

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Ferhat Abbas de Sétif - UFAS (Algérie)

Thèse

Présenté à la Faculté des Sciences

Département d'informatique

En vue de l'obtention du diplôme de

Doctorat en Sciences

Option : Informatique

Par

Chahrazed MEDIANI

Thème

**Modélisation des attitudes cognitives et affectives
chez l'individu en situation d'apprentissage
coopératif sur le Web**

Soutenu le : 24 Février 2013 devant la commission d'examen :

<i>Président</i>	A. Boucenna	Prof., Université de Sétif
<i>Examineurs</i>	M. Benmohamed	Prof., Université de Constantine
	A. Bilami	Prof., Université de Batna
	M. Touahria	MC., Université de Sétif
<i>Rapporteur</i>	M. Djoudi	MC., Université de Poitiers

Remerciements

Toute ma gratitude et mon remerciement vont, tout d'abord, à Dieu le tout puissant qui m'a aidé durant ce travail.

Que tous ceux qui ont contribué de près ou de loin par leurs conseils et leurs encouragements à l'aboutissement de ce travail, trouvent ici l'expression de ma gratitude et ma profonde reconnaissance.

Mes vifs remerciements accompagnés de toute ma gratitude vont également à mon directeur de thèse Monsieur Mahieddine DJOUDI, Maître de conférences à l'université de Poitiers (France). Je lui suis particulièrement reconnaissante pour son soutien, sa disponibilité, ses précieux conseils avisés, ses orientations et ses encouragements.

Je remercie les membres de jury : Pr. Boucenna Ahmed, Pr. Bilami Azzedine, Pr. Benmohamed Mohamed et Dr. Touahria Mohamed de m'avoir fait l'honneur d'accepter de participer à mon jury.

Je remercie ceux qui m'ont aidé à l'aboutissement de ce travail : tous mes collègues du département d'Informatique de l'Université Ferhat Abbas de Sétif (UFAS).

Je remercie aussi tous ceux que j'ai encadrés et avec lesquels j'ai eu l'honneur de collaborer.

Je remercie tous ceux qui ont participé aux enquêtes et aux expérimentations que j'ai réalisées.

Je remercie tou(te)s mes ami(e)s pour leur aide et encouragement.

Je remercie fortement tous les membres de ma famille, particulièrement mes parents pour leur affection et leur soutien.

Résumé

Dans le domaine des Technologies de l'Information et de la Communication (TIC), l'apprentissage coopératif à distance tient une place importante. Avec le développement de la technologie du web, c'est le savoir constamment actualisé qui va vers l'apprenant. Ce dernier peut décider de son cursus. Le formateur réapparaît à l'autre bout de la chaîne dans un rôle de régulateur. Le centre d'intérêt est bien désormais l'apprenant lui-même, or la modélisation de l'utilisateur en situation d'interaction avec la machine est un problème très difficile et un sujet très ouvert en recherche.

Notre travail vise à mettre en œuvre un outil à base d'agents de modélisation des attitudes cognitives et affectives de l'apprenant en situation d'apprentissage coopératif. Il permet d'observer et d'expérimenter de façon visuelle et interactive l'évolution des interactions dans un groupe d'apprenants en situation d'apprentissage avec un ou plusieurs formateurs. L'apprentissage est conçu comme un parcours de quête de connaissances de la part d'individus en interaction et parfois comme une résolution collective de problème.

Les individus seront formalisés en rapport avec les objets. Ils seront définis par une identité, un rôle, des modalités intentionnelles (vouloir, pouvoir, savoir, croire, devoir), des modalités affectives (plaisir, confiance en soi, confiance à l'autre, bienveillance), des variables situationnelles (type du formateur, compatibilité idéologique, degré de motivation, etc.). Le parcours sera défini par un scénario. Son déroulement sera commandé par les énoncés ou les actes successifs du formateur. Les apprenants réagiront chacun selon son état modal du moment. En fin de parcours, le bilan de chaque apprenant déterminera sa conduite d'apprentissage (correct, révolté, résigné, débordant, indéterminé, etc.). L'état du formateur se trouvera aussi modifié par sa propre performance et il sera donc différent à l'arrivée de ce qu'il était au départ.

Mots clés : Apprentissage coopératif, apprenant, situations d'apprentissage, modélisation de l'utilisateur, psychologie cognitive, agent, SMA, attitudes cognitives et affectives.

Abstract

In the field of Information and Communication Technologies (ICT), the cooperative learning holds an increasingly important place. With the development of the web technology, the learner benefits from knowledge constantly updated. This last can decide his learning path. The trainer reappears at the other side of the chain in a role of regulator. From now, the interest Focus is the learner himself, or the user modeling in an interactive situation with the machine is a very difficult and a very open research topic.

The subject aims to implement an agent based tool for dynamic modeling of the learner's cognitive and affective attitudes in cooperative learning situation. It allows observing and experimenting in a visual and interactive way the evolution of the learner interactions with a group of learners in a cooperative situation with one or more trainers. The training will be conceived like a search path of knowledge by individuals in interaction and sometimes like a collective resolution of the problem.

The individuals will be formalized in connection with the objects. They will be defined by an identity, a role, intentional attitudes (to want, to can, to know, to believe, to have to), emotional attitudes (pleasure, self-confidence, trust each other, benevolence) and situational variables (Type of trainer, ideological compatibility, motivation, etc.). The learning path will be defined by a scenario. Its unfolding will be controlled by successive acts or statements of the trainer. Each learner will react depending on his current modal state. At the end of the learning path, the assessment of each learner will determine his learning behavior (correct, revolted, resigned, overflowing, unspecified, etc.). The trainer status will be also modified by its own performance and it will be therefore different from its arrival from.

Keywords: Cooperative learning, learner, learning situations, user modeling, cognitive psychology, agent, multi agent system, cognitive and affective attitudes.

ملخص

يحتل التعلم التعاوني عن بعد مكانة هامة في مجال تكنولوجيا الإعلام و الاتصال. مع تطور تكنولوجيا الويب، وجدت المعرفة المحدثه أنيا طريقها إلى المتعلم الذي بات يملك قرار اختيار نمط دراسته و أصبح دور المدرس يتجلى في الطرف الثاني من الحلقة كمنظم. يحتل المتعلم أو الممهل مركز الاهتمام، مع أن عملية تصميم المستخدم في وضعية تفاعل واستعمال للآلة مشكلة صعبة جدا وموضوع بحث جد مفتوح.

يهدف موضوعنا إلى وضع وسيلة تصميم نشطة (ديناميكية) على أساس الأعوان أو الوحدات للتصرفات المعرفية و العاطفية أو الحسية في وضعية التعلم التعاوني عند الإنسان. كما يسمح بمراقبة وتجريب بطريقة عينية وتفاعلية تطور المتجاوبات لدى مجموعة من المتعلمين في حالة تعلم مع مدرس أو مجموعة من المدرسين. يتم تصوير التعلم كطريق بحث على المعارف من قبل أفراد في حالة تفاعل تعاقدى أو جدلي وأحيانا كحل جماعي لمشكل ما.

يستنبط الأفراد حسب الهيئات أو الأشياء و يتم تعريفهم بواسطة: هوية، دور، نوايا (الإرادة، القدرة، الإيمان أو الثقة، الواجب)، العاطفة أو الطر ق الحسية (السعادة، الثقة بالنفس، الثقة بالآخر، الحرص)، متغيرات ظرفية (نوع المدرس، تطابق الإيديولوجيات، درجة الدافع...). هذا الدرب يحدد بسيناريو حيث يعنى محتوى الدروس و كذا التصرفات المتتالية للمدرس بتحديد وجهته. يتفاعل كل متعلم حسب حالته اللحظية و في آخر مسيرته، تقدم حوصلة موضحة ومبينة للمسيرة التعليمية للمتعلم (صحيحة، ثائرة، مستسلمة، غير محددة...). حالة المدرس ستتغير بموجب كفاءته الخاصة وستكون بذلك في آخر المطاف مختلفة عن البداية.

مفاتيح: التعلم التعاوني، المتعلم، وضعية التعلم، تصميم المستخدم، السيكلوجيات المعرفية، الأعوان، الأنظمة متعددة الأعوان، السلوك المعرفية والحسية.

Table des matières

TABLE DES MATIERES.....	4
LISTE DES FIGURES.....	9
LISTE DES TABLES	13
INTRODUCTION GENERALE	15
INTRODUCTION.....	15
MOTIVATIONS.....	16
CONTRIBUTIONS	17
ORGANISATION DE LA THESE	18
CHAPITRE 1 : L'APPRENTISSAGE SUR LE WEB	21
1.1 INTRODUCTION	21
1.2 L'APPRENTISSAGE EN LIGNE OU LE E-LEARNING	21
1.3 THEORIES D'APPRENTISSAGE	22
1.3.1 L'approche behavioriste	22
1.3.2 L'approche cognitiviste.....	23
1.3.3 L'approche constructiviste	23
1.3.4 L'approche socioculturelle	24
1.4 L'EVOLUTION DES SYSTEMES PEDAGOGIQUES	24
1.4.1 Les systèmes pédagogiques programmés	25
1.4.2 Les systèmes pédagogiques intelligents	25
1.4.3 Les environnements informatiques pour l'apprentissage humain	26
1.5 L'APPRENTISSAGE COLLABORATIF/COOPERATIF.....	28
1.6 ENVIRONNEMENT INFORMATIQUE DE COLLABORATION OU DE COOPERATION.....	29
1.6.1 L'Apprentissage Collaboratif Assisté par Ordinateur.....	30
1.6.2 Le Travail Coopératif Assisté par Ordinateur	30
1.7 WEB2.0 : MODELES DE L' APPRENANT ET DE CONTENU	31
1.7.1 Les ontologies.....	32

1.8 CONCLUSION	34
CHAPITRE 2 : MODELISATION DE L'APPRENANT	36
2.1 INTRODUCTION	36
2.2 ROLE DU MODELE DE L'APPRENANT DANS LES ENVIRONNEMENTS D'APPRENTISSAGE	38
2.3 APPROCHES EXISTANTES DANS LA MODELISATION DE L'APPRENANT	40
2.3.1 <i>Modèle d'expertise partielle</i>	41
2.3.2 <i>Le modèle différentiel</i>	44
2.3.3 <i>Modèle de perturbation</i>	45
2.3.4 <i>Modèle de traçage</i>	46
2.3.5 <i>Modèle basé sur les contraintes</i>	48
2.3.6 <i>Le modèle social</i>	49
2.3.7 <i>Modèles ouverts (inspectable)</i>	50
2.4 CONTENU DU MODELE DE L'APPRENANT : CONNAISSANCES ET PREFERENCES.....	51
2.4.1 <i>Modélisation cognitive (des connaissances)</i>	53
2.4.2 <i>Modélisation affective (des préférences)</i>	56
2.5 LA GRANULARITE DE LA REPRESENTATION DE LA CONNAISSANCE DE L'APPRENANT	58
2.6 TECHNIQUES DE REPRESENTATION DE LA CONNAISSANCE	58
2.6.1 <i>Les systèmes à base de règles</i>	59
2.6.2 <i>Les réseaux sémantiques</i>	59
2.6.3 <i>Les systèmes basés sur la logique</i>	60
2.6.4 <i>Les systèmes basés sur les réseaux de Bayes</i>	61
2.7 ACQUISITION DU MODELE DE L'APPRENANT	65
2.7.1 <i>Le recueil centré serveur</i>	67
2.7.2 <i>Le recueil centré client</i>	67
2.7.3 <i>Outils spécifiques à l'environnement</i>	68
2.8 ANALYSE DES TRACES	69
2.8.1 <i>Trace</i>	69
2.8.2 <i>Les approches de traitement des traces numériques</i>	72
2.9 MODELISATION DE CONTENU.....	73
2.10 POSITION DE NOTRE APPROCHE DE MODELISATION	74
2.11 CONCLUSION	75
CHAPITRE 3 : AGENTS, SYSTEMES MULTI-AGENTS ET METHODOLOGIES DE CONCEPTION	77

3.1 INTRODUCTION	77
3.2 AGENT.....	77
3.2.1 Définitions.....	77
3.2.2 Les Types d'agents.....	79
3.3 LES SYSTEMES MULTI-AGENTS (SMA)	83
3.3.1 Définitions.....	83
3.3.2 Les caractéristiques d'un SMA	84
3.3.3 Les architectures des SMA	85
3.3.4 Les avantages des SMA	86
3.3.5 Les soucis des SMA	87
3.3.6 Les langages de communication entre Agents.....	87
3.4 METHODES D'ANALYSE ET DE CONCEPTION BASEES AGENTS.....	89
3.4.1 L'approche Voyelles (décomposition AEIO).....	89
3.4.2 Approche Agent Groupe Rôle "AGR":	91
3.4.3 La méthodologie MaSE.....	92
3.5 APPORTS DE LA TECHNOLOGIE AGENTS POUR L'APPRENTISSAGE	96
3.5.1 Modélisation de l'apprenant dans un groupe	96
3.5.2 L'assistance virtuelle et la délégation	97
3.6 CONCLUSION	98
CHAPITRE 4 : LA MODELISATION BAYESIENNE DE L'APPRENANT DANS	
L'ENVIRONNEMENT D'APPRENTISSAGE COLLABORATIF E-PACAD	100
4.1 INTRODUCTION	100
4.2 L'APPROCHE PACAD	100
4.2.1 Le modèle de contenu pédagogique	101
4.2.2 Le modèle de l'apprenant	105
4.2.3 Le modèle de communication	108
4.2.4 Le modèle de collaboration	108
4.3 APPLICATION D'UN RESEAU BAYESIEN A UN MODELE D'EXPERTISE PARTIELLE.....	109
4.3.1 Exemple d'un réseau bayésien	109
4.3.2 L'application du réseau bayésien dans notre modèle de l'apprenant	113
4.3.3 L'inférence du niveau de connaissance de l'apprenant.....	116
4.3.4 Le réseau bayésien de notre modèle cognitif	117
4.4 CONCLUSION	118

CHAPITRE 5 : L'APPROCHE A BASE D'AGENTS POUR LA MODELISATION	119
5.1 INTRODUCTION	119
5.2 DEMARCHE SUIVIE DANS LA CONCEPTION	120
5.3 LA PHASE D'ANALYSE.....	121
5.3.1 <i>Cahier des charges :</i>	121
5.3.2 <i>Le diagramme hiérarchisé des objectifs</i>	123
5.3.3 <i>Diagrammes de séquence</i>	124
5.3.4 <i>Le diagramme des rôles et des tâches</i>	133
5.3.5 <i>La description des tâches concurrentes</i>	135
5.4 LA PHASE DE CONCEPTION	147
5.4.1 <i>Le diagramme des classes d'agents</i>	147
5.4.2 <i>La construction des communications inter-agents</i>	149
5.4.3 <i>Implémentation des agents</i>	149
5.4.4 <i>Diagramme de déploiement</i>	149
5.5 CONCLUSION	150
CHAPITRE 6 : MISE EN ŒUVRE ET EXPERIMENTATION D'USAGE	152
6.1 INTRODUCTION	152
6.2 CHOIX DES OUTILS DE DEVELOPPEMENT UTILISES.....	152
6.2.1 <i>Choix du Java comme langage de programmation</i>	152
6.2.2 <i>Choix de JADE comme plateforme de développement</i>	153
6.2.3 <i>Les outils web</i>	155
6.2.4 <i>Les serveurs</i>	156
6.2.5 <i>Choix de MySQL comme un système de gestion de base de données</i>	156
6.3 L'ENVIRONNEMENT E-PACAD	157
6.4 DIDACTICIEL E-PACAD	157
6.4.1 <i>Accès à la formation recherchée</i>	157
6.4.2 <i>La connexion</i>	158
6.4.3 <i>Le lancement des agents de l'environnement</i>	158
6.4.4 <i>L'interface de l'environnement E-PACAD</i>	159
6.4.5 <i>L'accès aux différents espaces de l'environnement</i>	160
6.4.6 <i>Le détail des pages de l'environnement</i>	160
6.4.7 <i>L'accès aux concepts de la formation</i>	161
6.4.8 <i>Les ressources indexées</i>	163

6.4.9 Les ressources de différents espaces	165
6.4.10 L'historique.....	166
6.4.11 Le modèle cognitif de l'apprenant	167
6.4.12 La communication entre les agents	167
6.5 EXPERIMENTATION D'USAGE.....	168
6.5.1 Evaluations.....	169
6.5.2 Retour d'expérience	170
6.6 CONCLUSION	172
CONCLUSION GENERALE ET PERSPECTIVES.....	174
BILAN DES TRAVAUX ET APPORTS DE LA THESE	174
INTERET DU PROJET.....	177
CONCLUSION	178
PERSPECTIVES	179
Prise en compte d'autres informations dans le modèle de l'apprenant.....	179
Développement d'autres situations didactiques	179
Travail avec plusieurs apprenants.....	180
L'élargissement de l'architecture de notre environnement	180
Exploration de la collaboration.....	180
BIBLIOGRAPHIE	181
LISTE DES PUBLICATIONS PERSONNELLES	192
PUBLICATIONS DANS DES REVUES INTERNATIONALES	192
COMMUNICATIONS DANS DES CONFÉRENCES INTERNATIONALES	192
COMMUNICATIONS DANS DES CONGRÈS NATIONAUX (AVEC COMITÉ DE LECTURE).....	193
ANNEXES.....	194
ANNEXE 1 : PARTIE DU CODE SOURCE DU SYSTEME.....	194
ANNEXE 2 : QUESTIONNAIRES	200
ANNEXE 3 : FICHES D'ÉVALUATION.....	204

Liste des figures

Figure 1.	Schéma représentant le plan de la thèse	19
Figure 2.	Modélisation par recouvrement.....	41
Figure 3.	Exemple de modélisation par expertise partielle	42
Figure 4.	Modèle différentiel.....	44
Figure 5.	Modèle de perturbation.....	45
Figure 6.	Modèle social	49
Figure 7.	Exemple de relations AND-OR dans le modèle GM.....	55
Figure 8.	Réseau de Bayes de l'exemple	63
Figure 9.	Partage de tableau noir	85
Figure 10.	Communication par envoi de message.....	86
Figure 11.	Modèle Agent Groupe Rôle	91
Figure 12.	Les étapes et les phases de la méthodologie MaSE	92
Figure 13.	Approche conceptuelle du modèle de contenu	103
Figure 14.	Extrait de l'ontologie de domaine.....	104
Figure 15.	Extrait de l'ontologie d'application (AP)	104
Figure 16.	Réseau de Bayes de l'exemple	110
Figure 17.	Exemple d'un réseau bayésien	114
Figure 18.	Exemple d'un réseau bayésien	117
Figure 19.	Diagramme hiérarchisé des objectifs.....	124

Figure 20.	Diagramme de séquence «Accéder à l'unité de formation»	126
Figure 21.	Diagramme de séquence « Connexion d'un apprenant ».....	127
Figure 22.	Diagramme de séquence «Accéder à un espace de travail»	127
Figure 23.	Diagramme de séquence «Sélectionner un point d'accès».....	128
Figure 24.	Diagramme de séquence «Charger une ressource pédagogique»	129
Figure 25.	Diagramme de séquence «Ajouter un point d'accès »	129
Figure 26.	Diagramme de séquence « Ajouter une ressource »	130
Figure 27.	Diagramme de séquence « Travailler sur un exercice QCM ».....	131
Figure 28.	Diagramme de séquence « Charger la liste des utilisateurs ».....	131
Figure 29.	Diagramme de séquence « Envoyer un message ».....	132
Figure 30.	Diagramme de séquence « Recevoir un message »	133
Figure 31.	Diagramme de séquence «Déconnexion d'un apprenant».....	133
Figure 32.	Diagramme des rôles et des tâches	134
Figure 33.	États et transitions du rôle demandeur d'un service.	135
Figure 34.	États et transitions du rôle récepteur de la demande.....	135
Figure 35.	états et transitions du rôle R_Assistant demandant d'accéder à une unité de formation.....	136
Figure 36.	états et transitions du R_SBD envoyant l'unité demandée	136
Figure 37.	états et transitions de l'Interface émettant une demande de connexion d'un apprenant.....	137
Figure 38.	états et transitions du R_SBD envoyant le résultat de la connexion de l'apprenant.....	137
Figure 39.	états et transitions du rôle R_Assistant demandant le contenu d'un espace de travail.....	138

Figure 40.	états et transitions du R_SBD envoyant le contenu d'un espace de travail.....	138
Figure 41.	états et transitions du R_Assistant demandant la consultation d'un concept (notion).....	139
Figure 42.	états et transitions du R_SBD envoyant le concept courant au R_Assistant.....	139
Figure 43.	états et transitions du R_Assistant demandant le chargement d'une ressource pédagogique.....	140
Figure 44.	états et transitions du R_SBD envoyant la ressource pédagogique au R_Assistant.....	140
Figure 45.	états et transitions du R_Assistant demandant l'ajout d'un point d'accès au R_SBD.....	141
Figure 46.	états et transitions du R_SBD sauvegardant un point d'accès dans la base de données.....	141
Figure 47.	états et transitions du R_Assistant demandant l'ajout d'une ressource au R_SBD.....	142
Figure 48.	états et transitions du R_SBD sauvegardant une ressource dans la base de données.....	142
Figure 49.	états et transitions du R_Assistant demandant l'évaluation d'un exercice.....	143
Figure 50.	états et transitions du R_Evaluator évaluant un exercice.....	143
Figure 51.	états et transitions du R_Assistant demandant la liste des utilisateurs ...	144
Figure 52.	états et transitions du R_SBD envoyant la liste des utilisateurs au R_Assistant.....	144
Figure 53.	états et transitions du R_Assistant envoyant un message à son destinataire.....	145

Figure 54.	états et transitions du R_Collaborator transmettant le message au destinataire	145
Figure 55.	états et transitions du R_Assistant recevant un message	146
Figure 56.	états et transitions du R_Collaborator transmettant un message	146
Figure 57.	états et transitions de l'Interface émettant une demande de déconnexion d'un apprenant	146
Figure 58.	Diagramme des classes d'agents	148
Figure 59.	Diagramme de déploiement	150
Figure 60.	Interfaces graphiques de JADE	154
Figure 61.	L'accès à la formation	157
Figure 62.	La connexion des utilisateurs	158
Figure 63.	L'agent RMA de la plate-forme JADE	159
Figure 64.	Le choix d'un espace de travail	159
Figure 65.	La navigation au sein de l'ontologie d'application	161
Figure 66.	Visualisation d'une partie de l'ontologie d'application (AP)	162
Figure 67.	Les ressources indexées par le concept « variable »	163
Figure 68.	L'accès à une ressource.....	164
Figure 69.	L'ajout d'une ressource.....	165
Figure 70.	Visualisation des ressources de l'espace commun.	166
Figure 71.	Visualisation des ressources de l'espace privé.....	166
Figure 72.	Visualisation des ressources de l'espace du groupe 1.	166
Figure 73.	Le modèle cognitif de l'apprenant	167
Figure 74.	L'agent Sniffer de JADE	168

Liste des tables

Table 1.	Modèle de préférences de l'apprenant.....	57
Table 2.	La distribution de probabilités du nœud P	110
Table 3.	La distribution de probabilités du nœud A.....	111
Table 4.	Table des probabilités conditionnelles du nœud V ($P(V/P)$)	111
Table 5.	Table des probabilités conditionnelles du nœud J ($P(J/P,A)$)	111
Table 6.	Table des probabilités jointes $P(V,P)$	111
Table 7.	Table des probabilité jointe $P(J,P, A)$	111
Table 8.	La table des probabilités du nœud V.....	111
Table 9.	La table des probabilités du nœud J	111
Table 10.	Modification des valeurs des probabilités en prenant en compte le fait que J est vrai.....	112
Table 11.	Modification des valeurs des probabilités en prenant en compte le fait que V est vrai.....	112
Table 12.	Table des probabilités conditionnelles du concept V.....	115
Table 13.	Table des probabilités conditionnelles du concept E.....	115
Table 14.	Table des probabilités conditionnelles du concept F	115
Table 15.	La distribution de probabilités du concept L	116
Table 16.	La distribution de probabilité du concept G	116
Table 17.	Table de probabilité jointe $P(V,L,G)$	116

Table 18.	Table de probabilité jointe $P(V,L,G)$ en prenant en compte les faits $E=1$ et $F=1$	117
Table 19.	L'utilisation de l'historique.....	171
Table 20.	L'utilisation du modèle cognitif.....	171

Introduction générale

Introduction

L'avènement des Technologies de l'Information et de la Communication a permis de mettre en place un nouveau mode d'apprentissage connu sous le nom de l'apprentissage en ligne ou e-learning. Un apprentissage rapide et efficace, avec un minimum de problèmes d'organisation. Cependant, les besoins des acteurs de ces systèmes en termes d'adaptation, de parcours selon les profils et les préférences des utilisateurs, de partage et de possibilité de réutilisation des contenus et des fonctionnalités ne cessent de s'amplifier.

Notre travail s'inscrit dans le champ de recherches sur les environnements informatiques pour l'apprentissage humain (EIAH), il s'agit alors d'environnements informatiques conçus dans le but de « *favoriser la construction de connaissances chez un apprenant* » (Tchounikine, 2002). Nous nous intéressons plus particulièrement au traitement des interactions entre les apprenants et les environnements d'apprentissage pour envisager une personnalisation de l'apprentissage.

Dans le domaine des EIAH, la personnalisation des environnements d'apprentissage selon les préférences et les besoins de l'apprenant constitue aujourd'hui une problématique importante renouvelée récemment par l'évolution des technologies de l'information et de la communication. L'ordinateur est devenu un simple élément d'un système plus complexe impliquant la coopération de diverses entités humaines et artificielles partageant un ensemble de ressources pédagogiques. Face à ces ressources interactives, l'apprenant peut se trouver dans une situation où il est gagnant et perdant en même temps. D'un côté, il dispose d'un accès simple aux ressources numériques dont il a besoin pour son activité, il mène des expériences directes avec son environnement et il gère son temps et son rythme ; d'un autre côté, l'isolement de

l'apprenant, le manque de perception, le manque de vision et le manque des conseils immédiats de l'enseignant rendent l'apprenant beaucoup plus démuni qu'en situation d'apprentissage en présentiel.

Motivations

Lorsque la personnalisation de l'apprentissage est envisagée, plusieurs aspects à propos de l'apprenant d'ordre cognitif ou affectif, liés aux motivations ou aux préférences, sont pris en considération par l'enseignant au cours de l'apprentissage de manière naturelle et selon différents degrés d'importance.

Pour prendre en charge ces aspects, l'enseignant dans une situation d'apprentissage est basé sur son expérience, sur sa capacité d'établir un dialogue avec l'apprenant afin de promouvoir les interactions souhaitées et sur sa perception globale du comportement de l'apprenant qui lui permet de changer de plan et de proposer de nouvelles activités, donc de remédier aux erreurs éventuelles de l'apprenant.

Cependant, la complexité de la prise en compte de l'apprenant n'apparaît pas toujours évidente. Pourtant, cette complexité a été relevée depuis le début des recherches faites dans le domaine des EIAH. Une grande partie de la problématique de développement d'un environnement d'apprentissage autonome est concentrée sur la prise en compte de l'apprenant.

Notre hypothèse est que lorsqu'on propose aux apprenants des environnements d'apprentissage intégrés d'outils qui permettent d'offrir un suivi et une évaluation efficaces aux apprenants et qui peuvent s'adapter à leurs préférences et leurs besoins. Ceci les aide dans leur apprentissage.

Donc, pour individualiser l'apprentissage, les EIAH disposent d'un modèle de l'apprenant. Celui-ci contient les croyances du système sur l'apprenant, c'est-à-dire ses connaissances acquises ou non acquises, les erreurs commises, etc.

Au-delà d'un modèle de l'apprenant qui prend en compte certaines caractéristiques de l'apprenant à partir de ses performances et ses actions face à une interface informatique, la modélisation de l'apprenant a pour objectif de construire un modèle cognitif des

connaissances de l'apprenant afin d'envisager des explications au raisonnement de l'apprenant à partir de l'observation de son comportement.

Dans un EIAH, le modèle de l'apprenant n'est pas un composant autonome et isolé. Au début des interactions de l'apprenant, il est généralement assez générique, mais au cours du temps, il s'affine à partir des interactions de l'apprenant avec l'environnement. Ces interactions découlent des prises de décisions qui sont basées sur le modèle de l'apprenant et permettent aussi de valider ou refuser les croyances du système sur l'apprenant. Donc, il existe une forte liaison entre le modèle de l'apprenant et la prise de décisions qui conditionne l'impact de l'apprenant sur les stratégies d'enseignement.

Ainsi, le travail présenté dans cette thèse se propose d'étudier les problématiques liées à la conception d'un système multi-agents (SMA) permettant de gérer le modèle de l'apprenant. Nous devons donc réfléchir à une architecture multi-agents pour un environnement d'apprentissage coopératif qui intègre un ou plusieurs agents se chargeant du modèle de l'apprenant.

Nos problématiques seront à la fois de proposer une formalisation du modèle de l'apprenant dans une didactique particulière qui est la didactique de l'informatique ainsi que des mécanismes de construction de ce modèle. Par « gérer le modèle de l'apprenant », nous désignons à la fois, construire (ou modéliser) le modèle de l'apprenant et répondre aux interrogations venant des autres agents de l'environnement concernant les informations contenues dans ce modèle.

Contributions

Les travaux de cette thèse ont abouti aux contributions scientifiques suivantes :

- Une étude empirique sur la modélisation de l'apprenant montrant la dépendance entre le comportement de l'apprenant et l'apprentissage personnalisé.
- Une étude détaillée sur les systèmes multi-agents et sur les approches de conception et d'implémentation des SMA.

- Une étude sur un modèle de connaissances à base de ressources pédagogiques représenté par des ontologies de domaine et d'application et défini dans (Abel, 2008).
- La combinaison d'un modèle cognitif d'expertise partielle et d'un réseau de Bayes et l'attachement de ce modèle au modèle de connaissances précédent.
- Proposition d'une architecture multi-agents pour notre environnement d'apprentissage qui est un environnement centré sur la modélisation de l'apprenant en situation d'apprentissage coopératif.
- La programmation des agents en tant qu'entités autonomes à l'intérieur de notre environnement en utilisant le modèle d'architecture de *Java Agent Development Framework* (JADE).
- Réalisation d'un agent Assistant qui a parmi ses tâches la mémorisation des traces d'interaction de l'apprenant avec l'environnement et aussi avec les autres acteurs humains de la situation d'apprentissage.

Organisation de la thèse

Comme la figure 1 l'indique, la présente thèse est composée de six chapitres et trois annexes. Elle est organisée en deux parties:

La première partie concerne une étude d'un état de l'art relatif à notre problématique. Elle est constituée de trois chapitres:

- Le premier chapitre nous permet d'évoquer la notion d'apprentissage et de définir ce que nous entendons par l'apprentissage en ligne, autrement dit e-learning. Nous abordons également la notion d'apprentissage collectif en vue de comprendre comment dans un contexte de e-learning, les TIC peuvent favoriser un tel apprentissage. Cela nous a amené à traiter les notions de collaboration et de coopération dans la communauté d'apprenants.

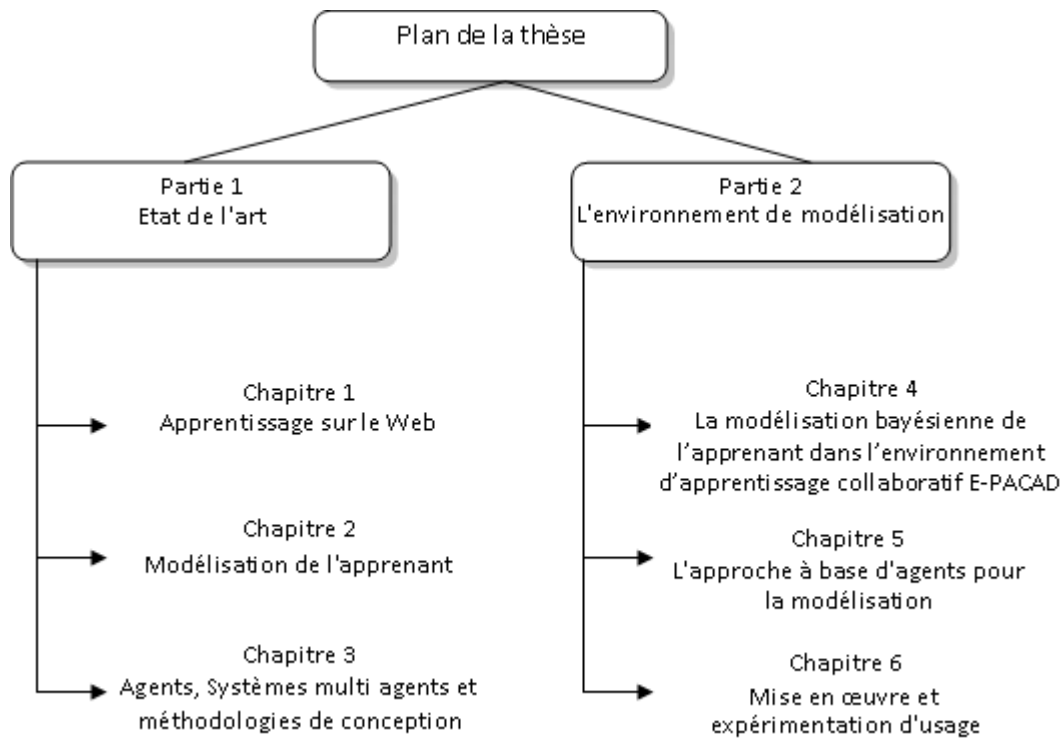


Figure 1. Schéma représentant le plan de la thèse

- Le deuxième chapitre présente la problématique et la motivation de notre thèse. Il traite les approches existantes dans la modélisation de l'apprenant, les techniques utilisées dans la représentation des connaissances, l'acquisition et l'analyse des traces récoltées dans une situation d'apprentissage afin de permettre une modélisation comportementale de l'apprenant.
- Le troisième chapitre présente les notions d'agent et de système multi-agents. Nous parcourons quelques méthodologies existantes pour l'analyse, la conception, la modélisation et la spécification des systèmes multi-agents. Nous présentons à la fin de ce chapitre l'approche utilisée pour l'analyse, la conception et la spécification de notre SMA de modélisation de l'apprenant, il s'agit de la méthodologie *Multi agent System Engineering* (MaSE).

