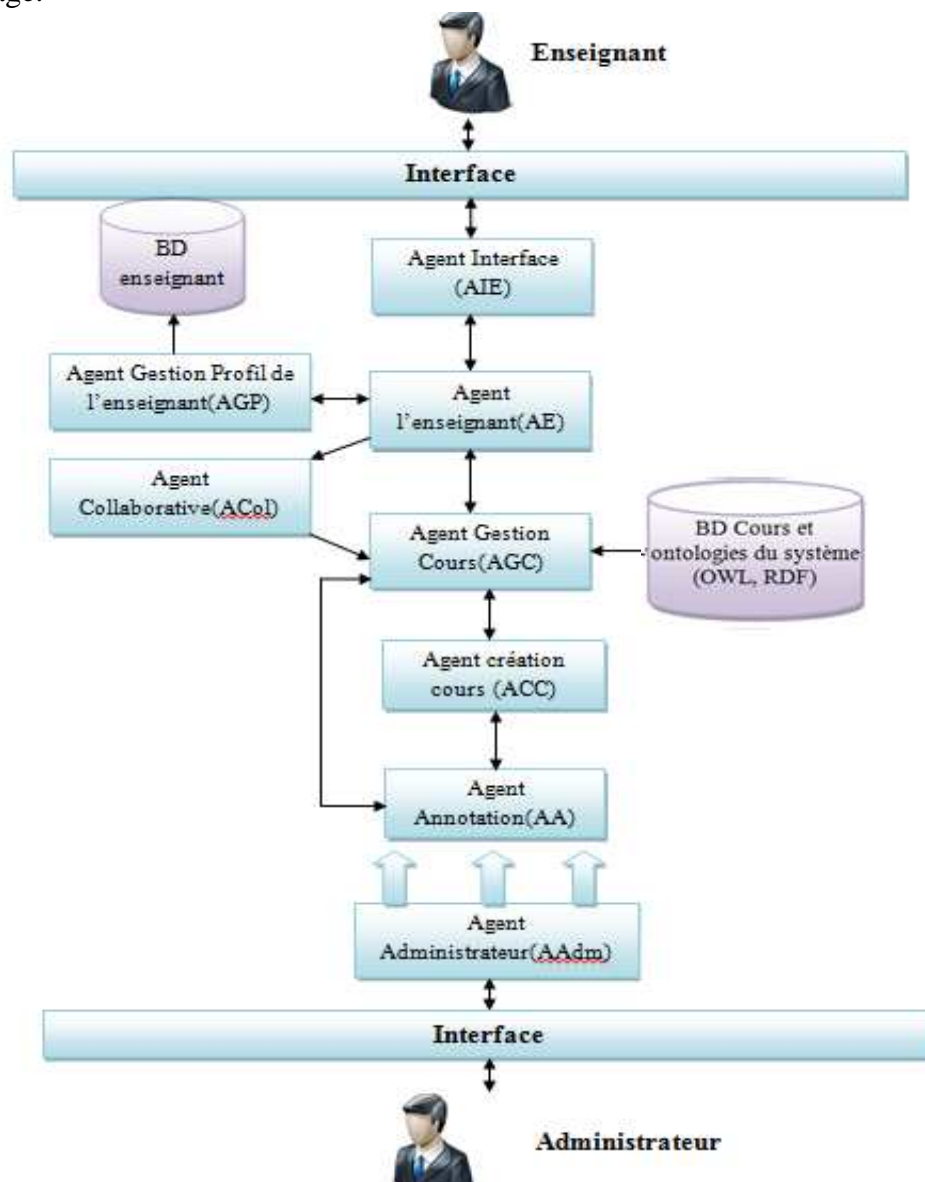


Figure IV.5 : Architecture détaillée du système (Création et annotation de contenu éducatif)

Dans ce qui suit nous présentons une description détaillée des différents agents de notre système et leurs communications :

IV.5.1. Agent interface enseignant (AIE)

Cet agent permet aux utilisateurs (enseignants) d'interagir avec le système. A travers cette interface, les enseignants peuvent exprimer leurs requêtes et de visualiser les résultats de celles-ci. Aussi, l'agent interface assure l'affichage de tous ce qui est dessiné aux utilisateurs (page d'accueil, les annonces d'inscription, création des cours ...etc). Il assure aussi l'inscription des nouveaux utilisateurs et l'identification des utilisateurs déjà inscrits. L'enseignant utilise cette interface pour la création de MainDoc et l'annotation de RefDoc ou modifier les documents déjà créés.

IV.5.2. Agent enseignant (AE)

Le rôle de cet agent est la préparation de tout ce qui est affiché à l'enseignant par l'agent d'interface enseignant. AE envoie le formulaire d'inscription à l'AIE, et envoie les pages (les informations) à AIE pour les afficher à l'enseignant. Si un enseignant change son profil, l'AIE envoie ces nouvelles informations à l'AE, ce dernier envoie les changements vers AGP pour les stocker dans la base de données. Il aide à l'identification des enseignants déjà inscrits. Il assure les services suivants :

- il contrôle l'autorisation de la connexion de l'enseignant par la vérification de mot de passe et l'identificateur de l'enseignant.
- Il fournit à l'enseignant le formulaire d'inscription dans notre plateforme.
- Il permet de vérifier ou modifier à partir de formulaire d'inscription les informations de l'enseignant.
- Il est responsable d'envoyer les informations des enseignants vers l'agent gestion profil (AGP).
- Il reçoit les informations nécessaires à la création et l'annotation des MainDoc et RefDoc, et il les envoie vers l'agent gestion cours.

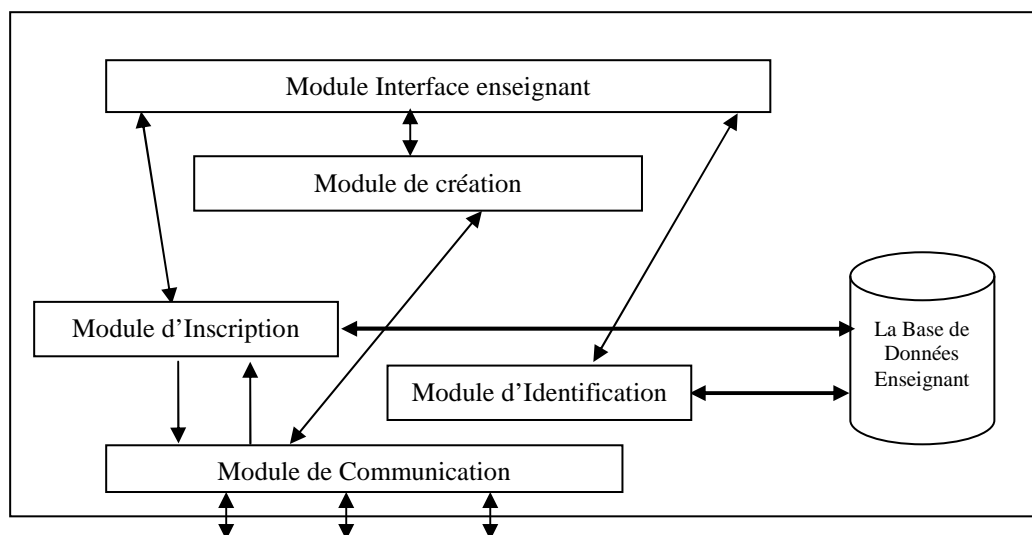


Figure IV.5.2 : Architecture structurelle de l'agent Enseignant

- **La base de données enseignant** : elle contient tous les profils des enseignants (le profil contient toutes les informations de l'enseignant : Diplôme, CV, le module qu'il est chargé...etc.) et les préférences de l'enseignant. Finalement il contient l'identificateur et le mot de passe.
- **Le module d'Inscription** : à travers ce module l'agent enseignant peut effectuer l'inscription des enseignants. Il reçoit les informations nécessaires à l'inscription et il les envoie vers l'agent Administrateur pour l'acceptation et l'initialisation du nouveau profil en fonction de ces informations. Si l'inscription est acceptée, ce module crée un nouveau profil pour l'enseignant et envoie un message d'acceptation vers le module Interface Enseignant.
- **Le module d'Identification** : ce module reçoit un message de demande d'identification par l'agent Interface Enseignant; ce message contient l'identificateur et le mot de passe de l'enseignant. Il vérifie la correspondance entre les informations données (ID + mot de passe) par celles-ci sauvegardées dans son profil pour permettre la connexion de l'enseignant. Ce module répond par l'envoi de message d'acceptation ou refus de connexion vers le module Interface Enseignant (et la raison de refus en cas d'échec).
- **Le module de création** : son principal rôle est d'aider l'enseignant dans le processus de la création et l'annotation du contenu éducatif par l'envoi des ontologies de vocabulaire, de MainDoc, et de RefDoc vers L'AIE. Aussi, il envoie les contenus créés, les informations générales et les annotations fournies par l'enseignant vers l'agent gestion cours pour compléter le processus de création et d'annotation.
- **Le module d'Interface enseignant** : il est responsable de l'envoi de ce qui est disponible pour l'enseignant dans le système et les différentes fonctionnalités du système vers l'interface pour la présentation sous forme d'une interface graphique tel que : page d'accueil, les annonces d'inscription, questions des apprenants, résultats des apprenants.
- **Le Module de Communication** : ce module gère les messages émis et les messages reçus. Il est responsable de la communication de l'agent Enseignant avec le système.

IV.5.3. Agent gestion profil enseignant (AGPE)

Cet agent est spécialisé dans la gestion de la base de données des profils des enseignants. AGPE reçoit les informations des nouveaux enseignants pour créer des nouveaux profils (AE envoie ces informations). Sa tâche principale est le stockage et l'envoi des informations des enseignants dedans la base de données. Il assure les services suivants :

- Stocker toutes les informations concernant le profil de l'enseignant dans la base de données.
- Gérer les informations des profils des enseignants (mettre à jour, modifier, supprimer, créer ...etc).
- Envoyer les informations de l'enseignant (mot de passe, Identificateur) vers AE pour vérifier l'autorisation de connexion.
- Envoyer toutes les informations nécessaires vers AE pour ouvrir la session des enseignants connectés.

IV.5.4. Agent gestion cours (AGC)

Cet agent effectue la gestion de la base de données de contenu d'apprentissage. Il stocke le contenu avec ses annotations (descriptions et l'ensemble de synonymes) et les prérequis de chaque cours, il offre les services suivants:

- Il envoie à ACC le MainDoc pour la structuration de cours (organisation et descriptions générales selon Ontologie de MainDoc)
- Il envoie à l'AA les annotations fournies par les enseignants pour annoter le RefDoc (selon l'ontologie RefDoc).
- Il stocke le contenu d'apprentissage créé avec ses annotations et synonymes (MainDoc et RefDoc, chaque type de document selon son ontologie définie précédemment).
- Il recherche sur un RefDoc à l'aide de la description et les synonymes.
- Il envoie à l'apprenant la RefDoc et les synonymes utilisé dans ce RefDoc.
- Il envoie le RefDoc en cas d'apprenant échoué ou excellente en utilisant l'annotation et les règles d'adaptation pour trouver les références pertinentes.

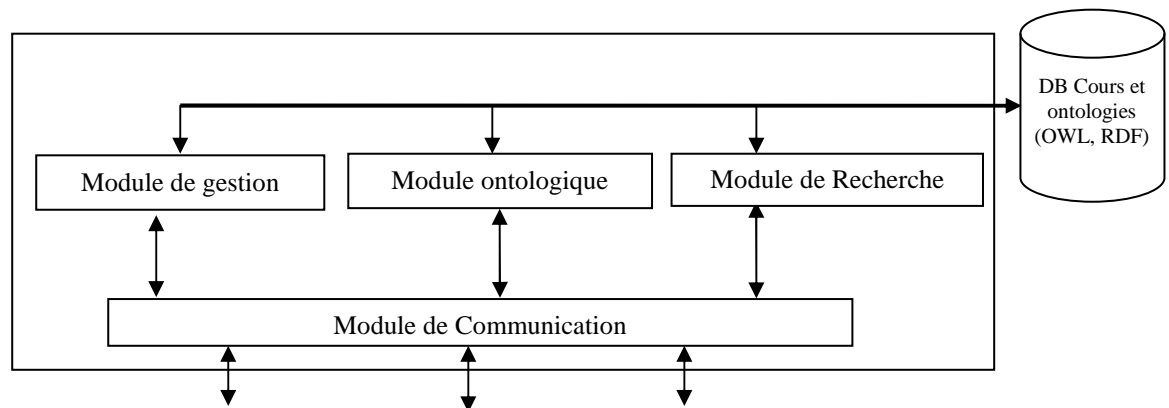


Figure IV.5.4 : Architecture structurelle de l'Agent Gestion Cours

- **Le module de Gestion :** dans le cas de la création du MainDoc, ce module envoie à ACC le MainDoc pour la structuration de cours (organisation et descriptions générales selon Ontologie de MainDoc). Dans le cas de la création du RefDoc, il envoie à l'AA les annotations fournies par les enseignants pour annoter le RefDoc (selon l'ontologie RefDoc). Finalement, il reçoit les MainDocs et RefDocs créés et annotés pour le stockage dans la base de données cours.
- **Le module ontologique :** il envoie l'ontologie vers les agents ACC et AA pour structurer et annoter les documents en se basant sur ces ontologies. Aussi, il vérifie l'annotation et la structuration des RefDoc et MainDoc par rapport les ontologies définies.
- **Le module de Recherche :** ce module a deux principaux rôles :1) la recherche et l'envoi de MainDoc (Concept suivant) vers l'apprenant; 2) il fait la recherche en utilisant les mots clés donnés par l'apprenant ou l'enseignant et leurs synonyme, puis il envoie les résultats vers le demandeur de la recherche.

IV.5.5. Agent création cours (ACC)

Cet agent est spécialisé dans la création du contenu, il offre plusieurs services avec la collaboration d'AA et AGC:

- Il reçoit le nouveau contenu ou ses mises à jour (modifications) à partir de l'AGC pour la création et structuration selon l'ontologie (nom, identificateur, dans quel chapitre, dans quel cours, concepts, le niveau, objectif ...etc).
- Il crée le contenu (structuration de contenu) et l'envoie à AA pour l'annotation (s'il est un RefDoc).
- Il envoie les mises à jour structurées à l'AGC pour mettre à jour le contenu éducatif dans la base de données.
- Il vérifie si le vocabulaire utilisé est le même que le vocabulaire défini en utilisant l'ontologie de domaine.

IV.5.6. Agent d'annotation (AA)

Le rôle de cet agent est d'assister l'ACC pour créer le contenu (RefDoc), il effectue l'annotation du contenu. Il reçoit le document pédagogique et ses annotations (définies précédemment) pour relier ses annotations avec le RefDoc. L'agent annotation se comporte comme suit:

- Il reçoit le contenu structuré non annoté de la CCA.
- Il reçoit les Annotations du contenu d'apprentissage de l'AGC (pour quel style d'apprentissage, type, référence pour quel MainDoc, niveau, détail, description, informations générales ...etc).
- Il reçoit l'ensemble des synonymes de chaque cours de CMA.
- Elle relie chaque annotation, description et synonyme avec la partie de cours associées pour former un RefDoc annoté selon son ontologie.
- Il envoie le contenu annoté final vers l'AGC pour l'enregistrer dans la base de données Cours.

IV.5.7. Agent collaborative (ACol)

Cet agent aide les enseignants à coopérer les uns avec les autres. Il informe les enseignants par la modification du contenu faite par un autre enseignant. Il effectue la modification ou l'ajout d'annotations.

IV.5.8. Agent administrateur (AAdm)

Cet agent permet de gérer l'interface entre le système et l'administrateur humain, il est chargé d'annuler l'inscription d'un enseignant, supprimer un enseignant, changer un enseignant. Ce module peut accéder à toutes les bases de données et gérer les autres modules administrativement ...etc.

IV.5.9. Scenario de fonctionnement du système

Dans cette section nous présentons les différents scénarios de fonctionnement de notre système. Ils aident à la compréhension et l'explication de processus de création des documents éducatifs. Nous avons choisi la connexion de l'enseignant et le processus de création de MainDoc et RefDoc qui sont considérés comme les deux processus le plus importants.

IV.5.9.1. Ouvrir une session

L'AI affiche la page d'accueil de l'enseignant qui va entrer son nom d'utilisateur et mot de passe. L'AI envoie, vers l'AE, les informations saisies par l'enseignant avec une demande d'ouverture d'une session. AE envoie cette demande ainsi que le mot de passe et nom d'utilisateur à la AGP pour les vérifier. AGP vérifie la correspondance entre le nom d'utilisateur et le mot de passe donnés par l'enseignant avec les informations enregistrées dans la base de données enseignant (profil de l'enseignant). Si le résultat du contrôle est «échec», alors l'AI affiche un message indiquant le type d'échec (mot de passe incorrect ou identificateur inexistant), sinon, la vérification est faite avec succès, cet agent va ouvrir la session. L'AI reçoit toutes les informations nécessaires pour les afficher à l'enseignant.