La deuxième partie présente la conception et la mise en œuvre de notre environnement de modélisation. Elle est composée de trois chapitres:

- Le quatrième chapitre présente les différentes composantes de notre environnement d'apprentissage collaboratif E-PACAD. Ensuite il traite notre

approche de modélisation cognitive de l'apprenant qui consiste à combiner un modèle d'expertise partielle de l'apprenant avec un réseau bayésien et l'attacher à un modèle de connaissances à base de ressources pédagogiques représenté par des ontologies de domaine et d'application.

- Le cinquième chapitre concerne l'expression des besoins et les phases d'analyse et de conception du cycle de développement de notre environnement de modélisation.
- Le sixième chapitre présente les choix techniques et d'implémentation ainsi que la réalisation informatique d'un point de vue utilisateur.

En guise de conclusion, nous présentons un bilan de nos travaux de thèse, les résultats obtenus et nous traçons des perspectives futures à nos travaux.

Trois annexes sont présentées à la fin de cette thèse.

Chapitre 1 : L'apprentissage sur le Web

1.1 Introduction

Ce chapitre présente une synthèse de travaux concernés par l'apprentissage sur le web ou e-learning et positionne notre approche de modélisation. Nous commençons par définir le concept de : « e-learning ». Nous poursuivons en présentant les théories d'apprentissage. Nous abordons l'évolution des systèmes pédagogiques vers les Environnements Informatiques de l'Apprentissage Humain. Ensuite, nous introduisons les concepts d'apprentissage collaboratif et coopératif, de l'Apprentissage Collaboratif Assisté par Ordinateur et de Travail Coopératif Assisté par Ordinateur afin de mieux justifier l'intérêt de la collaboration et la coopération dans le domaine de l'apprentissage. Dans la dernière section, nous abordons les notions du Web2.0 et des ontologies. Nous concluons finalement le chapitre.

1.2 L'apprentissage en ligne ou le e-Learning

L'apprentissage en présentiel présente un ensemble de caractéristiques qui le rendent moins efficace. En effet, il est centré sur le formateur plus que sur l'apprenant, puisque c'est le formateur qui sélectionne le contenu à présenter et le mode d'enseignement, il est à transmission unidirectionnelle : l'information et la connaissance sont transmises du formateur à l'apprenant. Il souffre du manque de personnalisation : le contenu doit satisfaire les besoins de l'ensemble des formés et enfin c'est un processus d'apprentissage statique et linéaire : le contenu reste inchangé (Abel et al., 2007).

L'avènement des Technologies de l'Information et de la Communication a permis de mettre en place un nouveau mode d'apprentissage connu sous le nom de l'apprentissage en ligne ou e-learning.

Le e-learning est un contexte d'apprentissage reposant sur la mise à disposition des contenus pédagogiques via un support électronique (Cédérom, Internet, intranet, etc.). A travers ce contexte d'apprentissage, on vise des publics différents, que ce soit des étudiants qui utilisent ces supports électroniques à des fins d'apprentissage académique ou bien des adultes qui veulent perfectionner leur formation et/ou mettre à jour leurs connaissances. Le e-learning étudie les contenus pédagogiques et les outils ou les applications permettant de mettre ces contenus à disposition des apprenants.

1.3 Théories d'apprentissage

Dans cette partie, nous présentons les différentes approches théoriques de l'apprentissage (Dillenbourg et al., 1996), telles que : l'approche behavioriste, l'approche cognitive, l'approche constructiviste et l'approche socioculturelle.

1.3.1 L'approche behavioriste

Cette approche s'intéresse au comportement observable de l'apprenant sans prendre en compte les processus mentaux intervenant dans l'apprentissage. Ces processus sont considérés comme une boîte noire à ne pas accéder.

Selon les behavioristes, on a un apprentissage lorsque l'apprenant répond correctement à un stimulus qui provient de l'environnement, et c'est la raison pour laquelle, l'idée centrale du behaviorisme est illustrée par la relation Stimulus-Réponse (S-R).

Dans cette approche, l'enseignant utilise des méthodes pédagogiques telles que les exposés, les pratiques répétées, les renforcements positifs augmentant la probabilité que le comportement antérieur se reproduise, les punitions diminuant la probabilité que le comportement antérieur se reproduise.

Pour atteindre l'apprentissage visé par l'approche behavioriste, l'enseignant définit les connaissances à acquérir en termes de comportements observables. L'apprenant dans ce cas devra être capable de reconnaître, d'identifier, de distinguer, de nommer, etc., mais pas de réfléchir, de comprendre, de savoir, etc.

L'apprentissage est donc défini comme un changement de comportement chez l'apprenant.

1.3.2 L'approche cognitive

L'approche cognitive se concentre sur les activités intérieures mentales. L'ouverture de la boîte noire de l'esprit humain est nécessaire pour comprendre comment les apprenants apprennent. Les processus mentaux tels que la pensée, la mémoire, le savoir et le besoin de résolution de problèmes doivent être explorés. La connaissance peut être considérée comme des schémas ou des symboliques des constructions mentales. L'apprentissage est défini comme un changement dans les schémas de l'apprenant.

Contrairement au behaviorisme, les apprenants ne sont pas des êtres passifs qui ne font que réagir à des stimuli de l'environnement, ce sont des êtres rationnels à qui on exige une participation active afin d'apprendre et dont les actions sont une conséquence de la pensée. Les changements de comportements sont observés, mais seulement comme une indication de ce qui se passe dans la tête de l'apprenant. Le cognitivisme considère la métaphore de l'esprit comme un ordinateur : l'information arrive, se traite et conduit à certains résultats.

1.3.3 L'approche constructiviste

L'approche constructiviste considère l'apprentissage comme un processus actif et constructif. L'apprenant est un constructeur de l'information. Les gens construisent activement ou créent leurs propres représentations subjectives de la réalité objective. De nouvelles informations sont liées à des connaissances antérieures.

Contrairement aux approches didactiques tels que le behaviorisme et l'enseignement programmé, le constructivisme annonce que l'apprentissage est un processus actif et contextualisé basé sur la construction des connaissances plutôt que sur son acquisition. La connaissance est construite sur la base des expériences personnelles et les hypothèses de l'environnement. Les apprenants vérifient en permanence ces hypothèses par la négociation sociale. L'interprétation et le processus de construction de connaissances se diffèrent d'une personne à l'autre.

Pour atteindre l'apprentissage visé par l'approche constructiviste, l'enseignant ne devra jamais dire directement quoi que ce soit aux apprenants, mais, au contraire, il faut toujours leur permettre de construire des savoirs pour eux-mêmes. Le constructivisme suppose que toute connaissance est construite à partir des connaissances antérieures de l'apprenant.

1.3.4 L'approche socioculturelle

L'approche socioculturelle est inspirée des travaux de (Vygotsky, 1978). Elle s'articule autour de *l'activité sociale intériorisée par l'individu et qui conduit à l'apprentissage*. Elle met l'accent sur la relation causale qui existe entre l'interaction sociale et les changements cognitifs.

Le développement est considéré comme le résultat du passage du plan social au plan interne, autrement dit le passage des interactions avec les autres au raisonnement (Dillenbourg, 1999). Il apparaît qu'une capacité se développe, s'intériorise et ensuite elle devient une partie intégrante du fonctionnement cognitif personnel.

Ainsi, pour Vygotsky, il existe une zone proximale de développement (ZPD) chez chaque individu qui est définie comme suit :

« La distance entre son niveau de développement actuel, en regard de sa capacité à résoudre individuellement un problème et son niveau de développement potentiel déterminé par la résolution guidée par un adulte ou en collaboration avec des pairs plus compétents ».

De ce fait, on appelle l'appropriation ou l'intériorisation le mécanisme par lequel la participation à une résolution de problème commune peut modifier la compréhension du problème.

1.4 L'évolution des systèmes pédagogiques

Les premiers systèmes d'Enseignement Assisté par Ordinateur (EAO) sont apparus au début des années 1960. L'objectif était de faciliter la transmission d'informations, diminuer le coût financier et humain de la formation et d'offrir à l'utilisateur les

dispositions nécessaires à la compréhension et à l'assimilation des informations diffusées.

Comme nous allons le constater, pour satisfaire ces objectifs, les techniques qu'on a utilisées ont beaucoup évolué au fil du temps.

1.4.1 Les systèmes pédagogiques programmés

Dans les années 50, une révolution scientifique de l'enseignement a été proposée par (Skinner, 1954) s'appuyant sur le fait qu'on peut enseigner n'importe quelle notion à un élève si on utilise la technique de l'enseignement programmé. Cela peut être réalisé en découpant la matière à enseigner en petites unités et en individualisant le rythme d'apprentissage. Des machines à enseigner ont été utilisées pour assurer l'enseignement.

Dans Les années soixante, ce furent les premiers systèmes d'Enseignement Assisté par Ordinateur (EAO) basés sur l'utilisation des logiciels relativement simples au départ. Ils étaient basés sur l'utilisation des graphes pour définir des scénarios. Ces graphes imposent une succession figée d'écrans pour une suite d'actions.

1.4.2 Les systèmes pédagogiques intelligents

Les années 70 ont connu l'apparition des systèmes d'Enseignement Intelligemment Assisté par Ordinateur (EIAO). Ces systèmes ont utilisé les techniques d'intelligence artificielle pour réaliser des logiciels interactifs, souples et qui s'adaptent mieux à leurs utilisateurs.

Les années 80, ce furent les systèmes tutoriels intelligents (STI), fortement liés au développement des systèmes à base de connaissances en intelligence artificielle. A la fin des années 70, le premier système expert qui a été réalisé dans ce sens par W.J. Clancey est le système GUIDON (Quéré et al., 1991). C'est un système de résolution de problèmes de niveau expert ayant une base de connaissances utilisé dans le but de former des étudiants.

L'idée principale était la conception des systèmes d'apprentissage individualisés et basés sur des activités de résolution de problèmes. D'autres STI basés sur la résolution

d'un problème proposé par le système ou par l'apprenant ont été développés. Citons, par exemple, le système APLUSIX dans le domaine de la factorisation de polynômes (Nicaud, 1987), le système QUIZ pour l'enseignement du bridge (Futtersack, 1990), (Labat, 1990), le système SCHOLAR (Carbonell, 1970) relatif à la géographie de l'Amérique du Sud utilisant les réseaux sémantiques pour la représentation des connaissances et doté de capacités de modélisation de l'apprenant.

De manière générale, nous pouvons dire que les systèmes tuteurs intelligents sont composés des modèles de l'apprenant, des modèles d'analyse et d'explication de raisonnement et des stratégies pédagogiques.

1.4.3 Les environnements informatiques pour l'apprentissage humain

Les années 80 ont été marquées par l'évolution rapide des interfaces graphiques et l'apparition de la souris, à travers la manipulation d'interfaces homme-machine par l'apprenant, l'informatique pédagogique commence à mettre l'accent sur le développement de l'interactivité. On parlait alors des Environnements Interactifs d'Apprentissage avec l'Ordinateur (EIAO) (Bruillard, 2000).

Par ailleurs, l'interactivité dans les EIAO est destinée à favoriser l'apprentissage autonome de l'utilisateur (Bruillard, 1997). Parmi les systèmes EIAO qui ont été développés, nous citons, par exemple, le système SIAM (Courtois, 1990) d'assistance à des activités de travaux pratiques de physique qui aident l'apprenant à comprendre pourquoi le montage qu'il a réalisé ne se comporte pas comme prévu, aussi le simulateur pédagogique APLUSIX (Nicaud, 1994) permettant à l'apprenant d'agir et de voir les conséquences de son action.

Une approche se basant sur des micro-mondes ou sur des environnements d'exploration est apparue. En effet, les micro-mondes s'intéressent à l'apprentissage plutôt qu'à l'enseignement, c'est-à-dire qu'une large initiative est laissée à l'apprenant. L'exploration du domaine ou du dispositif par l'élève se fait avec un minimum de contraintes de la part du système, par la combinaison des opérations élémentaires (déplacement, construction, sélection, etc.). L'exemple le plus connu de micro-monde

est le système de géométrie LOGO (Papert, 1980) dans lequel l'utilisateur peut apprendre à un robot virtuel à réaliser toutes sortes de tâches en le programmant.

Un autre exemple de micro-monde réactif est le système CABRI-Géomètre permettant à l'utilisateur de construire et de déformer une figure géométrique, en utilisant un ensemble de commandes symboliques et d'opérations de manipulation directe. De même, d'autres simulateurs de phénomènes ou de dispositifs ont été employés dans des situations de formation. Les micro-mondes se caractérisent par l'autonomie, la réactivité et la communication. L'autonomie est une hypothèse forte des EIAH, car l'apprenant doit pouvoir être acteur de sa formation. Mais pour être efficace, l'autonomie doit être associée à un contrôle de l'activité. Ce contrôle se manifeste dans les EIAH sous la forme de réactions formulées par le programme en fonction des actions de l'apprenant, telles que l'explication des erreurs, le guidage voire la perturbation (Piché et al., 1998).

Cependant, les limites des micro-mondes résident dans le fait que certains de ces environnements sont simplement réactifs, l'utilisateur ne fait qu'exécuter des commandes.

Les travaux de recherches actuelles évoluent vers la conception et le développement des Environnements Informatiques pour l'Apprentissage Humain (EIAH). Une diversité des systèmes est couverte par le terme EIAH, le point commun entre ces systèmes est la mise en relation d'une intention didactique et d'un environnement informatique.

Un EIAH est un environnement informatique conçu dans le but de favoriser l'apprentissage humain, c'est-à-dire la construction des connaissances chez un apprenant (Tchounikine, 2002). Ce type d'environnement intègre des agents humains (élève, enseignant) et/ou artificiels et leur offre des conditions d'actions localement ou à travers les réseaux informatiques ainsi que des conditions d'accès à des ressources formatives (humaine et/ou médiatisées) (Balacheff, 1997). L'environnement peut alors avoir différents outils comme, par exemple, un outil de type hypermédia pour la présentation de l'information, un outil de traitement de l'information (un système à base de connaissances résolvant les exercices avec l'élève) ou un outil de communication (entre l'homme et la machine ou entre les hommes à travers la machine) (Tchounikine, 2002). Nos travaux de recherche se cadrent dans le courant EIAH et plus

particulièrement dans le traitement des interactions de l'apprenant avec un EIAH, afin d'avoir un modèle de l'apprenant.

1.5 L'apprentissage collaboratif/coopératif

Parmi les définitions de l'apprentissage collaboratif, on donne celle de (Henri et Lundgren-Cayrol, 1998) :

« L'apprentissage collaboratif est une démarche active et centrée sur l'apprenant. Au sein d'un groupe et dans un environnement approprié, l'apprenant exprime ses idées, articule sa pensée, développe ses propres représentations, élabore ses structures cognitives et fait une validation sociale de ses nouvelles connaissances. La démarche collaborative reconnaît les dimensions individuelles et collectives de l'apprentissage, encourage l'interaction et exploite les cognitions réparties au sein de l'environnement. Le groupe, acteur principal et ressource première de la collaboration joue un rôle de soutien et de motivation. Il contribue à l'atteinte, pour chaque apprenant, d'un but commun et partagé. La collaboration qui s'y développe est faite de communication entre apprenants, de coordination de leurs actions et d'engagement de chacun face au groupe. »

Selon ces auteurs, l'apprentissage collaboratif est une démarche conduisant à la construction progressive des connaissances. Cet apprentissage est dans le courant du constructivisme permettant à une personne de construire ses connaissances en se basant sur le processus d'interaction entre lui et son entourage.

Une autre définition basée sur l'approche de la cognition partagée, est celle de (Roschelle et Teasley, 1995). Selon cette approche, *« la collaboration est un processus de construction et de maintien d'une conception partagée d'un problème. »*

L'apprentissage coopératif est une approche pour organiser les activités d'apprentissage en groupe, et les activités concernant les expériences sociales. Les apprenants doivent travailler ensemble dans des groupes pour accomplir collectivement les deux ensembles de tâches. L'apprenant réussit si le groupe réussit.

1.6 Environnement informatique de collaboration ou de coopération

Les deux notions collaboration et coopération dénotent parfois la même chose, Selon Dillenbourg (Dillenbourg et al., 1996): « *Cooperation & collaboration do not differ in terms of whether or not the task is distributed, but by virtue of the way in which it is divided; in cooperation the task is split (hierarchically) into independent subtasks; in collaboration cognitive processes may be (hierarchically) divided into intertwined layers* »

Une collaboration est un travail en commun, entre plusieurs personnes, qui génère la création d'une œuvre commune. Un environnement informatique peut favoriser la collaboration entre pairs en permettant d'échanger et de partager des connaissances ou compétences pour mieux réussir un projet commun. Plusieurs appellations sont utilisées, par exemple, on peut parler d'apprentissage collaboratif, de travail collaboratif, de formation collaborative, etc. (Lehtinen et al., 1999). Un environnement de collaboration doit permettre aux utilisateurs de réaliser leurs tâches collectives. Selon (Salber et al., 1995), cet environnement doit offrir des services couvrant trois espaces :

- L'espace de production désigne les services permettant de créer, de modifier ou de sauvegarder les productions qui résultent d'une activité de groupe, ainsi que l'espace privé de chaque utilisateur.
- L'espace de coordination permettant de définir les acteurs, d'identifier les activités et les tâches de chaque acteur (agenda, outils de planning, etc.).
- L'espace de communication offrant aux acteurs la possibilité d'échanger de l'information via les outils de communication (e-mails, chat, forum, etc.).

Alors que la coopération est un travail organisé de façon collective dans laquelle la tâche à accomplir est divisée en sous-tâches qui seront réparties entre les acteurs. Ces derniers agissent de façon autonome. L'acteur communique, partage ou échange des informations uniquement pour atteindre son objectif individuel. A la fin, les travaux des acteurs sont réunis pour atteindre l'objectif global du travail.

Plusieurs environnements de collaboration ou coopération ont été développés, on peut citer, par exemple, les travaux sur les : CSCL ou CSCW.

1.6.1 L'Apprentissage Collaboratif Assisté par Ordinateur

En anglais, *Computer-Supported Collaborative Learning* (CSCL). Plusieurs recherches d'origine et d'approches différentes ont été menées dans le domaine des CSCL (George, 2001). Le CSCL s'intéresse aux outils informatiques utilisables pour favoriser le travail collaboratif à distance.

Parmi les environnements de CSCL, on peut citer Acolad, qui consiste à représenter les différents lieux liés à une formation présentielle: un amphithéâtre (pour l'accès aux cours), un bureau personnel (pour le dépôt de ressources), un foyer (forum), des salles de travail (chat), etc., et cela, dans le but de faciliter l'apprentissage par la collaboration. Cet environnement sert à la fois à communiquer et à partager les connaissances acquises.

Les préoccupations des recherches faites dans le domaine des CSCL sont multiples, et se diffèrent dans les traits suivants :

- La nature de la tâche : discussion sur un sujet, résolution de problèmes, écriture de documents, recherche d'informations.
- Le niveau et l'âge des apprenants : enfants, adolescents, apprenants, adultes.
- La nature des acteurs de l'activité collective : apprenants entre eux, apprenants - enseignants, apprenants - ordinateur.
- Le nombre d'acteurs : de 2 à une centaine.

Deux domaines se croisent dans le CSCL, le CSCW et l'apprentissage collaboratif défini comme des groupes de personnes travaillant ensemble dans un but commun (Crawley, 1997).

1.6.2 Le Travail Coopératif Assisté par Ordinateur

En anglais, *Computer-Supported Cooperative Work* (CSCW), qui est un domaine de recherche relativement récent et utilisant les nouvelles technologies de l'information

comme support d'un travail coopératif (Bannon et Schmidt, 1991). Ce domaine des CSCW regroupe plusieurs disciplines tels que l'informatique, la psychologie, les sciences cognitives, la sociologie, les systèmes de gestion des informations (Rodden, 1994).

Le CSCW s'intéresse à la fois à la compréhension de la nature des groupes de travail, et aussi aux technologies informatiques permettant de soutenir ces groupes (Rodden, 1991). Parmi les questions posées par ce domaine, celles essayant de comprendre comment les technologies de l'information peuvent être utilisées pour améliorer les relations collaboratives entre les apprenants (Bannon et Schmidt, 1991).

Parmi les environnements développés de type CSCW, nous pouvons citer : ShareObject qui contient des outils de collaboration (espace de partage de fichiers, espace propre aux groupes de travail créés), de communication (chat, forum, mail) et de coordination (agenda personnel, agenda partagé). ShareObject a pour but de faciliter le partage de connaissances, l'échange de ressources et la coordination lors du travail en groupe.

Donc, on peut dire que les recherches dans le domaine de CSCW ont pour objectif la création d'environnement informatique support de la collaboration au sein du groupe de travail.

1.7 Web2.0 : Modèles de l'apprenant et de contenu

Web2.0 est une évolution de web vers un cadre conceptuel plus simple de telle sorte qu'il ne demande pas de grandes connaissances pour les utilisateurs que ce soit dans le domaine technique ou bien informatique, et aussi plus interactif permettant différentes façons de contribution à l'utilisateur. Les deux dimensions qui sont prises en considération par le Web 2.0 sont : le Web sémantique et le Web social (Gandon, 2006). Le Web sémantique a pour objectif d'exploiter efficacement les ressources du Web en les rendant accessibles et utilisables par des programmes et agents logiciels.

Pour rendre le Web sémantique, les ressources web doivent être décrites par des métadonnées formelles définies explicitement par les auteurs (Brooke et al., 2005). L'idée est de dépasser la simple analyse linguistique au profit des inférences (Alesso et

al., 2004) utilisant des représentations formelles standards comme RDF, RDFS et OWL (Shen et al., 2006). La base du Web sémantique est l'utilisation des ontologies. Une ontologie est définie comme « *une description de concepts clés dans un domaine incluant les règles, propriétés et relations existants entre ces concepts* » (Alani et al., 2003).

Le Web social est le terme qu'on associe aux activités en ligne qui nécessitent une participation collaborative des utilisateurs (Foell et al., 2007). Dans ce contexte, on appelle logiciels sociaux les outils permettant une génération coopérative de contenu (Tepper, 2003). Ces outils ont une structure sociale commune. Selon (Rodriguez et al., 2007), les contributions de tous les utilisateurs sont fusionnées dans des représentations partagées d'une croyance collective. Parmi ces outils, ils existent ceux qui exploitent des modèles de connaissances pour réaliser, par exemple, une classification des utilisateurs. D'autres exploitent les ontologies pour modéliser le profil des utilisateurs comme dans (Bermudez et al., 2006) où les auteurs recommandent vivement l'utilisation d'une ontologie afin de permettre la réutilisation et l'extension de modèles décrivant le profil d'une communauté d'utilisateurs.

Dans cette partie, nous commençons par définir la notion d'ontologie et ses utilisations puis nous présentons ses composantes.

1.7.1 Les ontologies

1.7.1.1 La notion d'ontologie

Introduite pour la première fois en Intelligence Artificielle (IA) dans les années 1990, la notion d'ontologie est cependant utilisée en philosophie depuis le XIX^{ème} siècle où elle signifie la *science ou théorie de l'être*. Plusieurs définitions ont été données pour les ontologies. Cependant la définition la plus simple et connue est celle donnée par (Gruber, 1993) « *une ontologie est une spécification explicite d'une conceptualisation* ». (Brost, 1997) a légèrement modifié cette définition, selon lui c'est : « *une spécification formelle d'une conceptualisation partagée* ». Une ontologie peut également être vue comme la distinction entre les entités du monde réel et les catégories tels que les concepts, propriétés et relations pour les décrire (Bourigault et Aussenac, 2003).

En IA, Les ontologies sont utilisées pour représenter et manipuler des connaissances et aussi pour appliquer des raisonnements sur ces connaissances. Il est aussi important de souligner que les ontologies apportent plusieurs novations, comme la favorisation de la réutilisation, le partage de données et l'amélioration de la communication entre les différents modules du système, entre les utilisateurs et entre les utilisateurs et les systèmes.

1.7.1.2 Composantes d'une ontologie

Une ontologie représente l'ensemble de concepts (des idées, des principes, des catégories d'objets ou des notions) et ainsi les relations sémantiques associées existant entre les concepts.

Selon (Gruber, 1993), les connaissances traduites par une ontologie, sont représentées par les cinq éléments: Concepts, Relations, Fonctions, Axiomes et Instances. Ces éléments sont définis par (Pérez et Benjamin, 1999) comme suit:

- **Concepts** : on les appelle aussi les termes ou les classes de l'ontologie. « *Un concept est un constituant de la pensée (un principe, une idée, une notion abstraite) sémantiquement évaluable et communicable. L'ensemble des propriétés d'un concept constitue sa compréhension ou son intention et l'ensemble des êtres qu'il englobe constitue son extension* » (Psyché, 2003).
- **Relations** : elles représentent les associations existant entre les concepts présents dans le segment analysé de la réalité. Elles permettent de structurer et d'inter relier les concepts, les uns par rapport aux autres. Ces relations peuvent être de différents types : sous-classe-de (spécialisation, généralisation); partie-de (agrégation ou composition); associée-à; instance-de; est-un, etc.
- **Fonctions**: ce sont des cas particuliers de relation, dans laquelle, on définit le n ème élément en fonction des $n-1$ éléments précédents.
- **Axiomes**: ce sont des assertions, acceptées comme vraies, à propos des abstractions du domaine traduites par l'ontologie.

- **Instances:** elles constituent la définition extensionnelle de l'ontologie, ces objets véhiculent les connaissances (statiques, factuelles) à propos du domaine du problème (Psyché, 2003).

1.8 Conclusion

Dans ce chapitre, nous avons présenté une synthèse sur les travaux menés dans le domaine de l'apprentissage sur le Web. Différents environnements pédagogiques ont été exposés. Parmi lesquels on a les CSCL et CSCW. Nous avons montré qu'il est clair qu'on ne peut pas dire qu'un environnement est mieux que l'autre. Ces environnements diffèrent à plusieurs égards. Selon (Crawley, 1997), les différences entre les deux domaines sont :

- Le but du CSCL est d'aider les apprenants à apprendre ensemble. Alors que celui du CSCW est de faciliter la communication et la production de document au sein d'un groupe de travail.
- Le CSCL se focalise sur ce qui est communiqué entre les acteurs, alors que le CSCW se focalise sur les techniques de communication entre les différents acteurs
- Le CSCL est dédié au monde de l'éducation, alors que le CSCW est utilisé principalement dans les entreprises et le monde du travail.

Selon (Stahl, 2002), les deux domaines se distinguent par :

- Leur contexte : le monde du travail ou le monde de l'éducation.
- Leur but : faciliter la collaboration au sein d'une équipe de travail ou favoriser l'apprentissage au sein d'un groupe d'apprenants.

Cependant, il est également vrai que dans certains cas, le travail individuel ou coopératif n'est pas de répéter ce qui a été appris, mais il comporte l'apprentissage de nouvelles tâches. De même l'apprentissage ne peut pas consister simplement en un transfert de la connaissance existant dans des esprits passivement réceptifs. Il doit guider les étudiants pour développer les qualifications personnelles et sociales qui leurs

permettront de trouver l'information appropriée aux problèmes imprévus et de s'engager dans des processus d'apprentissage.

Dans ce sens, les domaines d'application de CSCW et CSCL sont étroitement liés (Crawley, 1997). Ces deux domaines mènent au développement d'environnement informatique en ligne permettant la collaboration que ce soit pour travailler ou pour apprendre.

Chapitre 2 : Modélisation de l'apprenant

2.1 Introduction

De par sa nature, l'être humain ne se laisse pas aisément ranger dans des catégories statiques concernant son niveau, son type d'apprentissage et sa manière d'apprendre, d'une part. D'autre part, les connaissances de l'apprenant s'évaluent dans le temps. Et étant donné que les apprenants possèdent des attitudes diverses et variées, il faut bien les modéliser pour pouvoir adapter les environnements d'apprentissage à leurs besoins. La modélisation des caractéristiques d'un apprenant est essentielle pour les applications qui exigent une adaptation.

La modélisation de l'apprenant vise la création d'un modèle cognitif et affectif à partir de l'observation du comportement de l'apprenant à l'interface d'un environnement d'apprentissage. Un modèle de l'apprenant est un composant ou un système informatisé de gestion des données qui contient des informations sur les activités d'apprentissage de l'apprenant (Tanimote, 2005). Ce modèle doit représenter le profil de l'apprenant, ses buts, ses plans, ses actions, ses croyances et ses connaissances. Et doit servir par la suite à expliquer pourquoi un apprenant ne parvient pas à réaliser correctement son travail d'apprentissage et à intervenir pendant le processus de résolution d'un problème.

Jhon Self a été le grand défenseur du modèle de l'apprenant et il a montré qu'un modèle relativement imprécis de l'apprenant pouvait être utile dans les Environnement Informatiques pour l'apprentissage humain (EIAH) (Akhras et al., 1996). Un modèle cognitif selon (Ritt. et al., 2000) est un programme informatique qui sert à simuler un comportement cognitif humain lors de la résolution d'une tâche. Le principal défi de la modélisation cognitive est de représenter un état cognitif à partir des comportements

observables d'un apprenant et selon ses besoins à l'interface d'un environnement d'apprentissage.

Par comportement observable, on désigne la suite d'actions observables exécutées par l'apprenant à l'interface d'un environnement informatique et qui représentent les entrées du processus de modélisation de l'apprenant.

Les EIAH s'intéressent aux modèles de l'apprenant pour envisager une personnalisation de l'apprentissage. Dans ces systèmes, l'apprenant doit être perçu comme une entité à part ayant un niveau et des styles d'apprentissage, des besoins et des préférences, ainsi que des connaissances et des expériences qui lui sont propres. Grâce au modèle de l'apprenant, le système peut raisonner sur chaque apprenant pour donner réponse à ses besoins pour la réalisation de l'apprentissage visé.

Le modèle de l'apprenant relève des défis importants :

- D'abord il y a la question de la cohérence du modèle de l'apprenant par rapport à ses connaissances. Ce modèle est construit à partir des éléments observés à travers le système informatique et se réfère à un apprenant qui a un comportement dynamique (changeable, contradictoire, etc.).
- Ensuite, la problématique de l'intégration d'un modèle cognitif à d'autres composants de l'application qui doivent à leur tour interpréter ce modèle et de le prendre en compte lors de la prise des décisions d'enseignement.
- Finalement le modèle est construit à partir de décision prise à partir du contenu du modèle lui-même.

Cette forte liaison entre le modèle de l'apprenant et la prise de décision conditionne:

- D'une part, l'impact du modèle de l'apprenant sur les limites stratégiques et tactiques de l'enseignement défini a priori.
- D'autre part, la qualité des prises de décisions selon la richesse du modèle de l'apprenant.

Dans le présent chapitre, nous commençons tout d'abord par rappeler le rôle du modèle de l'apprenant dans les EIAH. Ensuite nous présentons les principales approches pour la modélisation de l'apprenant. Puis nous traitons le modèle de l'apprenant suivant les critères: contenu, granularité de représentation des connaissances. Ensuite nous présentons les techniques de représentation des connaissances. Et nous exposons les techniques d'acquisition et d'analyse des traces pour envisager une modélisation comportementale. Et enfin une conclusion sur le chapitre est donnée.

2.2 Rôle du modèle de l'apprenant dans les environnements d'apprentissage

Les modèles de l'apprenant ont été largement étudiés. Au début, l'intérêt était plutôt centré sur la compréhension du raisonnement et l'identification des conceptions erronées de l'apprenant, ce qui était particulièrement difficile puisque l'état cognitif de l'apprenant n'est pas du domaine de l'observable. En outre, l'évaluation de maîtrise de la connaissance d'un apprenant constituait un autre centre d'intérêt.

Un Environnement Informatique pour l'Apprentissage Humain (EIAH) a la capacité de générer de nouvelles situations d'apprentissage et d'intervenir pendant la résolution d'un problème, et cela, en prenant en compte la connaissance de l'apprenant lors de la modélisation de l'apprenant. Le système peut intervenir pour évaluer le modèle de l'apprenant et l'ajuster dans une boucle d'interaction tout au long d'un apprentissage (Wenger, 1987), (Greer et McCalla, 1994), (Dillenbourg, 1992).

La nécessité de modéliser l'apprenant dans un EIAH a fait l'objet de nombreux travaux de recherche à la fois par rapport à sa faisabilité et à son efficacité. La fin des années 80 a été marquée par des divergences idéologiques sur la possibilité de la construction automatique d'un modèle de l'apprenant représentant l'état des connaissances d'un apprenant (Self, 1988), (Webb et al., 2001). De plus les systèmes informatiques sont devenus plus interactifs. Afin de donner un rôle plus actif et autonome à l'apprenant, ainsi est apparue la nécessité d'un modèle de l'apprenant adapté à ces nouveaux systèmes interactifs et dont le contenu ne serait pas le même que celui des systèmes précédents.

A cette époque, l'énorme difficulté de cette modélisation de l'état des connaissances de l'apprenant a été reconnue par les chercheurs. Malgré cette difficulté, certains chercheurs ont continué à travailler dans ce domaine. Parmi eux, John Self qui a été le grand défenseur du modèle de l'apprenant et les recherches ont montré que la modélisation de l'apprenant ne s'opposait pas à l'approche constructiviste de l'apprentissage (Self, 1988), (Akhras et Self, 1996).

D'après (Self, 1994), le modèle de l'apprenant sert à :

- La correction d'une erreur commise par l'apprenant.
- Déterminer le prochain contenu à présenter à l'apprenant pour donner une continuité à son apprentissage.
- Contrôler l'efficacité de stratégie de l'enseignement choisie.
- Suivre les actions de l'élève et à prévoir la performance ou l'apprentissage de l'apprenant.
- Évaluer l'apprenant.
- Et finalement représenter l'évaluation des connaissances de celui-ci.

D'après (Ragnamalm, 1996) le modèle de l'apprenant est utilisé dans quatre catégories :

- La première catégorie traite l'importance du modèle pour la planification de l'enseignement (quels contenus doivent-ils être enseignés?).
- La deuxième traite la représentation du contenu d'enseignement (quelles expériences conviennent-elles en vue de l'apprentissage d'un contenu?).
- La troisième traite la rétroaction du système qui doit prendre en compte les connaissances mobilisées précédemment par l'apprenant et aussi bien le contexte d'apprentissage courant.
- Et enfin, la quatrième traite le traitement des conceptions erronées.

D'après VanLehn (VanLehn, 1988) le modèle de l'apprenant doit permettre :

- L'augmentation des connaissances de l'apprenant, et cela, en permettant à l'apprenant de passer au thème suivant après la maîtrise du thème en cours.
- L'intervention au moment des erreurs en offrant à l'apprenant des conseils non sollicités.
- La génération des problèmes de façon dynamique plutôt que seulement gérer l'enchaînement de problèmes prédéfinis.
- L'individualisation des explications suivant le niveau de la connaissance de l'apprenant.

2.3 Approches existantes dans la modélisation de l'apprenant

Le modèle de l'apprenant représente l'écart entre les connaissances diagnostiquées chez l'apprenant et les connaissances du domaine de l'apprentissage. La tâche de diagnostic doit prendre en compte l'état cognitif d'un apprenant comme ayant une nature dynamique. Les connaissances diagnostiquées peuvent être incomplètes, contradictoires et purement liées aux contextes d'un apprentissage spécifique. De plus, le modèle de l'apprenant est le composant intégré d'un environnement et vise surtout la création d'interactions opportunistes avec l'apprenant. Ces interactions visent à la fois à aider l'apprenant et à affiner le modèle lui-même. Pour ces raisons, la recherche informatique dans ce domaine bénéficie des résultats fructueux de plusieurs champs de problèmes d'Intelligence Artificielle tels que l'apprentissage automatique, le raisonnement incertain, la résolution des conflits et la reconnaissance des plans.

Une classification pour les modèles de l'apprenant a été proposée dans (Greer et McCalla, 1994) selon la connaissance représentée dans le modèle (la prise en compte des erreurs ou non) et plus particulièrement selon les méthodes informatiques utilisées pour la construction du modèle. Selon cette classification, le modèle de l'apprenant peut être d'expertise partielle, de perturbation, de traçage, basé sur les contraintes ou bien ouvert (Webber, 2003).

Une autre classification a été proposée dans (Nguyen, 2009) selon laquelle le modèle de l'apprenant peut être : d'expertise partielle, différentiel, de perturbation, ou bien social.

Ces modèles ainsi que leurs analyses sont présentés dans les sections suivantes.

2.3.1 Modèle d'expertise partielle

Le paradigme le plus fréquent pour représenter les connaissances de l'apprenant est le modèle d'expertise partielle (figure 2) qu'on appelle aussi le modèle de couches (*overlay model*), son principe est de représenter les connaissances de l'apprenant comme une couche d'une certaine épaisseur du modèle du domaine (Carr et al., 1977). Cette approche est facilement implémentée dans les domaines de connaissances qui sont décomposables en un ensemble d'éléments. Le modèle de l'apprenant va contenir le sous-ensemble d'éléments considérés comme maîtrisés par l'apprenant. Optionnellement, à chaque élément du domaine, s'associent des propriétés quantitatives pour indiquer, par exemple, le degré de maîtrise de l'élément par l'apprenant. Et dans ce cas, l'apprentissage est associé à l'acquisition des connaissances du domaine absentes du modèle partiel de l'apprenant, le système va donc créer des interactions avec l'apprenant afin d'enrichir son modèle partiel et de le rapprocher de celui du domaine. Dans un cas d'échec d'apprentissage, le système cherche à aider l'apprenant en présentant le contenu d'une autre manière jusqu'aux limites de ces ressources, ou jusqu'à la réussite effective de l'apprenant.

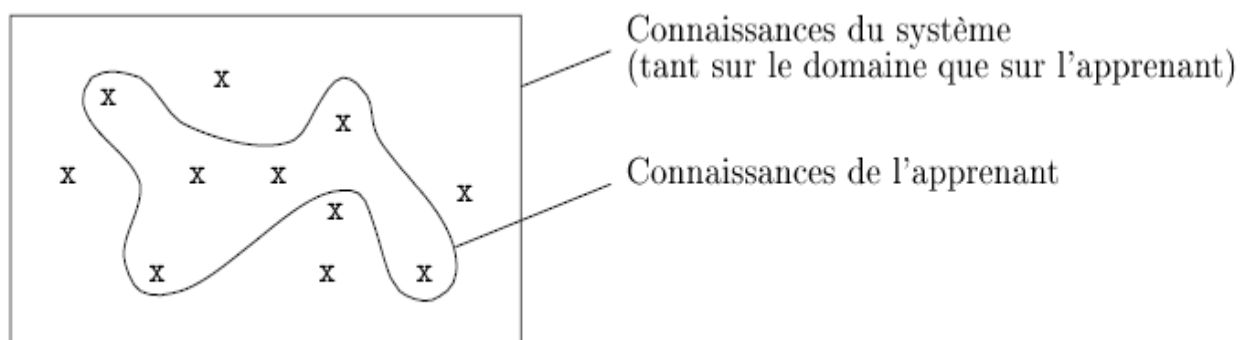


Figure 2. *Modélisation par recouvrement*

Dans la figure 3, nous montrons, à titre d'exemple, une représentation d'un modèle d'expertise partielle. Ici le domaine à étudier est décomposé en 5 concepts. Le niveau de connaissance de l'expert est égal à 1, cela exprime une maîtrise totale de chaque

concept. Pour l'élève modélisé, on remarque quelques défaillances sur les deux concepts 1 et 4.

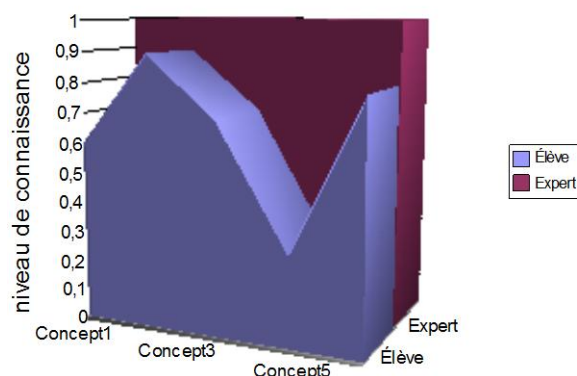


Figure 3. Exemple de modélisation par expertise partielle

On peut avoir différents types et différentes significations pour la valeur associée à chaque élément du modèle (Mayo, 2001). Dans certains systèmes, la valeur qui représente le niveau de connaissance de chaque élément est une variable modale. Ainsi, dans (De Bra, 1996) c'est une variable binaire ayant deux états acquis et non acquis. Le modèle de type expertise partielle pondérée est utilisé dans la majorité des systèmes. Dans ce cas, les variables possèdent plus de deux modalités. (Brusilovsky et al., 1998a; Grigoriadou et al., 2001) utilisent une valeur qualitative : Bon, Moyen, Mauvais pour distinguer plusieurs niveaux de connaissance.

Dans d'autres systèmes, et pour offrir une mesure du niveau de connaissance plus fine, chaque élément est représenté par une variable continue bornée (Brusilovsky et al., 1998b; De Bra et al., 2001). Cette valeur numérique peut représenter soit la fréquence avec laquelle l'élève a utilisé correctement ou incorrectement l'élément (Kay, 2000b), soit une probabilité comportant la notion d'incertitude dans la mesure (Millan, 2000). Ce dernier modèle est dit de type expertise partielle probabiliste.

Le modèle d'expertise partielle a été utilisé par plusieurs systèmes d'enseignement, chacun spécialisé dans un type d'information à maintenir (préférence, connaissances, navigation, etc.) (Frasincar et al., 2004). Par exemple, West (Burton et Brown, 1982) est un système destiné pour l'apprentissage de l'arithmétique basé sur une stratégie qui

consiste à combler les lacunes de l'apprenant. Guidon (Clanchy, 1983) est un système construit à partir de la base de connaissances du système expert MYCIN (Shortliffe, 1976) développé pour le diagnostic d'infections bactériennes. L'apprenant interagit avec Guidon et introduit ses conclusions puis le système vérifie les décisions de l'apprenant par rapport à celles que MYCIN aurait prises. STYLE-OLM (Dimitrova, 2001) est un système pour l'apprentissage de la terminologie qui a été étendu pour représenter d'autres connaissances que celles du domaine, de même il propose une modélisation ouverte de l'apprenant (section 2.3.6). Dans ce système, la connaissance correcte et incomplète de l'apprenant est représentée dans le modèle de l'apprenant, toute autre connaissance est représentée comme erronée. Donc, trois types de connaissances sont représentés dans STYLE-OLM, les connaissances correctes (reconnues par le domaine), les connaissances erronées (qui ne sont pas reconnues par le domaine) et les connaissances incomplètes (qui sont les éléments manquants, reconnus par le domaine et absents chez l'apprenant). Dans (Castillo et al., 2003), deux modèles sont utilisés, le premier modèle enregistre les croyances sur le niveau de connaissance de l'élève et un second modèle maintient les informations sur les interactions avec le système.

2.3.1.1 Analyse du modèle d'expertise partielle

Le modèle d'expertise partielle permet de situer la connaissance de l'apprenant comme correcte ou incorrecte par rapport à la connaissance du domaine. Mais il semble assez simple dès lors qu'il ne reconnaît pas la connaissance existant en dehors de la connaissance du domaine et l'interprète parfois comme erronée. Toutefois d'autres limitations demeurent comme des limitations causées par les méthodes de représentation des connaissances. Dans le domaine où l'enseignement peut être organisé selon une hiérarchie de concept, cette représentation peut être très satisfaisante. Cependant, elle est insuffisante pour la représentation des connaissances de l'apprenant dans des domaines dits complexes, comme pour l'enseignement de l'algèbre, de la physique, de la programmation ou encore de la géométrie. L'apprentissage dans ces domaines passe par la résolution du problème, la mise en œuvre de procédure de résolution et sa généralisation dans différents contextes d'apprentissage.

2.3.2 Le modèle différentiel

Une variante du modèle d'expertise partielle est le modèle différentiel (Holt et al., 1994). On considère ce modèle comme un modèle d'expertise partielle sur un niveau de connaissance demandé, qui est lui-même un recouvrement du niveau de connaissance du modèle du domaine (expert).

Dans la figure 4, les concepts qui sont en dessous du niveau demandé sont les concepts 3 et 4. Donc, on devra se focaliser sur ces deux concepts dans la séquence pédagogique qui suivrait.

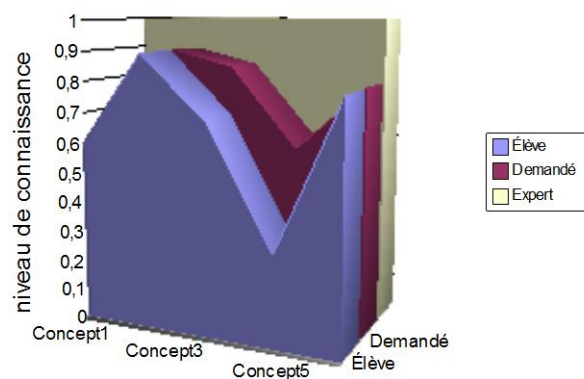


Figure 4. *Modèle différentiel*

2.3.2.1 Analyse du modèle différentiel

On peut dire que le modèle différentiel est plus flexible que le modèle d'expertise partielle, car les inférences sur le niveau de connaissance de l'élève sont focalisées uniquement sur les concepts visés. Ainsi, on ne tiendra en compte dans la séquence pédagogique qui lui serait proposée que les concepts réellement nécessaires pour un objectif d'apprentissage (Burton et al., 1978). Sauf que définir quels sont les objectifs d'apprentissage significatifs pour l'apprenant dans le domaine d'apprentissage constitue une difficulté pour le modèle. De plus, aucune solution quant à l'analyse d'une erreur persistante de l'apprenant sur l'acquisition d'un concept particulier n'est apportée par ce modèle.

2.3.3 Modèle de perturbation

La principale limitation du modèle d'expertise-partielle et du modèle différentiel est la considération des connaissances de l'élève comme une simplification de la connaissance experte, car dans ces deux modèles les mauvaises connaissances préalables et les erreurs de raisonnement lors de l'apprentissage sont ignorées. Par contre, le modèle de perturbation (figure 5) propose de représenter la connaissance de l'élève comme un sous-ensemble de la connaissance de l'expert complété avec un modèle des erreurs connues a priori à partir des analyses de protocoles d'apprenants ou découvertes lors de l'apprentissage par l'analyse de l'activité. Autrement dit, le modèle de perturbation perçoit la connaissance de l'apprenant comme un ensemble de perturbation dans la connaissance experte.

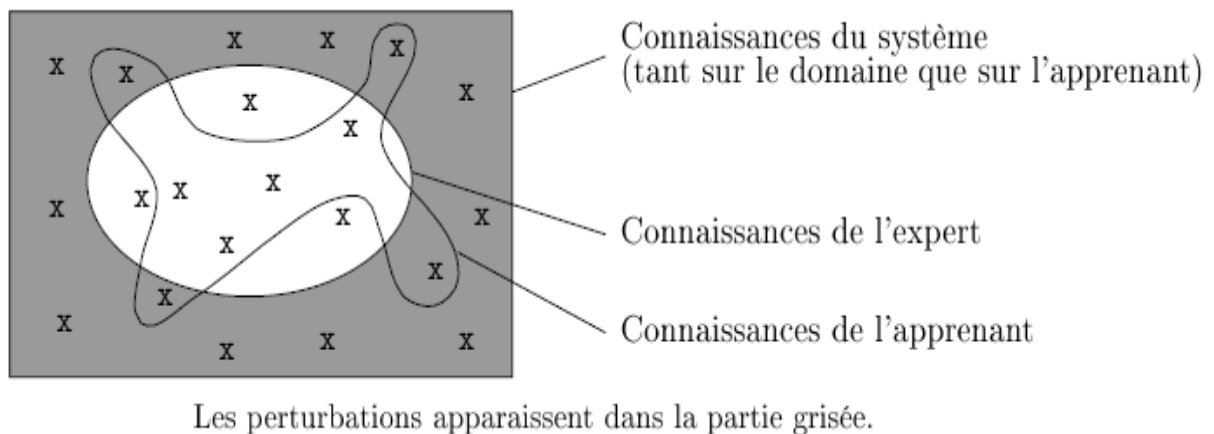


Figure 5. *Modèle de perturbation*

Ce modèle permet à l'apprenant d'avoir une représentation dynamique. Ce dynamisme procède des techniques issues des travaux sur l'apprentissage automatique et/ou la discrimination des erreurs systématiques et les procédures de résolutions. L'inconvénient des méthodes d'apprentissage est qu'elles nécessitent un nombre représentatif d'exemples pour le fonctionnement de l'algorithme. Cependant, les erreurs les plus fréquentes sont associées à des plans de résolution que l'apprenant pourrait mettre en œuvre dans le contexte d'un problème. Et chaque erreur systématique issue d'une action erronée ou incomplète de l'apprenant lors de la résolution du problème est aussi associée à une conception erronée dans le domaine de

l'apprentissage. Ainsi, des bibliothèques d'erreurs (bogues) sont construites et incorporées aux systèmes. Pour cette raison, on dit souvent que le modèle de perturbation combine un modèle d'expertise partielle et une bibliothèque d'erreurs (Greer et McCalla, 1994). Les bibliothèques de bogues permettent de déclencher une rétroaction particulière du système vers l'apprenant chaque fois qu'une procédure est reconnue comme erronée.

Parmi les systèmes construits sur le modèle de perturbation, nous pouvons citer :

- Le système Buggy qui est un système développé comme un jeu éducatif pour l'entraînement des futurs enseignants.
- Le système Andes (domaine physique) qui est un système de préceptorat destiné aux étudiants universitaires.

2.3.3.1 Analyse du modèle de perturbation

Un nombre important de systèmes d'apprentissage ont utilisé le modèle de perturbation. Cette technique a l'avantage d'utiliser les erreurs de l'apprenant pour raisonner sur les conceptions erronées apprises par cet apprenant. Ainsi, le système cherche une interprétation à l'erreur commise par l'apprenant qui va mener le système à identifier des bogues dans les procédures de résolution employées par l'apprenant. Une fois les bogues sont identifiés, le système d'enseignement peut mettre en place avec plus de pertinence une stratégie d'enseignement pour aider l'apprenant.

Cependant, l'inconvénient majeur des modèles de perturbation réside dans le fait de concevoir l'ensemble des perturbations aussi appelé bibliothèque d'erreurs car c'est très coûteux en temps. De même, la conception de ce catalogue peut par ailleurs requérir à une analyse manuelle détaillée d'un nombre important de réponses d'apprenant.

En conclusion, pour la modélisation par perturbation on utilise souvent les méthodes probabilistes telles que les réseaux bayésiens.

2.3.4 Modèle de traçage

Le modèle de traçage permet de construire une simulation dynamique de la résolution de l'apprenant en parallèle de son travail et dans ce cas le système peut interagir avec

l'apprenant à chaque pas de sa résolution à l'aide d'un ensemble de règles de production, le système reconstruit la solution de l'apprenant et l'utilise pour simuler et interpréter le comportement de l'apprenant.

Cet ensemble de règles est constitué par les règles correctes et incorrectes. De cette façon, dès qu'une erreur est détectée ou dès qu'une solution s'éloigne de la résolution correcte, une rétroaction est régénérée.

Un des principaux intérêts de cette approche- est de pouvoir interagir avec l'apprenant à chaque pas de sa résolution. De toute évidence ce modèle exerce un contrôle sur la solution construite par l'apprenant en le protégeant de la poursuite de la résolution dans une direction qui ne conduirait pas à la solution correcte.

Le modèle de traçage a d'abord été utilisé dans trois domaines d'apprentissage : le système Lisp Tutor (Anderson et Reizer, 1985) pour le langage Lisp, le système Géométrie Tutor pour la géométrie élémentaire et les systèmes Algebra I et Algebra II pour l'algèbre (Anderson et al., 1990 ; Anderson et al., 1995).

Un autre exemple basé sur le modèle de traçage est POLA (Probabilistic ON-Line Assessment) (Conati, 1996), il est appliqué dans les domaines où pour chaque problème proposé, il peut y avoir plusieurs solutions possibles. POLA a été validé dans le domaine de la physique et il génère pour chaque problème proposé à l'apprenant un graphe représentant l'ensemble des solutions pour ce problème. Et à partir de ce graphe, POLA va suivre le progrès de l'apprenant. Parallèlement et pour chaque action de l'apprenant qui est représentée par une règle, l'incertitude créée est manipulée par un réseau bayésien et elle peut être renvoyée à de multiples solutions. Donc, à chaque action de l'apprenant, POLA retrouve dans le graphe la solution ou les solutions applicables et les ajoute au réseau bayésien. Le réseau calcule la probabilité qu'une solution soit celle suivie par l'apprenant.

2.3.4.1 Analyse du modèle de traçage

Les systèmes basés sur le modèle de traçage sont connus par haut niveau d'intervention dans le travail de l'apprenant. Il est évident que le suivi de la solution de l'apprenant est souhaitable, mais le contrôle ne l'est pas. En effet, la diversité de solutions correctes et

incorrectes dépasse largement la capacité de représentation et de manipulation d'un système lorsqu'il doit agir sur chaque pas de la solution de l'apprenant. La solution proposée par ce modèle consiste justement à réduire le spectre d'action de l'apprenant en contrôlant ses actions à chaque étape de la solution afin d'éviter que l'erreur se propage. L'apprenant peut suivre les indications du système et apprendre par cœur la solution sans maîtriser la connaissance apprise.

2.3.5 Modèle basé sur les contraintes

Le modèle basé sur les contraintes (Ohlsson, 1994) permet de représenter la connaissance de l'apprenant et celle du domaine sous la forme de contraintes. On caractérise une contrainte par une clause de pertinence et une clause de satisfaction. La clause de pertinence est la condition qui doit être vraie pour que la contrainte soit adaptée à la solution d'un problème donné, une fois que la clause de pertinence est vraie, si la clause de satisfaction est également vraie, alors la solution est correcte.

Cette approche a été utilisée dans le système SQL-Tutor (Mitrovic, 1998) pour l'enseignement du langage SQL. Le domaine est représenté par un ensemble de contraintes de type syntaxique (pour vérifier la syntaxe de la solution de l'apprenant par rapport à la syntaxe d'SQL) et sémantique (pour vérifier que la réponse contient l'information demandée). Pour chaque problème, le système enregistre la solution idéale attendue. Un exemple de contrainte a été proposé dans (Martin, 1999) :

« Si l'énoncé du problème demande un sous-ensemble de colonnes, la solution ne doit pas contenir la commande SELECT *. »

2.3.5.1 Analyse du modèle basé sur les contraintes

Les critiques faites sur ce modèle concernent la construction de l'ensemble des contraintes, à chaque fois qu'on ajoute un nouveau problème, de nouvelles contraintes doivent s'ajouter. Et à chaque nouvel ajout de contraintes, la consistance de l'ensemble de contraintes doit être vérifiée.

2.3.6 Le modèle social

Le principe du modèle social est de classer un apprenant parmi un ensemble de stéréotypes prédéfinis. Ces stéréotypes sont utilisés pour guider le choix d'une séquence pédagogique. Pour définir les stéréotypes, on procède à une analyse des réponses données par un échantillon d'apprenants suite à un questionnaire orienté. Par exemple, (Milne et al., 1996) ont utilisé un test de personnalité psychologique pour définir une classification basée sur la personnalité des élèves.

Avant de débiter la séquence, tout nouvel apprenant doit répondre au questionnaire initial. L'analyse de ses réponses permettrait de calculer la probabilité qu'il appartienne à chaque stéréotype. Et il est donc associé au stéréotype le plus probable (figure 6).

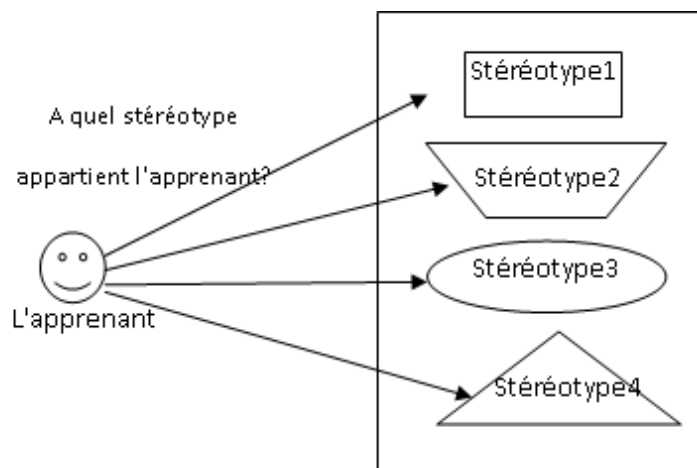


Figure 6. *Modèle social*

Deux sortes de modèles sociaux ont été utilisées :

- Le modèle « fixe »: un apprenant qui a été associé à un stéréotype, ne peut jamais être affecté à un autre stéréotype (Winter et al., 1999). Le problème se pose si, par exemple, l'apprenant a prétendu dans le questionnaire avoir une certaine compétence et qu'il s'avère au cours de la séquence qu'il n'en est rien. Il faudrait alors pouvoir lui réviser l'affectation.
- Le modèle « par défaut »: il consiste à considérer que l'affectation initiale de l'apprenant à un stéréotype n'est faite que pour avoir des paramètres initiaux. Ensuite, l'observation des activités menées permettra l'affinement de ces

paramètres afin de les personnaliser (Kay, 2000a). L'apprenant est alors affecté au nouveau stéréotype estimé le plus probable.

2.3.6.1 Analyse du modèle social

Ce modèle présente comme difficulté la complexité du nombre de stéréotypes différents à prendre en considération dans un environnement d'apprentissage intelligent, car la création des stéréotypes qui doivent être tous disjoints est pratiquement impossible. L'intersection de deux stéréotypes pose le problème de l'affectation de l'apprenant à l'un des stéréotypes plutôt qu'à l'autre. De même, dans le cas où l'apprenant ne parvient pas à suivre les activités pédagogiques qui lui sont proposées, le problème qui se pose avec ce modèle est l'identification de la cause et de la solution à apporter dans ce cas.

2.3.7 Modèles ouverts (inspectable)

A l'opposé des approches traditionnelles où le modèle de l'apprenant est caché aux yeux de l'apprenant, le modèle ouvert permet à l'apprenant d'inspecter, de discuter et de changer son modèle. Le modèle ouvert de l'apprenant peut être plus précis par rapport à celui obtenu sans l'aide de l'apprenant selon (Dimitrova, 2001), car il encourage la réflexion sur les concepts appris et la négociation sur les imperfections du modèle entre l'apprenant et le système. Évidemment, l'intérêt, la motivation de l'apprenant ainsi que son habilité à s'exprimer font partie des qualités indispensables pour l'obtention d'un modèle bien abouti.

Parmi les systèmes basés sur la modélisation ouverte, nous trouvons le modèle du système **STYLE-OLM** (Dimitrova, 2001) basé sur la méthode expertise partielle étendue pour l'apprentissage de la terminologie scientifique dans une langue étrangère. Dans ce système, l'apprenant dialogue à propos du contenu de son modèle avec le système. Ce dernier agit comme un guide de dialogue. Cette communication opère comme un outil de réflexion et d'appréhension du domaine par l'apprenant.

Un autre exemple, qui est le système I-Help (Vassileva et al., 1999) basé sur une architecture multi-agents pour l'apprentissage en autonomie et en favorisant l'aide

entre les apprenants. L'apprenant est modélisé selon sa capacité (connaissances sur les sujets de cours) et sa volonté pour aider les autres.

2.3.7.1 Analyse du modèle ouvert

L'un des avantages des modèles ouverts est que quand l'apprenant est engagé dans un dialogue à propos du contenu de son modèle, il réfléchit sur son domaine de connaissances et ses expériences de ce qui était passé inaperçu auparavant et il peut se rendre compte. Mais il existe certains risques lorsque l'apprenant a le contrôle sur son modèle (Kay, 2001). Ça peut être du, par exemple, au fait que l'apprenant peut saboter l'environnement ou, par accident.

2.4 Contenu du modèle de l'apprenant : connaissances et préférences

Plusieurs aspects du comportement de l'apprenant sont concernés par la tâche de la modélisation de l'apprenant : les concepts, règles ou procédures de résolution maîtrisées, les conceptions erronées, la motivation à apprendre, la vitesse de résolution de problèmes, etc.

Le choix du contenu du modèle de l'apprenant dépend de plusieurs facteurs parmi lesquels on peut citer : les caractéristiques du domaine d'enseignement, du modèle cognitif utilisé, des objectifs pédagogiques du système d'apprentissage, des types d'interaction avec l'apprenant et des types de rétroactions souhaitées par le système. Tous ces facteurs vont contribuer à déterminer le contenu initial du modèle de l'apprenant et son enrichissement à partir de l'engagement de l'apprenant.

Le modèle de l'apprenant est utilisé pour concevoir dynamiquement la présentation et l'organisation du contenu pédagogique au cours d'un apprentissage.

Selon (Cristea et al., 2004a), concevoir un modèle de l'apprenant consiste à répondre à la question suivante :

- Quels sont les attributs qui devraient être capturés par le modèle de l'apprenant?

Deux problèmes ont été soulevés par la conception du modèle de l'apprenant dans (Grandbastien et Labat, 2006) :

- Le premier problème consiste à choisir les caractéristiques individuelles (cognitives, comportementales) qui devraient être représentées dans le modèle de l'apprenant et de leur mode de représentation.
- Le second problème consiste à mettre en œuvre un mécanisme de mise à jour de ce modèle à partir des événements ou observables recueillis durant l'interaction.

Parmi les systèmes adaptatifs d'enseignement que nous avons étudiés pour répondre à ces questions, celui de (Trella et al., 2002) dans lequel le modèle de l'apprenant a été scindé en un modèle de connaissances et un modèle de préférences. Dans ce système, le modèle de l'apprenant utilisé est de type expertise partielle sur le modèle conceptuel de contenu. Le domaine à étudier est organisé sous la forme d'un graphe dont les nœuds représentent les concepts et les arcs représentent les notions de dépendances ou de composition entre les concepts.

Le modèle de connaissances de l'apprenant contient des informations sur le niveau de connaissance de l'apprenant correspondant à chaque concept du modèle conceptuel de contenu. C'est un modèle à deux couches : la couche «verified» qui contient le niveau de connaissance des concepts qui ont été évalués sommativement, et la couche «estimated» qui contient l'estimation du niveau de connaissance obtenue en traitant les diverses informations concernant le comportement de l'apprenant et collectées pendant la session d'apprentissage (ex : pages visitées, temps d'accès aux pages). Selon cette approche, les concepts non connus sont les concepts dont les pages n'ont pas été visitées.

Un modèle similaire a été utilisé dans le système ELM-ART-II de l'auteur Brusilovsky (Weber et al., 1997), mais il a plus de deux niveaux (concepts : visited, evaluated, inferred, etc.). Ces systèmes, on leur reproche la non-prise en compte des connaissances acquises par l'apprenant au préalable, car dans certains cas, la séquence pédagogique proposée risque de ne pas être adaptée.

Le modèle de préférences contient des informations concernant les caractéristiques de l'apprenant autres que le niveau de connaissance (l'identité, l'âge, la capacité linguistique, etc.). Les données de ce modèle sont décrites comme un ensemble de couples (attribut, valeur).

Dans (Schiaffino et al., 2000), le modèle utilisé contient l'ensemble des préférences de l'apprenant obtenues par un agent qui observe le comportement de l'apprenant dans son activité d'apprentissage et mémorise toutes les requêtes faites par l'apprenant sur une période donnée. Par la suite, des techniques de raisonnement sont utilisées afin de prédire l'information à proposer à l'apprenant. Par exemple, lorsqu'un apprenant interroge tous les jours le site de la météo pour connaître le temps qu'il fait. Lors des futures connexions de cet apprenant, le système sera capable de lui fournir cette information sans qu'il en fasse la demande.

2.4.1 Modélisation cognitive (des connaissances)

La modélisation des connaissances a été utilisée dans plusieurs systèmes d'enseignement adaptatifs. Son principe est d'associer des attributs à chaque concept du domaine à étudier. Ces attributs apportent différentes informations sur l'apprenant. La mise à jour de la valeur de chaque attribut est effectuée en fonction des activités de l'apprenant dans la séquence pédagogique.

Dans le modèle de type expertise partielle utilisé dans le système hypermédia adaptatif orienté Web AHA développé dans (De Bra et al., 2003), le niveau de connaissance d'un concept donné est représenté par un seul attribut appelé « knowledge ». La mise à jour de la valeur de cet attribut s'effectue de différentes façons, la première consiste à mémoriser que le contenu des pages reliées au concept a été parcouru par l'apprenant. La deuxième solution consiste à proposer à l'apprenant un questionnaire en ligne afin d'évaluer son niveau de connaissance, et cela, permet une évaluation plus fine que simplement enregistrer que le contenu a été lu.

ALEA est un autre système hypermédia adaptatif orienté Web de type expertise partielle pondéré qui est utilisé dans l'enseignement du cours de programmation logique et fonctionnelle (Kostelnik et al., 2003). Pour chaque concept abordé,

l'apprenant indique manuellement son niveau de connaissance en utilisant, par exemple, la valeur «understood» qui précise qu'il a bien compris ce concept. La même approche est utilisée dans le système ISIS-Tutor (Brusilovsky et al., 1994) où à chaque concept, on associe les valeurs : not-ready-to-be-learned, ready-to-be-learned, in-work et learned. Et aussi dans le système ELM-ART (Weber et al., 2001) où les valeurs utilisées sont : visited, learned, inferred et known.