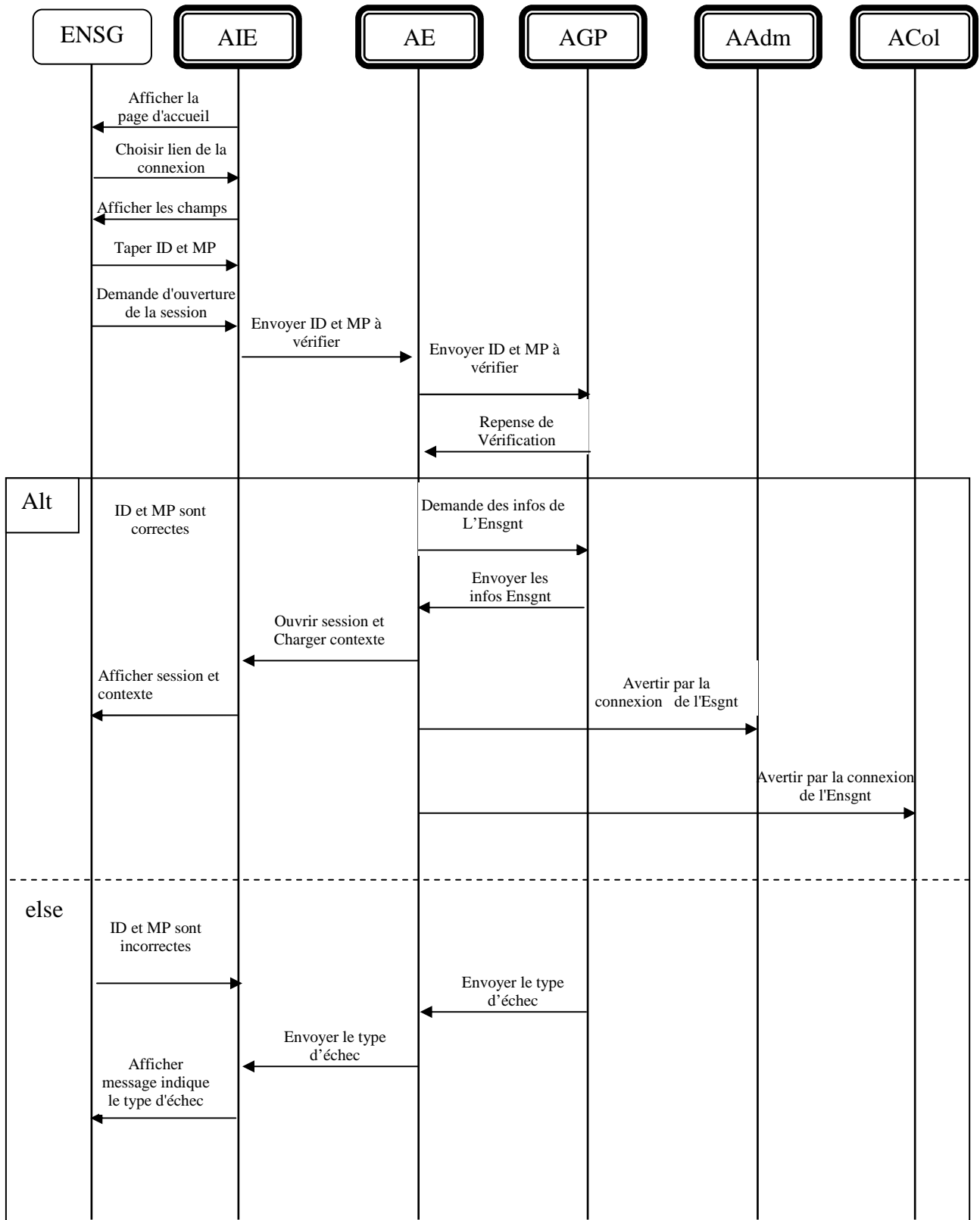


Figure IV.5.9.1. : Diagramme de séquence « ouvrir Session »

IV.5.9.2. Création et annotation d'un fragment de cours

Cette tâche est la plus importante, car c'est le noyau de notre système. Ce processus contient deux types : Création d'un MainDoc ou l'annotation de RefDoc. Donc, l'enseignant choisit l'un des deux processus par le clique sur les boutons affichés par l'agent interface enseignant.

- **Dans le cas de la création de MainDoc** : l'AGC envoie vers l'AE le vocabulaire de sous-domaine (l'ontologie de vocabulaire qui est utilisé pour la MainDoc). L'enseignant écrit une partie du cours, puis il donne toutes les informations nécessaires pour le MainDoc (titre, module, chapitre, ...etc). Lorsque l'enseignant clique sur le bouton «créer», l'AE envoie le contenu ainsi que les informations nécessaires vers AGC, ce dernier les envoie à l'ACC pour créer le MainDoc (compléter l'organisation du document, ajouter les informations sur le contenu, ...etc). L'ACC renvoie vers l'AGC le contenu éducatif organisé et structuré pour stocker le contenu final dans la base de données. Le MainDoc n'est pas annoté car il sera donné à tous les apprenants (contient les informations générales présentées dans l'ontologie de MainDoc). Par conséquent, l'AA ne sera pas l'annoter.
- **Dans le cas de Refdoc** : l'enseignant choisit une référence (livre, article, cours, ...etc) ou il la forme par lui-même (en utilisant l'AIE); aussi il remplit tous les champs (les informations nécessaires pour identifier et annoter le RefDoc). Ces informations sont nécessaires pour le Refdoc afin de l'annoter (selon l'ontologie) et de faciliter le processus de la recherche. Pour annoter le Refdoc, nous utilisons l'annotation principale (en utilisant les critères choisis de LOM : titre, mots-clés, degré de pertinence). Les autres informations sont : ce RefDoc est une référence de quel MainDoc, descriptions générales et toutes les informations générales (définies dans l'ontologie de RefDoc), ainsi que le type du Refdoc (des exemples explicatifs, schémas explicatifs, documents contiennent explication détaillée, résumé, livre, ... etc). Puis, on doit avoir le style d'apprentissage (auditif, visuel, collaboratif, détectif...etc). Le deuxième type d'annotation utilisée (donnée par l'enseignant) est les synonymes utilisés dans le Refdoc (les synonymes des termes de MainDoc). Enfin, l'enseignant définit de quelle règle d'adaptation est liée (règles d'adaptation de contenu est de stratégies). Dans ce niveau les informations nécessaires du Refdoc sont prêtes. L'AE envoie Le RefDoc (contenu éducatif) et toutes les informations fournies par l'enseignant vers l'AGC. Ce dernier envoie une demande de structuration de RefDoc (quel MainDoc, chapitre, cours, concept et informations générales ...etc) vers l'ACC pour créer le contenu de l'apprentissage. ACC prépare le contenu annoté par les informations générales seulement, et l'envoie à l'AA pour annoter le Refdoc par les autres types d'annotation. AA organise les trois types d'annotations (définies précédemment) pour le Refdoc et envoie le Refdoc final annoté vers AGC pour le stockage dans la base de données de contenu éducatif.

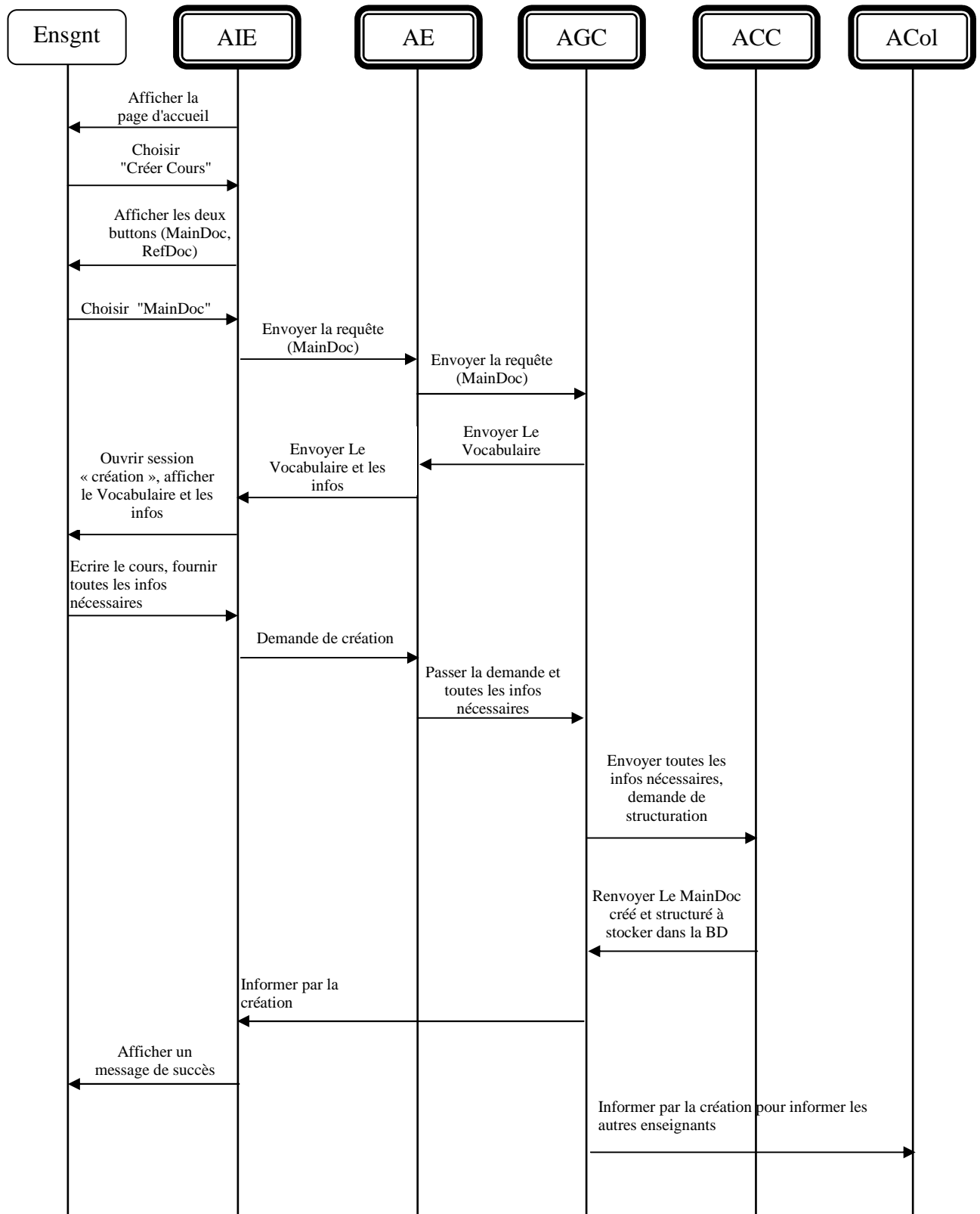


Figure IV.5.9.2.a : Diagramme de séquence « Création de MainDoc »

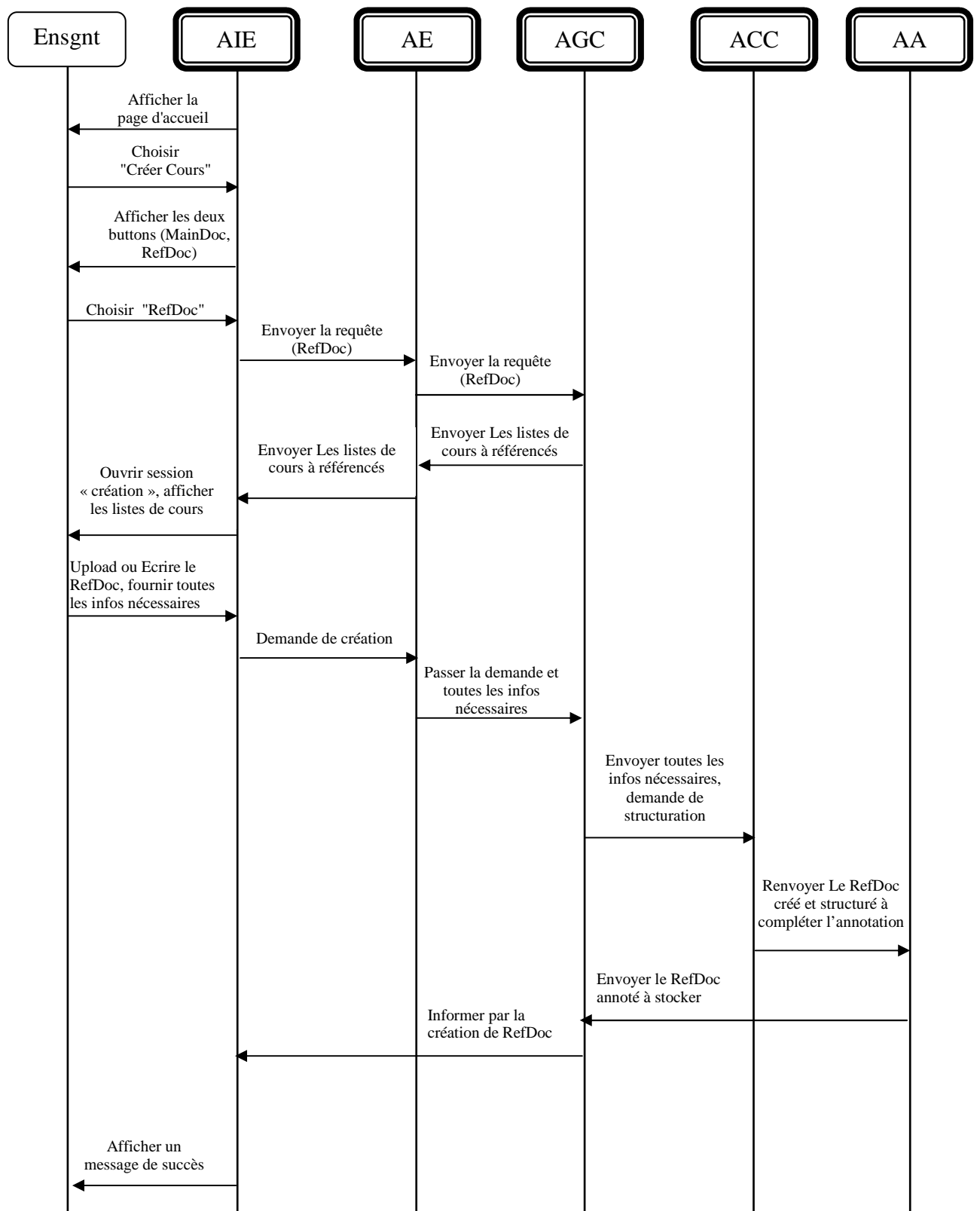


Figure IV.5.9.2.b : Diagramme de séquence « Création de RefDoc »

IV.6. Architecture de système d'apprentissage et d'adaptation

Dans cette section nous présentons la deuxième partie de notre approche qui assure : l'exploitation des MainDocs et RefDocs dans le processus d'apprentissage de l'apprenant et l'adaptation de contenu et de stratégies. L'architecture suivante modélise les différents agents qui sont capables de coopérer entre eux pour guider de manière bien structurée l'apprenant durant son processus de formation.

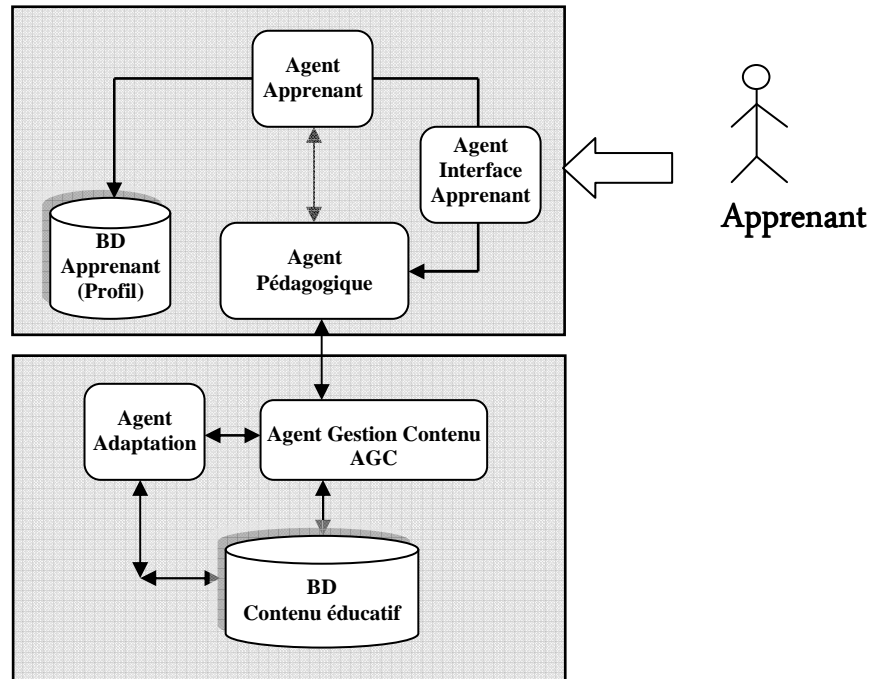


Figure IV.6 : Architecture de système (adaptation)

IV.6.1. Agent interface apprenant (AIA)

Cet agent permet à l'apprenant de réagir avec le système éducatif. Il porte les données de l'apprenant au système et il affiche les messages du système pour l'apprenant. AIA réagit comme des passerelles entre les apprenants et le système de formation à distance. L'agent interface apprenant assure l'affichage de tous ce qui est dessiné aux apprenants (page d'accueil, cours, tests et réponses des questions ... etc.). Il assure aussi l'inscription des nouveaux utilisateurs et l'identification des utilisateurs déjà inscrits et d'afficher les annonces des formations pour les apprenants.

Cet agent est la porte d'entrée au système pour l'apprenant, auquel plusieurs tâches sont attribuées :

- il contrôle l'autorisation de la connexion de l'apprenant et décide est ce que cet apprenant est déjà inscrit ou non.
- Il Affiche les avertissements du système tels que : date de test, période des questions, état de retard...etc.
- Il affiche aussi les fragments de cours et les activités selon le niveau et les préférences de l'apprenant.

- Il fournit à l'apprenant le formulaire d'inscription dans notre plateforme de la formation à distance.
- Il permet de vérifier les informations du formulaire d'inscription de l'apprenant.
- Il est responsable d'envoyer les informations nécessaires pour l'inscription vers l'agent Apprenant pour la création du profil de nouvel apprenant.

IV.6.2. Agent apprenant (AApp)

Le rôle principal de cet agent apprenant est de traiter les informations concernant le profil de l'apprenant (selon l'ontologie de profil). L'agent apprenant permet d'inscrire un nouvel apprenant et de modifier son niveau sur un cours, les résultats de tests, les styles ...etc. Cet agent permet de stocker les préférences des apprenants. Il est possible d'utiliser le profil pour connaître tous ceux qui concernent l'apprenant pour satisfaire ses besoins (état d'avancement, préférences, objectifs, styles d'apprentissage, niveauetc). En cas d'adaptation, l'agent pédagogique envoie vers l'AApp des requêtes pour avoir les informations de profil et pour modifier les informations de profil.

IV.6.3. Agent Pédagogique (AP)

Le principal rôle de cet agent est de fournir les cours et les références à partir de L'agent Gestion Cours. Il est considéré comme un intermédiaire entre l'apprenant et le contenu pédagogique. Cet agent reçoit des demandes à partir de l'AIA, puis il les envoie vers l'AGC (demande d'un MainDoc, RefDoc, recherche dans la base de données cours...etc). AP assure le suivi de l'apprenant durant son processus d'apprentissage.

IV.6.4. Agent gestion cours (AGC)

Cet agent effectue la gestion de la base de données de contenu d'apprentissage par l'envoi des cours demandés par l'agent pédagogique. Donc, il reçoit des demandes des MainDoc ou RefDoc à partir de l'agent pédagogique ou les requêtes de recherche. L'AGC répond par l'envoi des documents éducatifs demandés. Dans le cas d'adaptation, il envoie la demande de l'adaptation vers l'agent adaptation. Il offre les services suivants:

- Il envoie à l'AP le MainDoc et les RefDoc voulus.
- Il effectue l'opération de recherche à la demande de l'AP à l'aide de la description et les synonymes.
- Il envoie les résultats des recherches vers L'AP.
- L'intervention en cas d'échec pour adapter le contenu ou les stratégies à l'aide de l'agent adaptation.
- Il envoie toutes les informations vers l'agent adaptation pour les utiliser dans la détection de l'adaptation pertinente.

IV.6.5. Agent Adaptation (AAdp)

Il fait tous les types d'adaptation en utilisant les annotations de RefDoc, les informations de profil d'apprenant et les règles d'adaptation, ces différentes règles d'adaptation sont utilisées pour personnaliser l'apprentissage en utilisant les différentes ontologies.