LAOS est un framework théorique (Cristea, 2004b) créé dans le but de proposer le développement de systèmes auteurs hypermédias adaptatifs pour le Web. Il est basé sur le principe d'exploiter plusieurs modèles de type expertise partielle. Le modèle de connaissances de LAOS est du type différentiel. Il se compose des trois couches suivantes :

- Le modèle conceptuel de contenu (DM) : il est représenté sous forme d'un graphe de concepts auxquels sont associés des attributs. Selon les auteurs, le concept peut représenter une ressource (comme dans RDF [W3C]) sauf qu'on ne peut avoir des concepts qui représentent des ressources et des objectifs pédagogiques dans le même modèle.
- Le modèle d'objectifs et de contraintes (GM) : dans ce modèle, le modèle conceptuel de contenu (DM) est filtré, regroupé et restructuré en fonction d'un objectif pédagogique. Les nœuds du graphe sont reliés par des arcs de type AND-OR, qui peuvent être des relations (d'ordre, de dépendances, etc.) entre plusieurs concepts afin d'exprimer des contraintes pédagogiques pour la progression d'un élève dans le modèle de contenu (DM). L'exemple de la (figure 7) montre que le modèle GM est composé de cinq concepts et quatre relations. Dans ce graphe, pour étudier le concept C4, l'élève devra maîtriser les concepts C1 et C2, par contre l'élève n'est autorisé à étudier le concept C5 que lorsque son niveau de connaissance sera suffisant sur C2 ou C3.
- Le modèle de l'utilisateur (UM) : il est représenté par un graphe de concepts construit en se basant sur le modèle d'objectifs et de contraintes (GM). A chaque nœud du graphe, on associe différents attributs (Cristea et al., 2003) comme le

niveau de connaissance de l'élève sur le concept correspondant qui est considéré comme le principal attribut.

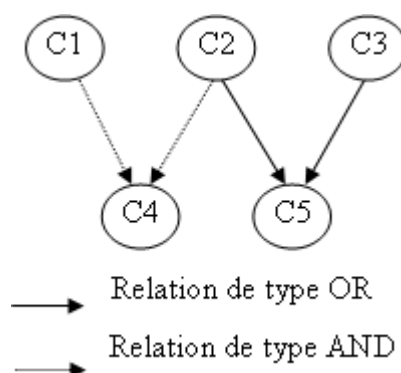


Figure 7. Exemple de relations AND-OR dans le modèle GM

Ce modèle de connaissances multicouches permet d'identifier les concepts qui sont la cause d'un échec dans la séquence pédagogique. En revanche, il est toujours impossible de détecter si la cause de l'échec est due à un niveau de connaissance insuffisant ou à la séquence pédagogique qui est inadaptée au style d'apprentissage de l'élève.

La détermination du niveau de connaissance de l'apprenant dans le modèle pour chaque concept est une tâche complexe (Kristofic et al., 2005). Pour aider à cette tâche, certains systèmes utilisent comme attributs (Cristea et al., 2004a) :

- Le temps passé par concept : la mesure utilisée est une estimation de la connaissance qualifiée généralement par l'utilisation d'un état à deux valeurs : « consulté » ou « non consulté ». L'hypothèse posée est que l'état « consulté » signifie que le concept est acquis.
- La performance mesurée par une évaluation sommative : la mesure utilisée est la performance de l'apprenant qui est généralement une valeur numérique appartenant à un intervalle (ex : [0,20]).

Le modèle de connaissances de l'apprenant proposé dans (Hibou et Labat, 2004) est basé sur un réseau bayésien composé de deux couches: domaine et activité.

- La couche domaine représente le profil de l'apprenant, elle se compose de nœuds «Knowledge» et «Know-How» qui expriment des règles de conception

pédagogique reliant les concepts à apprendre «Knowledge» aux savoir-faire «Know-How».

- La couche activité représente les activités pédagogiques de l'apprenant sur lesquelles il travaille, elle se compose de nœuds « Item » qui représentent les éléments d'évaluation des activités. Ces nœuds sont reliés aux nœuds « Know-How » de la couche domaine par une relation de dépendance.

La probabilité $P(item \mid Know - How)$ exprime le niveau de difficulté de l'exercice. L'hypothèse proposée ici est que dans un environnement pédagogique, le niveau de difficulté de la tâche est directement lié à l'objectif d'enseignement. L'observation des nœuds « Item » permet de montrer que l'information sur le niveau de connaissance de l'apprenant est propagée aux nœuds de la couche domaine.

2.4.2 Modélisation affective (des préférences)

La modélisation affective ou la modélisation des préférences a été utilisée dans les systèmes hypermédia adaptatifs. De par leur nature personnelle, les préférences ne sont pas modélisées de la même manière comme dans le cas de la modélisation des connaissances. Les préférences sont utilisées pour indiquer comment l'apprenant préfère percevoir et traiter l'information. Elles sont souvent représentées par des couples *attribut – valeur* (Piombo, 2007).

Dans le modèle utilisé dans (Klyne et al., 2003), l'attribut langue est utilisé pour définir la langue à utiliser pour les textes des contenus pédagogiques. D'autres modèles précisent si on doit inclure le son ou la vidéo dans la séquence pédagogique. Ou bien à inclure d'autres informations concernant le matériel utilisé pour suivre la séquence pédagogique (Résolution de l'écran, nombre de couleurs, etc.).

(Frasincar et al., 2004) proposent un modèle de présentation qui gère l'affichage des contenus selon les préférences de l'apprenant. Par exemple, pour permettre d'afficher des images, on assigne la valeur « yes » à l'attribut « imageCapable ».

La modélisation des préférences est généralement basée sur le principe de la manipulation de ces attributs afin d'adapter la séquence pédagogique aux préférences

de l'apprenant. Et ce, en utilisant un ensemble de règles exprimées sous forme de structures conditionnelles (Cristea et al., 2004c). Selon (Brusilovsky, 2003b), les techniques d'adaptation utilisées sont l'annotation, la gestion du contenu des pages à visualiser, la gestion de l'apparition des liens et enfin la gestion de matériels pédagogiques.

Dans (Cristea et al., 2004a), d'autres informations qui caractérisent le style d'apprentissage, le sexe, la personnalité, etc. sont inclus dans le modèle de l'apprenant.

Dans (Trella et al., 2002), les caractéristiques de l'apprenant représentées dans le modèle de préférences sont représentées dans le tableau suivant :

Caractéristiques de l'apprenant	valeur
Développement cognitif ou la capacité à formaliser et comprendre des concepts	Elevé, moyen, bas
Motivation	Elevé, moyen, bas
Style d'apprentissage	Théorie, exercice
Expérience avec l'ordinateur	Beaucoup, normal, peu, aucun
Débit internet	Elevé, moyen, bas

Table 1. *Modèle de préférences de l'apprenant*

Le système d'apprentissage des statistiques proposé dans (Schuyten et al., 2007) utilise un modèle de préférences qui précise, d'une part, le mode préféré de représentation de l'information (information textuelle, graphique ou symbolique), d'autre part, les aspects comportementaux de l'apprenant (degré d'intérêt pour les statistiques, l'aptitude à utiliser un ordinateur).

Pour recueillir ces informations, l'apprenant doit remplir des questionnaires en début d'année. Les problèmes de la validité et de la mise à jour des informations recueillies sont dans ce cas posés, car l'apprenant trouve généralement des difficultés à exprimer ses préférences. En plus, le système ne permet pas la modification en cours de l'année des informations recueillies.

2.5 La granularité de la représentation de la connaissance de l'apprenant

Lors de la modélisation de l'apprenant, l'aspect « granularité de la connaissance » doit être pris en considération. Ce terme fait référence au niveau de détail dans lequel un concept est présenté.

Articuler un cours autour de grains de connaissances est dû au fait que les apprenants raisonnent à plusieurs niveaux de granularité à la fois. Cela signifie qu'ils peuvent montrer une connaissance plus approfondie ou plus superficielle sur un concept dans la résolution d'un problème.

Dans le cadre de notre approche de modélisation, nous avons choisi de découper le contenu pédagogique autour des concepts à appréhender. Ainsi, *« les auteurs de ressources restent libres quant à la réalisation de leurs supports et n'ont pas à respecter une quelconque charte de rédaction, qu'elle soit graphique, structurelle ou relative au contenu »* (Abel, 2008). Ces concepts permettent d'indexer les documents qui traitent de ces derniers donnant ainsi la possibilité de réutiliser ces documents.

2.6 Techniques de représentation de la connaissance

Avec l'apparition des systèmes de tuteurs intelligents (STI), la modélisation des connaissances a connu un essor important. En effet, les informations du modèle de l'apprenant doivent avoir une représentation formelle pour pouvoir être exploitées par le système. Le modèle de connaissances de l'élève sert à représenter des informations sur le niveau de connaissance de l'élève au regard de chaque concept considéré dans le modèle conceptuel de contenu.

De nombreuses techniques de modélisation des connaissances ont été utilisées dans les STI.

Dans cette section, nous présentons les plus répandues ainsi que leurs avantages et inconvénients.

2.6.1 Les systèmes à base de règles

Un système à base de règles est composé de trois parties : une base de règles, une base de faits et un moteur d'inférences.

La base de règles contient l'ensemble des règles de production du système et représente la connaissance que l'expert a du domaine. Quant au moteur d'inférences, c'est lui qui contrôle l'activité du système. La règle de production est de la forme : SI <condition> ALORS <action>.

Cette technique de représentation est utilisée dans un grand nombre de STI pour modéliser les connaissances de type procédural (Haton et al., 1991). La connaissance est modélisée dans ce cas, par un ensemble de règles de production (Kass, 1987). Parmi les systèmes qui ont utilisé cette technique pour la représentation des connaissances: le système MYCIN pour le diagnostic médical, le tuteur (Anderson, 1983) basé sur la théorie ACT*, le tuteur Buggy (Brown et Burton, 1978).

2.6.1.1 *Avantages de la technique à base de règles*

Dans ces systèmes à base de règles, le processus de résolution de problèmes est exprimé facilement en termes de buts et de sous buts à atteindre par la réalisation des successions d'actions. Ces systèmes expriment également la connaissance de façon uniforme par des règles et des faits (Barr et Feigenbaum, 1981).

2.6.1.2 *Inconvénients de la technique à base de règles*

L'inconvénient majeur des systèmes à base de règles est que la connaissance n'est pas structurée. De plus, s'il y' a plusieurs règles qui parlent d'un même objet, alors l'information que représente cet objet se trouve éparpillée dans les différentes règles.

2.6.2 Les réseaux sémantiques

Un réseau sémantique est un graphe composé d'un ensemble de nœuds et d'arcs orientés. Les nœuds représentent des concepts, des entités, des objets, des attributs, des événements, des états, etc.

Un arc sert à lier entre deux nœuds et représente une relation binaire entre ces concepts. Les arcs sont généralement étiquetés par le type de la relation modélisée.

Un réseau sémantique permet de structurer hiérarchiquement la connaissance. Les réseaux sémantiques ont été utilisés pour la première fois par Quillian (Haton et al., 1991) pour modéliser la mémoire associative humaine.

Parmi les systèmes qui ont utilisé les réseaux sémantiques pour représenter les connaissances on cite le système Scholar dans le domaine de la géographie (Nicaud et Vivet, 1988), (Kass, 1987).

2.6.2.1 Avantages des réseaux sémantiques

La représentation à base de réseaux sémantiques est bien adaptée aux domaines où on a une forte liaison entre les concepts. Ces réseaux sont faciles à visualiser. Et comme les réseaux sémantiques sont représentés sous forme de graphe, la récupération des informations implicites ou explicites de la base de réseaux par le mécanisme de filtrage est rendue facile.

2.6.2.2 Inconvénients des réseaux sémantiques

Dans les réseaux sémantiques, chaque nœud représente une seule information qui est son nom, complété par une étiquette. Cette simplification rend l'expression des propositions ayant des quantificateurs universels, existentiels et numériques plus difficile. La représentation du monde par les réseaux sémantiques est statique, par conséquent la modélisation de l'évolution de l'information est aussi difficile.

2.6.3 Les systèmes basés sur la logique

Parmi les premières techniques utilisées par l'intelligence artificielle pour la formalisation de la connaissance, on trouve la logique mathématique. La logique des prédicats du premier ordre est la plus utilisée. Elle consiste à représenter la connaissance de l'apprenant sous la forme d'un ensemble de formules. Des valeurs de vérité sont associées à ces formules ainsi qu'un mécanisme d'inférence de nouvelles formules.

Parmi les STI qui ont utilisé la logique mathématique pour représenter les connaissances de l'apprenant, on peut citer :

- Le système Scent (Brecht et al., 1989), c'est un conseiller en programmation Lisp qui formalise la connaissance en se basant sur la logique classique.
- Le système SMIS (Student Modeling Inference System) (Ikeda et al., 1989), c'est un système qui utilise le formalisme de la logique multivaluée utilisant plus de deux valeurs de vérité (par exemple la valeur inconnue).
- Et enfin, le tuteur West (Burton et Brown, 1982) qui utilise la logique probabiliste pour la formalisation de la connaissance, cette logique associe à une formule une valeur de vérité répartie sur l'ensemble des réels compris entre 0 et 1, et représente la probabilité que cette formule soit vraie.

2.6.3.1 Avantages de la technique basée sur la logique

Le formalisme de la logique est clair et non ambigu, car d'une part, la signification donnée à une formule dépend uniquement de sa structure et de la signification de ses composants, d'autre part, le langage d'expression logique est proche du langage naturel.

2.6.3.2 Inconvénients de la technique basée sur la logique

Parmi les inconvénients de l'utilisation de la logique dans la représentation de la connaissance, on peut citer:

- La difficulté de la représentation des objets complexes, car les objets, les propriétés et les relations entre les objets sont tous décrits par leurs noms qui ne portent aucune information sur le sens ou la structure.
- L'inefficacité des algorithmes utilisés par les mécanismes de raisonnement pour la résolution de problèmes nécessitant un grand volume de connaissances.
- La difficulté de la représentation des connaissances incomplètes par la logique des prédicats.

2.6.4 Les systèmes basés sur les réseaux de Bayes

Les réseaux de Bayes, également appelés réseaux bayésiens ou réseaux de croyances sont les techniques les plus utilisées dans la modélisation de l'apprenant. Cette

technique probabiliste a été développée dans le cadre des recherches pour décrire l'incertitude de faits en intelligence artificielle.

Un réseau bayésien est un graphe orienté acyclique ayant un ensemble de nœuds et un ensemble d'arcs orientés. Les nœuds représentent des variables aléatoires (Pearl, 1988) et où la topologie du graphe sert à exprimer les dépendances entre ces variables.

Formellement, un réseau bayésien est un couple (G, V) où G est un graphe acyclique orienté, V est un ensemble $\{X_1, \dots, X_n\}$ de variables aléatoires.

La loi de probabilité jointe de $X_1, \dots, X_n$ se factorise en :

$P(X_1, \dots, X_n) = \prod (P(X_i \mid \text{pa}(X_i)))$ où $\text{pa}(X_i)$ est l'ensemble des parents de X_i dans le graphe.

Pour déterminer la loi de probabilité jointe, il suffit de connaître les probabilités a priori pour les nœuds n'ayant pas de parents et des probabilités conditionnelles des nœuds sachant leurs parents.-

L'exemple très pédagogique utilisé dans la présentation des réseaux bayésiens est celui de la pelouse de Sherlock Holmes (Becker et Naïm, 1999):

Sherlock Holmes sort de chez lui le matin pour aller travailler. En arrivant à sa voiture, il constate que sa pelouse est mouillée alors qu'il ne pleut pas. Il se dit qu'il a dû oublier de couper son système d'arrosage la veille au soir. Il se dirige vers la cave pour aller l'arrêter quand il jette un coup d'œil sur la pelouse de son voisin Watson : elle est mouillée. Il rebrousse alors chemin et monte dans sa voiture pour se rendre à son travail.

Son raisonnement peut être représenté comme un parcours dans un réseau causal comportant quatre faits et trois relations.

Voici les quatre faits (nœuds booléens) qui peuvent prendre la valeur "Vrai" ou "Faux" :

- P (pluie) représente le fait qu'il ait plu la nuit précédente.
- A (arroseur) représente l'oubli de l'arroseur automatique.
- J (jardin) représente l'état du jardin d'Holmes.

- V (voisin) représente l'état du jardin du voisin Watson.

Donc, la situation peut être représentée par le graphe suivant (figure 8) :

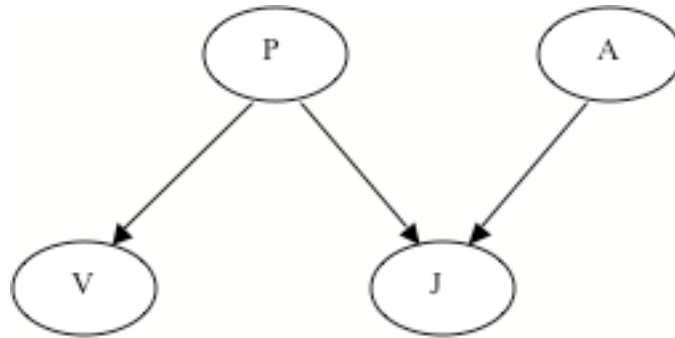


Figure 8. Réseau de Bayes de l'exemple

En termes de causalité, les arcs entre les variables s'interprètent ainsi :

La pluie est l'unique cause possible à l'humidité du jardin de Watson, alors que la pelouse d'Holmes peut avoir été arrosée soit par la pluie, soit par l'arroseur automatique.

Donc, voici les trois relations causales (arcs) qui lient les nœuds :

1. $P \rightarrow J$: s'il a plu, la pelouse de Holmes est mouillée
2. $P \rightarrow V$: s'il a plu, la pelouse de Watson est mouillée
3. $A \rightarrow J$: si le système d'arrosage de Holmes n'a pas été coupé alors la pelouse de Holmes est mouillée

On a donc : $P(A, P, J, V) = P(J | P, A) \times P(V | P) \times P(A) \times P(P)$

On obtient les lois de probabilités de nœuds P, V, J et A :

$P(P = \text{vrai}) ; P(V = \text{vrai}) ; P(J = \text{vrai}) ; P(A = \text{vrai})$

Ici l'objectif n'est pas de rentrer dans le détail des calculs, mais de faire une rapide présentation du concept.

Un événement de type $X_i = x_i$ représente une observation dans un réseau bayésien. Après avoir fait une conjonction d'observation dans le réseau, on peut calculer l'inférence dans le réseau qui consiste à mettre à jour les probabilités, c'est-à-dire on cherche à déterminer la loi d'une ou plusieurs variables aléatoires sachant que la variable aléatoire ou le vecteur de variables aléatoires E prend la valeur e , et on écrit $P(X|E = e)$. On peut dire donc que les croyances sont actualisées par les observations.

Pour construire un réseau de Bayes, il faut déterminer la structure du réseau et définir les tables de probabilités. Parmi les approches qui permettent la construction du réseau bayésien, on peut trouver le recueil d'expertise, l'apprentissage automatique à partir de données ou bien la combinaison des deux approches.

Dans le domaine de la modélisation de l'apprenant, la structure est généralement déterminée à l'aide d'experts du domaine et les tables de probabilités sont définies à partir de données recueillies en situation réelle.

Nous présentons deux environnements utilisant les réseaux de Bayes dans le domaine de la modélisation de l'apprenant : Hydrive et Andes.

Hydrive (Mislevy et Gitomer, 1996) est un système pour l'apprentissage de la maintenance du système hydraulique et qui prend en compte trois types de connaissances : les connaissances techniques en avionique, les connaissances stratégiques, et les connaissances procédurales.

Dans ce système, un arbre est utilisé pour structurer l'ensemble de ces connaissances. A la racine, on trouve le nœud « compétence générale », Les premières branches sont constituées par les trois types de connaissances mentionnées. La probabilité que la compétence ou la connaissance de l'apprenant soit acquise est inférée par le système en se basant sur les réponses de l'apprenant. Donc, le but étant d'inférer l'état des connaissances de l'apprenant à partir de ses actions face à un environnement. Dans ce cas, le diagnostic qui est effectué est épistémique.

Andes (Conati et al., 2002) est un environnement pour l'apprentissage de la résolution de problèmes de physique. Ce système permet d'évaluer les compétences d'un apprenant, de reconnaître le plan suivi lors de résolution de problèmes et de prédire ses

actions à venir. Un graphe obtenu à l'aide d'un résolveur à base de règles et contenant la description de toutes les solutions acceptables est associé à chaque problème. Ce graphe est ensuite converti en réseau de Bayes. Pour permettre une évaluation épistémique de l'apprenant, les probabilités des nœuds du domaine sont examinées et mises à jour.

2.6.4.1 Avantages des réseaux bayésiens

Les réseaux de Bayes permettent la représentation facile des liens de causalité dans le modèle de l'apprenant. Généralement, les informations du modèle de l'apprenant sont liées les unes aux autres. Autrement dit, les connaissances de l'apprenant influent les unes sur les autres, par exemple, la motivation à apprendre a une influence directe sur la capacité d'effectuer la tâche. Et comme le modèle de l'apprenant a un caractère hypothétique, en utilisant les probabilités, l'incertitude est traitée facilement.

2.6.4.2 Inconvénients des réseaux bayésiens

Les réseaux de Bayes présentent quelques inconvénients liés à leur construction. En effet, comme nous l'avons dit, la structure du réseau est généralement déterminée à l'aide des experts du domaine et les probabilités sont définies à partir de données recueillies en situation réelle. Donc, dans quelques cas et selon le domaine, la construction du réseau par des experts est vraiment nécessaire.

2.7 Acquisition du modèle de l'apprenant

Nous abordons dans cette section le recueil des informations qui doivent être dans le modèle de l'apprenant, deux questions se posent :

- Comment recueillir les valeurs des attributs du modèle de l'apprenant ?
- Quand effectuer ce recueil (en début, à la fin ou bien en cours de la session) ?

Pour répondre aux questions précédentes, des solutions ont été envisagées dans (Cristea et al., 2004a). La première solution consiste à saisir directement quelques attributs concernant l'apprenant en début de session (ex : background knowledge, learning preference). La deuxième solution consiste à saisir indirectement des

informations par le biais d'un questionnaire (ex: style d'apprentissage) comme dans (Papanikolaou et al., 2003).

La troisième solution est effectuée par le système pour évaluer en cours de la session les attributs indirects de l'apprenant en analysant, par exemple, sa navigation (pages ou liens visités), ainsi que le temps passé sur un concept donné. Comme dans (Brusilovsky, 2004a) qui mémorise, entre autres, les activités de l'apprenant dans la session (ex : page lue, réponse à une question, etc.).

Le recueil des informations concernant l'apprenant s'effectue donc d'une manière : statique, dynamique ou mixte.

- **Recueil statique** : c'est le mode de recueil le plus simple qui consiste à demander explicitement à l'apprenant ses préférences, généralement par le biais d'un questionnaire soumis à l'apprenant en début de l'activité. Selon (Manber et al., 2000), ce mode d'acquisition trouve vite ses limites, car les apprenants ont du mal à exprimer leurs besoins et leurs préférences à travers les questionnaires. Pour y remédier, certains auteurs demandent à l'apprenant d'exprimer son sentiment sur un ensemble de documents pour en déduire les préférences (Billsus et Pazzani, 1999).
- **Recueil dynamique** : le recueil dynamique consiste à observer l'apprenant en situation d'apprentissage et mémoriser ses traces d'activités pour inférer le modèle de l'apprenant (Broisin et al., 2006). Ce mode de recueil est plus intéressant, car il permet de capter les interactions de l'apprenant sans le détourner de sa principale tâche. Selon (Kristofic et al., 2005), la meilleure façon d'acquérir les informations caractérisant l'apprenant est celle basée sur les fichiers de logs qui sont créés tout au long de la session.
- **Recueil mixte** : le recueil mixte combine à la fois une phase statique pour initialiser le modèle de l'apprenant et une phase dynamique pour la mise à jour du modèle en observant les interactions de l'apprenant avec le système.

Selon les sources d'observations, nous pouvons distinguer plusieurs méthodes de recueil pour les environnements d'apprentissage : méthodes centrées serveur, méthodes

centrées client, ou bien méthodes utilisant des outils spécifiques à l'environnement d'apprentissage.

2.7.1 Le recueil centré serveur

L'approche de recueil centrée serveur est basée sur l'analyse des fichiers logs présents sur les serveurs Web. Ces fichiers logs contiennent l'ensemble des événements et des actions marqués temporellement et effectués sur le serveur. L'analyse des fichiers logs est un processus très complexe nécessitant plusieurs opérations, à savoir, le nettoyage, la structuration, le filtrage, l'identification des sessions, etc. Parmi les informations qui peuvent être récupérées à partir des fichiers logs, on peut citer : la date et l'heure d'accès, le navigateur et le système d'exploitation utilisés, l'adresse IP du client, la durée de consultation d'une page, la réponse du serveur, le type d'erreur rencontré, etc.

Cette démarche a été utilisée par plusieurs systèmes, parmi lesquels nous pouvons citer les travaux de (Iksal et Choquet, 2005) qui s'intéressent à l'acquisition des traces à partir des fichiers logs. Ainsi que les travaux de (Stermsek et al., 2007) portant sur l'exploitation des fichiers *logs* des apprenants et les métadonnées des pages HTML pour déduire leurs profils.

Le recueil côté serveur est très intéressant. Cependant, il présente quelques inconvénients : le premier inconvénient est que les informations issues d'applications s'exécutant sur le poste client ou d'applications produisant du contenu dynamique ne sont pas enregistrées dans les fichiers logs, ainsi les informations réellement consultées par les utilisateurs ne sont pas déterminées selon (Fansler et Riegel, 2004). Le deuxième réside dans le fait que les fichiers logs ne sont généralement pas disponibles pour tous les utilisateurs (juste les administrateurs réseau et système).

Pour pallier à ces inconvénients, d'autres approches de recueil coté utilisateur ont été proposées.

2.7.2 Le recueil centré client

Les interactions concernant les recherches effectuées par l'apprenant sur le Web lors de son travail d'apprentissage ne sont pas enregistrées sur le serveur, malgré leur

importance dans l'explication du parcours de l'apprenant. Il est donc intéressant de recueillir toutes les interactions de l'apprenant sur le poste utilisateur. Cette approche de recueil est encore un domaine de recherche (Beauvisage, 2004).

Parmi les travaux utilisant cette approche, nous pouvons citer le travail de (Blanchard et al., 2005) qui collecte les traces des utilisateurs (pages visitées, mode de navigation). (Beauvisage, 2004) utilise des sondes de recueil de trafic Internet installées sur les postes des utilisateurs pour décrire les parcours des internautes à partir des URLs visitées. Dans le domaine des EIAH, un agent *keylogger* a été proposé par (Loghin, 2008). Cet agent collecte tous les événements clavier et souris produits par l'apprenant, les processus exécutés ainsi que les titres et le contenu des boîtes de dialogue affichées sur l'écran.

En EIAH, cette approche est implémentée par très peu de travaux. Généralement, on utilise les *logs* serveur ou les outils spécifiques à l'environnement tracé.

2.7.3 Outils spécifiques à l'environnement

Dans cette approche, la collecte s'effectue à travers un outil spécifique à l'environnement dont on veut tracer.

Parmi les travaux utilisant cette approche, nous pouvons citer l'environnement collaboratif eMédiathèque (Michel et al., 2005) qui traite les interactions de l'utilisateur à partir de deux modèles : l'un pour les outils mis à disposition tels que les ressources, le navigateur Internet, la messagerie instantanée, etc. ; et l'autre pour les actions réalisables par l'utilisateur (création, modification, suppression de contenus, etc.).

L'environnement Knowledge Pool System (KPS) de la fondation ARIADNE enregistre les interactions de l'utilisateur (ressources indexées, téléchargées ou supprimées, l'utilisateur effectuant l'action, date de création, etc.) (Cram et al., 2007) avec l'interface de KPS dans un fichier au format XML (Najjar et al., 2004).

Certains atouts sont présentés par les approches spécifiques. Cependant, leur aspect spécifique et leur cloisonnement constituent un inconvénient, car les informations collectées ne peuvent pas être traitées par d'autres systèmes à base de traces (SBT).

2.8 Analyse des Traces

L'analyse des traces d'interaction de l'apprenant avec un environnement d'apprentissage est devenue un thème de recherche en forte évolution, et ce, pour plusieurs raisons parmi lesquelles on peut citer :

- La multiplicité des acteurs concernés par l'analyse en tant qu'observés ou qu'observateurs (apprenant ou groupe d'apprenants, enseignant, tuteur, administrateur, concepteur, agents virtuels, etc.).
- Le volume important d'informations numériques recueillies.
- Et enfin, la diversification des modalités de communication avec les environnements d'apprentissage.

L'analyse des traces vise à comprendre et suivre les apprentissages d'un apprenant ou d'un groupe d'apprenants et à qualifier l'utilisation, l'utilisabilité et l'acceptabilité de l'environnement pour le rendre plus adaptatif (Cottier et Choquet, 2005). Pour atteindre ces objectifs, de nombreux travaux de recherche sont menés sur ce thème et portant sur le recueil, au cours des sessions d'apprentissage, des données appelées *traces* et définies comme une séquence temporelle d'observés. L'exploitation de ces traces fournit des connaissances sur l'activité qu'on appelle indicateurs d'apprentissage.

2.8.1 Trace

2.8.1.1 Définition

Il existe plusieurs définitions que l'on peut donner au terme trace. Une trace est une « *observation ou un enregistrement de l'interaction de l'apprenant avec un système en vue d'une analyse* » selon (Jermann et al., 2001).

Pour (Pernin, 2005) une trace est définie comme un « *indice de l'activité des acteurs d'une situation d'apprentissage, qu'elle soit ou non instrumentée* ». Il rajoute aussi qu'il s'agit « *d'un résultat obtenu au cours ou au terme d'une activité, d'un événement ou d'un ensemble d'évènements relatif au déroulement de la situation d'apprentissage* ».

Pour (Champin et al., 2004) une trace est une séquence d'états et de transitions représentant l'activité de l'utilisateur : *«la séquence temporelle des objets et opérations mobilisés par l'utilisateur lorsqu'il utilise le système est appelée trace d'utilisation ».*

(Choquet et Iksal, 2007) définissent la trace comme *« un objet pédagogique au même titre que les ressources ou les scénarios pédagogiques ».*

Et selon (Cram et al., 2007) une trace d'interactions est définie comme *« tout objet informatique dans lequel s'accumulent des données à propos des interactions entre un système informatique et son utilisateur ».*

De manière abstraite et générale, la trace est définie comme *« une séquence temporelle d'observés »* (Settouti, 2006). Le terme *« séquence temporelle »* dénote l'existence d'une relation d'ordre organisant les données de la trace par rapport à un repère de temps. Le terme *« observé »* dénote que les données de la trace sont issues d'une observation.

Notons par les définitions précédentes qu'une trace est une trace d'activité, d'utilisation ou d'interaction, et qu'aucune interprétation sur la situation d'apprentissage ne lui est associée. On dit alors des traces primaires, brutes, de base ou de *« bas niveau »*. Des traces pertinentes et de *« plus haut niveau »* sont construites par le traitement et la transformation des traces de *« bas niveau »*.

2.8.1.2 Objectifs des traces dans le EIAH

Dans le cadre des EIAH, l'étude à base de traces ne vise pas seulement comment analyser les traces, mais aussi comment les compléter et exploiter pour améliorer l'apprentissage (settouti, 2006) ?

Parmi les travaux qui ont été menés dans le contexte des EIAH afin de soutenir l'observation on peut citer les travaux qui traitent l'analyse du comportement de l'apprenant et la caractérisation de ses activités (Sanderson et Fisher, 1994), (Georgeon et al., 2006) et (Loghin, 2008). Et aussi ceux qui traitent l'interprétation des interactions de l'apprenant avec les environnements informatiques et avec les autres acteurs (Siebra et al., 2005), (Barros et Verdejo, 2000) et (George, 2004).

Les traces sont aussi utilisées dans la scénarisation de l'activité pédagogique de l'apprenant (Ferraris et al., 2005). La scénarisation pédagogique doit permettre à l'enseignant de décrire une séquence d'apprentissage destinée aux apprenants et adaptée à ses objectifs pédagogiques. Une comparaison entre le parcours prescrit par l'enseignant et le parcours effectif de l'apprenant contribue à la régulation du processus d'apprentissage (Heraud et al., 2005). Pour pouvoir effectuer cette comparaison, la mise en œuvre d'un langage dans lequel les observables sont retranscrits est proposée par (Iksal et al., 2005).

Récemment, des travaux considérables de recherches ont été faits dans le domaine d'automatisation de l'acquisition et de la distribution des connaissances. Il y a plusieurs environnements d'apprentissage où les interactions entre le système et les utilisateurs sont tracées, on peut citer, par exemple : Drew (Dialogical Reasoning Educational Web tool) (Séjourné et al., 2004) qui est un environnement d'apprentissage collaboratif. APLUSIX (Nicaud, 1987) qui est un environnement pour aider les élèves à apprendre le calcul arithmétique et algébrique. Synergo (Avouris et al., 2005) qui est un environnement permettant le suivi et l'analyse des activités de petits groupes d'apprenants distants engagés dans des situations de résolution de problèmes collaboratifs. Et ColAT (Collaboration Analysis Tool) (Avouris et al., 2005) qui est un outil indépendant de tout système d'apprentissage permettant l'analyse des activités collaboratives à partir des fichiers logs et de leurs enregistrements vidéo.

Le point commun entre tous ces systèmes est que les traces obtenues doivent passer par différentes étapes avant d'être analysées et exploitées. Dans ces systèmes, les interactions entre les utilisateurs et l'environnement sont tracées dans plusieurs fichiers peu exploitables par l'utilisateur lui-même ou un analyste de la situation d'interaction, ce sont les fichiers logs (Ollagnier-Belbame, 2007). Ces fichiers logs fournissent un matériel très conséquent, facile à collecter, mais plus difficile à analyser. Le temps passé par le tuteur pour suivre et analyser les activités de l'apprenant est beaucoup plus que celui dépensé dans la situation de classe (Al-Sakran, 2006). Pour cela, la nécessité d'aider et d'assister les utilisateurs et les analystes des traces dans leurs activités est mise en évidence.

2.8.2 Les approches de traitement des traces numériques

Le besoin d'analyser et de traiter des traces complexes n'a cessé de s'accroître. Afin d'être exploitées, les traces doivent passer par plusieurs étapes. Le nombre d'étapes et les appellations attribuées aux étapes se différent d'un travail à l'autre. Mais malgré cette différence, on peut distinguer dans le processus de traitement des traces, trois phases principales: la phase de collecte, la phase d'analyse et la phase d'exploitation.

Les traitements des traces sont formalisés à l'aide de Systèmes à Base de Traces (SBT). Un SBT est un « système informatique, avec ou sans mémoire, gérant une base de traces modélisées dans le cadre d'un système observé et offrant des services de transformation, de requête, de partage, etc. » (Hulin, 2011). Parmi les SBT proposés, on peut citer :

Le modèle de (Jermann et al., 2001) qui a été conçu pour permettre de réguler des situations d'apprentissage collaboratif. Ce modèle est un cycle qui traite les traces en quatre étapes : la phase de collecte des données, la phase de structuration en indicateurs, la phase de diagnostic et la phase de remédiation.

L'approche Musette (*Modelling USEs and Tasks for Tracing Experience*) (Champin et al., 2004) a été conçue dans un objectif de permettre la création d'une assistance à l'apprenant basée sur la réutilisation d'expérience (Mille et Prié, 2006). L'approche MUSETTE permet de définir trois phases différentes pour le traitement des traces : la première phase est celle de collecte des traces primaires en format natif. Dans la deuxième phase qui est la phase d'analyse, la description des signatures de tâches permet de structurer les traces primaires en traces de haut niveau qualifiées d'épisodes. En fonction du contenu des épisodes, des algorithmes d'assistance de l'apprenant sont appliqués lors de la dernière phase qui est la phase d'assistance.

Et enfin, l'approche de (Setoutti et al., 2006) qui proposent de définir et de formaliser le processus de traitement par un système à base de traces (SBT). Ils définissent le Système à Base de Traces comme « un système informatique permettant et facilitant la manipulation des traces ». Dans ce système, on parle du modèle de trace qui décrit le contenu de la trace en utilisant un ensemble de concepts et de relations types.

2.9 Modélisation de contenu

Le modèle de contenu des systèmes d'enseignement est constitué d'un ensemble d'éléments représentant des fragments élémentaires de connaissances du domaine étudié (Brusilovsky, 2003c). Ces éléments souvent organisés en une hiérarchie sont nommés (concepts, éléments de connaissances, sujets). Et ils peuvent être de différents types.

Le modèle de contenu hiérarchique du système ALEA est composé de concepts de types :

Text, Programming scheme, Exercise, Test. Chaque concept est décomposé en éléments suivants: Text, Exercise definition, Exercise hint, Exercise solution, Source code.

Le système MEDEA (Trella et al., 2002) possède un modèle de contenu organisé sous la forme d'un réseau sémantique de connaissances. Le système propose de représenter le domaine à étudier par un réseau sémantique de concepts et des données pédagogiques. MEDEA exploite trois éléments : *les concepts*, *les relations* entre les concepts et *les types d'évaluation* pour représenter le niveau de connaissance de l'apprenant.

A chaque concept est assigné un attribut dont sa valeur indique le seuil minimum requis pour estimer la maîtrise du concept étudié. Plusieurs types de relation peuvent être utilisés (*prerequisite*, *part of*, *is a*, *belongs to*, *is useful to understand*, *is similar to* et *is opposite to*).

Cette représentation ne permet pas d'associer des activités pédagogiques à un concept et de définir le matériel pédagogique utilisé par l'apprenant pendant la session d'apprentissage.

Le système proposé dans (Brusilovsky et al., 2003a) possède un modèle de contenu similaire au précédent. Les nœuds représentent les éléments de connaissances (*concepts*, *sujet*, *règles*, etc.), et les arcs représentent des opérateurs booléens (ET, OU).

Deux nœuds A et B sont reliés au nœud C par un opérateur de type ET, cela signifie que pour commencer l'étude du concept associé au nœud C, il faut que les concepts associés

aux noeuds A et B soient acquis. Si c'est de type OU, cela signifie que la maîtrise de l'un des deux concepts suffit.

Un autre attribut a été ajouté aux arcs. Il prend les valeurs suivantes : *temporel, causal, analogie, pré-requis, agrégation ou généralisation*.

Les nœuds et les arcs sont reliés aux matériels pédagogiques constituant les ressources manipulées par l'apprenant pendant la session. Ces ressources peuvent être (*introduction, explication, exemple, exercice ou test*) avec différents médias (*texte, image, animation, vidéo, etc.*).

2.10 Position de notre approche de modélisation

Notre volonté, avec notre approche de modélisation de l'apprenant, est d'opérationnaliser les connexions entre *elarning* et *learner modelling*. Pour ce faire, notre objectif est de modéliser et concevoir un environnement d'apprentissage tenant compte à la fois de ces deux aspects.

Côté e-learning, ces dernières années, la modélisation des environnements d'apprentissage a été étudiée en ingénierie éducative selon deux principales approches (Abel et al., 2007) :

- L'approche par les ressources, basée sur le paradigme des objets pédagogiques.
- L'approche par les activités, basée sur les notions d'unités d'apprentissage, d'activité et de scénario pédagogique.

Côté modélisation de l'apprenant « un modèle de l'apprenant représente l'état de l'ensemble des connaissances de l'apprenant par rapport aux connaissances du domaine de l'apprentissage » (Webber, 2003).

Dans le cadre de cette approche, nous proposons de modéliser et de concevoir un environnement d'apprentissage collaboratif selon l'approche par les ressources définie dans (Abel, 2007) et en tenant compte de certaines caractéristiques de l'utilisateur apprenant. Parmi ces caractéristiques : le niveau de connaissance que pour l'estimer,

nous avons choisi de combiner un réseau de Bayes avec un modèle d'expertise partielle associé au modèle de connaissances basé sur les objets pédagogiques.

Nous avons fait le choix de tester et d'évaluer l'apport des agents et des ontologies dans un contexte d'apprentissage. Rappelons qu'un tel environnement que nous avons baptisé E-PACAD (Plateforme d'apprentissage collaboratif à distance) doit encourager l'apprentissage à différents niveaux (individu, groupe et organisation).

2.11 Conclusion

Dans ce chapitre, nous avons présenté un état de l'art relatif à notre problématique. Pour individualiser l'apprentissage, les EIAH élaborent un modèle de l'apprenant permettant de personnaliser et d'adapter l'environnement d'apprentissage aux besoins de l'apprenant.

Nous avons choisi de présenter les modèles de l'apprenant suivant plusieurs critères :

Le premier critère concerne le type du modèle de l'apprenant. On distingue plusieurs types, parmi lesquels on peut citer le modèle d'expertise-partielle, différentiel, de traçage, basé sur les contraintes, de perturbation, social, etc.

A partir de ces méthodes de la modélisation de l'apprenant, nous avons pu observer la complexité de la tâche du diagnostic cognitif de l'apprenant dont les résultats ne permettent pas toujours de créer des conditions d'une évolution dans le sens de l'apprentissage recherché.

Le deuxième critère concerne le contenu du modèle. Généralement, deux catégories de données peuvent être représentées par le modèle de l'apprenant : le niveau de connaissances de l'apprenant ou les préférences de l'apprenant.

Le troisième critère concerne l'acquisition du modèle qui peut être statique, dynamique ou mixte.

Il est compréhensible que les cadres théoriques de caractère cognitif employés couramment selon un modèle informatique simplifié envisagent seulement la modélisation de certains aspects cognitifs de l'apprenant, cependant, nous soutenons

l'idée que le traitement informatique des interactions de l'apprenant requiert aussi un cadre théorique capable de modéliser certains aspects épistémiques du comportement observé chez l'apprenant. Comme conséquence de ce fait, un modèle de base épistémique permettrait la représentation des connaissances associées aux actions de l'apprenant. Plus précisément le modèle de l'apprenant pourrait représenter les actions de ce dernier en termes de connaissances investies dans la résolution d'un problème.

Donc, Nous y avons distingué quatre catégories à faire figurer dans notre modèle : les informations générales sur l'apprenant, les préférences, les aspects comportementaux et les connaissances de l'apprenant.

Nous avons également parlé des différentes méthodes de représentation des connaissances. Parmi les quelles, on trouve les réseaux bayésiens qui constituent la technologie la plus puissante de ces dernières années en IA et en apprentissage automatique selon (Moore, 2004). Dans le chapitre 4, nous allons parler de notre approche de modélisation cognitive qui consiste à combiner un modèle d'expertise partielle avec un réseau bayésien.

Chapitre 3 : Agents, systèmes multi-agents et méthodologies de conception

3.1 Introduction

Les notions d'agents et systèmes multi-agents (SMA) ne sont pas faciles à définir pour certains raisons : l'une est que plusieurs significations sont données à ces deux notions. Une autre est que les systèmes à base d'agents sont très peu implémentés.

Ce chapitre a ainsi pour principal objectif de présenter l'architecture multi-agents retenue pour notre environnement E-PACAD. Nous commençons par présenter les notions d'agents et de systèmes multi-agents pour situer notre travail. Ensuite, nous présentons quelques méthodes d'analyse et de conception des systèmes multi-agents en se focalisant sur la méthodologie *Multi agent System Engineering* (MaSE) que nous avons choisie pour analyser et concevoir notre environnement E-PACAD au sein duquel opèrent nos agents gestionnaires de notre modèle de l'apprenant. Les détails concernant l'application de cette approche sur notre environnement ne seront exposés dans le détail qu'aux chapitres suivants.

3.2 Agent

3.2.1 Définitions

Définir la notion d'agent n'est pas un sujet simple. Il n'en existe pas une définition ou signification unique donnée à cette notion. C'est la raison pour laquelle plusieurs auteurs essaient d'en donner une définition avant d'utiliser le paradigme dans tel ou tel contexte, ainsi la notion d'agent est utilisée dans plusieurs domaines: sociologie, biologie, psychologie cognitive, psychologie sociale et bien sûr informatique.

Aujourd'hui avec l'avènement de nouvelles technologies et de l'expansion de l'Internet, ce concept est associé à plusieurs applications comme agent ressource, assistant personnel, agent interface, agent ontologique, etc.

On trouve donc une multitude de définitions qui se ressemblent toutes, mais diffèrent selon le type d'application pour laquelle l'agent est conçu. L'une des premières définitions de l'agent est due à (Ferber, 1995) :

Il définit l'agent comme étant une entité physique ou virtuelle:

- a. qui est capable d'agir dans un environnement.
- b. qui peut communiquer directement avec d'autres agents.
- c. qui est mue par un ensemble de tendances sous la forme d'objectifs individuels ou d'une fonction de satisfaction qu'elle cherche à optimiser.
- d. qui possède des ressources propres.
- e. qui est capable de percevoir de manière limitée son environnement.
- f. qui ne dispose que d'une représentation partielle de cet environnement.
- g. qui possède des compétences et offre des services.
- h. qui peut éventuellement se reproduire.
- i. dont le comportement tend à satisfaire ses objectifs en tenant compte des ressources et des compétences dont elle dispose et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.

(Jennings et Wooldridge, 1998) ont proposé la définition suivante pour un agent :

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.

Les notions "situé", "autonomie" et "flexible" sont définies comme suit :

Situé : l'agent est capable d'agir sur son environnement à partir des entrées sensorielles qu'il reçoit de ce même environnement.

Exemples : systèmes de contrôle de processus, systèmes embarqués, etc.

Autonome : l'agent est capable d'agir sans l'intervention d'un tiers (humain ou agent) et contrôle ses propres actions ainsi que son état interne.

Flexible : l'agent dans ce cas est :

- Capable de répondre à temps : l'agent doit être capable de percevoir son environnement et d'élaborer une réponse dans les temps requis.
- Proactif : l'agent doit exhiber un comportement proactif et opportuniste tout en étant capable de prendre l'initiative au "bon" moment.
- Social : l'agent doit être capable d'interagir avec les autres agents (logiciels et humains) quand la situation l'exige afin de compléter ses tâches ou d'aider ces agents à accomplir leurs tâches.

3.2.2 Les Types d'agents.

Dans le paragraphe précédent, nous avons essayé de rechercher une définition consensuelle pour la notion d'agent servant de base de travail, et aussi nous permettant de nous positionner par rapport aux types d'agents qui nous intéressent. Selon les auteurs, il existe plusieurs typologies d'agents.

Nous présentons quelques-unes :

Des chercheurs en intelligence artificielle comme dans (Shoham, 1997) et (Wooldridge et Jennings, 1995) catégorisent souvent les agents en des agents dits faibles et des agents forts. Conceptuellement, les agents forts possèdent des propriétés mentales et émotionnelles.

(Moulin et Chaib-Draa, 1996) caractérisent les agents par leur capacité à résoudre les problèmes, selon eux un agent peut être :

- Réactif : il réagit aux changements de son environnement ou aux messages provenant des autres agents.

- intentionnel : il est capable de raisonner à propos de ses intentions, ses croyances et peut concevoir et exécuter des plans d'actions.
- social : il possède en plus des propriétés d'un agent intentionnel, des modèles explicites des autres agents.

La classification proposée dans (Ferber, 1995) catégorise les agents en fonction de leur caractère cognitif où chaque agent dispose d'une base de connaissances ou bien réactif où chaque agent dispose des mécanismes de réactions aux événements. Ces deux grandes familles d'agents sont aussi classées par Ferber selon leurs capacités de résolution de problèmes :

3.2.2.1 Les agents cognitifs

Les agents cognitifs sont capables d'anticipation et peuvent planifier leur environnement. Ils sont constitués d'une base de connaissances qui contient toutes les informations sur leur fonctionnement. Les agents cognitifs ont la capacité de raisonner sur cette base de connaissances, sur les intentions et sur les buts pour prendre des décisions suivant une planification. L'agent acquiert de nouvelles connaissances soit par l'échange d'informations avec les autres agents soit par le biais de ses perceptions sur son environnement.

3.2.2.2 Les agents réactifs

Contrairement aux agents cognitifs, les agents réactifs sont incapables d'anticipation et ne peuvent pas planifier leur environnement. Ils n'ont pas aussi une grande capacité de raisonnement, car ils ne disposent pas de connaissances explicites de l'environnement et des autres agents. Ils ne disposent pas de connaissances sur leurs passés et sur leurs buts. La seule capacité qu'ils possèdent répond au principe de stimulus/action.

Une autre typologie des agents est proposée dans (Nwana, 1996), selon laquelle les agents sont classés selon plusieurs critères de classification :

- Selon la mobilité : où l'agent est soit statique soit mobile.
- Selon la présence d'un modèle de raisonnement symbolique que ce soit délibératif ou que ce soit réactif.

- Selon l'existence d'un objectif et de propriétés initiales comme l'autonomie, la coopération et l'apprentissage.
- Selon les rôles joués par l'agent comme, par exemple, la recherche d'informations ou le travail sur Internet ;
- Selon la philosophie hybride c.à.d. combinaison de deux ou de plusieurs approches dans un seul agent.

Les sept catégories d'agents proposées dans cette typologie sont : agents collaboratifs, agents d'interface, agents mobiles, agents d'information ou d'Internet, agents réactifs, agents hybrides, agents intelligents.

3.2.2.3 Les agents hybrides

Les agents hybrides ou mixtes sont généralement utilisés afin de minimiser les faiblesses et d'augmenter les forces de tel ou tel type d'agents que l'on mettrait en œuvre dans une application. Les agents hybrides utilisent les deux approches réactive et cognitive. Ils consistent donc en la combinaison de plusieurs caractéristiques au sein d'un même agent. Ces caractéristiques concernent la mobilité, la collaboration, l'autonomie, la capacité à apprendre, etc.

3.2.2.4 Les agents collaboratifs

Les agents collaboratifs sont caractérisés par l'autonomie, la coopération avec les autres agents dans la réalisation de leurs objectifs et aussi la capacité de négociation afin d'arriver à des compromis acceptables.

Dans le domaine de l'intelligence artificielle, plusieurs chercheurs prêtent aux agents collaboratifs des états mentaux comme les croyances, les désirs et les intentions. Dans ce cas, ces agents sont qualifiés d'agents collaboratifs de type BDI (beliefs, desires, intentions).

3.2.2.5 Les agents d'interface

Les agents d'interface sont caractérisés par la *métaphore d'assistant personnel collaborant avec l'utilisateur dans le même environnement* selon (Maes, 1997). Contrairement aux agents collaboratifs qui collaborent avec les autres agents, les agents d'interfaces

collaborent avec l'utilisateur. L'existence explicite d'un langage de communication entre agents n'est pas nécessaire dans cette catégorie d'agents. Les agents d'interfaces disposent aussi d'autonomie et de capacité à apprendre. Au départ, l'agent d'interface est doté d'un minimum de connaissances, par la suite, il acquiert lui-même, la connaissance dont il en a besoin pour assister l'utilisateur.

3.2.2.6 Les agents mobiles

Les agents mobiles sont des processus capables de se déplacer dans de grands réseaux d'information comme Internet pour accomplir des tâches qui leurs sont confiées comme, par exemple, le recueil des informations pour leurs propriétaires. Pour ne pas rendre publiques toutes les informations et données le concernant, l'agent mobile dispose des caractéristiques d'autonomie et de coopération lui permettant de communiquer ou de coopérer avec un autre agent chargé d'informer les autres agents de sa localisation, de ses attributs et ses méthodes. Notons que la mobilité n'est pas une caractéristique des agents. Quoique les agents mobiles coopèrent et sont autonomes, ils diffèrent des agents collaboratifs.

3.2.2.7 Les agents d'information

L'avènement de l'internet, l'explosion de l'information et le besoin de disposer des outils et des techniques de manipulation de ces informations ont permis de mettre en œuvre un nouveau type d'agents. Il s'agit des agents d'information ou d'Internet.

Ces agents permettent l'administration, la manipulation ou la collecte des informations à partir de plusieurs sources. Comme pour les autres types d'agents tels que les agents d'interface ou de collaboration, les agents d'information manipulent aussi des informations. Mais ils se diffèrent dans le fait qu'ils soient définis par ce qu'ils font, par contre les autres sont définis par ce qu'ils sont, c'est-à-dire par leurs attributs et leurs propriétés caractéristiques.

3.2.2.8 Les agents logiciels réactifs

Les agents réactifs sont des agents situés qui n'ont pas besoin d'avoir de modèles internes et symboliques de leur environnement, ni d'une représentation de cet environnement. Ils ont pour rôle de réagir aux stimuli provenant de l'environnement.

L'interaction de ces agents avec les autres agents se fait de façon basique. Ces agents peuvent être considérés comme une collection de modules autonomes ayant chacun une tâche spécifique. Contrairement aux autres types d'agent, les agents réactifs sont plus robustes et plus tolérants aux pannes, car la perte d'un tel agent n'entraîne pas de conséquences graves.

3.3 Les Systèmes Multi-Agents (SMA)

Les systèmes multi-agents sont apparus dans les années 80, on peut aussi dire que les systèmes multi-agents ont vu le jour avec l'avènement de l'intelligence artificielle distribuée (IAD) dont l'un de ses objectifs est la définition de systèmes pouvant modéliser des connaissances, raisonner et planifier des actions afin de résoudre des problèmes complexes.

Comme pour l'agent, ici aussi plusieurs auteurs essaient de donner une définition. Nous présentons quelques-unes :

3.3.1 Définitions

Un système multi-agents (SMA) est un système distribué composé d'un ensemble d'agents. Ferber quant à lui l'a défini comme étant un système composé des éléments suivants : un ensemble d'agents, un ensemble de tâches à réaliser et un ensemble d'objets associés à l'environnement (Ferber, 1995).

On définit généralement un système multi-agents comme étant un modèle informatique composé d'entités de base : les agents. Ces derniers possèdent une autonomie en terme de décisions et d'actions et sont organisés en société au sein d'un environnement dans lequel et avec lequel ils interagissent. Ils doivent, de ce point de vue, être capables de percevoir, de décider, d'agir et de communiquer

Cette définition fait apparaître des composants de base aux SMA : les agents vus précédemment, l'environnement, les interactions et l'organisation.

- L'environnement : c'est un espace représentant le monde dans lequel les agents évoluent. On fait généralement une distinction entre les agents qui sont les entités actives et les objets passifs qui se situent dans l'environnement.
- Les interactions : L'interaction est définie comme étant la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions réciproques (Ferber, 1995). Une interaction peut se caractériser par une simple trace qu'un agent laisse dans l'environnement (déplacement d'un objet, par exemple) et la perception de cette trace par un autre agent. Elle peut également se caractériser par un message que s'échangent deux agents ou encore par un acte de langage entre ces deux derniers.
- L'organisation : elle est définie comme une structure décrivant la manière dont ses membres sont en relation et interagissent afin d'atteindre un but commun. Dans les SMA, les aspects interaction et organisation ne sont pas dissociables. Une organisation ne peut pas exister si elle est découplée des interactions entre agents et réciproquement une interaction entre deux agents ne peut s'exprimer qu'à l'intérieur d'une organisation.

Comme définition de travail, nous dirons qu'un SMA est un ensemble d'agents travaillant ensemble (coopérativement ou parallèlement) pour remplir les objectifs du système.

3.3.2 Les caractéristiques d'un SMA

Un SMA est caractérisé comme suit :

- Chaque agent a des informations ou des capacités de résolution des problèmes limités, ainsi chaque agent a un point de vue partiel.
- Il n'y a aucun contrôle global du système multi-agents.
- Les données sont décentralisées.
- Le calcul est asynchrone.

3.3.3 Les architectures des SMA

Il existe plusieurs types d'architectures pour la conception des SMA. En effet le choix du type d'architecture est très important, car il a beaucoup d'impact sur le niveau de complexité de la modélisation, du développement et surtout de l'implémentation du système qui est la plus grande difficulté. Notons aussi que la modélisation des architectures qui tendent vers la décentralisation est plus difficile que celle des systèmes qui adoptent une architecture plus centralisée ou hiérarchisée. Nous présentons les architectures les plus utilisées. Il existe d'autres architectures hybrides empruntant des caractéristiques à plusieurs architectures.

3.3.3.1 Les SMA à contrôles centralisés (*Blackboard-based*)

Les architectures multi-agents à base de tableau noir sont les premières apparues. En effet dans ce type d'architecture les agents ne se communiquent directement aucune information, mais interagissent indirectement en partageant un espace de travail qu'on appelle tableau noir (Ferber, 1995). Ce tableau représente une base de données structurée en objets regroupant les données initiales du problème, les solutions et les alternatives des divers agents ainsi que la fonction finale (figure 9).

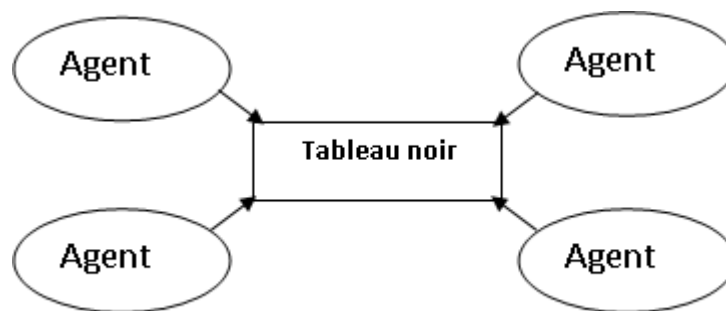


Figure 9. Partage de tableau noir

3.3.3.2 Les SMA Hiérarchisés (*horizontal*)

Cette architecture est utilisée par plusieurs systèmes pour la réalisation des SMA, les SMA hiérarchisés se basent sur une structure où les entités répondent à leurs supérieures hiérarchiques. Ce type de système est relativement simple à implémenter.

Cependant, cette architecture comporte des faiblesses. On note, par exemple, le peu de tolérances aux fautes, la limite imposée par les capacités de son supérieur hiérarchique, etc.

3.3.3.3 Les SMA à contrôle distribué

Les systèmes à tableau noir ont certes servi pour le développement des premiers SMA, mais la complexité des besoins actuels a fait introduire les SMA à contrôle distribué qui sont caractérisés par un ensemble d'agents complexes dotés d'une grande capacité de raisonnement, d'une distribution totale des connaissances, de décentralisation du contrôle et d'une communication directe (figure 10) par envoi de messages (Ferber, 1995).

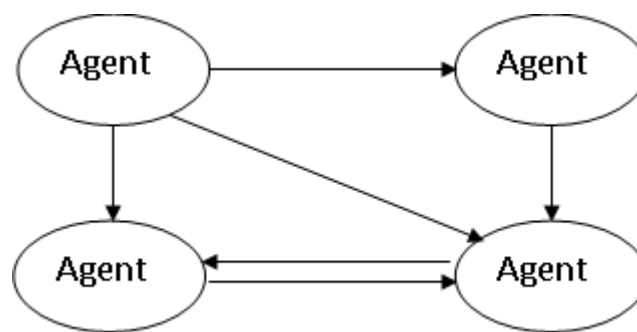


Figure 10. Communication par envoi de message

3.3.4 Les avantages des SMA

Les SMA présentent beaucoup d'intérêts justifiant leur utilisation. Nous présentons les principaux intérêts :

- L'intégration des connaissances diverses et complexes adoptant les formalismes hétérogènes.
- Les connaissances insuffisantes peuvent être complétées suivant la résolution.
- La possibilité de résolution des problèmes de taille et de complexité.
- L'adaptation à la réalité dans la mesure où de nombreux problèmes sont de nature distribuée.

3.3.5 Les soucis des SMA

Bien que les SMA offrent beaucoup d'avantages potentiels, ils doivent aussi relever beaucoup de défis. Nous présentons les problèmes inhérents à la conception et à l'implémentation des SMA. Ces problèmes concernent :

- La formalisation, la description, la décomposition, l'allocation des problèmes et la synthétisation des résultats.
- La communication et l'interaction entre agents? Comment, quoi et quand communiquer ?
- Les agissements cohérents des agents.
- Le raisonnement des agents individuels sur les actions, les plans et les connaissances des autres agents afin de se coordonner avec eux.
- Le compromis entre le traitement local au niveau d'un seul agent et le traitement distribué entre plusieurs agents. Plus généralement, la gestion de la répartition des ressources limitées.
- La gestion d'un comportement nuisible du système global.
- La conception des plates-formes technologiques et les méthodologies de développement pour les SMA.

3.3.6 Les langages de communication entre Agents

La communication est l'un des points fondamentaux dans l'étude des systèmes multi-agents. Les agents l'utilisent comme moyen pour se mettre en relation. Les agents communiquent entre eux à l'aide d'un langage de communication et ils utilisent des protocoles de communication. Les échanges entre agents se font par envoi de messages selon deux modes de transmission : la transmission directe (point à point) et la transmission par diffusion. Dans le premier mode l'émetteur de message connaît l'adresse du ou des destinataire (s). Dans le second mode l'émetteur envoie le message à tous les agents du système. Pour communiquer, les agents utilisent un langage commun et doivent comprendre son vocabulaire sous tous ses aspects concernant :

- La syntaxe : qui précise le mode de structuration des symboles.
- La pragmatique : pour interpréter les symboles.
- L'ontologie : pour utiliser les mêmes mots d'un vocabulaire commun.

Trois langages de communication et de représentation de l'information sont devenus des standards en SMA: KQML, FIPA ACL et KIF.

3.3.6.1 *Le langage KQML*

KQML (Knowledge Query and Manipulation Language) a été le premier langage de communication entre agents qui a eu une reconnaissance internationale (Finin, 1994). Ce langage est basé sur la théorie des actes de langage et permettait à des bases de données de s'interroger mutuellement.

KQML permet le partage des informations entre les agents du système afin de coopérer pour résoudre un problème. C'est un langage qui détermine un format pour les messages et un protocole pour l'envoi et la réception de ces derniers.

Conceptuellement, nous pouvons identifier trois couches dans un message de KQML : contenu, communication et message.

Les messages de KQML concernent non seulement des phrases dans un langage quelconque, mais ils sont enrichis d'une attitude sur le contenu (affirmation, désengagement, requête, question, etc.).

3.3.6.2 *FIPA ACL*

FIPA¹ (Foundation For Intelligent Physical Agent) est une organisation créée en 1996 qui tente d'instaurer des standards dans le domaine des SMA.

FIPA ACL (Agent Communication Language) est un langage syntaxiquement similaire à KQML. Il repose sur la définition de deux ensembles : un ensemble d'actes de

¹ <http://www.fipa.org/specs/fipa00061/index.html>

communication primitifs ou composés de ces actes primitifs. Un autre ensemble de messages prédéfinis et compréhensibles par tous les agents.

3.3.6.3 KIF (*Knowledge Interchange Format*)

Le troisième standard KIF permet de représenter le contenu des messages par des prédicats et la logique du premier ordre, car pour communiquer les agents ont besoin d'un langage entre eux, mais ils ont aussi besoin de comprendre le contenu des messages qu'ils reçoivent.

3.4 Méthodes d'analyse et de conception basées agents

Disposer d'une méthodologie pour l'analyse et la conception de logiciels basés sur le concept de systèmes multi-agents devient désormais un besoin inévitable. Ce besoin est apparu avec l'augmentation de la mise en œuvre de solutions multi-agents pour la conception de systèmes informatiques évolués et complexes. Le développement *ad hoc* doit permettre la mise en place des approches d'analyse et de conception reproductibles permettant ainsi d'introduire la notion de génie logiciel orienté systèmes multi-agents. Des travaux récents de recherche vont dans ce sens. Nous pouvons citer quelques approches d'analyse et de conception du contexte du génie logiciel comme : AEIO (Occello et Koning, 2001), « agent-groupe-rôle » (Guthnecht et Ferber, 1999) et MaSE (DeLoach et al., 2001).

3.4.1 L'approche Voyelles (décomposition AEIO)

Dans cette approche, on fait la distinction entre les deux étapes du génie logiciel qu'on appelle analyse et conception orientée systèmes multi-agents.

3.4.1.1 *Analyse orientée systèmes multi-agents*

Cette approche est connue sous le nom de VOYELLES (Demazeau, 1995). Son principe consiste à décomposer le problème en une solution multi-agents en identifiant les quatre composantes fondamentales pour la modélisation des SMA : Agents, Environnement, Interaction et Organisation (EIAO). Cette modélisation peut être orientée selon trois axes :

- *Orientée agent* : quand la résolution de problème est influencée par la parfaite identification du nombre, du rôle et des fonctionnalités de chaque agent et aussi du type même de l'agent (cognitif, réactif, mixte).
- *Orientée environnement* : quand on modélise une représentation du monde en mettant l'accent sur l'environnement (exemple des robots, des couloirs aériens de navigation).
- *Orientée interaction-organisation* : quand la résolution de problèmes doit mettre l'accent sur les mécanismes de coordination et de coopération entre agents ou quand on forme des groupes d'intérêts. Ou plus généralement, lorsque les besoins en communication au sein du système sont importants.

3.4.1.2 Conception orientée systèmes multi-agents :

Dans cette phase de conception, et pour produire un SMA opérationnel, les quatre composantes A, E, I, et O constituant les spécifications globales du système doivent être intégrées des modèles opérationnels choisis pour chaque composante comme :

- Les modèles d'agents ou architecture d'agents comme, par exemple, l'automate à états finis pour les agents réactifs ou les systèmes à base de connaissances pour les agents cognitifs.
- Les modèles d'environnement qui dépendent des domaines d'application.
- Les langages et les protocoles d'interaction entre agents (exemple : les actes de langage).
- les modèles d'organisation comme, par exemple, les modèles inspirés par les lois sociales et qui structurent les agents en groupes, hiérarchies, relations.

En résumé, on peut dire que pour un problème à résoudre, l'utilisateur choisit le modèle d'agents, le modèle d'environnement, le modèle d'interaction et le modèle d'organisation pour engendrer un système multi-agents qui résoudra le problème dans le domaine considéré.

3.4.2 Approche Agent Groupe Rôle "AGR":

L'approche Agent Groupe Rôle du projet AAlAADIN (Gurknecht et Ferber, 1999) s'articule autour des trois concepts : l'agent, le groupe et le rôle. L'agent est une entité autonome et communicante qui joue des rôles au sein de différents groupes. Un groupe est un regroupement d'agents. Un rôle est une représentation abstraite d'une fonction, d'un service ou de l'identification d'un agent au sein d'un groupe particulier. Un même rôle peut être tenu par un ou plusieurs agents. Ces principes sont illustrés par la figure 11 qui suit.

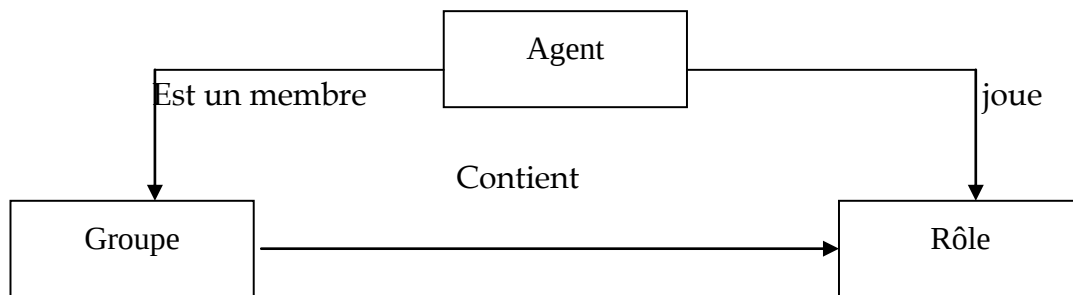


Figure 11. *Modèle Agent Groupe Rôle*

Cette approche est composée de trois phases : d'analyse, de conception et de réalisation. La phase d'analyse consiste à analyser fonctionnellement le système, à analyser les dépendances, à identifier des communautés pour préparer la conception des groupes et à choisir les mécanismes de coordination et d'interaction.