Le Refdoc annoté est utilisé dans des cas particuliers (Pour enrichir les règles de l'adaptation il suffit d'ajouter d'autres règles dans la base de règles de cet agent). Donc, on présente quelques règles:

- L'apprenant échoue (moins de 50%): ici nous cherchons RefDoc qui a des annotations: des exemples, des schémas et des exercices avec solutions, utiliser le style auditif.
- L'apprenant échoue (moins de 25%): nous cherchons RefDoc qui contient des annotations: plus de détails, explication simple, utiliser le style auditif et visuel... etc. S'il ne suit pas les conseils des enseignants et du système nous cherchons des RefDoc pour les apprenants indépendants (style indépendant).
- L'apprenant échoue (moins de 10%): nous cherchons Refdoc qui a des annotations: les explications des prérequis, quiz, utiliser le style active. S'il fait des communications avec ses amis : on utilise des RefDoc qui exigent un travail collectif (style collaboratif).
- Pour les excellents apprenants : nous cherchons Refdoc qui a des annotations: plus d'informations, résumés, articles, ...etc.
- Quand un apprenant fait une recherche en utilisant des mots-clés qui existent dans le sous domaine: ici nous allons utiliser ces mots-clés et leurs synonymes dans le processus de recherche.
- Un apprenant qui pose plusieurs questions : nous cherchons Refdoc qui a des annotations: plus de détails, utiliser le style collaboratif.
- Un apprenant qui a échoué et qui manipule plusieurs activités et exercices (pratique) : on cherche les documents de référence qui ont le style détectif et on change le style de son profil par ce nouveau style.
- Si un apprenant a échoué, et il participe dans forum : on change son profil par le style actif et on cherche des RefDoc qui ont l'annotation pour Actifs.
- Si un apprenant échoue et qui a le style séquentiel dans le profil, on essaye de le changer au style global (cherchons des RefDoc qui ont comme annotation style global).
- Dans toutes les situations précédentes (particulièrement dans le deuxième échec dans le même concept) : on utilise des questionnaires qui contiennent plusieurs questions pour la détection automatique de style pertinent pour chaque apprenant (questions psychologiques et sociologiques, question sur le style préféré ...etc).

IV.6.6. Fonctionnement du système d'adaptation

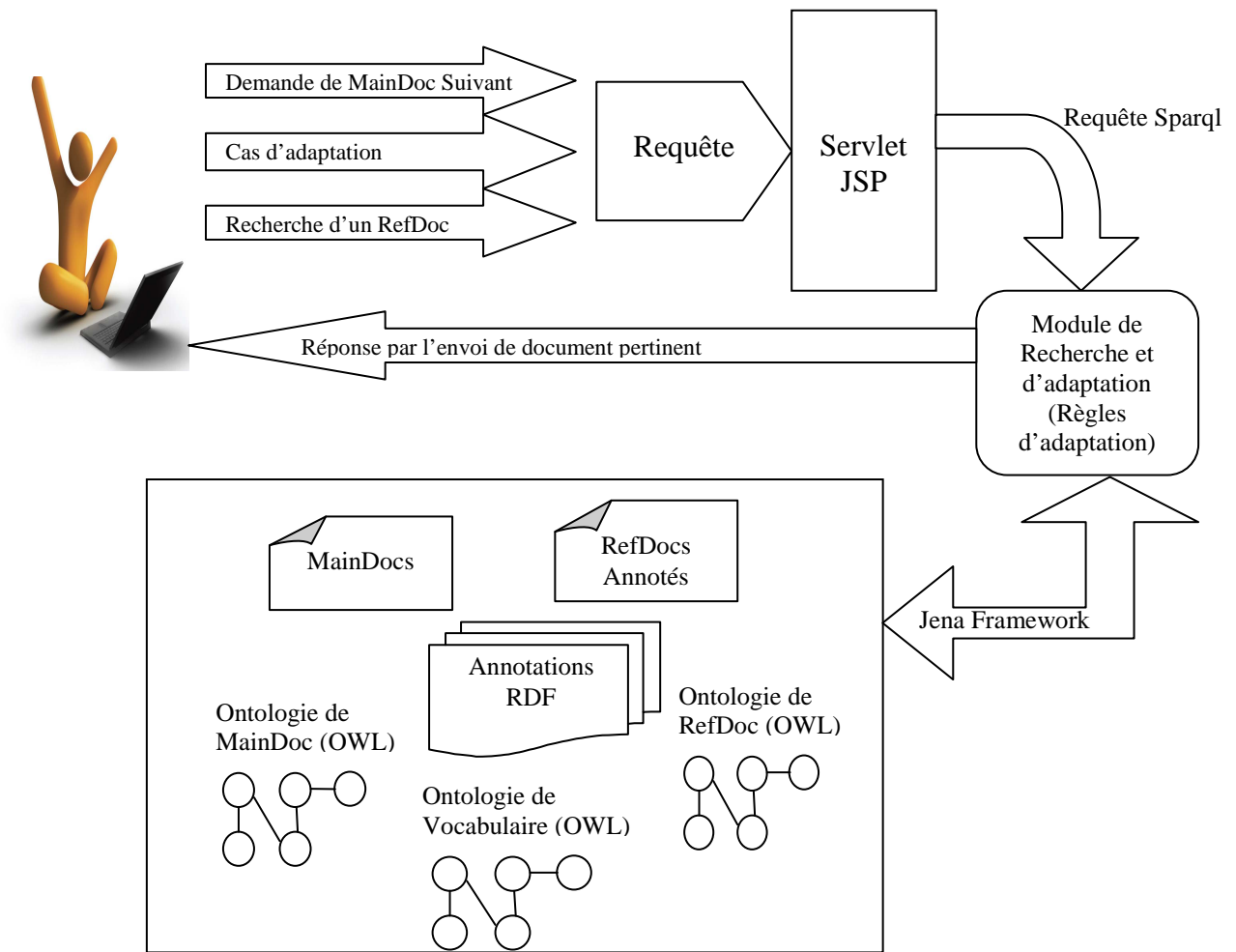


Figure IV.6.6 : Fonctionnement du système d'adaptation

L'apprenant demande un cours (concept suivant du MainDoc), ou il demande une opération de recherche, ou il est dans un cas qui exige une adaptation de contenu ou de style d'apprentissage. Donc, Une requête va être transmise via servlet (JSP) vers l'agent Gestion cours (dans le cas de demande d'un concept de MainDoc), ou vers l'agent adaptation (dans les deux autres cas). Ce dernier utilise Framework Jena pour extraire ou décrire dedans la base de données cours en RDF (en utilisant Sparql pour le langage de requête pour RDF). En appliquant les règles d'adaptation on forme une requête adéquate pour obtenir le RefDoc pertinent dans chaque cas. Dans le cas de recherche, on utilise les mots donnés et leurs synonymes pour trouver les documents pertinents.

IV.7. Conclusion

Nous avons exploité la technologie de Web sémantique pour améliorer l'apprentissage des apprenants. La représentation sémantique des contenus pédagogiques, en utilisant l'ontologie cela facilite la recherche et la réutilisation des documents éducatifs. Ainsi, elle aide la machine à comprendre et manipule les documents d'apprentissage. Les apprenants ont des caractéristiques hétérogènes, donc il faut assurer un apprentissage personnalisé pour chaque apprenant. L'adaptation de contenu et de stratégies (style d'apprentissages) s'avère très pertinente pour cet objectif. Finalement nous avons présenté une architecture pour la formation à distance. Cette architecture est basée sur les agents qui sont considérés comme un paradigme efficace (modularité, autonomie,... etc) dans les systèmes sur le Web (E-commerce, E-formation,...etc). Notre architecture assure la création et l'annotation des documents éducatifs, en utilisant la technologie de Web sémantique, et le bon guidage de l'apprenant durant sa formation et essaie de satisfaire les besoin des apprenants selon les niveaux et selon ses préférences.

Dans le chapitre suivant, nous allons présenter la démarche suivie et les outils utilisés pour développer un outil implémentant de telle approche.

Chapitre V : Implémentation

V.1. Introduction

Nous avons présenté dans le chapitre précédent la conception d'une plateforme de formation à distance en utilisant les systèmes multi-agents basée sur le Web sémantique. Pour exploiter cette approche nous présentons les différents outils utilisés pour le développement et les ontologies utilisées pour créer le contenu éducatif.

Nous essayerons d'élaborer un prototype et les différentes ontologies existantes dans notre système et les exploiter pour créer les documents éducatifs. Ces documents sont utilisés dans un système de la formation à distance à base d'agent capable de répondre aux besoins de l'apprenant tout en mettant en œuvre les procédures nécessaires pour satisfaire les besoins des apprenants.

Pour la réalisation de notre plateforme multi-agents destinée à la formation à distance, il est nécessaire d'utiliser un ensemble d'environnements spécifiques à la construction et à la mise en service d'agents. Ces outils peuvent servir également à l'analyse et au test du SMA ainsi créé. Ces outils peuvent être sous la forme d'environnement de programmation (API) et d'applications permettant d'aider le développeur. Nous allons utiliser dans notre implémentation la plate-forme JADE (Java Agent DEvelopment framework).

L'objectif de ce présent chapitre est de mettre en œuvre notre architecture. Tout d'abord, nous commençons par une brève présentation de l'environnement de développement, puis nous présentons les interfaces de notre application.

V.2. Environnement de développement

Pour le développement de l'outil qui exprime notre approche, nous utilisons Eclipse «juno». Eclipse est l'Environnement de Développement Intégré (EDI:Environment development integrated). Il est particulièrement bien adapté pour le développement d'applications Web, ainsi de supporter différents autres langages, comme Python, C, C++, Java Script, XML, HTML. C'est un IDE moderne qui offre un éditeur avec des codes couleurs et un ensemble de signes, des modèles de projets multi-langage et de différents types (application indépendante, distribuée, plugin, mobiles, ...etc), l'éditeur graphique d'interfaces et de pages web pour supporter le programmeur dans son travail.

Conçu en Java, Eclipse est disponible sous Windows, Linux ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Développent Kit JDK est requis pour les développements en Java.

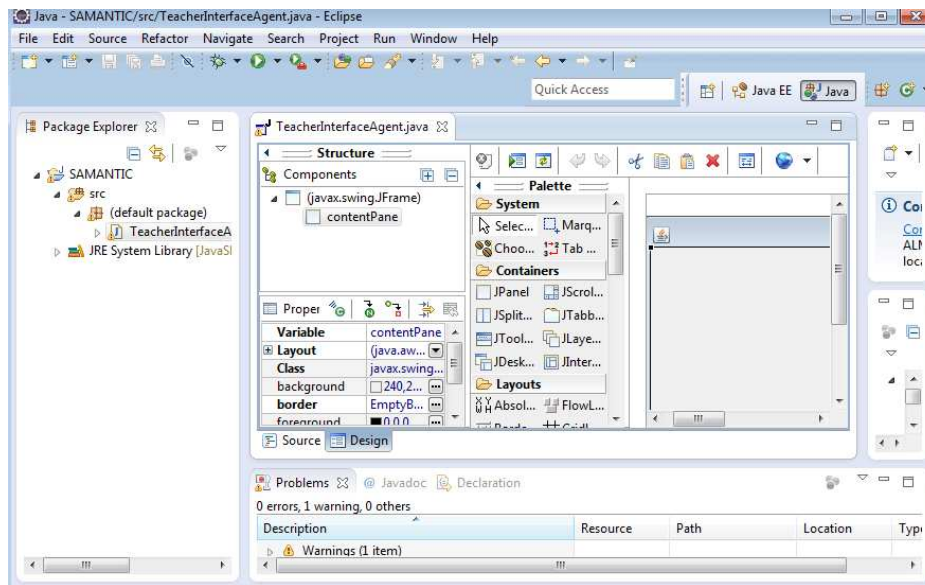


Figure V.2 : la fenêtre principale de l'Eclipse

V.3. Choix du langage de programmation

Nous avons choisi le langage Java. Java est un langage orienté objet développé par la société Sun (maintenant Oracle). La syntaxe générale du langage Java est très proche de celle du langage C. Dans ce qui suit, nous utilisons le terme Java pour désigner le langage de programmation. Ce choix a été motivé par les raisons suivantes :

- Les agents développés sous la plate-forme JADE, sont entièrement écrits en Java. Ce langage s'est donc imposé comme étant une conséquence de nos précédents choix en termes de plate-forme de développement du SMA (JADE).
- Un accès simplifié aux bases de données, soit à travers la passerelle JDBC-ODBC ou à travers un pilote JDBC spécifique au SGBD.
- Java assure une totale indépendance des applications vis-à-vis de l'environnement d'exécution : c'est à dire que toute machine supportant Java est en mesure d'exécuter un programme sans aucune adaptation (ni recompilation, ni paramétrage de variables d'environnement).
- Une conception pensée dès l'origine vers les réseaux : lors de l'exécution d'une application sur une machine-client depuis une machine-serveur, l'environnement Java téléchargera automatiquement les morceaux de programmes dont il a besoin.

- Une programmation orientée objet et une bibliothèque immense d'objets: dès sa naissance, les programmeurs de Sun ont doté leur langage d'une des plus grandes bibliothèques d'objets prêts à l'emploi.

V.4. La plateforme JADE

JADE (Java Agent Développement Environnement) permet le développement de systèmes multi-agents, la création des agent et l'envoi des messages entre eux. Elle est implémentée en JAVA et fourni des classes à base d'agent qui traite toutes les tâches liées aux fonctionnements des agents, telles que l'enregistrement, la configuration, la gestion à distance, ...etc. JADE contient un ensemble de méthodes qui peuvent être appelées pour implémenter les tâches spécifiques à l'agent, comme l'envoi des messages, utilisation des protocoles d'interaction standards, ...etc. En utilisant JADE nous pouvons gérer tous les aspects de contrôle des comportements. JADE propose également des classes de bases pour des types de comportements spécifiques (CyclicBehaviour, ParallelBehaviour, ReceiverBehaviour, SenderBehaviour, SequentialBehaviour...).

La plateforme multi-agents JADE a été choisie pour sa conformité aux normes de la FIPA et pour l'ensemble d'outils inclus facilitant les différentes phases de développement de systèmes multi-agents. Elle fournit ainsi une infrastructure agent, gérant la communication entre agents et la gestion de leurs cycles de vie, et sert de support pour le développement d'agent au comportement simple.

Lorsqu'on lance la plate-forme, on remarque que JADE a trois modules qui sont active au démarrage de la plate-forme, ces modules sont :

- **DF** « Director Facilitator » fournit un service de « pages jaunes» à la plate-forme ;
- **ACC** «Agent Communication Channel » gère la communication entre les agents ;
- **AMS** « Agent Management System » supervise l'enregistrement des agents, leur authentification, leur accès et l'utilisation du système.

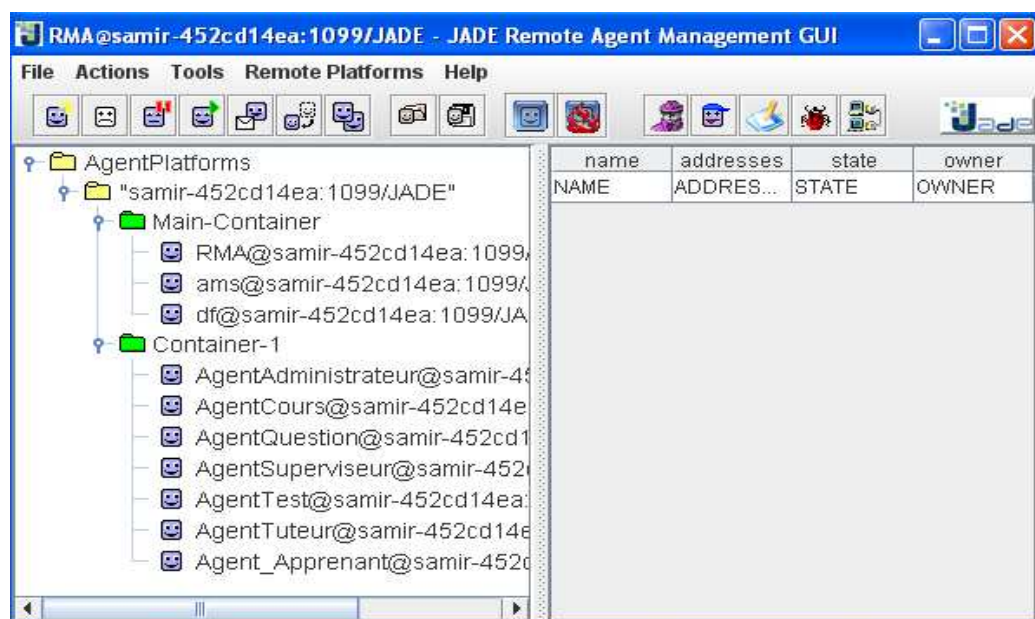


Figure V.4 : la fenêtre principale de la plateforme JADE

V.4.1. La création d'un agent

La création de l'agent JADE est faite à l'aide de la classe *jade.core.Agent* et l'implémentation de la méthode *setup()*.

```
import jade.core.Agent;
public class Apprenant extends Agent {
    protected void setup() {
        //l'initialisation et le comportement lorsque l'agent est lancé
    }
}
```

La méthode *setup()* est l'initialisation de l'agent et le comportement qui est fait par cet agent après sa création.

La méthode *dodelete()* doit être appelé à la fin de l'exécution de l'agent, cette méthode indique la terminaison de cet agent. La méthode *takeDown()* est appelé juste après la terminaison de l'agent pour la suppression des instances et les variables et les méthodes de l'agent.

V.4.2. Identificateur

Chaque agent a son identificateur qui représente l'instance de la classe *jade.core.AID*. La méthode *getAID()* de la classe *agent* permet l'obtention de l'identificateur d'un agent quelconque. La forme de l'identificateur d'un agent est *<nickname>@<platform-name>* où *nickname* est le nom de l'agent et *platform-name* est le nom de la plateforme qui contient cet agent.

```
AID id = new AID(nom, AID.ISLOCALNAME);
```

Lorsqu'on utilise la constante *ISLOCALNAME*, alors on veut avoir le premier nom dans l'identificateur c'est à dire *nickname* seulement.

V.4.3. Comportement d'un agent

L'agent peut exécuter plusieurs comportements concurrents, mais dans JADE l'exécution n'est pas préemptive comme celle de *thread* de JAVA mais Coopérative c'est à dire lorsqu'un comportement commence, donc il s'exécute jusqu'à la fin de l'exécution du contenu de son comportement.

Les méthodes de l'agent sont exécutées dans "Comportements = behaviours". Un *comportement* représente la tâche qu'un agent doit faire et il est un objet d'une classe qui hérite *jade.core.behaviours.Behaviour*. Pour permettre à l'agent d'exécuter un comportement (behaviour) il est nécessaire d'appeler la méthode *addbehaviour()* de la classe *agent*. On peut ajouter le comportement *behaviour* à l'agent dans n'importe où dans la méthode *setup()* qui lance l'agent.