La phase de conception consiste à construire le modèle organisationnel qui est une représentation abstraite des structures de groupe et de rôles. Et également à définir des schémas d'interaction (protocole, messages) et aussi à définir les autres entités manipulées (actions, tâches, objectifs).

La phase de réalisation quant à elle consiste à choisir l'architecture d'agent en fonction des phases précédentes et du domaine d'application, et aussi à choisir les stratégies d'implémentation des protocoles d'interaction.

Le modèle Agent Groupe Rôle permet donc d'exprimer et d'analyser les systèmes multi-agents en se basant sur des concepts organisationnels.

3.4.3 La méthodologie MaSE

La méthodologie MaSE a été développée par Scott DeLoach et ses collaborateurs au sein du laboratoire d'intelligence artificielle de Air Force Institute (AFIT) (DeLoach et al., 2001). Cette méthodologie fait partie des méthodologies qui s'inspirent des méthodes de spécification et de développement des systèmes à base d'objets.

3.4.3.1 L'objectif de MaSE

Le but de la méthodologie MaSE est d'aider le concepteur à analyser, à concevoir et à implémenter un SMA à partir d'un cahier des charges initial. MaSE comporte sept étapes réparties en deux phases : la phase d'analyse et la phase de conception. Chacune des étapes a pour résultat un ou plusieurs diagrammes. Toutes les étapes et les phases de la méthodologie MaSE sont illustrées dans la figure 12.

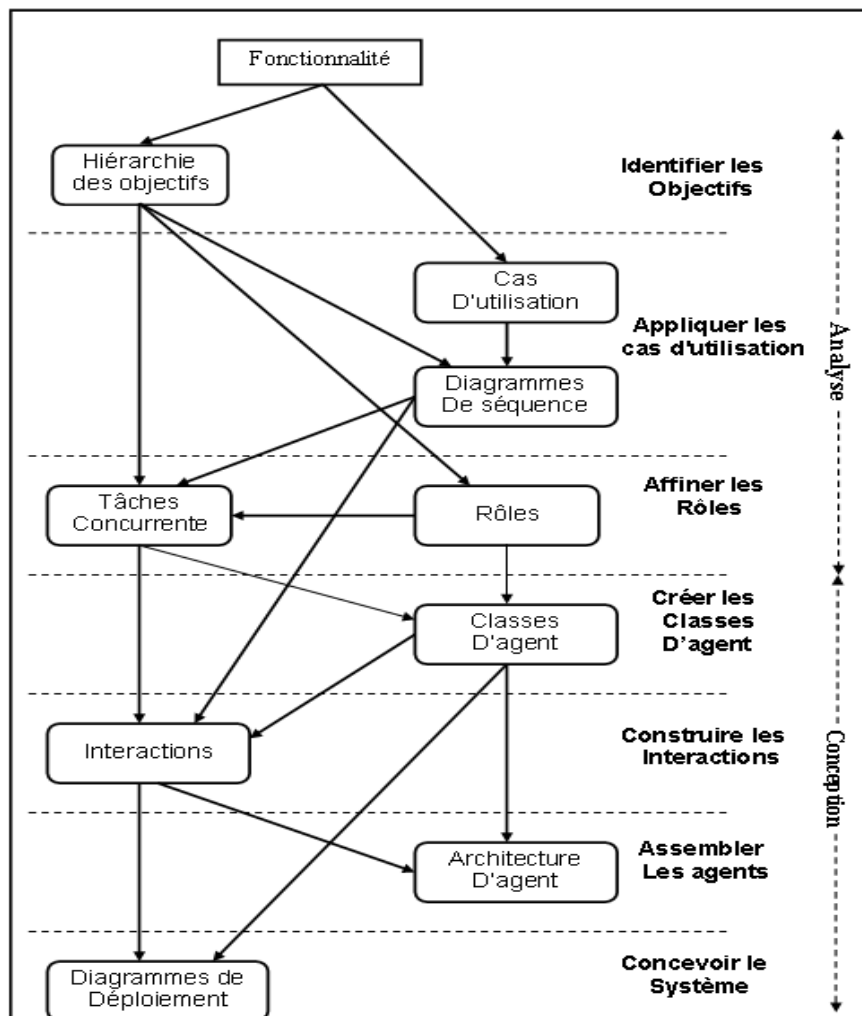


Figure 12. Les étapes et les phases de la méthodologie MaSE

3.4.3.2 La phase d'analyse

Elle comprend trois étapes qui sont : l'identification des objectifs, l'identification des cas d'utilisation et l'affinage des rôles.

Le but de cette phase est de définir les objectifs et aussi de produire un ensemble de rôles nécessaires pour leur accomplissement et un ensemble de tâches. Les tâches décrivent les exigences du cahier des charges initial, autrement dit ce que doit faire le système pour atteindre ses objectifs.

3.4.3.2.1 L'identification des objectifs

L'identification des objectifs du système est la première étape de la phase d'analyse. Son but est de transformer les spécifications initiales du cahier des charges du SMA en un ensemble structuré d'objectifs. Cette transformation s'effectue en deux sous-étapes à savoir le recensement des objectifs d'une part et leur structuration d'autre part.

Le recensement des objectifs consiste à extraire l'ensemble des scénarios à partir de l'ensemble des fonctionnalités contenues dans le cahier des charges initial, ensuite on décrit l'objectif de chaque scénario. Les objectifs doivent rester à un bon niveau d'abstraction et s'affranchir des détails.

La deuxième sous étape consiste quant à elle à structurer les objectifs de façon hiérarchique produisant ainsi un diagramme hiérarchisé d'objets. L'objectif global du système est placé au sommet du diagramme. Les objectifs dits « pairs » sont les objectifs placés au même niveau.

3.4.3.2.2 L'identification des cas d'utilisation

L'identification des cas d'utilisation est la deuxième étape de la phase d'analyse. Elle consiste à déterminer un ensemble de cas d'utilisation à partir du cahier des charges initial et à créer un ensemble de diagrammes de séquence. Les cas d'utilisation permettent l'explicitation des interactions dans le système et reflètent la vision du comportement du SMA offerte à l'utilisateur.

Les diagrammes de séquence sont extraits à partir des propos des utilisateurs, des fonctionnalités initiales ou de toute autre source existante. Ils facilitent l'identification

d'un ensemble initial de rôles et les différentes interactions présentes dans le système. Les cas d'utilisation sont décrits par les diagrammes de séquence comme un ensemble d'événements survenant entre les rôles constituant le système.

3.4.3.2.3 L'affinage des rôles

Dans cette troisième étape de la phase d'analyse s'effectuent la transformation des objectifs hiérarchisés du système et les diagrammes de séquences en un diagramme de rôles et de tâches associées. Les rôles représentent les objectifs du système durant la phase de conception et servent comme fondation de la définition des classes d'agents. Dans l'affinage des rôles, on a plusieurs considérations. Chaque objectif est associé à un rôle, mais des objectifs similaires ou voisins doivent être combinés en un rôle unique.

Dans le *MaSE Role Model*, on définit les rôles et aussi les informations sur les interactions entre les tâches. Les rôles sont représentés dans des rectangles et les tâches associées dans des ellipses reliées aux rectangles. Les protocoles de communication entre les tâches sont représentés par des lignes fléchées entre les tâches indiquant l'initiateur et le destinataire. Un fil de contrôle (thread) unique est associé à chaque tâche. Il définit un comportement particulier du rôle dans ses interactions. Les rôles ne doivent pas partager ou dupliquer des tâches.

3.4.3.3 La phase de conception

Cette phase comporte quatre étapes : la création des classes d'agents, la construction des interactions entre agents, l'assemblage des classes d'agents et la conception du système.

3.4.3.3.1 La création des classes d'agents.

Dans cette première partie de la phase de conception, on associe les rôles définis lors de la phase d'analyse à des types d'agents spécifiques. Ces rôles constituent la fondation de la création des classes d'agents. Ces dernières sont définies à partir des rôles qu'elles vont jouer et les interactions les concernant. Notons qu'une classe d'agent peut jouer plusieurs rôles. Un diagramme de classes d'agents comparables au diagramme de classes d'objets est donc défini par les classes d'agents et les interactions associées. Ces deux diagrammes sont différents dans les deux points suivants : le premier point est

que la définition de la classe d'agent ne se fait pas par les attributs et les méthodes la constituant, mais plutôt par les rôles qu'elle joue. Le deuxième point est que toutes les relations se trouvant dans ce diagramme des classes d'agents représentent des interactions entre les classes d'agents.

Notons que le diagramme des classes d'agents montre l'organisation du système en classes d'agents et les interactions entre elles. Il représente la première entité de MaSE décrivant le SMA tout entier et la façon dont il sera éventuellement implémenté.

3.4.3.3.2 La construction des interactions

Cette deuxième étape permet de définir les interactions entre les différentes classes d'agents qui sont créés à partir des rôles définis lors de la phase d'analyse. Les détails des interactions entre les agents sont obtenus à partir des informations contenues dans les diagrammes des tâches concurrentes.

Une interaction MaSE décrit un protocole de coordination entre deux agents. Pour la représenter, on se sert de deux diagrammes d'interaction, l'un pour l'émetteur et l'autre pour le destinataire. Chaque diagramme est un automate à états finis. La communication est initiée par un message de l'émetteur. À la réception du message, le destinataire le compare avec ses communications actives. S'il y a une correspondance, la transition nécessaire est opérée et le destinataire migre vers un nouvel état et effectue les actions nécessaires contenues dans la transition ou le nouvel état dans lequel il se trouve. S'il n'y a pas de correspondance avec le message reçu, c'est une nouvelle communication qui sera comparée par le destinataire avec toutes les interactions pouvant apparaître avec l'agent émetteur. Si le destinataire trouve une correspondance, il engage une nouvelle communication.

3.4.3.3.3 L'assemblage des classes d'agents.

La troisième étape quant à elle consiste à définir l'architecture interne et les processus de raisonnement des agents. Elle comporte deux sous étapes : la définition de l'architecture et la définition des composants de cette architecture. Les concepteurs peuvent définir leur propre architecture, peuvent utiliser des architectures prédéfinies

comme, par exemple, l'architecture BDI ou bien peuvent créer ou utiliser des composants prédéfinis.

Un composant est un ensemble d'attributs et de méthodes et s'il est trop complexe, il peut avoir une sous-architecture.

3.4.3.3.4 La conception du système

La dernière étape de MaSE est l'étape d'instanciation des classes d'agents définies préalablement. Un diagramme de déploiement est utilisé pour montrer le nombre, le type et la localisation des agents dans le système.

3.5 Apports de la technologie agents pour l'apprentissage

Les SMA présentent comme caractéristiques : de permettre de partager ou de distribuer de la connaissance, et aussi, de faire correspondre un ensemble d'agents et de coordonner leurs actions dans un environnement pour l'accomplissement d'un but commun. Le domaine des EIAH n'y échappe pas. Dans un tel environnement, la complexité du pilotage de l'apprentissage humain, les besoins de résolution de problèmes et de coopération des différents acteurs d'un EIAH sont tels qu'il est nécessaire de dépasser les modèles classiques autonomes au profit de modèles de coopérations d'agents humains et artificiels pour réaliser ce pilotage.

Parmi les nombreux aspects prometteurs des solutions agents, on peut citer les domaines de :

3.5.1 Modélisation de l'apprenant dans un groupe

La modélisation des groupes et des interactions entre individus s'appuie sur les observations du monde réel. Elle pourra servir au développement d'outils de simulation de mouvement de groupe ou bien à la mise en œuvre de systèmes informatiques multi-agents.

L'activité d'un groupe d'individus se structure par le biais des interactions directes entre les individus eux-mêmes et également par le biais des interactions directes via

l'environnement qu'ils modifient par leurs actions. Un ensemble d'individus que l'on peut représenter de plusieurs façons :

Chacun des N individus possède un état évoluant au cours du temps $X_i(t)$, $i = 1$ à N , où chaque X_i peut être une grandeur scalaire ou vectorielle : l'état peut décrire à la fois certaines caractéristiques « externes » comme la position de l'individu, ainsi que des caractéristiques « internes » correspondant à son état interne (fatigue, volonté, etc.). Cette représentation du collectif permet de suivre les individus un par un et d'étudier leur évolution temporelle aussi bien que spatiale. Elle permet aussi de traiter de la division du travail, l'organisation sociale, etc. Il s'agit là sans doute de la représentation la plus exhaustive et la plus précise. Elle peut servir à une modélisation par équation différentielle (ordinaires déterministes ou stochastiques). La simulation se révèle souvent utile pour ne pas dire nécessaire.

3.5.2 L'assistance virtuelle et la délégation

Les tâches relatives à la gestion du flux du travail et les tâches répétitives des utilisateurs seront déléguées à des assistants virtuels dotés d'une certaine intelligence. Un assistant est un logiciel qui accomplit la mission qui lui a été confiée de manière autonome au moment opportun. Un tel assistant doit prendre toutes sortes de petites et grandes décisions sans réclamer d'incessantes approbations. Les caractéristiques d'un assistant intelligent sont :

- 1) l'assistant doit être autonome, une fois déclenché par l'homme ou par un événement, il agit sans la moindre intervention extérieure.
- 2) il est auto-adaptable, il est capable donc d'apprendre en contact de l'utilisateur et ses capacités se développent ainsi d'elles-mêmes au fur et à mesure du temps.
- 3) il doit fonctionner en arrière-plan sans gêner l'utilisateur dans son travail.
- 4) il possède enfin une intelligence conditionnelle. Ainsi, dans des contextes complexes et changeants, il est capable de prendre des décisions à la place de l'utilisateur.

Un assistant possède un état évoluant dans le temps, un récepteur et un émetteur. Il peut ainsi produire et réagir à des événements provenant de l'application ou des

utilisateurs. A la réception d'un événement, l'assistant modifie son état en conséquence et émet d'autres événements (Coutaz, 1990).

L'architecture interne de l'assistant est basée sur une approche qui combine à la fois des règles qui définissent sa tâche (Fournier, 1994) et une fonction intelligente d'observation des actions et des habitudes de travail des utilisateurs.

L'assistant scrute l'utilisateur et lorsqu'il repère un comportement, une action qui se répète, il la compare avec sa base de connaissances. S'il existe une forte probabilité d'un lien entre le comportement repéré et les connaissances de l'assistant, ce dernier propose alors d'automatiser la séquence d'actions. L'utilisateur peut ou ne pas accepter les conseils de l'assistant et la délégation de responsabilités. L'utilisateur peut à tout moment intervenir sur l'assistant non pour modifier sa programmation, mais pour orienter ses déductions. Il revient de trouver une technique simple interaction entre l'assistant et l'utilisateur.

On peut donc conclure qu'un assistant est un compagnon de l'utilisateur qui peut jouer plusieurs rôles. C'est un observateur qui observe les actions de l'utilisateur. C'est un scruteur de l'utilisateur. C'est un facilitateur qui facilite l'accès et l'usage de l'environnement grâce à la personnalisation et à la recommandation. Pour réaliser ses tâches, l'assistant gère une structure permettant de garder toutes les informations concernant l'utilisateur plus particulièrement l'apprenant. Cette structure est appelée le modèle de l'utilisateur.

3.6 Conclusion

Les notions d'agents et de SMA ont été abordées dans ce chapitre. La multitude de significations données à ces deux termes nous a permis de faire une synthèse et de garder des définitions pour notre travail. Nous avons vu également les langages de communication entre les différents agents dans un environnement SMA.

La technologie agents est considérée comme une nouvelle méthode pour développer des logiciels lorsque des critères comme la distance, la coopération entre les entités, la réutilisation ou l'extensibilité sont à prendre en compte. Un système complexe pourrait

être considéré comme une communauté d'agents autonomes de petite taille et interagissant entre eux. Chaque agent a ses propres propriétés et fonctionnalités. L'idée est donc de se servir des agents pour élaborer de nouveaux systèmes plus autonomes, plus distribués, plus intelligents, etc.

Nous avons ensuite présenté quelques méthodes de conception des SMA. Parmi ces méthodes MaSE, que nous allons l'utiliser dans la conception de notre environnement.

La méthodologie MaSE est une succession d'étapes où l'étape actuelle se sert des résultats de l'étape précédente. Mais dans la pratique, MaSE est itérative. On a souvent besoin des allers et des retours entre les différentes étapes. Un des avantages de MaSE est que ses entités apparaissent dans toutes les étapes. Par exemple, On trouve un objectif identifié dans la première étape dans un rôle, dans une tâche et dans une classe d'agent.

Nous allons voir dans le chapitre 5, comment la notion des systèmes multi-agents et les aspects méthodologiques d'analyse et de conception orientées multi-agents contribuent-ils à l'élaboration de notre environnement E-PACAD ?

Chapitre 4 : La modélisation bayésienne de l'apprenant dans l'environnement d'apprentissage collaboratif E-PACAD

4.1 Introduction

Dans ce chapitre, nous précisons en quoi consiste notre environnement E-PACAD, comment l'exploiter et les choix de modélisation effectués. Nous présentons d'abord l'environnement E-PACAD en insistant sur le rôle joué par les ontologies pour structurer et explorer les connaissances et aussi sur le rôle joué par les réseaux bayésiens dans la modélisation des connaissances. Nous présentons ensuite notre démarche de modélisation cognitive de l'apprenant qui consiste à combiner un modèle d'expertise partielle avec un réseau bayésien. Enfin nous concluons.

4.2 L'approche PACAD

Avec l'approche PACAD, nous nous sommes intéressés à la modélisation des informations concernant l'apprenant (son profil, son parcours, ses spécificités, ses connaissances, etc.) dans le contexte d'un apprentissage au sein d'une organisation partageant des ressources pédagogiques liées à ces connaissances.

Nous avons fait le choix de gérer ce modèle au moyen d'un environnement d'apprentissage centré et adapté apprenant. L'environnement entier est modélisé comme une interaction entre plusieurs agents humains (l'apprenant, l'enseignant, et l'administrateur, etc.) et artificiels qui partagent des espaces (de travail, de communication, de collaboration) ainsi qu'un ensemble de ressources pédagogiques. Nous nous sommes focalisé sur son utilisation pour en déterminer la modélisation (de contenu, d'apprenant, etc.).

La réalisation d'un tel environnement que nous avons baptisé E-PACAD (Plateforme d'Apprentissage Collaboratif à Distance) passe par les étapes suivantes :

- la conception d'un modèle de l'environnement permettant à un usager de disposer d'outils adaptés et compatibles aux particularités du monde dans lequel il évolue.
- l'étude de contrainte pour permettre un réel apprentissage collaboratif. Cela nécessite en particulier la modélisation du contenu à étudier, la modélisation de l'utilisateur et la formalisation des règles d'interaction.

Dans le modèle que nous avons défini, les ressources pédagogiques sont indexées aux moyens de connaissances définies au sein d'ontologies:

- Ontologie de domaine : définit les concepts spécifiques au domaine traité.
- Ontologie d'application : définit les concepts propres à une application particulière.

L'utilisation de ces ontologies permet aux acteurs de l'environnement d'acquérir les connaissances, et cela, en accédant aux différentes ressources (cours, exercice, forum, etc.) et en réalisant les différentes tâches (consulter cours, exemples, définitions ; résoudre des problèmes ou exercices ; poser des questions, etc.).

Le dialogue homme-machine-homme exige la prise en compte de divers facteurs d'ordre cognitif, ergonomique ou conceptuel. L'apprentissage collaboratif engendre des difficultés supplémentaires relatives à la psychologie du groupe. La réaction d'un utilisateur est double, l'une face à la machine et l'autre face aux autres utilisateurs du groupe. L'architecture proposée vise à augmenter la productivité du travail collectif, à trouver les solutions pour rendre son utilisation plus efficace. Elle comporte quatre composantes :

4.2.1 Le modèle de contenu pédagogique

Le contenu pédagogique de notre environnement de formation est représenté par trois niveaux hiérarchiques : objectif d'apprentissage, notions à appréhender (concepts) et les ressources (documents) pédagogiques indexées par ces notions.

4.2.1.1 Objectif pédagogique

Il correspond à une unité d'enseignement du domaine à étudier que l'apprenant choisit pour débiter son étude. Chaque objectif est relié à un ensemble de concepts du domaine (notions à appréhender) à étudier, de même, chaque concept est relié à un ensemble de ressources pédagogiques. Les concepts sont reliés entre eux par des liens sémantiques.

4.2.1.2 Notion à appréhender

C'est une notion particulière qui doit être assimilée par l'apprenant durant l'apprentissage. Les notions permettent d'indexer les ressources pédagogiques les traitant. Cela offre un moyen de réutiliser ces ressources.

Une notion à appréhender peut référencer plusieurs ressources offrant ainsi la possibilité de l'évoquer de différentes façons. Plusieurs notions peuvent référencer la même ressource donnant ainsi la possibilité de l'appréhender de différentes manières.

Notons que ce n'est pas parce qu'une notion est traitée par une ressource qu'elle indexera obligatoirement cette ressource. Il faut que la ressource ait été jugée suffisamment intéressante pour l'apprentissage de cette notion. Ces décisions découlent de l'objectif pédagogique mis en œuvre par le responsable de la formation.

4.2.1.3 Ressources pédagogiques

Ce sont généralement les documents pédagogiques (cours, exercices, livres électroniques, diaporamas, liens vers les sites web, rapports, etc.). Parmi ces ressources, certaines sont internes c.à.d. directement enregistrées dans l'environnement (cours, exercices, rapports) alors que d'autres sont des ressources externes et ne sont représentées que par des liens ou des références (ex. livres électroniques).

Chaque ressource pédagogique est indexée par les concepts qu'elle traite. Cette indexation a pour objectifs :

- La réutilisabilité des ressources pédagogiques.
- La facilité de repérage des concepts et des parties qui les traitent dans les documents pédagogiques.

- La présentation aux utilisateurs d'un concept de différentes manières, et ce, en présentant différents documents traitant ce concept.
- La rentabilité économique dans la production des ressources pédagogiques.

Le modèle de contenu (figure 13) que nous proposons pour notre environnement permet d'indexer les ressources pédagogiques par une ontologie du domaine et une ontologie d'application.

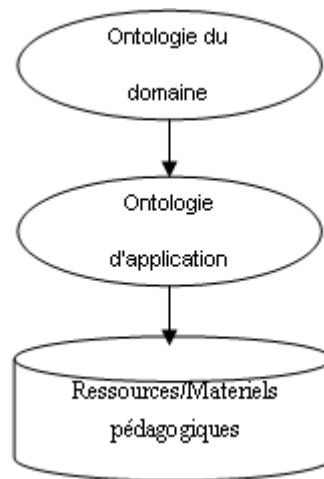


Figure 13. *Approche conceptuelle du modèle de contenu*

4.2.1.4 L'ontologie de domaine

L'ontologie de domaine dont un extrait est représenté dans la figure 14 est la composante de l'environnement d'apprentissage qui permet de décrire les concepts spécifiques au domaine de la formation. Ces concepts sont structurés dans des sous ontologies telles que : Personnes (apprenant, enseignant, tuteur, administrateur, etc.), ressources (livres, sites web, diaporamas, etc.), Type média (texte, image, audio, vidéo, etc.). Unité d'enseignement (Algorithmique et programmation, Analyse de données, Bases de données, etc.), Groupes (Groupe1, Groupe2, etc.) et Relations (a-pour-auteur, est-inscrit, est-responsable-de-unité, etc.), etc.

Les éléments de cette ontologie peuvent être de formes : concept, attribut ou relation. Par exemple, la relation est-responsable-de-unité qui associe les deux concepts Personne et Unité d'enseignement est utilisée pour représenter le responsable d'une unité d'enseignement. De même, l'élément nom est un attribut du concept Personne.

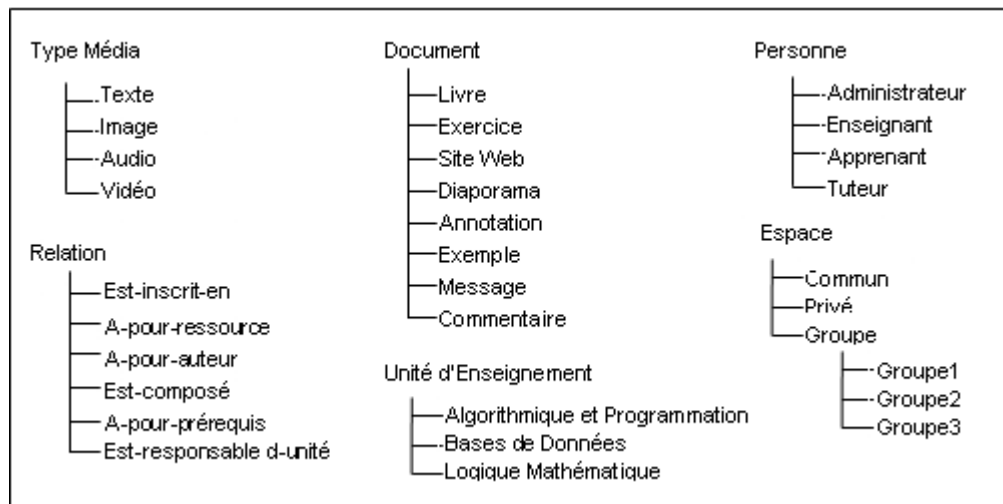


Figure 14. Extrait de l'ontologie de domaine

4.2.1.5 Les ontologies d'application

L'ontologie d'application spécifie les concepts d'une application particulière. Ces concepts représentent des notions à appréhender d'une unité de formation. Ces notions sont organisées dans une hiérarchie qui représente aussi plusieurs types de relations (spécialisation et autres). Parmi les ontologies d'applications construites dans le cadre de notre environnement E-PACAD: L'ontologie pour l'unité d'enseignement algorithmique et programmation suivie par les étudiants inscrits en première année licence académique pour la filière Maths Informatique (MI) à l'UFAS (figure 15).

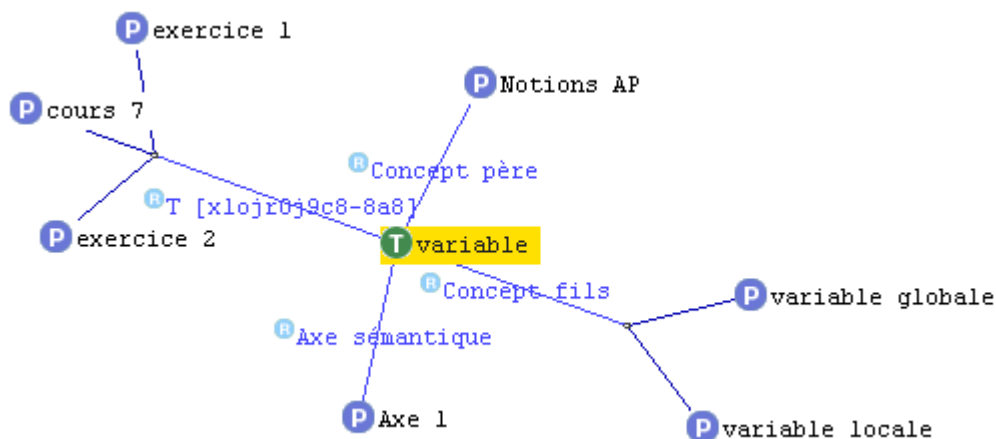


Figure 15. Extrait de l'ontologie d'application (AP)

Pour construire ces ontologies d'application, nous avons choisi la méthode OntoSpec (Ikeda et al., 2002). Son principe est d'organiser les concepts en une taxonomie souvent selon une relation de spécialisation (subsumption) : le concept fils hérite de toutes les propriétés du concept père. Ces concepts peuvent aussi être organisés suivant d'autres types de relations qui ne nécessitent pas un héritage de propriétés. Pour améliorer l'organisation des concepts, la méthode définit des axes sémantiques qui servent à grouper les concepts frères selon différents points de vue (Benayache, 2005).

Nous avons choisi Les Topics Maps (XTM, 2001) comme formalisme de représentation de nos ontologies. Un extrait montrant la déclaration du topic variable locale se trouve en annexe 1.

Notons que l'ontologie d'application est intégrée dans l'ontologie de domaine. Cela permet de lier les concepts de la première ontologie aux concepts de la deuxième permettant ainsi d'exprimer, par exemple, le fait qu'un document particulier est utilisé pour introduire une notion de la formation sachant que ces deux concepts appartiennent à deux ontologies différentes.

4.2.2 Le modèle de l'apprenant

L'utilisateur est une personne désignée pour accomplir une tâche. Dans ce modèle sont décrits l'identité, les droits, les devoirs, les compétences et les disponibilités de l'utilisateur. Chaque usager a accès à l'environnement commun via une interface graphique personnalisable selon ses propres caractéristiques. De même, un utilisateur quelconque peut à tout moment s'isoler pour accomplir momentanément une tâche personnelle. Notre environnement est centré et adapté apprenant. L'apprenant est donc l'utilisateur principal de l'environnement.

Le modèle de l'apprenant est un composant essentiel de tout système d'enseignement ou d'apprentissage adaptatif. Il permet d'identifier les caractéristiques de l'apprenant par l'observation de son comportement au cours de la session d'apprentissage. Certaines des données du modèle de l'apprenant peuvent être recueillies directement par l'analyse du choix de l'apprenant ou indirectement par l'observation et l'analyse de

son comportement. Par exemple, en observant le contenu des documents consultés on peut conclure la langue préférée ou bien le type de média préféré (texte, son, vidéo). D'autres caractéristiques de l'apprenant peuvent être obtenues en lui demandant de réaliser une activité particulière. Par exemple, l'évaluation sommative permet d'indiquer le niveau de connaissance sur un domaine particulier. Mais il reste des données non observables comme, par exemple, le style d'apprentissage qui est un paramètre de base pour les systèmes hypermédias adaptatifs. Ce paramètre non mesurable peut être estimé par un ensemble d'indicateurs observables.

Nous constatons donc que le modèle de l'apprenant est un ensemble de couples paramètre-valeur. Certains paramètres sont mesurables alors que d'autres ne le sont pas. Et dans ce dernier cas, un ensemble d'indicateurs observables est utilisé pour estimer ces paramètres.

Dans ce contexte, la modélisation de l'apprenant est un processus qui consiste à définir des indicateurs observables et des paramètres non observables pour caractériser l'apprenant ainsi que les relations d'influence entre ces indicateurs et ces paramètres.

Plusieurs travaux ont utilisé les réseaux bayésiens pour indiquer le niveau de connaissance de l'apprenant et prédire les prochaines actions (Gamboa et Fred, 2001) (Carberry, 2001).

Inspiré de ces travaux, pour notre modèle de connaissances, nous proposons de combiner un modèle probabiliste utilisant un réseau bayésien avec un modèle d'expertise partielle sur le modèle de contenu. Cela signifie que la structure du modèle de contenu est représentée dans le modèle de connaissances. A chaque élément du modèle de l'apprenant qui correspond à un élément du modèle de contenu, on associe un ensemble de valeurs qui mesurent le niveau de connaissance de l'apprenant pour cet élément.

Le modèle de l'apprenant que nous proposons se compose de quatre modèles suivants:

4.2.2.1 Le profil de l'apprenant

Il contient les informations personnelles sur l'apprenant. L'apprenant doit remplir le formulaire de saisie concernant ces informations au niveau de l'inscription, parmi ces

informations nous identifions: Nom d'utilisateur, mot de passe, nom, prénom, date de naissance, sexe, adresse, e-mail, degré de motivation de l'apprenant (fort, moyen, faible), etc.

4.2.2.2 Modèle affectif

Il est appelé aussi modèle de préférences. Il mémorise les préférences de l'apprenant au sujet des ressources pédagogiques ainsi que des médias utilisés : type de média préféré, type d'exercice préféré (exercice classique, exercice semi-assisté, QCM), etc. Un exemple de questionnaire pour identifier quelques caractéristiques et préférences de l'apprenant se trouve en annexe 2.

4.2.2.3 Modèle de l'interaction

Il enregistre les informations issues des interactions de l'apprenant avec l'environnement (ex : nombre de visites par concept, ou par ressource pédagogique, nombre de messages postés ou reçus par l'apprenant via les moyens de communication, performance obtenue à une évaluation sommative, etc.)

4.2.2.4 Modèle cognitif

Appelé aussi modèle de connaissances qui mémorise les croyances du système sur la connaissance de l'apprenant sur les concepts définis dans le modèle de contenu. De ce fait, pour chaque concept du modèle du contenu, on le qualifie par : maîtrisé ou non maîtrisé par l'apprenant. Notre modèle est basé sur trois couches :

- La couche *visited* : qui indique si le concept a été consulté par l'apprenant ou non.
- La couche *evaluated* : qui contient pour chaque concept du domaine le niveau de connaissance représenté par une performance mesurée P_m , que l'apprenant a obtenue à l'évaluation sommative correspondante.
- La couche *inferred* : qui représente un modèle de prédiction qui permet d'estimer la performance théorique P_t envisageable pour un apprenant dans un concept donné en fonction de ses connaissances actuelles dans le domaine. Le concept sera recommandé uniquement quand sa P_t est supérieure à un seuil s fixé.

4.2.3 Le modèle de communication

Il concerne les échanges formels et informels entre les utilisateurs dans le but de se préparer à la réalisation d'un travail commun. Le gestionnaire de communication cherche à avoir une vue plus globale sur l'environnement pour gérer au mieux les conflits entre utilisateurs. Il est chargé d'informer chaque intervenant sur le travail des autres. Les échanges conversationnels se feront essentiellement à travers les outils de communication.

4.2.4 Le modèle de collaboration

Les possibilités d'interaction au sein d'un groupe permettent aux individus qui le composent de corrélérer leurs comportements de manière adaptative pour faire face à des situations diverses. Cette corrélation fonctionnelle implique l'existence d'une cause commune : le problème à résoudre (Boukelif, 2008). La résolution collaborative est un processus cyclique qui s'arrête lorsque le but est atteint.