Chaque classe héritant de la classe *behaviour* doit implémente la méthode *action()* c'est à dire chaque comportement doit avoir la méthode *action()*. Cette méthode indique les opérations faites par l'agent lorsque le comportement *behaviour* est exécuté. La méthode *done()* retourne une valeur booléenne qui indique la fin du comportement *behaviour* ou la continuation de l'exécution du comportement.

```

public class LireCours extends Behaviour {
    public void action() {
        while (true) {
            // Comportement
        }
    }
    public boolean done() {
        return true;
    }
}

```

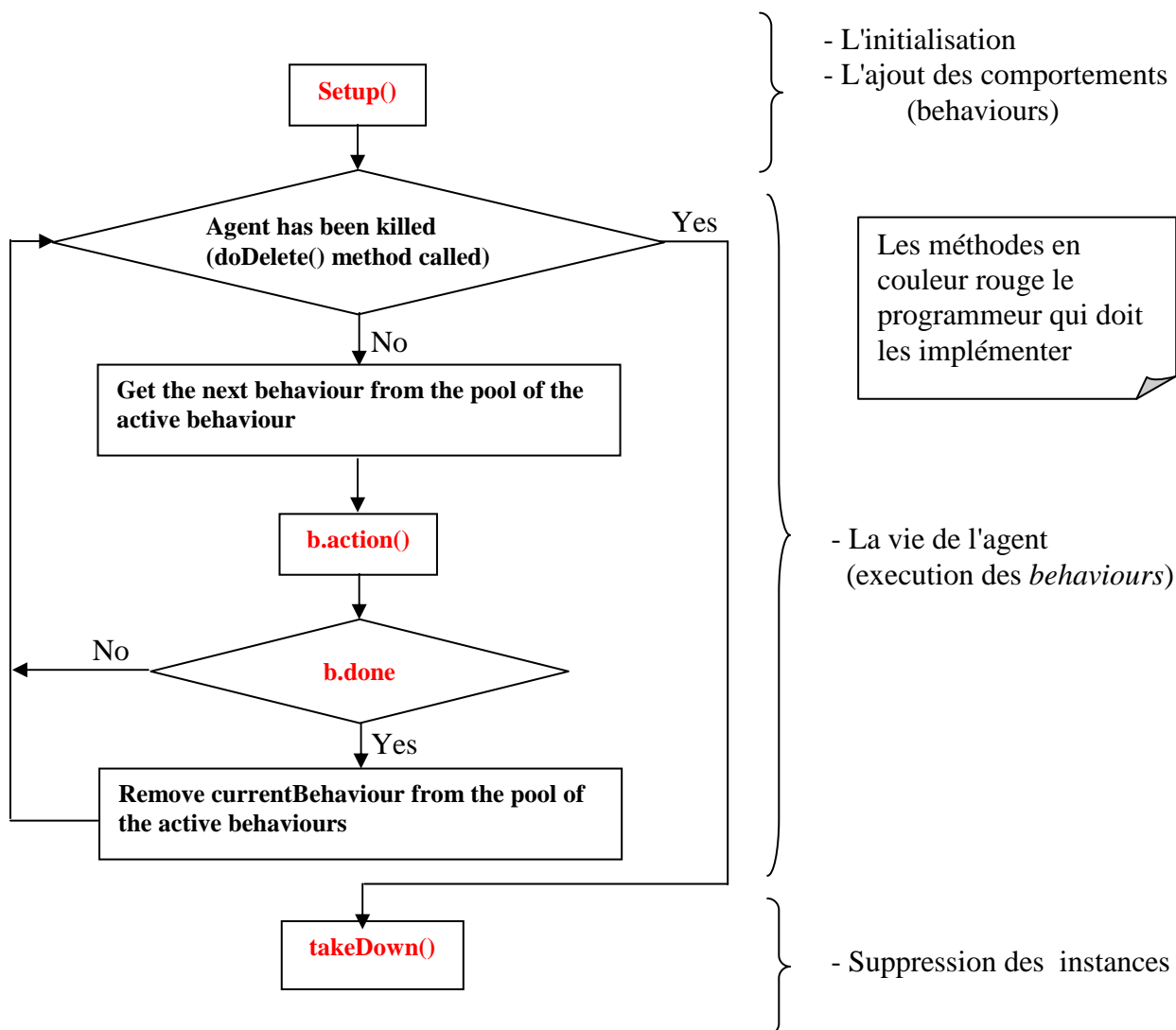


Figure V.4.3: Les chemins d'exécution des comportements d'un agent afin de libérer le processeur. Dans le cas où un comportement est encore disponible, alors l'agent se réveille pour exécuter un *Behaviour*.

On peut trouver non seulement un seul type de Behaviour mais il y a trois types essentiels à savoir:

1. **One-shot behaviours**: ce type de comportement est terminé immédiatement après l'exécution une fois de la méthode action.

```
Import jade.core.behaviours.OneShotBehaviour
public class MyOneShotBehaviour extends OneShotBehaviour {
    public void action() {
        // exécuter l'opération X
    }
}
```

L'opération X est faite une seule fois seulement.

2. **Cyclic behaviour**: dans ce type de comportement le *Behaviour* ne se termine jamais et il est exécuté chaque fois qu'il y a un appel de ce comportement.

```
Import jade.core.behaviours.CyclicBehaviour
public class MyCyclicBehaviour extends CyclicBehaviour {
    public void action() {
        // exécuter l'opération Y
    }
}
```

L'opération Y s'exécute toujours (s'il y a un appel) jusqu'à la terminaison de l'agent.

3. **Generic behaviours**: le comportement s'exécute selon des états. La terminaison de ce comportement est réalisée selon une condition donnée (si la condition est réalisée alors terminaison du comportement).

```
public class MyThreeStepBehaviour extends Behaviour {
    private int step = 0;
    public void action() {
        switch (step) {
            case 0:
                // perform operation X
                step++;
                break;
            case 1:
                // perform operation Y
                step++;
                break;
            case 2:
                // perform operation Z
                step++;
                break;
        }
    }
}
```

```

    }
  }
  public boolean done() {
    return step == 3;
  }
}

```

Les opérations X, Y et Z s'exécutent une après l'autre puis il y a la terminaison du comportement.

Ils existent d'autres types de comportement tels que: *SequentialBehaviour*, *ParallelBehaviour* and *FSMBehaviour*.

Type de comportement	Son nom en JADE	Sa classe JADE
Cyclique	CyclicBehaviour	jade.core.behaviours.CyclicBehaviour;
OneShot	OneShotBehaviour	jade.core.behaviours.OneShotBehaviour;
Périodique	TickerBehaviour	jade.core.behaviours.TickerBehaviour;
Séquentiel	SequentialBehaviour	jade.core.behaviours.SequentialBehaviour;

Tableau V.4.3 : Types de comportements utilisés

V.4.4. Le langage ACL

La communication est l'une des plus importantes opportunités offertes par la plateforme JADE. Le paradigme adopté est le passage des messages asynchrones "asynchrone message passing". Chaque agent détient une sorte de boîte aux lettres "the agent queue" destinée à recevoir les messages provenant des autres agents. Tout message arrivant sur cette boîte est notifié pour qu'il puisse être traité par l'agent. Les messages échangés dans JADE ont le format spécifié par le langage ACL défini par FIPA.

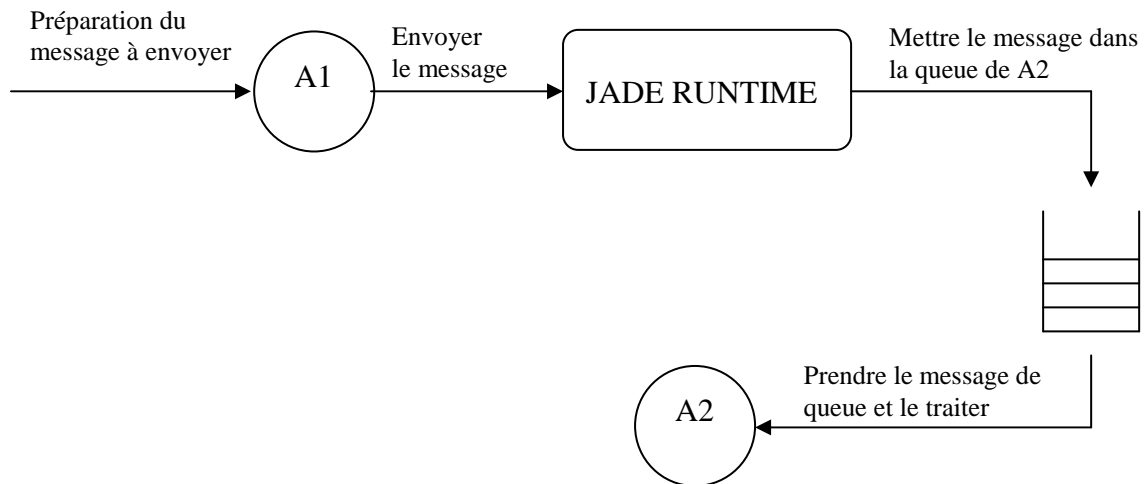


Figure V.4.4 : Le paradigme du passage d'un message asynchrone

Le format de messages envoyés par JADE est spécifié par le langage ACL (asynchronous message passing). Ce langage est défini par FIPA (Foundation for Intelligent

Physical Agents (<http://www.fipa.org>) le standard international pour l'interopérabilité des agents. Les champs de ce message sont:

- **Sender** : contient le nom de celui qui envoie le message.
- **Receiver** : contient la liste des noms des agents qui vont recevoir le message.
- **Performative** : ce champ peut prendre les valeurs suivantes : REQUEST, INFORM, PROPOSE, ACCEPT_PROPOSAL, REJECT_PROPOSAL .
- **Content** : ce champ contient le message à envoyer.

V.4.4.1. L'envoi d'un message

Pour envoyer un message, premièrement, il faut donner les valeurs des champs de ACLMessage motionnés ci-dessus. Puis il faut faire un appel de la méthode *send()* de la classe *agent*.

```
ACLMessage msg = new ACLMessage(ACLMessage.INFORM); // déclaration du message
msg.addReceiver(new AID("Sam", AID.ISLOCALNAME)); // sam est le nickname de l'agent
msg.setContent("Hello Sam"); // Message = "Hello Sam"
send(msg); // Envoyer le message
```

V.4.4.2. La réception du message

Le JADE Runtime met le message dans la queue de l'agent récepteur. Dans ce cas l'agent récepteur peut prendre ce message pour appeler la méthode *receive()*. Cette méthode retourne le premier message de la queue de l'agent puis elle le supprime; si la queue est vide, alors cette méthode retourne *null*.

```
ACLMessage msg = receive();
if (msg != null) {
    // traiter le message
}
```

Type de contenu	Lire (Prendre) contenu	Ecrire (Poser) contenu
String	getContent()	setContent()
Java Objects	getContentObject()	setContentObject()

Tableau V.4.4 : Communications entre les agents

V.4.5. La création d'un agent avec une interface graphique

Dans la plateforme JADE, nous pouvons utiliser deux types d'agents, donc on a deux classes abstraites JADE qui ont été utilisées pour déclarer les agents non graphiques et les agents graphiques (ayant des interfaces utilisateurs) c'est à dire des agent avec Gui (Graphical User Interface):

- La classe *jade.gui.GuiAgent* pour les agents d'interfaces. Gui (Graphical User interface) est utilisé pour la déclaration d'un agent avec une interface graphique.

- La classe *jade.core.Agent* pour déclarer les agents non graphiques.

Pour recevoir les événements des composants de l'interface (Button, Edit, label...etc), *GuiAgent* doit implémenter la classe *ActionListener* et il doit aussi utiliser la méthode *actionPerformed()* pour chaque composant d'interface.

a. Classe Interface

```
public class A_Interface_App extends JFrame implements ActionListener{
/*----- */
private A_App myAgent; // référence vers la classe Agent
/*----- */
    public A_Interface_App(A_App a) {
        this.myAgent = a;
    }
/*----- */
    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == Inscire) {
            GuiEvent ge = new GuiEvent(this, myAgent.NEW_APPRENANT);
            myAgent.postGuiEvent(ge); // envoyer un évènement "new apprenant"
        }
/*----- */
        else if (ae.getSource() == Inscrivez) {
            GuiEvent ge = new GuiEvent(this, myAgent.INSCRIRE);
            myAgent.postGuiEvent(ge); // envoyer un autre évènement "Inscrivez"
        }
    }
}
```

b. Classe Gui (Graphic User Interface)

```
/*----- */
transient protected A_Interface_App myGui; //référence vers la classe Gui
/*----- */
protected void setup() {
    myGui = new A_Interface_App(this);
    myGui.setVisible(true);
}
/*----- */
protected void onGuiEvent(GuiEvent ev) { // procédure de la réception des évènements
    command = ev.getType();
    switch (command) {
        case NEW_APPRENANT: // Recevoir d'un évènement "new apprenant"
            try {
                // procedure
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }
}
```

```

}
return;
.....etc.

```

V.5. L'éditeur d'Ontologie Protégé

Protégé est un éditeur d'ontologies distribué en open source par l'université en informatique médicale de Stanford. Protégé n'est un outil spécialement dédié à OWL, mais un éditeur hautement extensible, capable de manipuler des formats très divers.

Le support d'OWL, comme de nombreux autres formats, est possible dans protégé grâce à un plugin dédié. Protégé est un outil employé par les développeurs et des experts de domaine pour développer des systèmes basés sur les connaissances (Ontologies). Des applications développées avec Protégé sont employées dans la résolution des problèmes et la prise de décision dans un domaine particulier. Protégé est aussi une plate-forme extensible, grâce au système de plug-ins, qui permet de gérer des contenus multimédias, interroger, évaluer et fusionner des ontologies, etc. L'outil Protégé possède une interface utilisateur graphique (GUI) lui permettant de manipuler aisément tous les éléments d'une ontologie : classe, méta-classe, propriété, instance,...etc. Protégé peut être utilisé dans n'importe quel domaine où les concepts peuvent être modélisés en une hiérarchie des classes.

Protégé permet aussi de créer ou d'importer des ontologies écrites dans les différents langages d'ontologies tel que : RDF-Schéma, OWL, DAML, OIL, ...etc. Cela est rendu possible grâce à l'utilisation de plugins qui sont disponibles en téléchargement pour la plupart de ces langages.

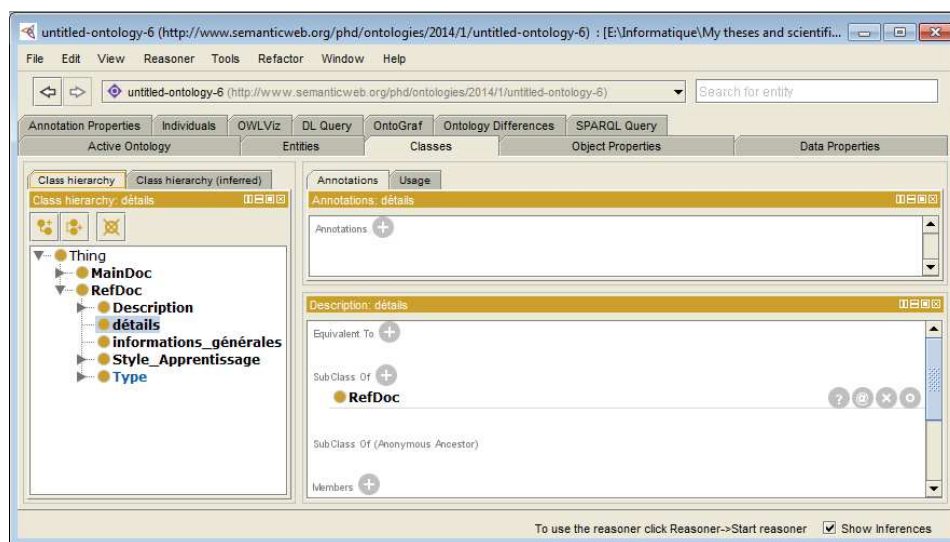


Figure V.5 : La fenêtre principale de Protégé 4.3

Protégé 4.3 offre la possibilité de créer les ontologies utilisées dans notre système. La hiérarchie des classes est réalisée à l'aide d'utilisation de *SubClass*. Aussi, les annotations sont réalisées par Protégé 4.3 en utilisant *AnnotationProperties*. Nous pouvons ajouter des

propriétés et des relations entre les différents éléments de l'ontologie en utilisant *ObjectProperties*, *DataProperties* et *AnnotationProperties*.

V.6. JENA

Jena est un « open source framework » de Web sémantique pour Java. Il fournit une API pour extraire des données et d'écrire des données sur des graphes RDF. Les graphes sont représentés comme un modèle abstrait. Un modèle peut être obtenu auprès de données à partir de fichiers, bases de données, des URL ou une combinaison de ceux-ci. Un modèle peut également être interrogé par SPARQL et mis à jour par SPARUL.

Jena est similaire à Sesame, bien que, Jena fournit un support pour OWL (Web Ontology Language). Jena a plusieurs raisonneurs internes et le raisonneur Pellet (un open source Java OWL-DL raisonneur) peut être mis en place pour travailler à Jena.

Jena peut supporter la sérialisation de graphes RDF pour:

- Une base de données relationnelle
- RDF / XML
- Turtle (tortue)
- Notation 3

Jena est une API Java pour les applications Web sémantique. Le paquet-clé RDF pour le développeur d'applications est *com.hp.hpl.jena.rdf.model*. L'API a été définie en termes d'interfaces donc le code de l'application peut fonctionner avec différentes implémentations sans changement. Ce paquet contient des interfaces pour représenter des modèles, des ressources, des propriétés, des littéraux, des déclarations et tous les autres concepts-clés de RDF, et une *ModelFactory* pour créer des modèles. Ainsi ce code d'application reste indépendant de l'implémentation, il est préférable s'il utilise des interfaces dans la mesure du possible et pas les implémentations de classes spécifiques.

Le paquet *com.hp.hpl.jena.tutorial* contient le code source du travail pour tous les exemples utilisés dans ce tutoriel.

Les paquets *com.hp.hpl...jenaimpl* contiennent l'implémentation des classes qui peuvent être communes à plusieurs implémentations. Par exemple, elles définissent les classes *ResourceImpl*, *PropertyImpl* et *LiteralImpl*, qui peuvent être utilisées directement ou dérivées par différentes implémentations. Les applications doivent rarement, voire jamais, utiliser ces classes directement. Par exemple, plutôt que de créer une nouvelle instance de *ResourceImpl*, il vaut mieux utiliser la méthode *createResource* quel que soit le modèle utilisé. Ainsi, si l'implémentation du modèle a utilisé une implémentation optimisée de *Resource*, alors aucune conversion entre les deux types ne sera nécessaire.

V.7. JSP

Le JavaServer Pages ou JSP est une technique basée sur Java qui permet aux développeurs de créer dynamiquement du code HTML, XML ou tout autre type de page web. Cette technique permet au code Java et à certaines actions prédéfinies d'être ajoutés dans un contenu statique. Depuis la version 2.0 des spécifications, la syntaxe JSP est complètement conforme au standard XML. Page JSP est une combinaison entre le code Java et balises HTML avec des balises spéciaux, chacun a des spécifications spéciales. Les JSP permettent donc d'écrire facilement des servlets, en incluant dans des balises spécifiques le code JSP au sein du fichier HTML. De cette façon, elles fournissent une technologie rapide afin de créer des pages dynamiques. De plus, les JSP sont basées sur Java côté serveur, elles possèdent toutes les caractéristiques faisant la force de Java :

- Efficace : il sépare le contenu statique et dynamique.
- Intégrer Java et HTML
- facile pour les développeurs du web.

V.8. Architecture d'application Web

L'architecture d'application web correspond à une demande d'utilisateur pour une page web et le serveur web retourné le résultat au client pour afficher la page correspondante comme la figure suivante qui représente un architecture d'application Web :

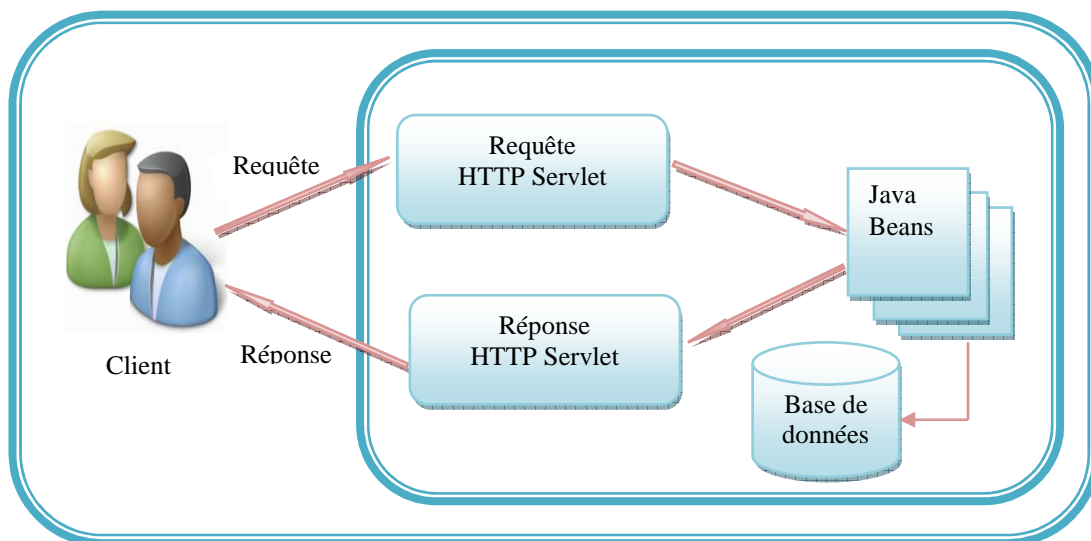


Figure V.8: Architecture application Web.

V.9. Serveur TOMCAT

Tomcat est un servlet, c'est à dire un programme tournant sur un serveur (**Apache**) qui gère d'autre servlets permettant de générer des pages web dynamiques. Il gère en particulier des servlets Java et des pages serveur, Il peut être utilisé seul ou couplé avec un serveur. Tomcat est paramétrable par des fichiers XML et de propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur http.

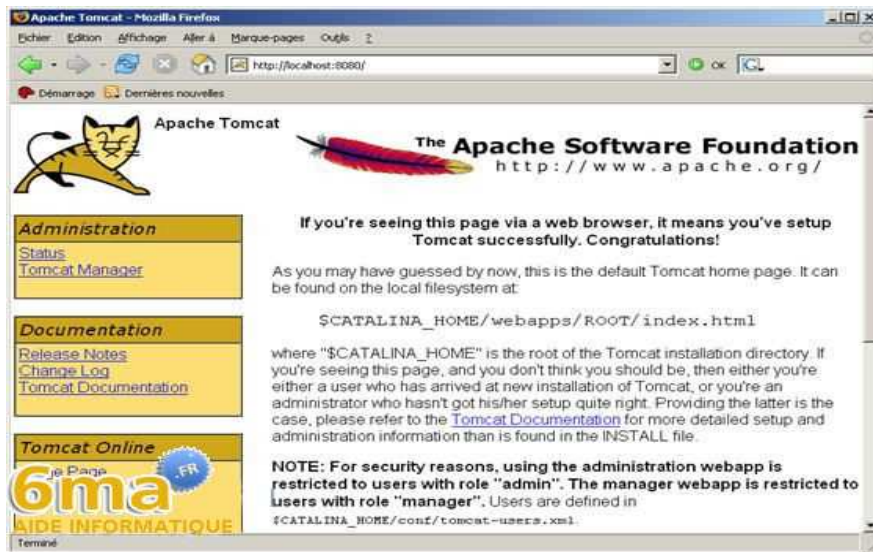


Figure V.9 : Serveur Tomcat

V.10. Principe de l'application

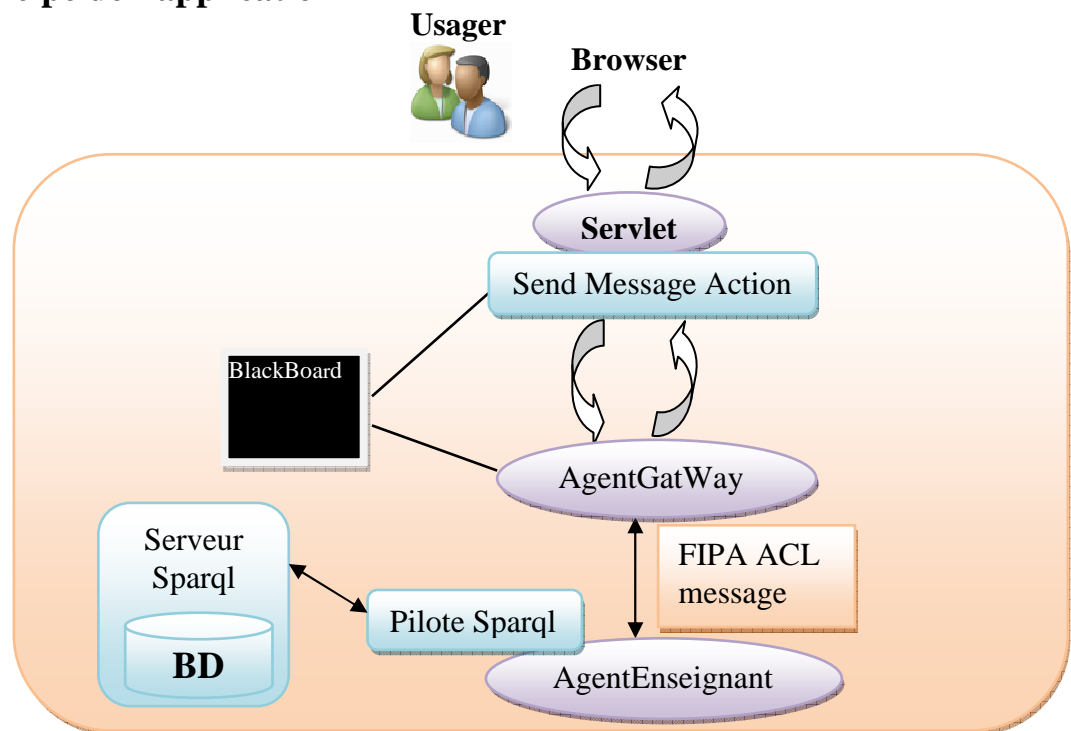


Figure V.10 : principe de l'application

1. Dans le navigateur de l'utilisateur provoque un événement et il génère un message POST.
2. La servlet reçoit les informations à partir d'une page HTML (par la méthode doPost), il va invoquer une action de type SendMessageAction.
3. L'action crée un objet BlackBoard qui sera le message canal entre l'AgentGateway et la servlet.

4. L'AgentGetWay récupère l'objet de tableau de bord créé précédemment et extraits qui en est le destinataire et quel est le message. Après, il envoie le message de type FIPA-ACL.
5. AgnetEnseignant qui est maintenant le bénéficiaire et les réponses à l'AgentGetWay.
6. A l'inverse de l'AgentGetWay quand il reçoit un message FIPA-ACL du destinataire, il va créer un objet et le pose dans le tableau noir puis il informe la servlet par la méthode releaseCommend.
7. On utilise Jena et Sparql pour extraire des données et d'écrire des données sur des graphes RDF.

V.11. Quelques ontologies de notre système

Puisque notre système est basé sur le Web sémantique et les ontologies, nous avons développé les ontologies (hiérarchie des classes, propriétés et les relations).

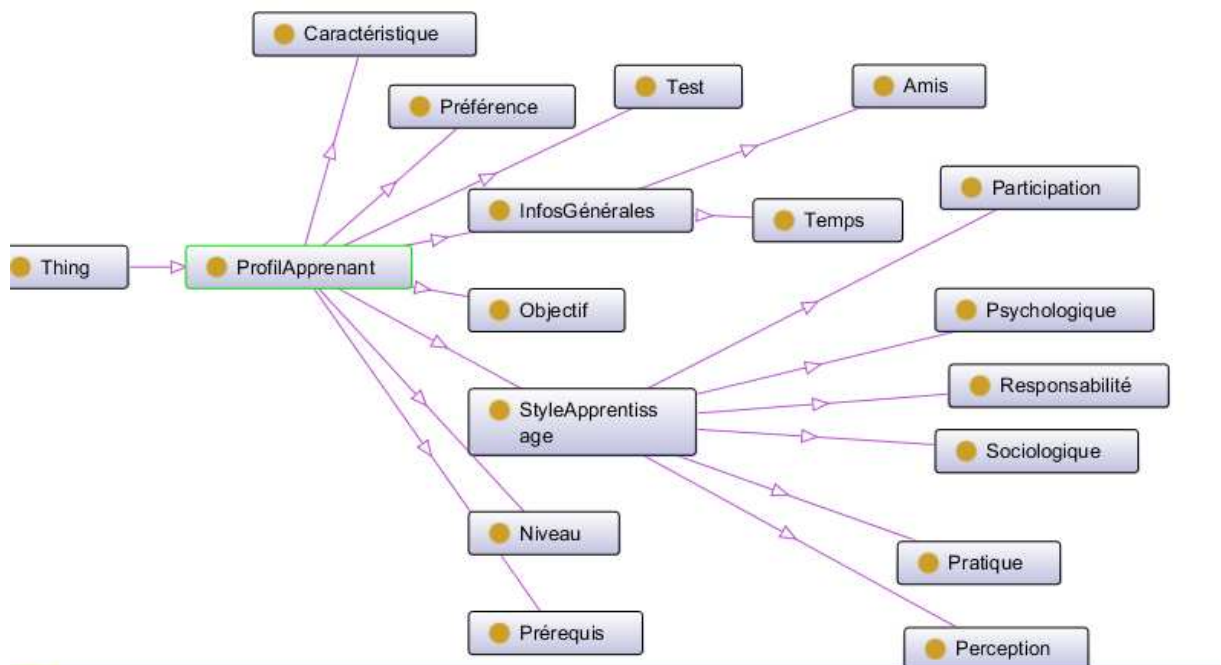


Figure V.11.a : ontologie Profil apprenant

Protégé offre la possibilité d'avoir le code de cette ontologie en langage XML-RDF comme suit :

```

<?xml version="1.0"?>
<!DOCTYPE Ontology [
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY xml "http://www.w3.org/XML/1998/namespace" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
]>
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-15"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  
```

```

xmlns:xml="http://www.w3.org/XML/1998/namespace"
ontologyIRI="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-15">
<Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#" />
<Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema#" />
<Prefix name="xsd" IRI="http://www.w3.org/2001/XMLSchema#" />
<Prefix name="owl" IRI="http://www.w3.org/2002/07/owl#" />
<Declaration> <Class IRI="#ProfilApprenant"/> </Declaration>
<Declaration> <DataProperty IRI="#Identificateur"/> </Declaration>
<DataPropertyRange>
  <DataProperty IRI="#Identificateur"/>
  <Datatype abbreviatedIRI="xsd:int"/>
</DataPropertyRange>
</Ontology>
<!-- Generated by the OWL API (version 4.3) http://owlapi.sourceforge.net -->

```

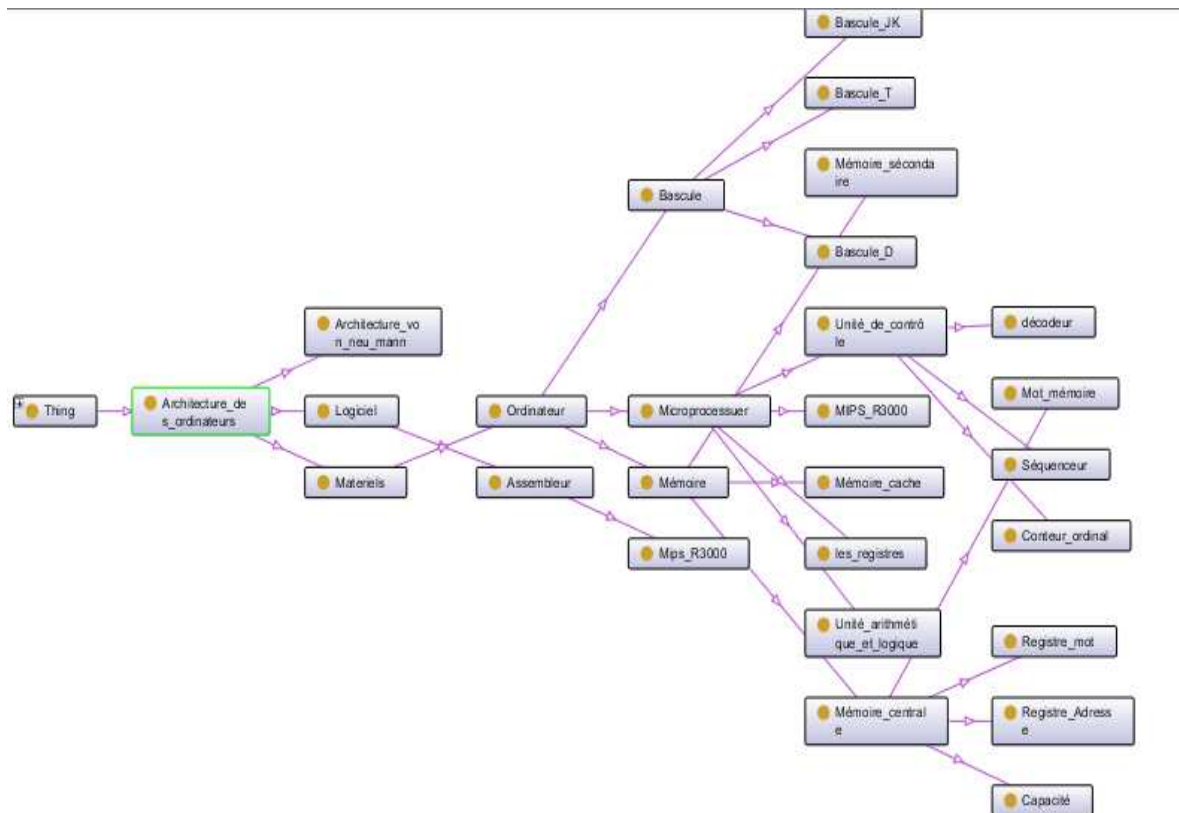


Figure V.11.b : l'ontologie de domaine (vocabulaire)

Protégé offre la possibilité d'avoir le code de cette ontologie en langage OWL comme suit :

```

<!--
// Classes
-->
<!-- http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-
10#Architecture_des_ordinateurs -->
<owl:Class rdf:about="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-
10#Architecture_des_ordinateurs"/>
<!-- http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-
10#Architecture_von_neu_mann -->

```

```

<owl:Class rdf:about="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Architecture_von_neu_mann">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Architecture_des_ordinateurs"/>
</owl:Class>
<!-- http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Assembleur -->
<owl:Class rdf:about="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Assembleur">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Logiciel"/>
</owl:Class>
<!-- http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Bascule -->
<owl:Class rdf:about="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Bascule">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/phd/ontologies/2014/2/untitled-ontology-10#Ordinateur"/>

```

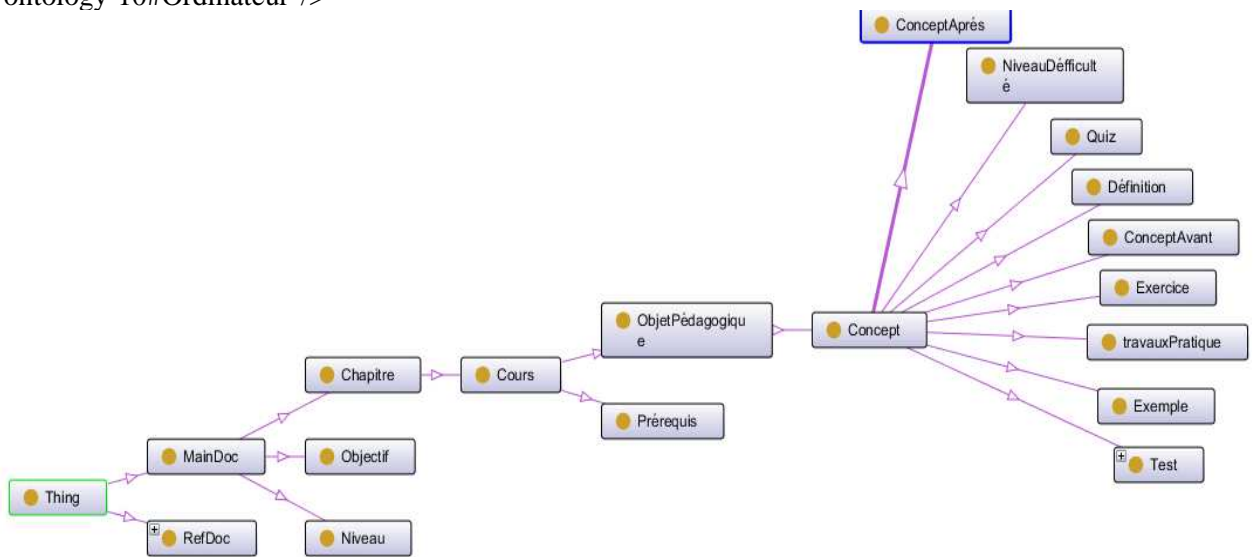


Figure V.11.c : Organisation de MainDoc (Module)