Notre modèle permet d'organiser des espaces de collaboration dédiés aux apprenants travaillant en groupe sur un même problème. Formant ainsi un lieu de travail et d'échange pour le groupe, et permettant, d'une part, à chaque membre du groupe d'accéder aux ressources (documents et autres) destinées au groupe et, d'autre part, de mémoriser son travail (documents, idées, connaissances, solutions, etc.) relatif au problème traité.

Cette façon de voir le problème permet une grande flexibilité, car aucune solution n'est imposée; les modifications spatio-temporelles de l'environnement entraînent au retour des modifications du comportement du groupe qui peut ainsi s'adapter au nouvel environnement. Elle permet aussi une plus grande robustesse, car l'erreur individuelle n'est pas exclue et devient au contraire porteuse de solutions nouvelles pour le groupe en lui permettant de se sortir de configurations pièges. Le mécanisme de rétroaction permet la coordination des comportements individuels.

4.3 Application d'un réseau bayésien à un modèle d'expertise partielle

Un réseau bayésien est un graphe orienté acyclique de n variables aléatoires $X_1, \dots, X_n$ tel qu'il existe une bijection entre l'ensemble des variables aléatoires et l'ensemble des sommets du graphe et que :

$P(X_1, \dots, X_n) = \prod (P(X_i \mid \text{pa}(X_i)))$ où $\text{pa}(X_i)$ est l'ensemble des parents de X_i dans le graphe.

Les réseaux bayésiens reposent sur la formule de Bayes qui relie des probabilités conditionnelles avec des probabilités jointes :

$$P(A|B) = \frac{P(A, B)}{P(B)}$$

Avec : A et B sont deux variables aléatoires ayant respectivement les probabilités $P(A)$ et $P(B)$ qu'elles soient vraies.

$P(A \mid B)$ est la probabilité conditionnelle que A soit vraie sachant que B est vraie.

$P(A, B)$ est la probabilité jointe entre A et B .

La construction d'un réseau bayésien nécessite de préciser, d'une part, sa structure représentée par un graphe orienté acyclique et, d'autre part, les tables de probabilités associées à chaque nœud du graphe.

Pour la définition de la structure du réseau, il faut recenser les nœuds possibles et distinguer entre les variables observables et les variables non observables. L'existence d'un arc entre deux variables peut exprimer l'influence (la causalité) d'une variable sur une autre.

4.3.1 Exemple d'un réseau bayésien

Revenons à l'exemple de la pelouse de Sherlock Holmes cité dans le chapitre 2 qui comporte quatre nœuds et trois relations (figure 16).

Les quatre nœuds qui peuvent prendre la valeur "Vrai" ou "Faux" sont :

- P (pluie) représente le fait qu'il ait plu la nuit précédente.
- A (arroseur) représente l'oubli de l'arroseur automatique.
- J (jardin) représente l'état du jardin d'Holmes.
- V (voisin) représente l'état du jardin du voisin Watson.

Et les trois relations causales (arcs) qui lient les nœuds sont :

- $P \rightarrow J$: s'il a plu, la pelouse de Holmes est mouillée.
- $P \rightarrow V$: s'il a plu, la pelouse de Watson est mouillée.
- $A \rightarrow J$: si le système d'arrosage de Holmes n'a pas été coupé alors la pelouse de Holmes est mouillée.

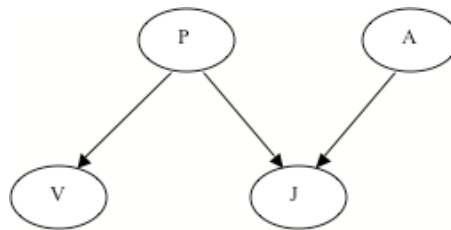


Figure 16. Réseau de Bayes de l'exemple

Pour évaluer la probabilité qu'il ait oublié de couper le système d'arrosage, Holmes a du prendre deux décisions contradictoires :

- Il constate que J est vrai. Cela augmente la probabilité du fait que A soit vrai, donc il retourne à la cave pour couper le système d'arrosage.
- Il constate que V est vrai. Cela diminue le risque que A soit vrai. Il rebrousse chemin et monte dans sa voiture.

Pour modéliser le comportement du réseau, on associe à chaque fait ayant plusieurs causes une table des probabilités donnant toutes les probabilités conditionnelles :

P=1	0.2
P=0	0.8

Table 2. La distribution de probabilités du nœud P

A=1	0.3
A=0	0.7

Table 3. La distribution de probabilités du nœud A

	P=1	P=0
V=1	1	0.2
V=0	0	0.8

Table 4. Table des probabilités conditionnelles du nœud V ($P(V/P)$)

	P=1		P=0	
	A=1	A=0	A=1	A=0
J=1	0.95	0.90	0.90	0.10
J=0	0.05	0.10	0.10	0.90

Table 5. Table des probabilités conditionnelles du nœud J ($P(J/P,A)$)

On obtient alors les deux tables des lois des probabilités jointes $P(V,P)$ et $P(J,P,A)$.

	P=1	P=0
V=1	0.20	0.16
V=0	0.00	0.64

Table 6. Table des probabilités jointes $P(V,P)$

	P=1		P=0	
	A=1	A=0	A=1	A=0
J=1	0.06	0.13	0.22	0.06
J=0	0.00	0.01	0.02	0.50

Table 7. Table des probabilité jointe $P(J,P, A)$

A partir de ces deux tables, on peut calculer les probabilités marginales $P(V)$ et $P(J)$

V=1	0.36
V=0	0.64

Table 8. La table des probabilités du nœud V

J=1	0.47
J=0	0.53

Table 9. La table des probabilités du nœud J

Le calcul d'inférence dans le réseau consiste à déterminer la loi d'une ou plusieurs variables aléatoires sachant que la variable aléatoire ou le vecteur de variables aléatoires E prend la valeur e , c'est à dire $P(X | E = e)$.

Un réseau bayésien nous permet, par exemple, de répondre à la question suivante: comment déterminer quelle est la cause (l'arrosage ou la pluie) la plus possible pour que le jardin soit mouillé. Par conséquent, nous allons calculer les deux probabilités $P(A = 1/J=1)$ et $P(P = 1/J=1)$.

$$P(P = 1 | J = 1) = \frac{\sum_{A,P} P(J = 1, A, P = 1, V)}{\sum_{A,P,V} P(J = 1, A, P, V)} = \frac{0.19}{0.47} = 0.40$$

$$P(A = 1 | J = 1) = \frac{\sum_{A,P} P(J = 1, A = 1, P, V)}{\sum_{A,P,V} P(J = 1, A, P, V)} = \frac{0.28}{0.47} = 0.60$$

Dans ce cas, la cause la plus probable de l'humidité du jardin de Holmes est le system d'arrosage.

Maintenant, on dispose de la connaissance supplémentaire que J est vrai. Donc J devient la variable que l'on observe le fait que le jardin est mouillé, alors J a la valeur 1. Nous multiplions les probabilités conjointes de $P(J,P,A)$ par un coefficient pour que la première ligne ait pour somme 1 et que la deuxième ligne soit de somme nulle. On réactualise aussi les deux tables de $P(A)$ et $P(P)$ par sommation, ce qui donne les tables suivantes.

P(P)		P(A)		P(V,P)		P(J,P,A)					
P=1	0.4	A=1	0.6	V=1	P=1	P=0	P=1		P=0		
P=0	0.6	A=0	0.4	V=0	0.4	0.12	A=1	A=0	A=1	A=0	
					0	0.48	J=1	0.13	0.27	0.47	0.13
							J=0	0.00	0.00	0.00	0.00

Table 10. Modification des valeurs des probabilités en prenant en compte le fait que J est vrai

En lisant le tableau, on constate que $P(A=1)$ est passée de 0.3 à 0.6. Ainsi, la prise en compte de l'information sur J a augmenté la probabilité que A soit vrai. Ce qui a poussé Holmes à faire un détour par sa cave.

Maintenant, si on prend en compte la certitude que V est vrai. Et de la même façon, on met à jour les tables des probabilités, on obtient les tables (Table 11).

P(P)		P(A)		P(P,V)		P(P,A,J)					
P=1	0.77	A=1	0.43	V=1	P=1	P=0	P=1		P=0		
P=0	0.23	A=0	0.57	V=0	0.77	0.23	A=1	A=0	A=1	A=0	
					0.00	0.00	J=1	0.25	0.52	0.18	0.05
							J=0	0.00	0.00	0.00	0.00

Table 11. Modification des valeurs des probabilités en prenant en compte le fait que V est vrai

On remarque que la probabilité de A a redescendu à 0.43. Ce qui a poussé Holmes à repartir sans passer par la cave.

On peut donc calculer :

$$P(P=1/J=1) = 0.77 \quad \text{et} \quad P(A=1/J=1) = 0.43$$

Dans ce dernier cas, La cause la plus probable de l'humidité du jardin du Holmes est devenue la pluie.

4.3.2 L'application du réseau bayésien dans notre modèle de l'apprenant

Notre modèle de l'apprenant est de type expertise partielle, cela signifie que le modèle de l'apprenant est un sous-ensemble du modèle de domaine. Le domaine est décomposé en un ensemble d'éléments et le modèle de l'apprenant est représenté par un ensemble de valeurs mesurables associées à ces éléments. Ces valeurs varient entre 0 (non maîtrisé) et 1 (maîtrisé). Les relations qui existent entre les éléments du modèle de l'apprenant peuvent être de différents types. Nous nous intéressons plus particulièrement à la relation de type spécialisation « est-un ». Deux éléments (concepts) A et B sont liés par une relation de type spécialisation signifie que le concept B est un cas spécial du concept A et il hérite de toutes ses propriétés.

La construction de notre modèle passe par les étapes suivantes :

- La structure du modèle d'expertise partielle est la même que celle du réseau bayésien. Les éléments (concepts et ressources) du modèle d'expertise partielle deviennent des nœuds dans le réseau bayésien. Le poids de chaque élément est remplacé pour chaque variable par une probabilité d'estimer le degré de maîtrise des connaissances de l'apprenant. Ces probabilités varient entre 0 (non maîtrisé) et 1 (maîtrisé).
- Les relations de type « est-un » dans le modèle d'expertise partielle deviennent des relations de dépendance conditionnelle entre les variables formant ainsi les arcs du réseau bayésien. Par conséquent, nous associons à chaque nœud une table de probabilité conditionnelle.
- Les éléments de connaissances ou les concepts représentent les variables non observables tandis que les autres éléments qui sont soit des ressources pédagogiques soit des activités d'apprentissage utilisées pour évaluer

l'apprenant (tests, exercices, forums, etc.) représentent les variables observables qui sont ajoutées au réseau bayésien.

Nous essayons de montrer le fonctionnement de notre modèle de l'apprenant qui utilise un réseau bayésien par l'exemple suivant :

Supposons que dans l'unité d'enseignement algorithmique et programmation « AP » un concept « variable » est constitué de deux concepts « variable globale » et « variable locale » considérés comme des variables non observables dont les liens sont des relations de type « est-un » (figure 17).

La valeur comprise entre 0 et 1 qui mesure l'importance relative d'un concept est définie par l'expert ou l'enseignant. Cette valeur représente le poids de l'arc qui relie le nœud père au nœud fils. La somme des poids de tous les arcs entrants ou sortants de/vers un nœud père ou fils est égale à 1.

Dans cet exemple, le nœud V « variable » a deux parents : L « variable locale » et G « variable globale ». Les relations de type « est-un » ont respectivement les poids : $w_1 = 0.6$, $w_2 = 0.4$. Le nœud L « variable locale » quant à lui, a deux relations avec les deux variables observables : E « résolution d'exercice » et F « Poster plus de 5 questions sur le forum » ayant respectivement les poids $w_3 = 0.8$, $w_4 = 0.2$.

Que l'apprenant poste plusieurs questions sur le forum montre son inculcation dans le processus d'apprentissage. Même chose pour la « résolution d'un exercice » qui prouve si l'apprenant a bien compris le concept ou non.

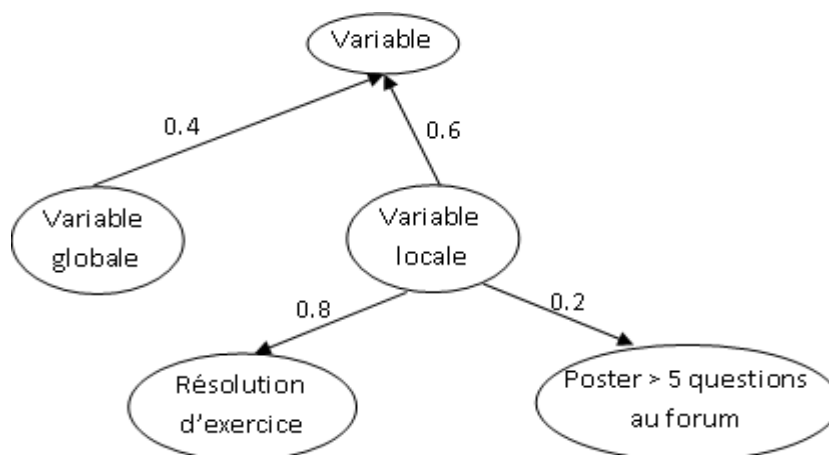


Figure 17. Exemple d'un réseau bayésien

Dans le cas général, la probabilité conditionnelle d'avoir un certain niveau de connaissance pour un concept X donné sachant les niveaux de connaissance de ces sous-concepts $Y_1, \dots, Y_n$ est calculée par la formule suivante (Nguyen, 2009) :

$$P(X = 1 | Y_1, Y_2, \dots, Y_n) = \sum_{i=1}^n w_i * h_i$$

Avec:

$$h_i = \begin{cases} 1 & \text{si } Y_i = X \\ 0 & \text{sinon} \end{cases}$$

Où les $w_1, \dots, w_n$ sont les poids des relations entre X et $Y_1, \dots, Y_n$ respectivement.

Pour l'exemple précédent, la probabilité conditionnelle de V est calculée comme suit :

$$P(V | L, G) = w_1 * h_1 + w_2 * h_2$$

avec:

$$h_1 = \begin{cases} 1 & \text{si } Y_1 = X \\ 0 & \text{sinon} \end{cases} \quad h_2 = \begin{cases} 1 & \text{si } Y_2 = X \\ 0 & \text{sinon} \end{cases}$$

En appliquant les calculs précédents, nous obtenons les tables de probabilités conditionnelles associées à chaque concept :

	L=1		L=0	
	G=1	G=0	G=1	G=0
V=1	$0.6*1 + 0.4*1 = 1.0$	$0.6*1 + 0.4*0 = 0.6$	$0.6*0 + 0.4*1 = 0.4$	$0.6*0 + 0.4*0 = 0.0$
V=0	0.0	0.4	0.6	1.0

Table 12. Table des probabilités conditionnelles du concept V

E	P(E=1)	P(E=0)
1	$0.8*1 = 0.8$	0.2
0	$0.8*0 = 0.0$	1.0

Table 13. Table des probabilités conditionnelles du concept E

F	P(F=1)	P(F=0)
1	$0.2*1 = 0.2$	0.8
0	$0.2*0 = 0.0$	1.0

Table 14. Table des probabilités conditionnelles du concept F

Les concepts L et G n'ont pas des concepts reliés par des relations de spécialisation, donc ils suivent la distribution de probabilités normale avec un medium de 0.5 :

P(L=1)	P(L=0)
0.5	0.5

Table 15. La distribution de probabilités du concept L

P(G=1)	P(G=0)
0.5	0.5

Table 16. La distribution de probabilité du concept G

Par exemple, si on calcule la table de la loi des probabilités jointes $P(V,G,L)$ en utilisant la formule $P(V,L,G) = P(V/L,G) * P(L) * P(G)$ on obtient :

	L=1		L=0	
	G=1	G=0	G=1	G=0
V=1	0.25	0.15	0.10	0.00
V=0	0.00	0.10	0.15	0.25

Table 17. Table de probabilité jointe $P(V,L,G)$

A partir de cette table, on peut calculer la probabilité marginale $P(V=1)$ qui est égale à 0.5.

De la même façon, on peut calculer la table des probabilités jointes globale en utilisant la formule :

$$P(V, L, G, E, F) = P(L) * P(G) * P(E | L) * P(F | L) * P(V | L,G)$$

4.3.3 L'inférence du niveau de connaissance de l'apprenant

Supposons que pour le concept « variable locale », un apprenant a bien résolu son exercice et a bien participé au forum en postant plusieurs questions, c.à.d. $E=1$ et $F=1$. A titre d'exemple, le réseau bayésien peut servir à répondre aux questions suivantes :

Combien le concept « variable locale » a-t-il participé dans la maîtrise du concept « variable » ?

Et dans ce cas, combien l'apprenant a-t-il maîtrisé le concept « variable » ?

La probabilité conditionnelle de la variable non observable V avec des événements fixes $E = 1$ et $F=1$, $P(V = 1 | L, G, E = 1, F = 1)$ est calculée comme suit :

$$P(V = 1 | L, G, E = 1, F = 1) = \frac{\sum_{L,G} P(V = 1, L, G, E = 1, F = 1)}{\sum_{L,G,E,F} P(V = 1, L, G, E, F)}$$

Où $P(V, L, G, E, F)$ est la distribution de probabilités jointes globale.

Pour répondre à la question précédente, nous appliquons les probabilités conditionnelles dans les tables 12, 13, 14, 15, et 16 pour déterminer $P(V, L, G, E, F)$. Ensuite nous calculons $P(V=1 | L, G, E = 1, F = 1)$

Après la mise à jour de la table des probabilités jointes $P(V,G,L)$ en prenant en compte les faits $E=1$ et $F=1$, on aura la nouvelle table des probabilités jointes $P(V,G,L)$

	L=1		L=0	
	G=1	G=0	G=1	G=0
V=1	0.50	0.30	0.00	0.00
V=0	0.00	0.20	0.00	0.00

Table 18. Table de probabilité jointe $P(V,L,G)$ en prenant en compte les faits $E=1$ et $F=1$

Donc $P(V=1 | L, G, E = 1, F = 1) = 0.80/0.80=1$

Pour $E=1$ et $F=1$, le concept «variable locale» a participé à 100% dans la maîtrise du concept « variable ». Et dans ce cas, le degré de la maîtrise du concept « variable » par l'apprenant est égale à $P(V=1) = 0.50+0.30=0.80$

4.3.4 Le réseau bayésien de notre modèle cognitif

Pour notre modèle cognitif, nous avons inversé le sens des arcs qui lient entre les concepts et les ressources pédagogiques de la figure précédente comme suit (figure 18):

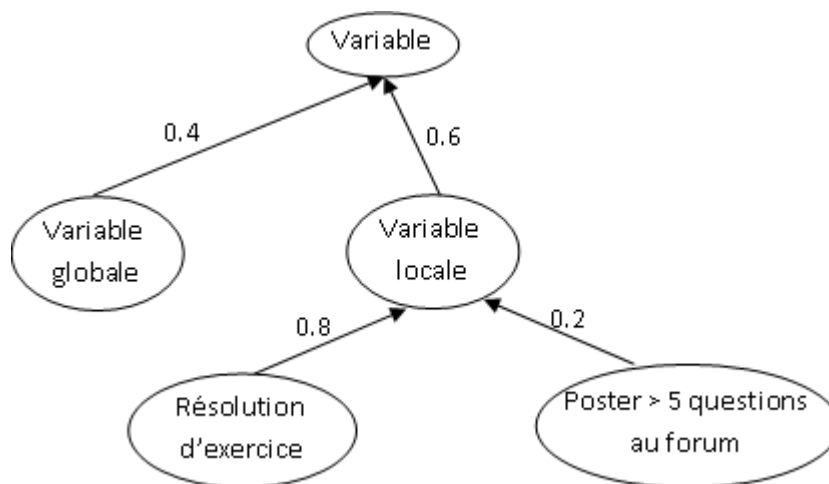


Figure 18. Exemple d'un réseau bayésien

Nous avons choisi de travailler avec l'hypothèse suivante : « Pour maîtriser un concept il faut réaliser les activités pédagogiques associées à ce concept ».

Pour le calcul de la table des probabilités jointes globale de l'exemple de la figure 18, on utilise la formule suivante :

$$P(V, L, G, E, F) = P(E) * P(F) * P(G) * P(L | E, F) * P(V | L, G)$$

4.4 Conclusion

La collaboration ne cesse d'influencer l'architecture des environnements et c'est dans ce sens qu'on a présenté une architecture d'environnement collaboratif interactif. Cette architecture est souple et modulaire, car ses différentes composantes se relaient par les événements.

Notre environnement consiste à proposer à l'apprenant des ressources pédagogiques organisées dans différents espaces (commun, privé, du groupe) et selon différents points de vue. L'accès aux ressources se fait par des ontologies intégrant des concepts, des relations et des éléments pédagogiques permettant ainsi à l'apprenant de consulter rapidement les ressources pertinentes. L'objectif visé est de permettre à plusieurs utilisateurs (apprenants, formateurs, administrateurs, gestionnaires, etc.) d'interagir collectivement avec l'environnement.

Pour ce modèle de domaine, nous avons associé un modèle cognitif de l'apprenant de type expertise partielle combiné avec un réseau bayésien afin d'inférer le niveau de connaissance de l'apprenant correspondant à chaque concept du domaine.

Les réseaux bayésiens permettent de construire des probabilités jointes à partir desquelles, on peut retrouver toutes les probabilités conditionnelles souhaitées. Les réseaux bayésiens constituent une simple technique graphique et qui sert d'exprimer pratiquement ce de quoi on est certain ou incertain. Les difficultés essentielles des réseaux bayésiens sont la description au préalable de la structure de ce réseau ainsi que la définition de toutes les valeurs de probabilités conditionnelles.

Chapitre 5 : L'approche à base d'agents pour la modélisation

5.1 Introduction

Dans ce chapitre, nous présentons une description générale des différentes phases d'analyse et de conception mises en œuvre dans notre environnement d'apprentissage E-PACAD que nous avons développé en vue de modéliser les interactions dans des groupes de formation à distance. Ce système est modélisé sous forme d'un SMA. Au chapitre 3, nous avons évoqué les raisons pour lesquelles nous avons eu recours au paradigme d'agents pour implémenter les fonctionnalités attendues.

Nous présentons dans ce chapitre l'application de la méthodologie d'analyse, de conception et de spécification des systèmes multi-agents MaSE. Ensuite, nous sommes passés à la programmation concrète de nos agents en utilisant l'architecture JADE. Nos agents ont été écrits en Java et communiquent par des messages FIPA ACL. L'utilisateur interagit avec son agent via une interface graphique. L'architecture actuelle d'E-PACAD repose sur une base de données qui sert d'interface entre le SMA et l'environnement et qui contient entre autres le modèle de l'apprenant qui représente des informations concernant l'apprenant.

Pour élaborer notre environnement d'apprentissage collaboratif E-PACAD, nous avons commencé par identifier l'ensemble des fonctionnalités que doit posséder E-PACAD pour répondre entièrement aux différents problèmes identifiés. Ensuite, nous avons dégagé un sous-ensemble d'objectifs pilotes. C'est ce sous-ensemble que nous avons effectivement implémenté dans le cadre de cette thèse.

5.2 Démarche suivie dans la conception

Les systèmes multi-agents (SMA) présentent comme caractéristiques, d'une part, de permettre le partage ou la distribution de la connaissance et, d'autre part, de faire correspondre un ensemble d'agents et de coordonner leurs actions dans un environnement pour l'accomplissement d'un but commun.

Dans notre analyse et conception, nous avons choisi la méthodologie MaSE. C'est une méthodologie qui part d'une définition des SMA très modeste et très pragmatique : dans ce cas un SMA est considéré comme un ensemble de processus informatiques qui communiquent entre eux pour atteindre un objectif global donné.

La méthodologie MaSE a été choisie pour les raisons suivantes:

- MaSE est une méthodologie qui couvre complètement le cycle de vie d'un logiciel dans les systèmes multi-agents.
- MaSE est une méthodologie rigoureuse, car les liens entre les étapes et les résultats attendus pour chaque étape sont assez bien spécifiés.
- MaSE utilise pour ses représentations graphiques les techniques d'UML (Unified Modeling Language).
- MaSE utilise des techniques propres au développement des objets et les applique au développement des SMA.
- MaSE a la qualité de décrire dans le détail les différentes étapes de la méthodologie.

MaSE comporte sept étapes réparties en deux phases. Chacune des étapes a pour résultat un ou plusieurs diagrammes.

La phase d'analyse comprend trois étapes :

- Identifier des objectifs dont le résultat est le diagramme hiérarchisé des objectifs.
- Identifier les cas d'utilisation dont le résultat est un ensemble de cas d'utilisation exprimés par un ou plusieurs diagrammes de séquence (à la UML).

- Affiner les rôles dont le résultat est le modèle de rôles de MaSE et un ensemble de diagrammes de tâches concurrentes.

La phase de conception comprend quatre étapes :

- Créer les classes d'agents dont le résultat est le diagramme de classes d'agents.
- Construire les interactions entre agents dont le résultat est un ensemble de diagrammes d'interactions.
- Assembler les classes d'agents dont le résultat est l'architecture interne des agents.
- Déployer le système dont le résultat est le diagramme de déploiement (à la UML).

5.3 La phase d'analyse

5.3.1 Cahier des charges :

Dans le but de délimiter le périmètre fonctionnel de l'application et de décrire les principaux besoins à inclure dans l'environnement, nous élaborons un cahier des charges précisant ces besoins.

5.3.1.1 Acteurs du système

L'identification des acteurs participants dans le système est primordiale pour décrire et préciser comment chacun d'eux se comporte pour servir ou être servi. Nous distinguons trois types d'acteurs : apprenant, enseignant et administrateur. Chaque utilisateur (apprenant, enseignant, administrateur) est connecté à son ordinateur et accède aux programmes et à toutes les ressources pédagogiques à travers le protocole http et les autres protocoles d'Internet. L'utilisateur a donc besoin d'un navigateur Internet pour accéder à la plate-forme et réaliser ses activités d'apprentissage dans cette plateforme. Nous allons détailler les besoins fonctionnels de chaque acteur :

5.3.1.1.1 Acteur Apprenant

Le rôle principal de cet acteur est de transformer les informations en connaissances. Cet acteur a besoin de :

- Accéder à l'unité de formation recherchée.
- Connecter et déconnecter de l'environnement
- Accéder aux différents espaces de travail (espace de l'apprenant, espace du groupe, espace de formation).
- naviguer dans l'ontologie d'application.
- Accéder aux ressources pédagogiques indexées par les notions.
- Ajouter une ressource dans l'espace privé ou l'espace du groupe.
- Ajouter un point d'accès dans l'espace privé ou l'espace du groupe.
- Demander l'évaluation.
- Communiquer avec ses paires (créer un message, accéder à un message, répondre à un message).
- Sauvegarder la trace (notions consultées, ressources chargées, communications avec apprenants/enseignants, etc.).

5.3.1.1.2 Acteur Enseignant

Son rôle principal est de rendre disponibles les informations pour l'apprentissage et de faciliter cet apprentissage sur le plan pédagogique, il a besoin de :

- Connecter et déconnecter de l'environnement.
- Proposer les cours et les exercices relatifs à une unité d'enseignement.
- Accéder à un espace de travail.
- Naviguer dans l'ontologie d'application.
- Évaluer le travail de l'apprenant.

- Communiquer avec les autres utilisateurs.

5.3.1.1.3 Acteur Administrateur

Le rôle de l'administrateur est de :

- Connecter et déconnecter du système.
- Gérer les formations.
- Gérer les comptes des utilisateurs.
- Gérer les groupes des utilisateurs.
- Gérer les espaces de travail.
- Gérer les ressources pédagogiques.

Dans ce qui suit, nous ne parlerons que de l'acteur apprenant.

5.3.2 Le diagramme hiérarchisé des objectifs

Le diagramme hiérarchisé des objectifs a pour principal but de résumer et d'organiser les besoins du système sous la forme d'une arborescence des objectifs et sous objectifs. Un environnement d'apprentissage est conçu pour fournir aux apprenants un espace de travail et un espace de communication. Donc, les principales fonctionnalités que doit offrir un tel environnement sont :

- Surveiller les connections et les déconnections de l'apprenant.
- Gérer les différents espaces de travail et l'espace de communication.
- Naviguer dans l'ontologie d'application.
- Accéder aux ressources pédagogiques de l'environnement.
- Ajouter des ressources pédagogiques.
- L'extraction et la sauvegarde des données dans les bases de données.

Mais les objectifs principaux de notre travail de modélisation de l'apprenant sont :

- La collection des informations concernant l'apprenant (profil et préférences).
- La sauvegarde de ses traces de navigation dans l'environnement d'apprentissage (modèle comportemental).
- L'inférence de son modèle de connaissances en fonction de ses activités (modèle cognitif).

Ces objectifs ont été ajoutés aux fonctionnalités principales de l'environnement d'apprentissage. Donc, les objectifs qui indiquent les services que doit fournir notre environnement sont organisés dans le diagramme hiérarchisé des objectifs suivant (figure 19) :

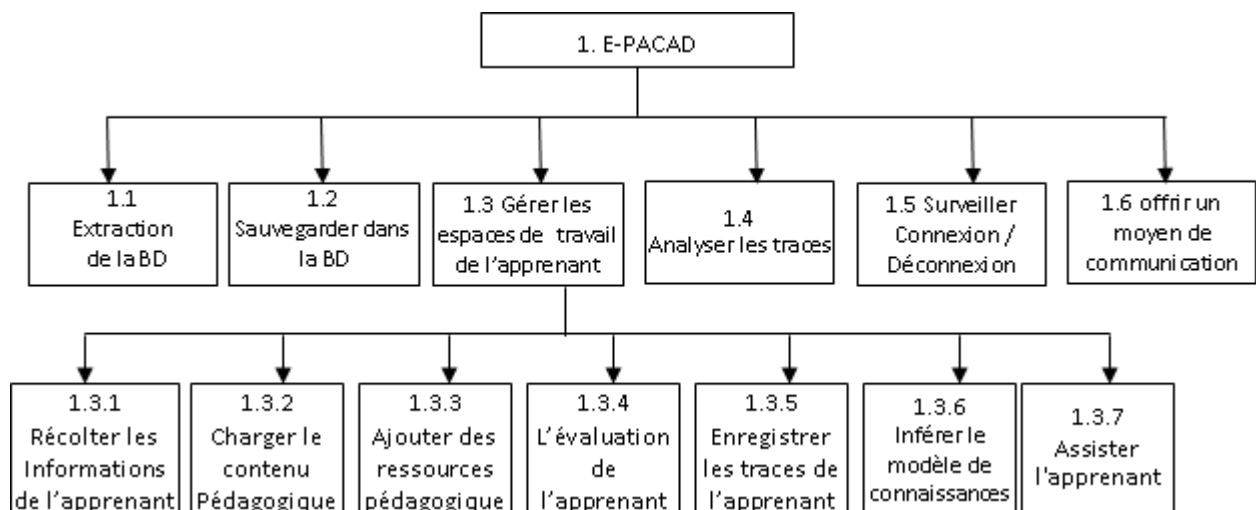


Figure 19. Diagramme hiérarchisé des objectifs.

5.3.3 Diagrammes de séquence

Les diagrammes de séquence dans MaSE indiquent les interactions entre les rôles. Pour l'ensemble des cas d'utilisation de notre environnement d'apprentissage à distance, nous avons défini les rôles suivants :

R_Interface : encapsule l'interface entre l'apprenant et le système. C'est lui le responsable des interactions avec l'apprenant lui présentant le contenu pédagogique. Il transfère également les actions et les réponses de l'apprenant au rôle assistant.

R_Assistant : encapsule les tâches effectuées par l'agent de l'apprenant qui consistent à adresser des requêtes relatives au savoir (charger une ressource, ajouter une ressource, etc.).

R_Spy : encapsule les tâches effectuées par l'agent de l'apprenant qui consistent à adresser des requêtes pour sauvegarder les traces des activités de l'apprenant dans le modèle de l'apprenant.

R_Collector : il collecte les informations concernant l'apprenant et les sauvegarde dans le modèle de l'apprenant.

R_Collaborator : il accompagne l'apprenant durant son interaction avec les autres (apprenants ou enseignants) via les outils de communication et envoie les traces d'interaction au rôle Assistant.

R_Evaluator : il effectue l'évaluation automatique des exercices en comparant les connaissances de l'apprenant à celles de l'expert et en envoyant les résultats d'évaluation au rôle assistant pour les sauvegarder dans le modèle de l'apprenant.

R_Analyser : il analyse les traces enregistrées dans le modèle de l'apprenant afin de fournir à l'agent demandeur l'état de l'apprenant (nombre de messages reçus, nombre de messages envoyés, nombre de messages lus, le type de message, l'émetteur du message, nombre d'accès à une ressource pédagogique, pourcentage de lecture d'une ressource pédagogique, durée de consultation, les activités de l'apprenant, etc.).

R_DB_Supervisor (R_DBS) : encapsule les tâches de l'agent superviseur de la base des données qui consistent principalement à répondre aux requêtes venant des agents enseignants et apprenant concernant le savoir (les concepts, les ressources pédagogiques, etc.). Il est le seul à faire des transactions avec les bases de stockage des données.

R_DB (R_DB) : encapsule l'interface SYSTEME-BD (BD est la base des données qui contient le contenu pédagogique).

R_Learner_model (R_LM) : encapsule l'interface SYSTEME-LMD (LMD est la base de données qui contient le modèle de l'apprenant).

Nous présentons dans les figures suivantes les diagrammes de séquence décrivant les différents cas d'utilisation.