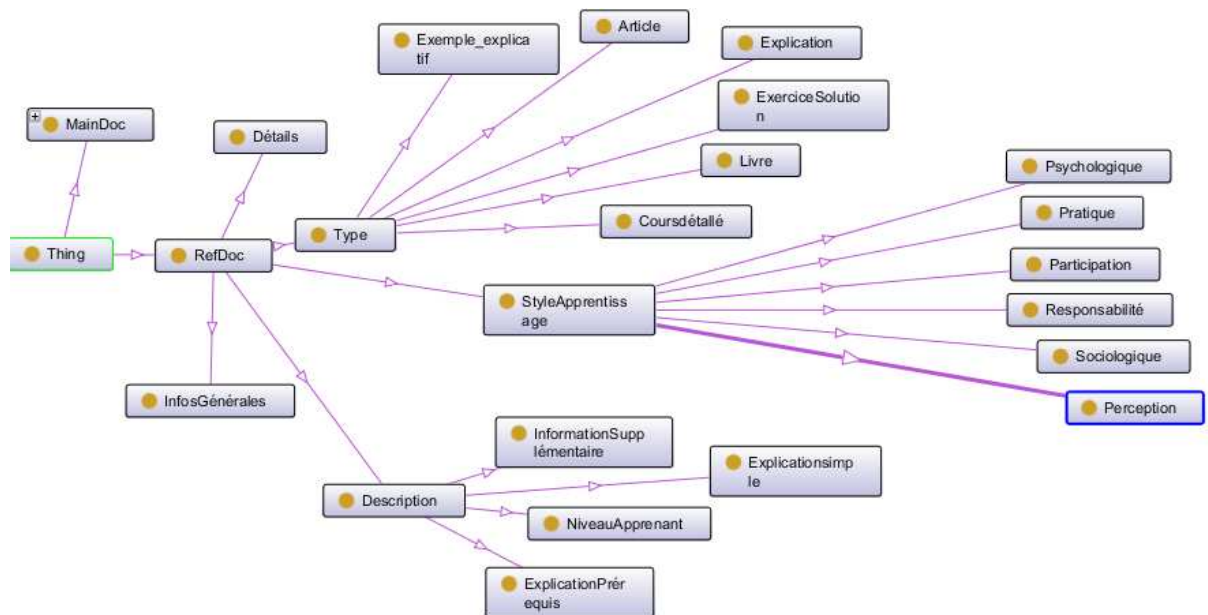
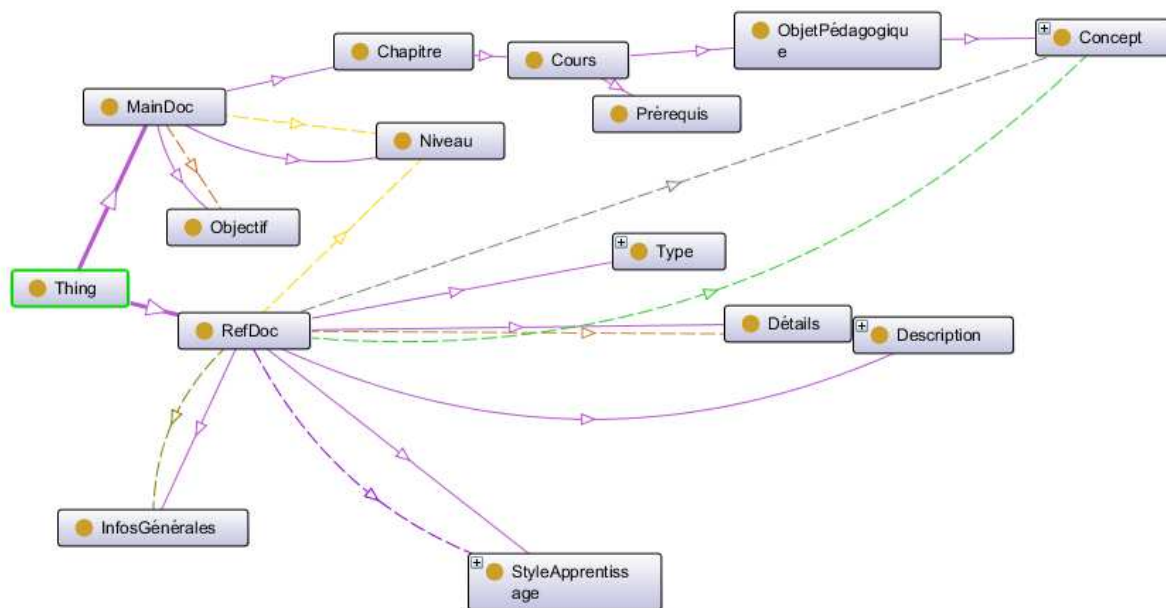


Figure V.11.d : Ontologie de RefDoc



FigureV.11.e : Ontologie RefDoc et MainDoc et leurs relations

Nous devons ajouter les propriétés de chaque élément de l'ontologie à titre d'exemple l'identificateur, le nom, la date de création ...etc, comme suit :

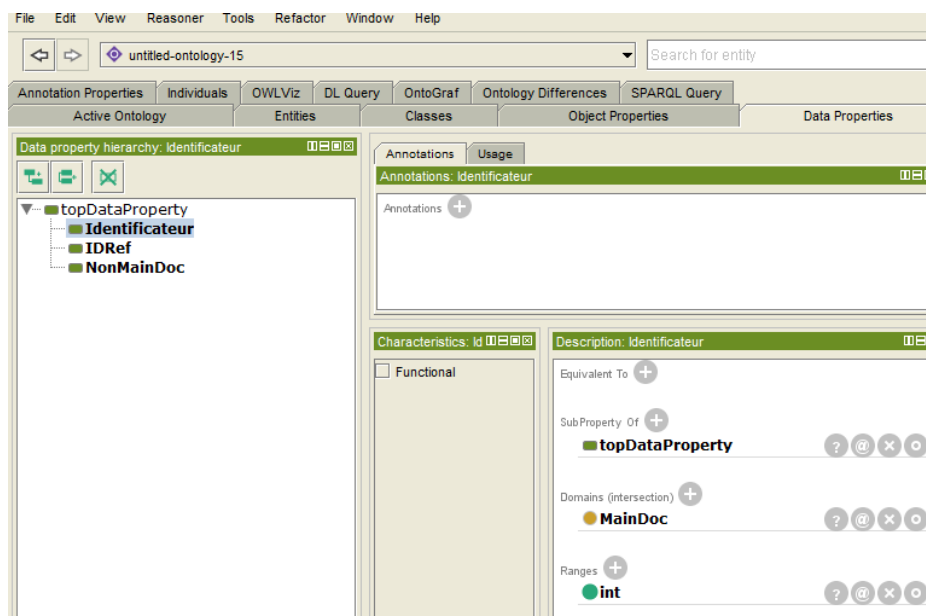
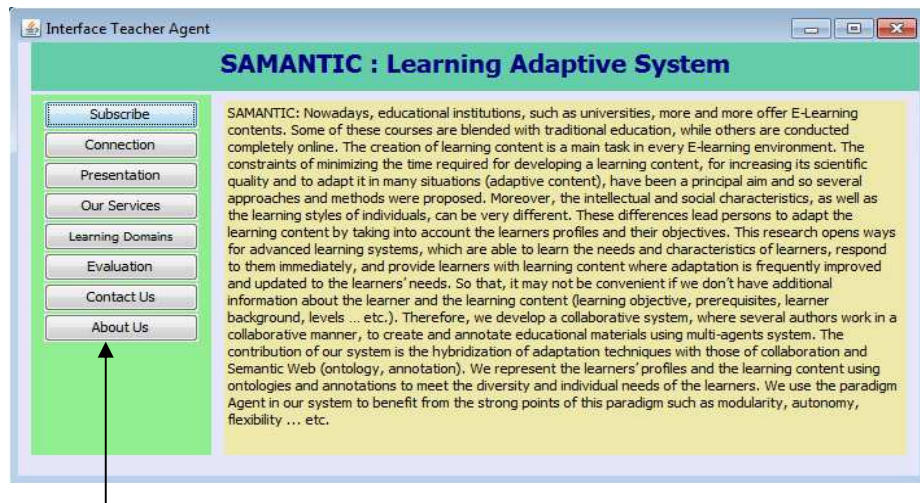


Figure V.11.f : exemple de propriété de classe d'une ontologie

V.12. Présentation de quelques fenêtre de notre système

Notre architecture conçue est une plateforme de formation à distance. Elle utilise la technologie des agents, par conséquent il faut installer sur chacun des postes le JDK (Java Developer Kit) et les classes JADE. En effet, ces classes permettent de gérer les agents. Notre outil se base sur les agents avec Gui (Graphical User Interface).

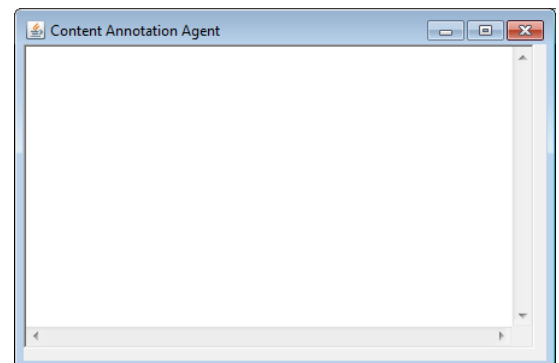
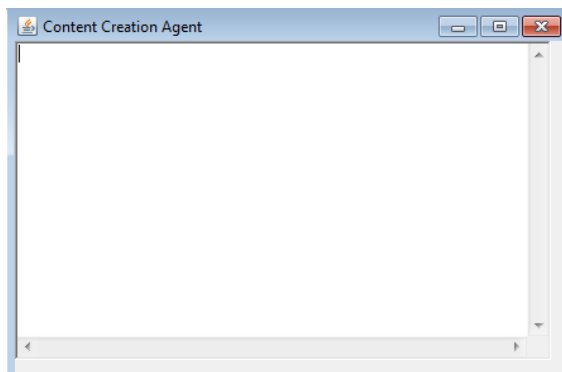
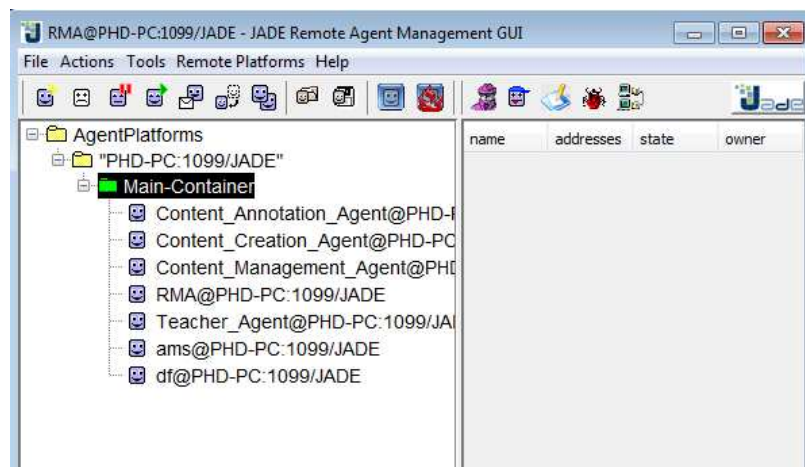
Dans notre prototype, l'interface principale est *Teacher Interface Agent* et *Teacher Agent* :



La zone de choix
(Fonctionnalités possibles)

Figure V.12.a : fenêtre principale de notre application

Ils existent d'autres fenêtres (les interfaces des agents de notre plateforme "GUI Agent"). Chaque agent a une fonctionnalité comme elle a été définie dans le chapitre de la conception. Ces Agent qui interviennent à la création des MainDoc et RefDoc.



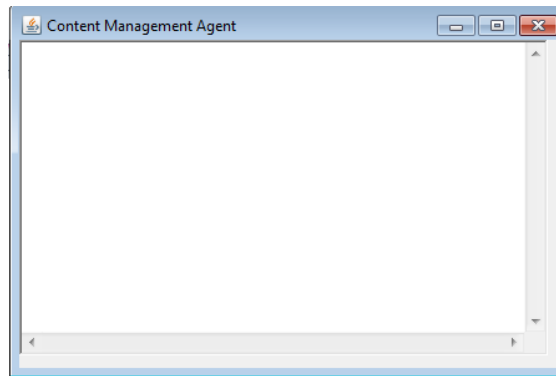


Figure V.12.b : les agents de notre système

Pour l’inscription des nouveaux enseignants l’Agent interface Enseignant affiche les zones d’inscription après la clique sur le bouton « subscribe » :

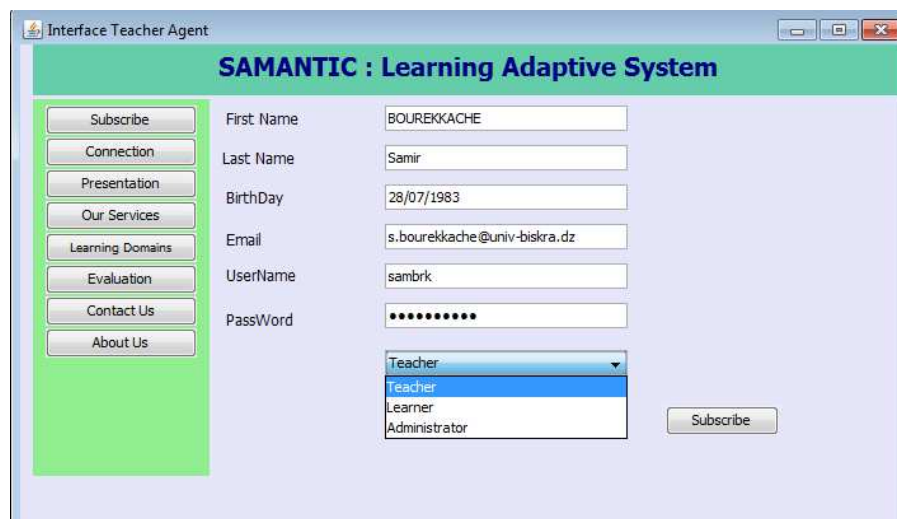


Figure V.12.c : fenêtre d’inscription de l’enseignant

Après la clique sur « subscribe », l’enseignant doit fournir d’autres informations personnelles et professionnelles qui aident à l’affectation de module (sous-domaine) à enseigner. Donc, l’enseignant peut connecter à notre système à travers la fenêtre suivante :

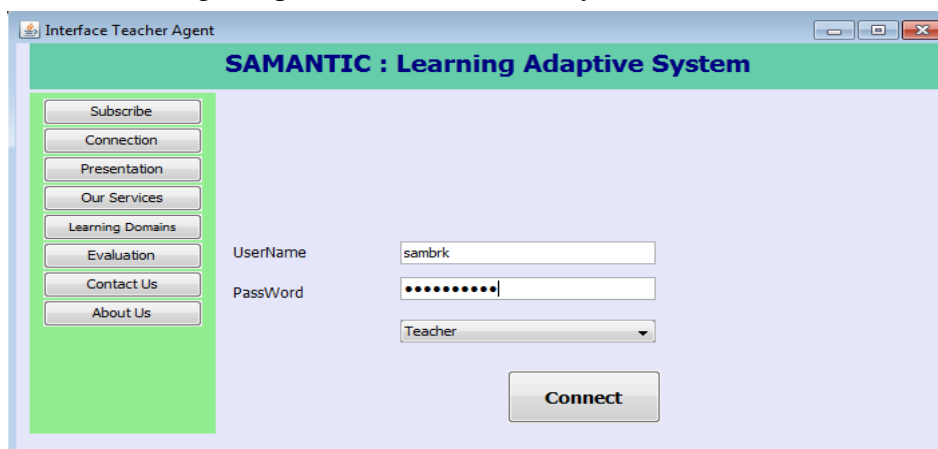
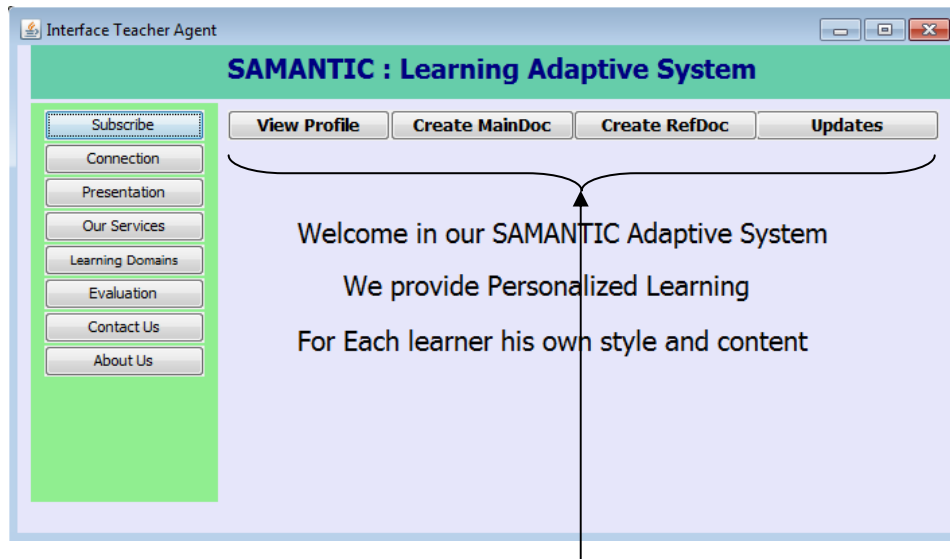


Figure V.12.d : fenêtre de connexion de l’enseignant

L’enseignant remplit les champs (Username, Password) puis clique sur « Connect ». Un espace de travail s’affiche pour aider l’enseignant dans les fonctionnalités possibles pour lui.



Les fonctionnalités de l’enseignant

Figure V.12.e : fenêtre de session de l’enseignant

Les fonctionnalités des enseignants sont :

- **View Profile** : Pour consulter le profil et modifier et ajouter ses informations.
- **Create MainDoc** : Pour voir les MainDoc ou commence un processus de modification.
- **Create RefDoc** : Pour voir les RefDoc ou commence un processus de Création.
- **Updates** : Pour consulter tous les mises à jour dans le système (modification de contenu, message d’un autre enseignant ou un apprenant...etc)

La fenêtre suivante présente le processus de création d’un RefDoc :

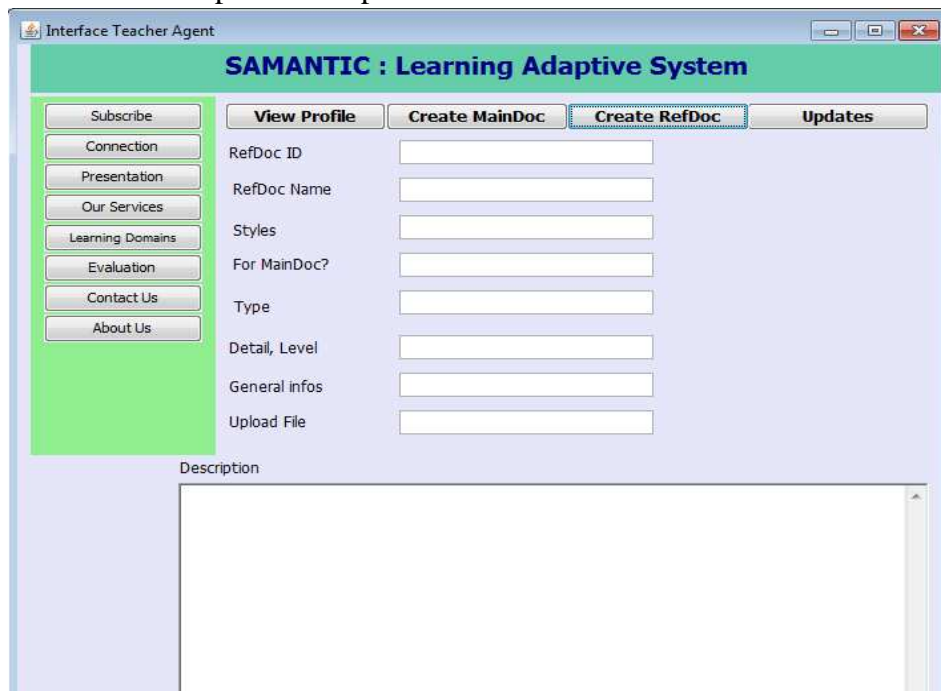


Figure V.12.f : fenêtre de création de RefDoc

La vérification de la circulation des messages entre les agents de notre plateforme, est faite grâce à l'outil graphique de JADE. Cet outil permet la visualisation des messages échangés dans le système. Il s'agit de l'agent "sniffer" qui permet de garder la trace des messages dans la plate-forme et de donner une interface graphique pour afficher les échanges des messages entre les différents groupes d'agents. La Fenêtre suivante présente les messages envoyés entre les agents pour la création d'un RefDoc (diagramme de séquence).

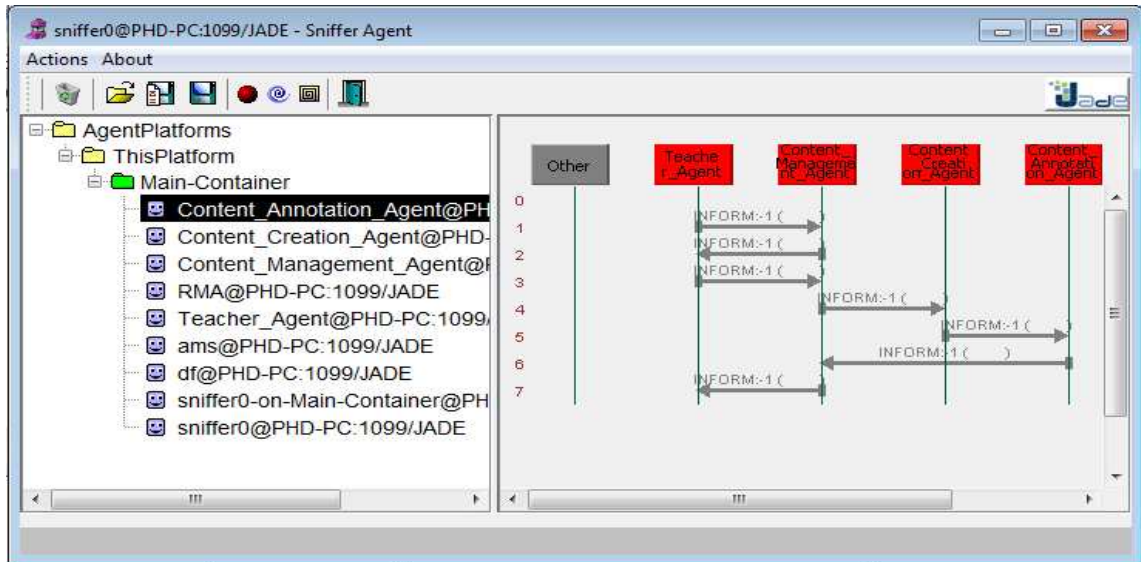


Figure V.12.g : le diagramme de séquence (création RefDoc) en utilisant l'agent Sniffer

Pour confirmer qu'un message soit bien envoyé vers la destinataire et que le contenu du message soit correct on clique sur la flèche indiquant ce message dans la fenêtre de l'agent sniffer, donc une nouvelle fenêtre s'affiche (la figure suivante):

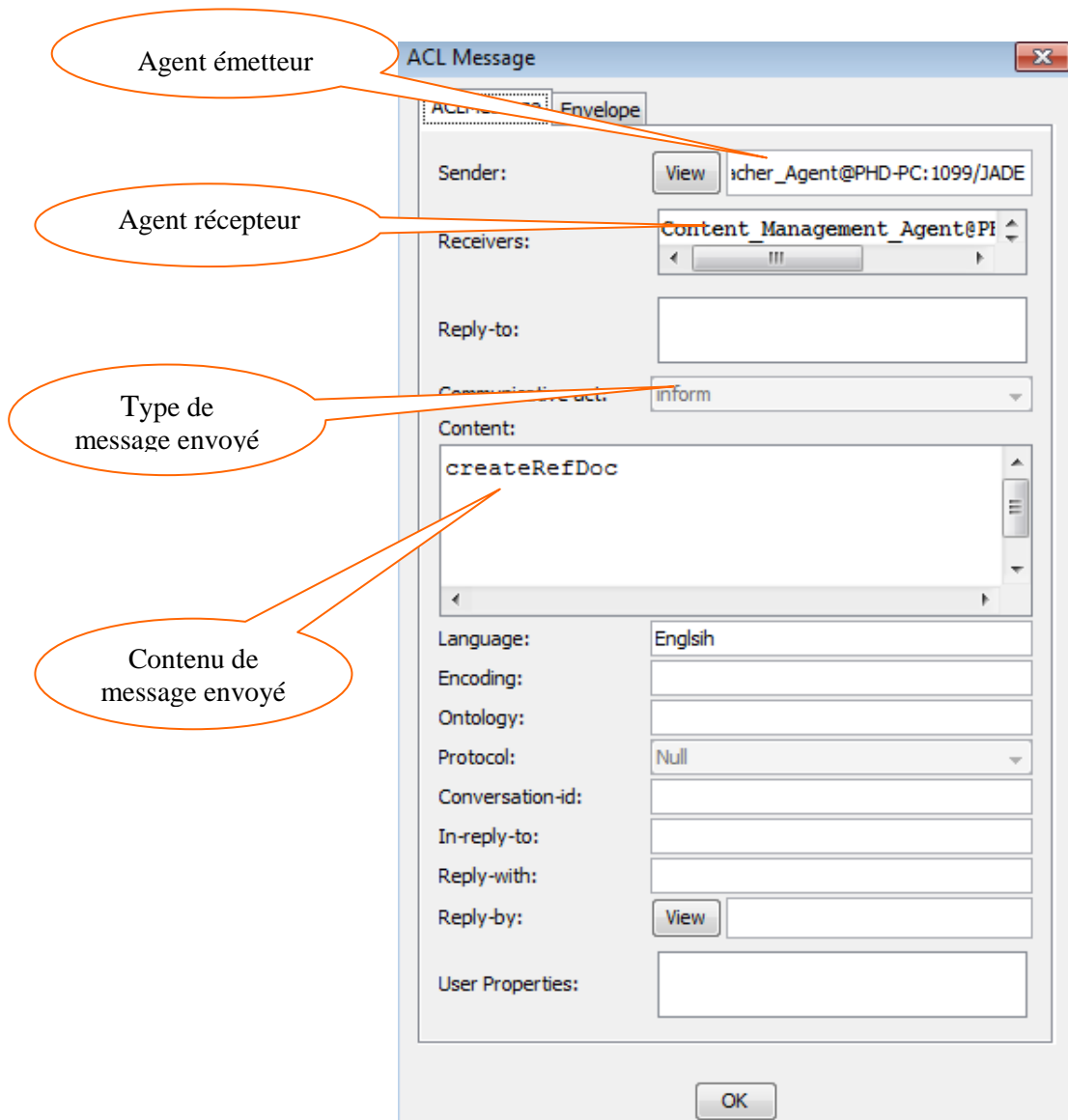


Figure V.12.h : le contenu de message envoyé entre agents, en utilisant l’agent Sniffer

V.13. Conclusion

Durant ce chapitre, nous avons présenté l'architecture logicielle de notre système de formation à distance et ses différents composants et les différents outils utilisés pour son développement. Nous avons essayé de mettre en œuvre l'ensemble des idées qui caractérise l'architecture proposée en se concentrant sur l'implémentation des agents et le Web sémantique.

Protégé OWL est un outil efficace pour le développement des ontologies. Il offre plusieurs caractéristiques pour la représentation sémantiques des informations. Le langage java s'avère pertinent pour l'implémentation de notre plateforme multi-agents. JADE est un outil pour le développement des agents. Jena est une API qui facilite le développement des environnements sémantiques (Web sémantique et ontologie) en utilisant java.

Conclusion Générale

Dans cette thèse, nous avons présenté un environnement sémantique à base d'agents pour la formation à distance (E-Learning). L'objectif principal est de créer, de façon collaborative, les documents d'apprentissage ; ces documents vont être annotés sémantiquement. On s'intéresse à la construction des ontologies pour ajouter une couche sémantique et l'ensemble des relations aux contenus éducatifs. Cette ontologie facilite l'utilisation de contenu éducatif pour offrir un apprentissage personnalisé pour chaque individu (apprenant).

L'utilisation d'Internet dans le domaine d'enseignement a influencé sur les secteurs de l'éducation puisqu'il offre des avantages dont celui notamment de faciliter l'enseignement à distance ou l'E-Learning. L'E-Learning implique un changement de rôle du formateur car la connaissance est disponible en dehors de lui : il devient un médiateur. Le E-Learning implique aussi l'autonomie de l'apprenant qui est une conséquence de son isolement et de l'absence du contact direct avec l'enseignant ou les autres apprenants. Donc, il est nécessaire de gérer les ressources éducatives (documents pédagogiques) afin d'augmenter leur qualité et de fournir des documents répondant aux besoins des apprenants hétérogènes. Ainsi que, le système d'apprentissage doit gérer les interactions entre les apprenants, les rythmes et les styles d'acquisition des savoirs. Cependant, la création des documents pédagogiques est considérée comme une tâche difficile. Donc nous avons proposé une méthode collaborative pour créer ces documents, et pour les annoter en utilisant le web sémantique et les ontologies. L'utilisation des ontologies offre une technique solide et adéquate pour représenter les contenus éducatifs car elles facilitent le processus d'adaptation de contenu et de stratégies et la réutilisation de ces documents. Nous avons créé deux types de contenu éducatif : 1) le document principal qui est créé par un ensemble d'auteurs (enseignants); ce document est le support principal de cours pour tous les apprenants. 2) Le document de référence qui est utilisé dans des cas particuliers d'adaptation. Nous avons construit une ontologie de domaine (vocabulaire) pour unifier les termes du document principal et une ontologie pour les documents de références et les relations entre eux et les ensembles des synonymes utilisés dans ce type de document.

Il est impossible de fournir le même document éducatif par le même style à tous les apprenants qui sont de nature hétérogène (social, psychologique, intellectuelle, préférences, objectifs ...etc). Donc, l'adaptation de contenu et de stratégies selon les besoins et les préférences des apprenants à tout moment est indispensable pour pouvoir les motiver et les guider et ainsi éviter l'échec ou l'abandon de ces apprenants.

Le paradigme agent s'avère pertinent dans les applications sur le web tel que la FAD à cause de ses points forts (flexibilité, modularité, l'autonomie, ...etc.). Pour l'implémentation de notre système nous avons utilisé le langage de programmation JAVA et la plateforme JADE pour l'implémentation des agents et Protégé OWL pour les ontologies.

Comme bilan de notre travail, nous citons :

- L'utilisation des ontologies dans la phase de la représentation des profils des apprenants, des styles d'apprentissage et de la création des documents éducatifs ceci offre des avantages prometteurs à exploiter facilement et efficacement ces documents par la machine et aide à la phase d'adaptation et réutilisation.
- La technologie multi-agents est l'une des points fort dans les systèmes réseaux car elle fournit un haut degré de l'autonomie, la modularité et elle est adéquate dans les cas des environnements ouverts.
- Une plateforme de formation à distance qui répond aux besoins des apprenants et assure le bon guidage durant la formation, et garde l'apprenant toujours motivé.
- L'adaptation du cours au profil de l'apprenant qui permet la satisfaction des besoins de plusieurs apprenants qui ont des niveaux et des préférences et des backgrounds différents.
- L'adaptation de stratégie d'apprentissage offre une chance aux apprenants d'apprendre par le style adéquat pour chaque apprenant.
- L'utilisation des pré-requis (les concepts obligatoires avant de commencer le cours). Ces prérequis sont utilisés pour la préparation des apprenants pour entrer dans une formation sur un cours.
- Les questions des apprenants donnent une vue globale sur les points qui ne sont pas claires à l'apprenant, l'objectif des questions est aussi de donner des explications additionnelles.
- Les tests sont des carrefours pour le guidage de l'apprenant, à partir des tests on peut décider le type d'adaptation adéquate.
- Le système est basé sur un tuteur logiciel et sur l'aide d'un tuteur humain.

Avec la notion de mobiquité (mobilité et ubiquité) une perspective devient aussi incontournable et concerne le M-Learning et ceci soit sur le web classique ou via le Cloud Computing. Nous envisageons d'enrichir et d'étendre la plateforme par d'autres fonctionnalités telles que les fonctionnalités de l'interface et l'intégration des formations ayants des projets de fin d'étude où l'apprenant aura un projet dans un domaine spécifique et il sera sous la supervision d'un encadreur à distance.

Références de l'auteur

- [121] Bourekkache S., Kazar O., “*Agent-Based Approach for E-Learning*”. International Journal of Emerging Technologies in Learning (iJET), Vol 4, No 4, 2009.
- [122] Bourekkache S., Kazar O., Benharkat N., Kahloul L., “*A cooperative multi-agent approach for the creation and annotation of adaptive content for e-learning*”, Journal of e-Learning and Knowledge Society (Je-LKS), Vol 10, No 1, 2014.
- [123] Kahloul L., Bourekkache S., Kazar O., “*E-Learning “Java Programming” on Facebook!!! Big Ambitions and Important Challenges*”, International Arab Journal of e-Technology (IAJeT), Vol 3, No 3, 2014.

Bibliographie

- [1] Tatyana I., “*Adaptive Open Corpus E-Learning and Authoring, Using Collaborative Ontology Learning*”, ICETA 2011, 9th IEEE International Conference on Emerging eLearning Technologies and Applications, Stará Lesná, Slovaquie, 2011.
- [2] Hyun-Sook Ch., Jung-Min K., “*Ontology Design for Creating Adaptive Learning Path in e-Learning Environment*”, Proceedings of the International Multi-Conference of Engineers and Computer Scientists, Hong Kong, 2012.
- [3] Sylvain D., “*Exploiting Semantic Web and Knowledge Management Technologies for E-learning*”, thèse pour obtenir le titre de Docteur en Sciences, université de Nice-Sophia Antipolis, soutenue le 2 février 2007.
- [4] Nassim K., Nafise S., Razie M., “*Ontology-based e-learning*”, IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 2, No.4, August 2012.
- [5] Mahdi H., Attia S., “*Developing and Implementing a Multi-Agent System for Collaborative E-learning*”, Dans : E-learning, Maja Jakobovic (eds.), ISBN : 978-953-7619 95-4, InTech. 2010.
- [6] Cubric M., Tripathi M., “*A Framework for Creating Semantically Adaptive Collaborative E-learning Environments*”, International journal of emerging technologies in Learning, Vol 4, No 3, 2009.
- [7] Bremgartner V., de Magalhães Netto J. F., “*Improving Collaborative Learning by Personalization in Virtual Learning Environments Using Agents and Competency-Based Ontology*”, Frontiers in Education Conference (FIE), IEEE, 2012.
- [8] Bouquet P., Molinari A., “*Semantic Technologies and e-Learning: towards an entity-centric approach for Learning Management Systems*”, Journal of e-Learning and knowledge society Je-LKS, Vol 8, N2, 2012.
- [9] Guillermo Valente G., “*Enrichissement de requêtes et visualisation sémantique dans une coopération de systèmes d'information : méthodes et outils d'aide à la recherche d'information*”, Thèse Pour l'obtention du titre de Docteur de l'Université de Bourgogne, Soutenue le 14 Décembre 2010.
- [10] Mazyad H., “*Une Approche Multi-agents à Architecture P2P pour l'Apprentissage Collaboratif*”, Thèse Pour l'obtention du titre de Docteur de l'Université du Littoral Côte d'Opale, Soutenue le 31 janvier 2013.
- [11] Sabine G., “*Adaptivity in Learning Management Systems Focussing on Learning Styles*”, Thèse Pour l'obtention du titre de Docteur de l'université de technologie, Vienna, Décembre 2007
- [12] Siddhartha K., A., Devshri R., “*Learning Content Management Using Machine Learning*”, International Journal of Information and Education Technology, Vol. 2, No. 5, October 2012.
- [13] Boyce S., Pahl C., “*Developping domain ontologies for course content*”, Educational Technology & Society, vol. 10, pp. 275-288, 2007.
- [14] Yarandi M., Tawil Ar., Jahankhani H., “*Adaptive E-Learning System Using Ontology*”, 22nd International Workshop on Database and Expert Systems Applications (IEEE), 2011.
- [15] Kabel S., de Hoog R., Wielinga B., Anjewierden., “*Indexing Learning Objects: Vocabularies and Empirical Investigation of Consistency*”, Journal of Educational Multimedia and Hypermedia, 13(4),pp. 405-425, Norfolk, VA: ACE, 2004.

- [16] Hammache S., Ahmed-Ouamer R., “*Système d'Inférence pour une Indexation de Documents Basée sur une Ontologie de Domaine.*”, In Proceedings of INFORSID, pp. 895-910, 2006.
- [17] Benayache A., “*Construction d'une mémoire organisationnelle de formation et évaluation dans un contexte e-Learning*”, Thèse de Doctorat, l'université de technologie de Compiègne, le 5 décembre 2005.
- [18] Ferber J., “*Les Systèmes Multi-Agents Vers une intelligence collective*”, livre, 1995.
- [19] Russel S., Norvig P., “*Artificial Intelligence: A Modern Approach*”, Prentice-Hall, 1995.
- [20] Jennings N., Wooldridge M., Sycara K., “*A roadmap of agent research and development*”, Journal of Autonomous Agents and Multi-Agent Systems, 1998.
- [21] RuiMin S., YiYang T., TongZhen Z., “*The intelligent assessment system in web-based distance learning education*”, Reno, NV. 31th ASEE/IEEE Frontiers in Education Conference T1F-7, October 10 - 13, 2001.
- [22] Bayne, S. and Cook, J.(2006). “*WebCT vs BlackBoard? An Evaluation of Two Virtual Learning Environments*”, 2006.
- [23] Oye N.D., Mazleena Salleh., Iahad N., “*E-Learning Methodologies and Tools*”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No.2, 2012.
- [24] Yücel Y., “*Knowledge management in E-learning practices*”, TOJET: The Turkish Online Journal of Educational Technology , volume 11 Issue 2, April 2012.
- [25] Neepa K. Shah., “*E-Learning and Semantic Web*”, International Journal of e-Education, e-Business, e-Management and e-Learning, Vol. 2, No. 2, April 2012
- [26] Ljiljana Stojanovic., Steffen Staab., Rudi Studer., “*eLearning based on the Semantic Web*”, In WebNet2001-World Conference on the WWW and Internet, Orlando, Florida, USA, 2001.
- [27] Drucker P., “*Need to Know: Integrating e-Learning with High Velocity Value Chains*”, A Delphi Group White, 2000.
- [28] Neeraj Kumar Chaurasia., Ganesh Kumar., “*E-Education : An Online Perspective*”, 2nd International CALIBER-2004, New Delhi, 11-13 February, 2004.
- [29] Seyed Yahya Seyed Danesh., Shahram Hashemini., “*Evaluation of Effective Factors on Electronic Learning and Satisfying Learners in Virtual Universities of Tehran*”, INTERDISCIPLINARY JOURNAL OF CONTEMPORARY RESEARCH IN BUSINESS, Institute of Interdisciplinary Business Research, Vol 3, No 9, JANUARY 2012.
- [30] Olojo Oludare Jethro., Adewumi Moradeke Grace., Ajisola Kolawole Thomas., “*E-Learning and Its Effects on Teaching and Learning in a Global Age*”, International Journal of Academic Research in Business and Social Sciences, Vol. 2, No. 1., January 2012.
- [31] Saleh N., “*What is e-Learning? In Information literacy within a Quality Assurance Framework: An Ideas Exchange for Librarians*”, 2012.
- [32] Rezaei Sharifabadi S., “*How Digital Libraries Can Support E-learning?*”, The Electronic Library , V. 24, No3. pp.389-401. 2006
- [33] Holmes B., Gardner J.R., “*E-Learning: Concepts and Practice*”. London: Sage Publications Ltd. 2006.
- [34] Rosenberg M.C., “*Beyond E-Learning: Approaches and Technologies to Enhance Organizational Knowledge, Learning, and Performance*”. San Fransisco: Pfeiffer. 2005.

- [35] Learning Technology Standards Committee of the IEEE, Draft Standard for Learning Object Metadata, IEEE 1484.12.1-2002, 15 July 2002, IEEE Standards Department, Copyright and Permissions, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA
- [36] Marie-Chantal Dufour., “*Présentation de la norme internationale ISO/IEC 19788 – Métadonnées pour les ressources d'apprentissage, Ressources d'enseignement et d'apprentissage*”, Atelier MLR (partie 1) (v1.1), 2011.
- [37] OUBAHSSI L., “*Conception de plates-formes logicielles pour la formation à distance, présentant des propriétés d'adaptabilité à différentes catégories d'utilisateurs et d'interopérabilité avec d'autres environnements logiciels*”, Thèse de doctorat, Université René Descartes – Paris V, Soutenue le 01/12/2005.
- [38] Gathany Nancy., “*Expert Instructional Designers' Views of the Impact of Accessibility Requirements on E-learning Instructional Strategies*”, 2012.
- [39] Dossou Anani Koffi DOGBE-SEMANOU., Anne Durand., “*Etude comparative de plates-formes de formation à distance*”, dans le cadre du Projet @2L, avril 2008.
- [40] Bardini T., Proulx S., “*Les promesses du cyberspace : méditations, pratiques et pouvoirs à l'heure de la communication électronique*”, Sociologie et sociétés.2000.
- [41] Keita A., “*Conception Coopérative d'Ontologies Pré-Consensuelles: Application au domaine de l'Urbanisme*”. Thèse pour l'obtenir le grade de docteur, INSA de Lyon, France, Soutenue le 06 Juin 2007.
- [42] Connolly D., Harmelen F. V., Horrocks I., McGuinness D. L., Patel-Schneider, P. F., Stein L. A., “*DAML+OIL : Reference Description. Rapport technique*” W3C : World Wide Web Consortium, 2001.
- [43] Zghal S., “*Contributions à l'alignement d'ontologies OWL par agrégation de similarités*”, Thèse pour l'obtenir le grade de docteur, faculté des science Tunis, Soutenu le 21 décembre 2010.
- [44] Bachimont B., “*Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances*”. Ingénierie des connaissances : évolutions récentes et nouveaux défis, pages 305–323, 2000.
- [45] Vandecasteele A., “*Modélisation ontologique des connaissances expertes pour l'analyse de comportements à risque - Application à la surveillance maritime -*”, thèse pour obteneur le grade de docteur délivré par l'école nationale supérieur des mines de Paris, France, soutenue le 30 octobre 2012.
- [46] De Nicola A., Missikoff M., Navigli R., “*A software engineering approach to ontology building*”. Information Systems 34, 258–275. 2009.
- [47] Staab S., Studer R., Schnurr H.P., Sure Y., “*Knowledge processes and ontologies*”. Intelligent Systems, IEEE 16, 26 – 34.2001.
- [48] Corcho O., “*Methodologies, tools and languages for building ontologies. Where is their meeting point?*” Data & Knowledge Engineering 46, 41–64. 2003.
- [49] Horrocks I., Patel-Schneider P. F., van Harmelen F., “*From SHIQ and RDF to OWL : The making of a web ontology langage*”. Journal of Web Semantics, 1(1):7-26. 2003.
- [50] <http://www.educnet.education.fr/superieur/glossaire.htm>
- [51] Rheaume J., “*Apprivoiser la technologie éducative*”, Québec, Université Laval. <http://www.fse.ulaval.ca/mediatic>. 2001.
- [52] Nipper S., “*Third generation distance learning and computer conferencing*”, In R. Mason and A. Kaye (Eds.), Mindweave: Communication, computers and distance education, Permagon, Oxford, 1989.

- [53] Laskri Mohamed T., Beggas M., “*Modélisation basée agents d'un hypermédia adaptatif dynamique*”. Laboratoire GRIA/LRI, Département d'informatique, Université Baji Mokhtar Annaba, Algérie.
- [54] Laskri Mohamed T., Bouhadada T., “*Design of interactive pedagogical activities for distance learning environment*”. Laboratoire GRIA/LRI, Département d'informatique, Université Baji Mokhtar Annaba, Algérie.
- [55] Després., “*Modélisation et Conception d'un Environnement de Suivi Pédagogique Synchrones d'Activités d'Apprentissage à Distance*”. Thèse de l'Université du Maine, Le Mans, France. 2001.
- [56] Mbala Hikolo A., “*Analyse, conception, spécification et développement d'un système multiagents pour le soutien des activités en formation à distance*”. Thèse de Doctorat en informatique, Université de Franche-Comté. 2003
- [57] ELFE. Rapport de la 1ère Conférence du Forum eLearning européen pour l'éducation (ELFE). Bruxelles, les 22-23 novembre 2004.
- [58] Roisin C., “*Documents structurés multimédia*”, Habilitation à diriger les recherches Institut National Polytechnique de Grenoble, septembre 1999.
- [59] Bourda Y., Helier M., “*Métadonnées et XML : Application aux objets pédagogiques*”, Conférence TICE 2000, Troyes, octobre 2000.
- [60] Taylor J., Swannel., “*Internet based ODL: Initiatives Where From, Where Now, Where To*”, Texte présenté à la rencontre ICDE SCOP Meeting. 1997.
- [62] “*L'enseignement à distance en mutation: Diagnostic et perspectives en Communauté française de Belgique*”, Recherche en éducation n° 92/01, Université de Mons Hainaut. Août 2003.
- [63] Le groupe de développement IUFM. “*Formation à distance et nouvelles modalités de formation*”. France. 2002.
- [64] Delestre N., Pécuchet J., Gréboval C., “*L'architecture d'un hypermédia adaptatif dynamique pour l'enseignement*”. PSI LIRINSA, INSA de Rouen, 1998.
- [65] Brusilovsky P., “*Methods and techniques of adaptive hypermedia, In User Modeling and User Adapted Interaction*” (6). 2001.
- [66] Edmonds E., “*Adaptive Man-Computer Interfaces*”. In computer and People series, Academic Press, 1981.
- [67] Bursilovsky P., “*Methods and Techniques of Adaptive Hypermedia, In User Modeling and User-Adapted Interaction*” 6 (2-3), pp 87-129, Kluwer Academic Publishers, 1996.
- [68] Dorothée R., Coffinet, S., “*Un environnement support à l'appropriation d'une formation à distance par l'apprenant*”. Thèse de doctorat de l'Université du Maine. Soutenue le 5 novembre 2004
- [69] Christian F., “*Une première étude de plates-formes d'enseignement pour l'université paris 5*”. Cellule TICE – Direction des Systèmes d'Informations de l'Université Paris 5, 45 rue des saints-pères 75270 PARIS Cedex 06, France. Juillet 2005.
- [70] Thot, Nouvelles de formation à distance. Disponible en ligne sur l'adresse: <http://thot.cursus.edu/>. Date dernière consultation: 07/12/2007.
- [71] La plateforme de la formation à distance CLAROLINE. Disponible en ligne sur l'adresse: <http://www.claroline.net/>. Date dernière consultation: 03/03/2014.
- [72] La plateforme de la formation à distance GANESHA. Disponible en ligne sur l'adresse: <http://www.anemalab.org/ganisha/index.htm>. Date dernière consultation: 03/03/2014.
- [73] Glikman V., “*Formations à distance: au nom de l'usager*”, DistanceS. vol. 3, n°2. , p: 101-118. 1999.

- [74] La plateforme de la formation à distance MOODLE. Disponible en ligne sur l'adresse: <https://moodle.org/> Date dernière consultation: 03/03/2014.
- [75] La plateforme de la formation à distance SAKAI. Disponible en ligne sur l'adresse: <https://sakaiproject.org/>. Date dernière consultation: 03/03/2014.
- [76] La plateforme de la formation à distance Sakai de canada est Disponible en ligne sur l'adresse: <http://www.sakaiquebec.org/>. Date dernière consultation: 03/12/2013.
- [77] La plateforme SPIRAL de l'Université Lyon 1. Disponible en ligne sur l'adresse: <http://spiral.univ-lyon1.fr/>. Date dernière consultation: 03/03/2014.
- [78] Durfee E., Lasser V., "*Partial global planing: A coordination framework for distributed hypothesis formation*". IEEE Transactions on systems, Man and cybernetics, 21(5):1167-1183, 1991.
- [79] Labidi S., Lejouad W., "*De l'Intelligence Artificielle Distribuée aux Systèmes Multi-Agents*". Rapport de recherche n 2004. INRIA Sophia-Antipolis, France. Août 1993.
- [80] S. Deloach S., "*Analysis and design using MaSE and agentTool*". The 12th Midwest Artificial Intelligence and Cognitive Science Conference, 2001.
- [81] Chaib-draa B., I. Jarras I., Moulin B., "*Systemes multiagents : Principes generaux et applications*". Editions J.P. Briot, Y. Demazeau, Agent et systemes multiagents, Hermes. 2001.
- [82] Ferber J., "*Objets et Agents : Une étude des structures de représentation et communication en intelligence artificielle*". Thèse de doctorat, 1989.
- [83] Odell J., Van Dyke P., Bauer B., "*Extending UML for Agents*". Proceedings of the Agent-Oriented Information Systems Workshop at the 17th National conference on Artificial Intelligence, Gerd Wagner, Yves Lesperance, and Eric Yu eds., Austin, TX, pp. 3-17, 2000.
- [84] Wooldridge M., "*An Introduction to Multiagent Systems*". Editeurs: John Wiley & Sons Ltd, England, ISBN 047149691X. 2002.
- [85] Jarras I., Chaib-Draa B., "*Aperçu sur les systèmes multiagents*". Série Scientifique, Montréal, 2002.
- [86] Moulin B., Chaib-Draa B., "*An overview of distributed artificial intelligence*". Dans: G.M.P.O'Hare and N.R. Jennings, editors, Foundations of Distributed AI, pages 3-54. John Wiley & Sons: Chichester, England. 1996.
- [87] Nwana H., Lee L., Jennings N., "*Coordination in software agent systems*". BT Laboratories internal report, (1994).
- [88] Guessoum Z., "*Modèles et architectures d'agents et de systèmes multi-agents adaptatifs*". Rapport de soutenance de HDR, Université Pierre et Marie Curie. 2003.
- [89] Demazeau Y., "*From Interactions to Collective Behaviour in Agent-Based Systems*". Dans: Proceedings of the 1st European Conference on Cognitive Science. Saint-Malo, France. 1995.
- [90] Klein F., "*Contrôle d'un système multi-agents réactif par modélisation et apprentissage de sa dynamique globale*". Rapport de thèse, Université de Nancy 2, Décembre 2009.
- [91] Wooldridge M., Ciancarini P., "*Agent-oriented software engineering: the state of the art*". Editeurs: P. Ciancarini and M. Wooldridge, Agent-Oriented Software Engineering. Springer-Verlag Lecture Notes dans AI Volume, Janvier 2001.
- [92] Ferber J., "*Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*". Addison Wesley, London. 1999.
- [93] Chaib-draa B., Demolombe R., "*Dans : Information-Interaction-Intelligence*", 2002.

- [94] Durfee E., Lesser V., “*Negotiating task decomposition and allocation using partial global planning*”. Dans: Distributed Artificial Intelligence Volume II, pages 229-244. Pitman Publishing: London and Morgan Kaufmann: San Mateo, CA, 1989.
- [95] Jennings N., “*Coordination Techniques for Distributed Artificial Intelligence*”. Foundations of Distributed Artificial Intelligence, Editions G. M. P. O’Hare et N. R. Jennings, Wiley, 1996.
- [96] Ferber J., “*Les Systèmes Multi-Agents – Un aperçu général*”. Technique et science informatiques. Volume 16 – n° 8/1997.
- [97] Chaib-draa B., “Agents et systèmes multiagents”. Département d’informatique Faculté des sciences et de génie. Université Laval Québec, Canada. 1999.
- [98] Bellefemine F., Caire G., Trucco Y., “*Rimassa G., JADE programmer’s guide (JADE 3.2)*”. Rapport, TILAB S.p.A, 2004.
- [99] Genin T., “*Stratégies de formation de coalitions dans les systèmes multi-agents*”. Thèse de doctorat, Université Pierre et Marie Curie, 2010.
- [100] Gutknecht O., Ferber J., “*Madkit : une architecture de plateforme multi-agent générique*”. Rapport de recherche, Laboratoire d’Informatique, de Robotique et de Microelectronique de Montpellier (LIRM), Université de Montpellier II, 2000.
- [101] Pesty S., Webber C., Balacheff N., “*Baghera: une architecture multi-agents pour l’apprentissage humain*”. Cognitique, (P. Aniorde, S. Gouarderes eds), Cepadeus Edition, Toulouse, 2003.
- [102] Fougère D., “*Vers un système de médiation pour les systèmes coopératifs*”, Mémoire d’Habilitation à diriger des Recherches, Laboratoire M3M, Université de Technologie de Belfort-Montbéliard. 2010.
- [103] Khandaker N., Kiat L., Miller L., Jiang H., “*Lessons Learned from comprehensive deployments of multi-agent CSCL applications I-MINDS and classroom wiki*”. IEEE Transactions on Learning Technologies, Vol 4, Num 1 (TLT4(1)), pages: 47-58, Janvier 2011.
- [104] Christopher B., Kei-Hoi Ch., “*Knowledge discovery for biology with Taverna*”. Producing and consuming semantics in the Web of science. Semantic Web, chapter 16, pages 355-395. Springer-Verlag New York Inc, 2007.
- [105] Christopher B., Hei-Hoi Ch., “*Semantic Web*”. Springer, 2007.
- [106] Nagarajan M., “*Semantic annotations in Web Services*”. In Semantic Web Services, Processes and Applications. Springer, 2006
- [107] Cardoso J., Sheth A., “*The semantic web and its applications*”. In Semantic Web Services, Processes and Applications, volume 3 of Semantic Web and Beyond, pages 3-33. Springer US, 2006.
- [108] Sheth A., “*Semantic Meta Data for enterprise information integration*”. DM Review magazine, July 2003.
- [109] Neches R., Fikes R., Finin T, Gruber T., Patil R., Senator T., Swartout W., “*Enabling technology for knowledge sharing*”. AI Magazine, 12(3) :36-56, 1991.
- [110] Sowa J. F., “*Knowledge Representation: Logical, Philosophical and Computational Foundations*”. Brooks Cole Publishing Co., Pacific Grove, California. 2000.
- [111] Borst W.N., Akkermans M., Top J., “*Engineering ontologies*”. International Journal of Human-Computer Studies, vol. 46, pp. 365-406, 1997.
- [112] GRUBER T. R., “*Toward Principles for the Design of Ontologies Used for Knowledge Sharing*”, presented at International Workshop on Formal Ontology, Padova, Italy, 1993.

- [113] Van Heijst G., Schreiber Th., Wielinga B., “*Using explicit ontologies in KBS development*”. International Journal of Human-Computer Studies, 47(2-3) :183-292, 1997.
- [114] Guarino N., “*Semantic Matching : Formal ontological distinctions for information organization, extraction, and integration*”. Lecture Notes in Computer Sciences, Technology LNAI, 1299 :139-170, 1997.
- [115] Fernandez-Lopez M., Gomez A., Juristo N., “*METHONTOLOGY: From Ontological Art Towards Ontological Engineering*”, 33–40.1997.
- [116] BERNERS-LEE T., HENDLER J., LASSILA O., “*The Semantic Web*”. Scientific American Press, May 2001.
- [117] Dunn R., Dunn K., “*Learning Style as a Criterion for Placement in Alternative Programs*”, Phi Delta Kappan, 56 (4), 275-278, 1974.
- [118] Dunn R., Griggs S., “*Synthesis of the Dunn and Dunn Learning Styles Model Research: Who, What, When, Where and So What – the Dunn and Dunn Learning Styles Model and Its Theoretical Cornerstone*”. St John’s University, New York. 2003.
- [119] Grasha, A. F., Riechmann, S., “*Student Learning Styles Questionnaire*”, University of Cincinnati, Faculty Resource Center, Cincinnati, OH, 1975.
- [120] Felder R. M., Silverman, L. K., “*Learning and Teaching Styles in Engineering Education*”. Engineering Education, 78 (7), 674–681, 1988.

Références de l’auteur

- [121] Bourekkache S., Kazar O., “*Agent-Based Approach for E-Learning*”. International Journal of Emerging Technologies in Learning (iJET), Vol 4, No 4, 2009.
- [122] Bourekkache S., Kazar O., Benharkat N., Kahloul L., “*A cooperative multi-agent approach for the creation and annotation of adaptive content for e-learning*”, Journal of e-Learning and Knowledge Society (Je-LKS), Vol 10, No 1, 2014.
- [123] Kahloul L., Bourekkache S., Kazar O., “*E-Learning “Java Programming” on Facebook!!! Big Ambitions and Important Challenges*”, International Arab Journal of e-Technology (IAJeT), Vol 3, No 3, 2014.