5.3.3.1 Cas d'utilisation « Accéder à l'unité d'enseignement »

Ce cas d'utilisation (figure 20) décrit la façon dont le système gère l'accès à une unité d'enseignement. La demande, initiée par l'apprenant et portant le nom de l'unité choisie, est adressée au rôle R_Interface puis au superviseur de la base de données. Ce dernier envoie une requête SQL au R_DB qui va, à son tour, envoyer le résultat de la requête (la description de l'unité et le contenu) au superviseur puis à l'interface qui va, à son tour, afficher une nouvelle page à l'utilisateur.

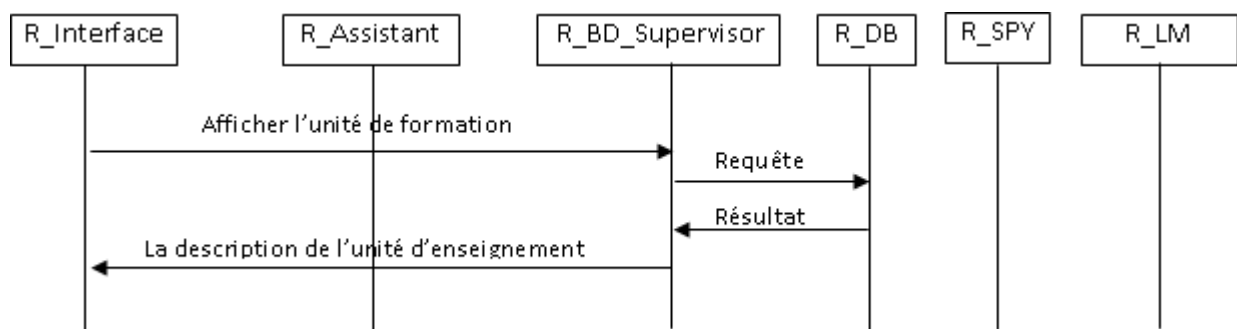


Figure 20. Diagramme de séquence «Accéder à l'unité de formation»

5.3.3.2 Cas d'utilisation « Connexion de l'apprenant »

Ce cas d'utilisation (figure 21) décrit la façon dont le système gère la connexion de l'apprenant. La demande, initiée par l'apprenant, est adressée au rôle R_Interface. Ce dernier envoie le nom d'utilisateur et le mot de passe au rôle superviseur de la base de données pour vérification. Le résultat de cette vérification sera transmis au rôle R_Interface. Si la connexion est établie, les rôles R_Assistant et R_Spy vont être créés. Le rôle R_Interface informe le rôle R_Spy pour qu'il sauvegarde la trace dans le modèle de l'apprenant.

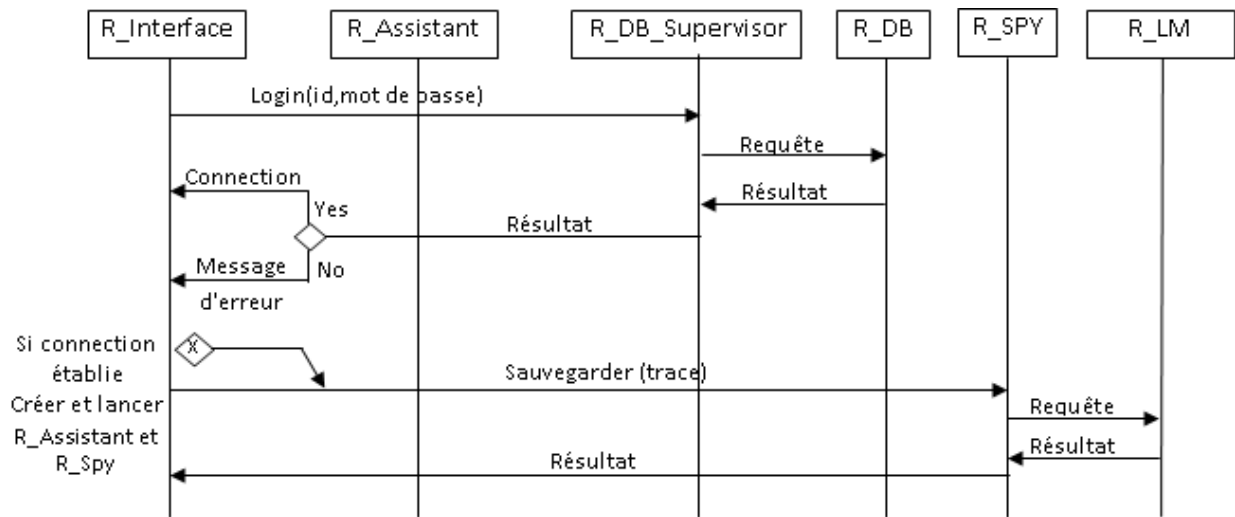


Figure 21. Diagramme de séquence « Connexion d'un apprenant »

5.3.3.3 Cas d'utilisation « Accéder à un espace de travail »

Ce cas d'utilisation (figure 22) décrit la façon dont le système charge le contenu pédagogique représenté par une ontologie d'application qui correspond à un espace de travail (formation, privé ou groupe) d'une unité de formation. La demande, initiée par l'apprenant et portant le nom de l'unité et l'espace choisi, est adressée aux rôles R_Interface et R_Assistant puis au superviseur de la base de données. Ce dernier envoie une requête SQL au R_DB qui va, à son tour, envoyer le résultat de la requête (le contenu) au superviseur puis au R_Assistant et enfin à l'interface. La trace est sauvegardée par le rôle R_Spy dans le modèle de l'apprenant.

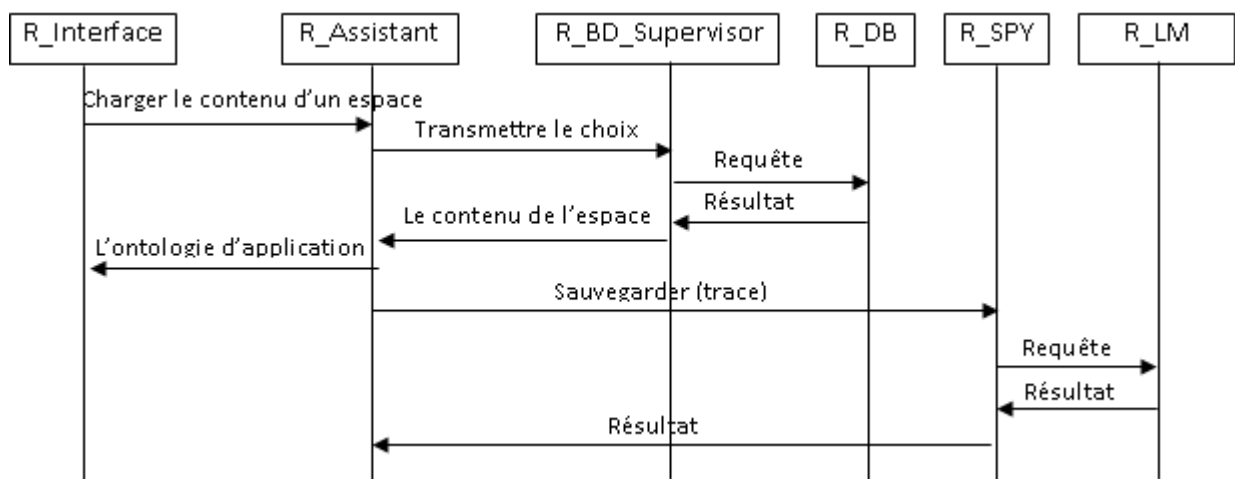


Figure 22. Diagramme de séquence « Accéder à un espace de travail »

5.3.3.4 Cas d'utilisation « Sélectionner un point d'accès »

Ce cas d'utilisation (figure 23) décrit la façon dont le système initie la navigation au sein de l'ontologie d'application. La demande, initiée par l'apprenant et portant le nom de la notion choisie, est adressée aux rôles R_Interface et R_Assistant puis au superviseur de la base de données. Ce dernier envoie une requête SQL au R_DB qui va, à son tour, envoyer le résultat de la requête (la liste des ressources pédagogiques indexées par la notion en question) au superviseur puis au R_Assistant et enfin à l'interface. Le rôle R_Assistant infère le modèle de connaissances de l'apprenant et sauvegarde le résultat de cette inférence dans le modèle de l'apprenant (mise à jour des couches visited et inferred). La trace est sauvegardée par le rôle R_Spy dans le modèle de l'apprenant.

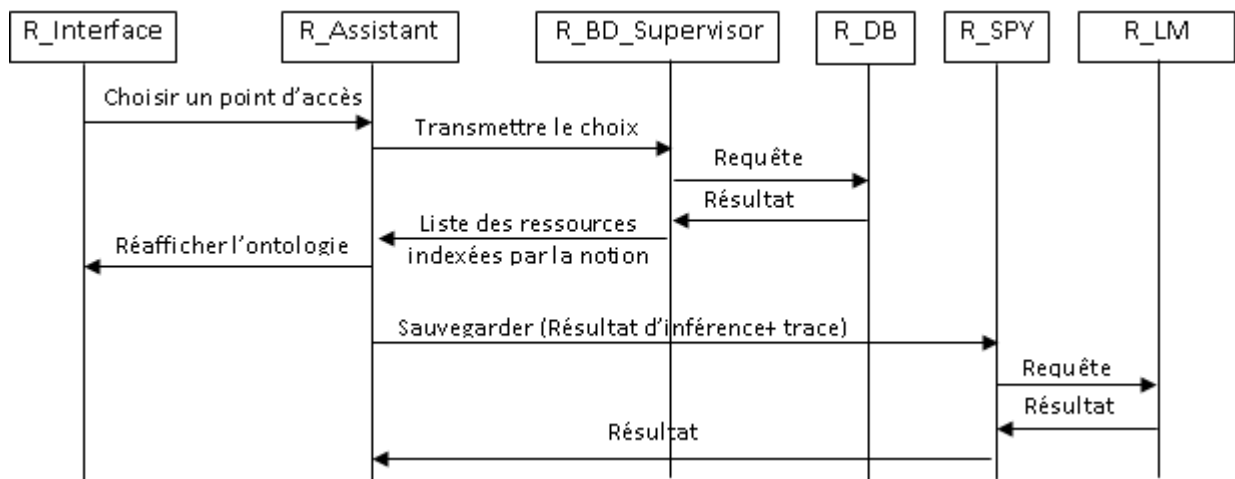


Figure 23. Diagramme de séquence «Sélectionner un point d'accès»

5.3.3.5 Cas d'utilisation « Charger une ressource pédagogique »

Ce cas d'utilisation (figure 24) décrit la façon dont le système charge une ressource pédagogique. La demande, initiée par l'apprenant et portant le nom de la ressource choisie, est adressée aux rôles R_Interface et R_Assistant puis au superviseur de la base de données. Ce dernier envoie une requête SQL au R_DB qui va, à son tour, envoyer le résultat de la requête (la ressource) au superviseur puis au R_Assistant et enfin à l'interface. Le rôle R_Assistant infère le modèle de connaissances de l'apprenant et sauvegarde le résultat de cette inférence dans le modèle de l'apprenant (mise à jour de la couche inferred). La trace est sauvegardée par le rôle R_Spy dans le modèle de l'apprenant.

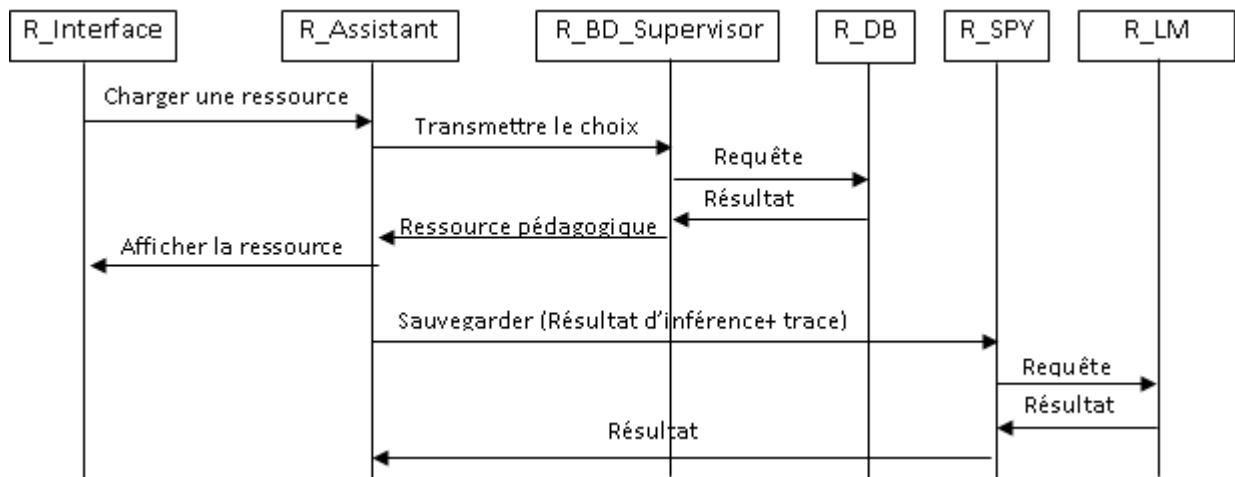


Figure 24. Diagramme de séquence «Charger une ressource pédagogique»

5.3.3.6 Cas d'utilisation « Ajouter un point d'accès »

Ce cas d'utilisation (figure 25) décrit la façon dont le système gère l'ajout d'un point d'accès dans l'espace privé de l'apprenant ou l'espace du groupe. Quand un apprenant désire ajouter un point d'accès dans son espace privé ou l'espace de son groupe courant, son agent interface informe l'assistant puis le superviseur de la base de données qui effectue la tâche. La trace est aussi sauvegardée dans le modèle de l'apprenant par le rôle R_Spy.

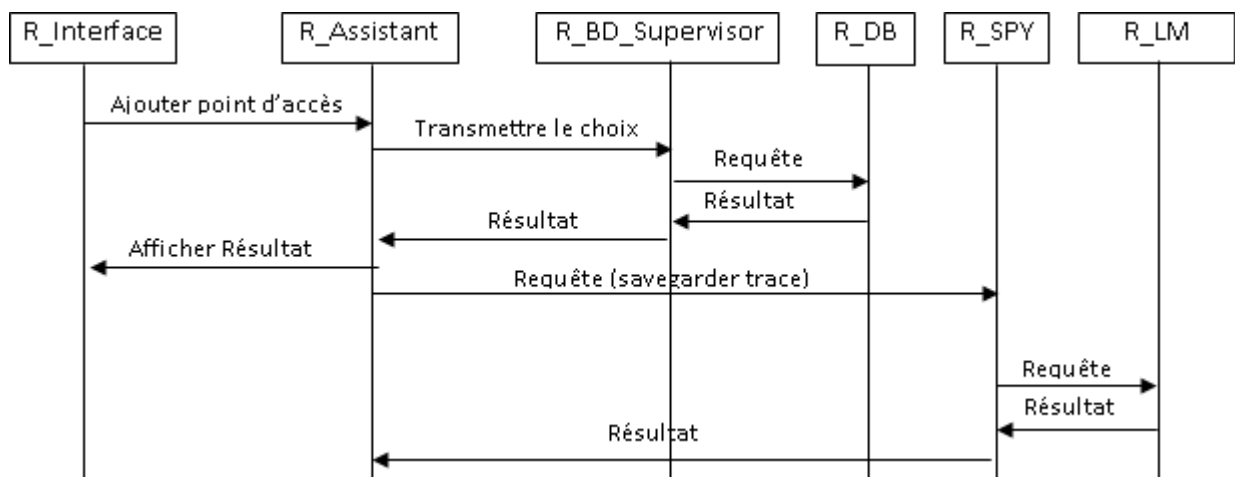


Figure 25. Diagramme de séquence «Ajouter un point d'accès »

5.3.3.7 Cas d'utilisation « Ajouter une ressource »

Ce cas d'utilisation (figure 26) décrit la façon dont le système gère l'ajout d'une ressource dans l'espace privé de l'apprenant ou l'espace du groupe. Quand un

apprenant désire ajouter une ressource dans son espace privé ou l'espace de son groupe courant, son agent interface informe l'assistant puis le superviseur de la base de données qui effectue la tâche. La trace est aussi sauvegardée dans le modèle de l'apprenant par le rôle R_Spy.

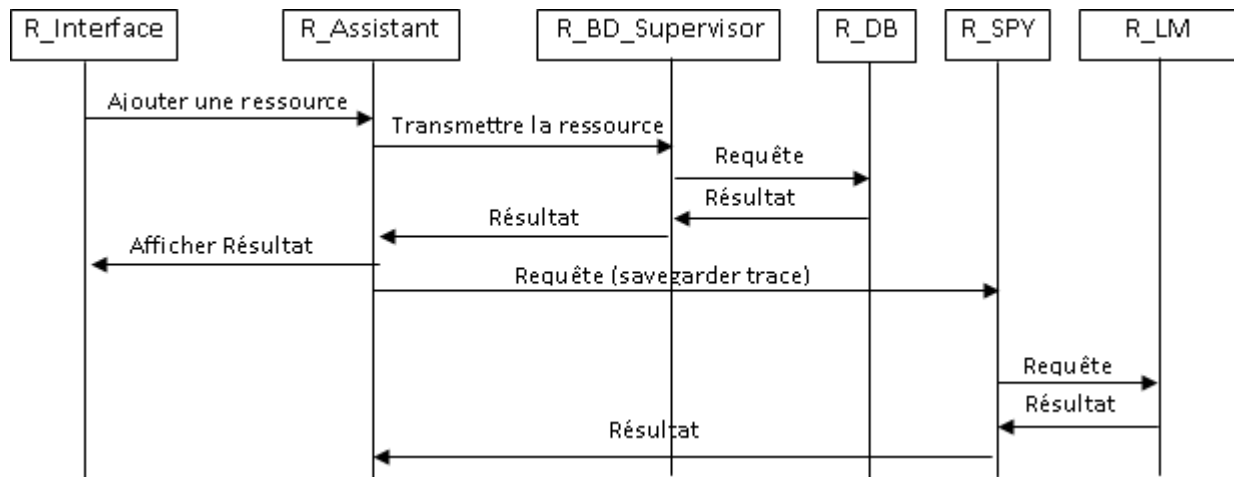


Figure 26. Diagramme de séquence « Ajouter une ressource »

5.3.3.8 Cas d'utilisation « Travailler sur exercice de type QCM »

Ce cas d'utilisation (figure 27) décrit la façon dont le système gère le travail de l'apprenant. Quand un apprenant désire travailler sur un exercice de type QCM, son choix est transféré au superviseur de la base de données par le biais du rôle interface et du rôle assistant. Le superviseur envoie l'exercice choisi à l'apprenant, ce dernier effectue le travail. Une fois la solution est validée par l'apprenant, une évaluation automatique est lancée par l'évaluateur qui compare la solution de l'apprenant et celle de l'expert. Le résultat de cette évaluation est envoyé à l'assistant qui, à son tour, lance un processus d'inférence du modèle de connaissances de l'apprenant en se basant sur ce résultat et bien sûr une sauvegarde de la trace de l'activité ainsi que ses résultats (la notion, la solution, etc.) est effectuée dans le modèle de l'apprenant par le rôle R_Spy. Le rôle R_Assistant effectue la mise à jour de la couche évaluée correspondante à la notion choisie.

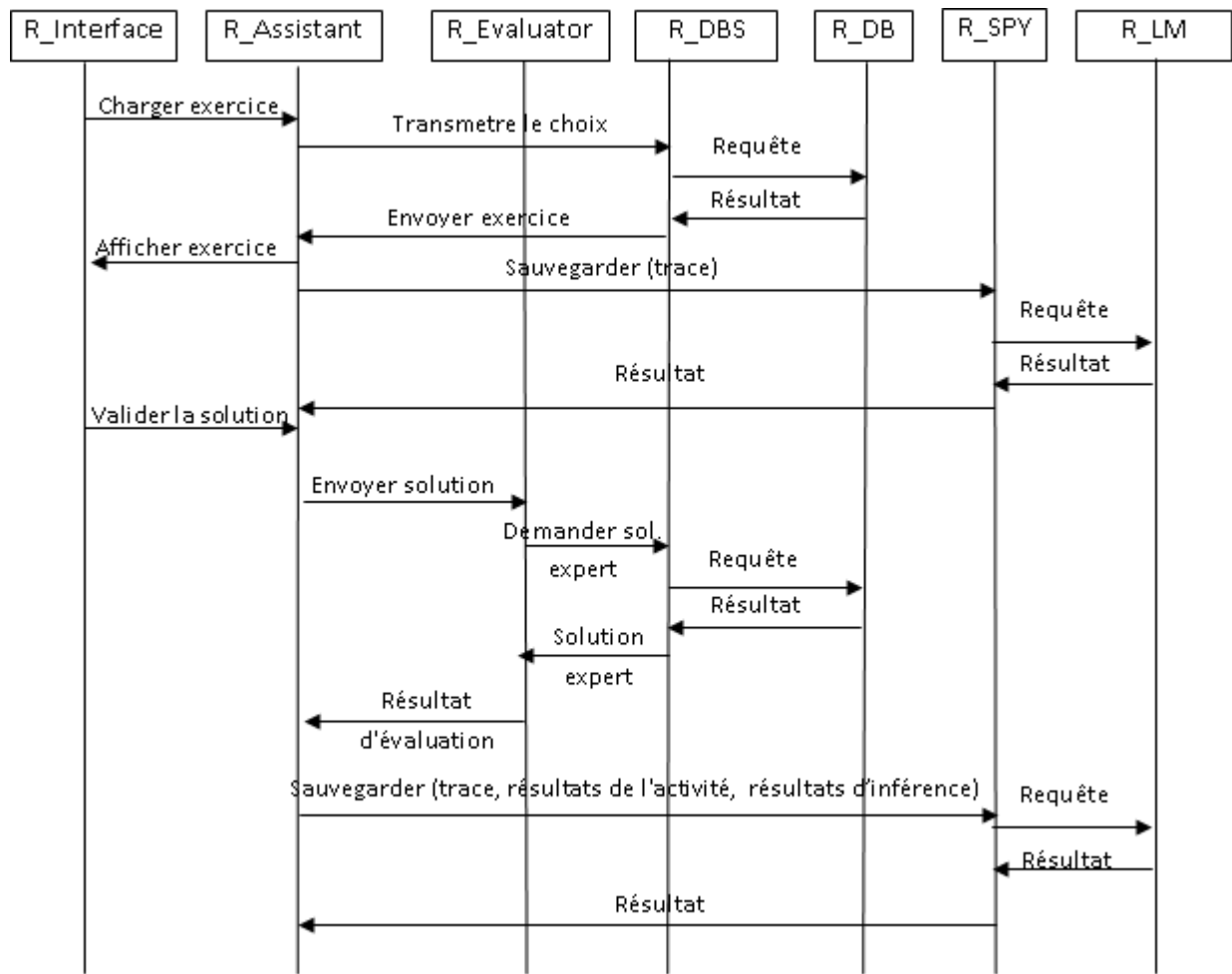


Figure 27. Diagramme de séquence « Travailler sur un exercice QCM »

5.3.3.9 Cas d'utilisation « Charger la liste des utilisateurs »

Ce cas d'utilisation (figure 28) décrit la façon dont le système charge la liste des utilisateurs (apprenants et enseignants) de l'environnement.

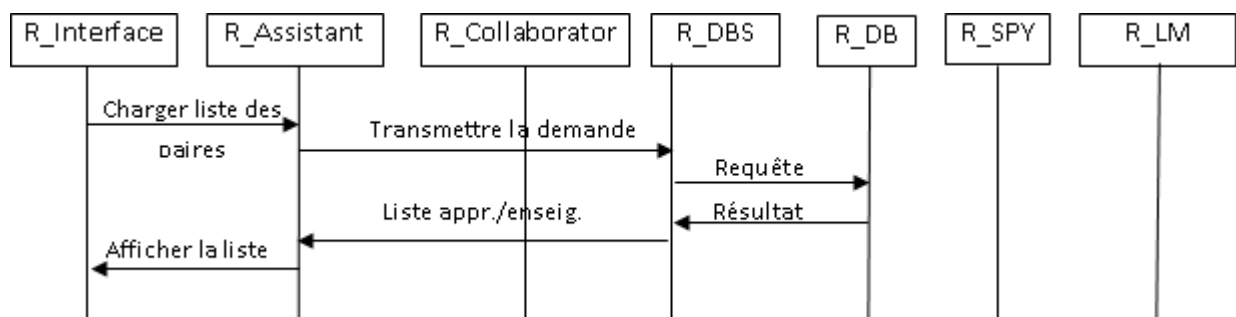


Figure 28. Diagramme de séquence « Charger la liste des utilisateurs »

Quand un apprenant désire afficher la liste de ses paires, il informe son agent interface qui, à son tour, informe l'assistant puis le superviseur de la base de données pour lui envoyer la liste des destinataires. La liste des apprenants et des enseignants lui est envoyée par le superviseur de la base de données.

5.3.3.10 Cas d'utilisation « Communiquer avec ses paires »

Ce cas d'utilisation (figure 29) décrit la façon dont le système gère la communication entre l'apprenant et ses paires (les membres de son groupe et ses enseignants). Quand un apprenant désire envoyer un message à un de ses pairs. Il informe son agent interface qui, à son tour, informe l'assistant puis le superviseur de la base de données pour lui envoyer la liste des destinataires. La liste des apprenants et des enseignants lui est envoyée par le superviseur de la base de données. En choisissant son destinataire, son agent assistant demande une autorisation de communication de l'agent collaborateur. Si la communication est autorisée, l'apprenant envoie son message à son destinataire sous la supervision du rôle collaborateur. Ce dernier informe le R_Spy pour sauvegarder la trace et le message dans le modèle de l'apprenant.

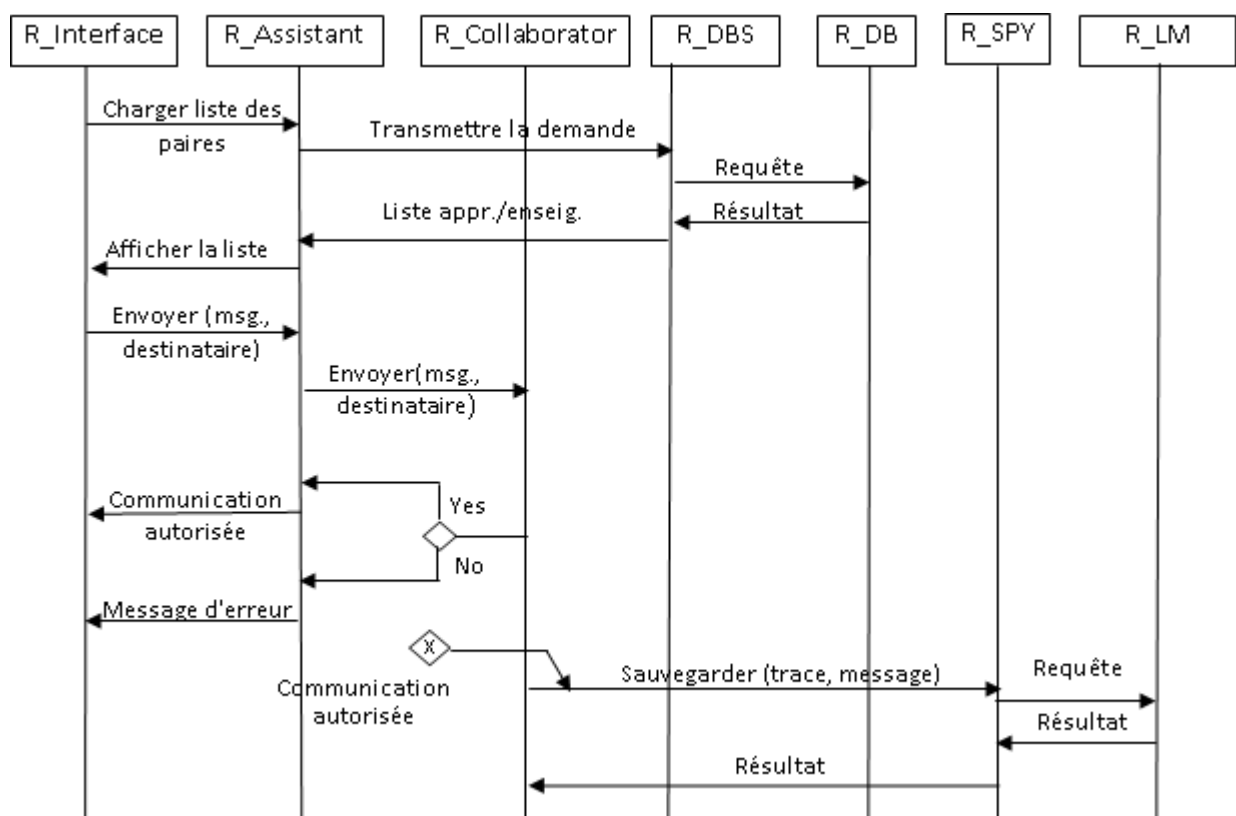


Figure 29. Diagramme de séquence « Envoyer un message »

S'il y a un message destiné à l'apprenant, le rôle collaborator informe l'assistant qui va transmettre le message à l'interface et lance une requête pour sauvegarder cette activité y compris le contenu du message dans le modèle de l'apprenant (figure 30).

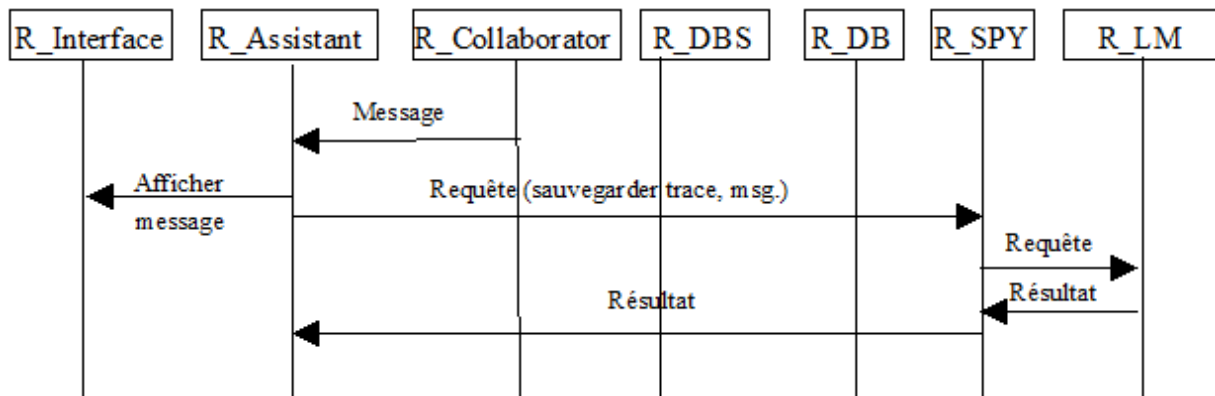


Figure 30. Diagramme de séquence « Recevoir un message »

5.3.3.11 Cas d'utilisation « Déconnexion de l'apprenant »

Ce cas d'utilisation (figure 31) décrit la façon dont le système gère la déconnexion de l'apprenant. La demande, initiée par l'apprenant, est adressée au rôle R_Interface puis au rôle R_Assistant qui va, à son tour, informer le rôle R_Spy pour sauvegarder la trace de déconnexion dans le modèle de l'apprenant. Ensuite, les rôles Interface, Assistant et Spy seront supprimés du système.

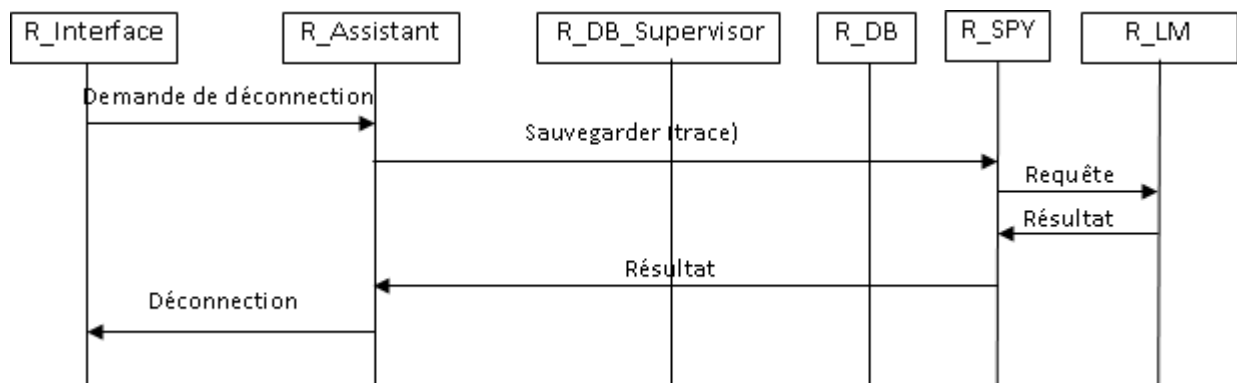


Figure 31. Diagramme de séquence « Déconnexion d'un apprenant »

5.3.4 Le diagramme des rôles et des tâches

L'objectif de cette étape est de transformer le diagramme hiérarchisé des objectifs et les diagrammes de séquence issus des différents cas d'utilisation en un diagramme de rôles

associés aux tâches concurrentes. Les tâches sont des processus indépendants que le rôle doit exécuter pour atteindre ses objectifs.

Les interactions inter-rôles sont dessinées par des chemins de communication entre les tâches. Chacun des objectifs ou des sous objectifs du diagramme hiérarchisé des objectifs est assigné à un rôle particulier. La méthodologie MaSE indique que chaque objectif doit être assigné à au moins un rôle. La figure 32 montre l'ensemble des rôles et des tâches associées à ces rôles de notre environnement.

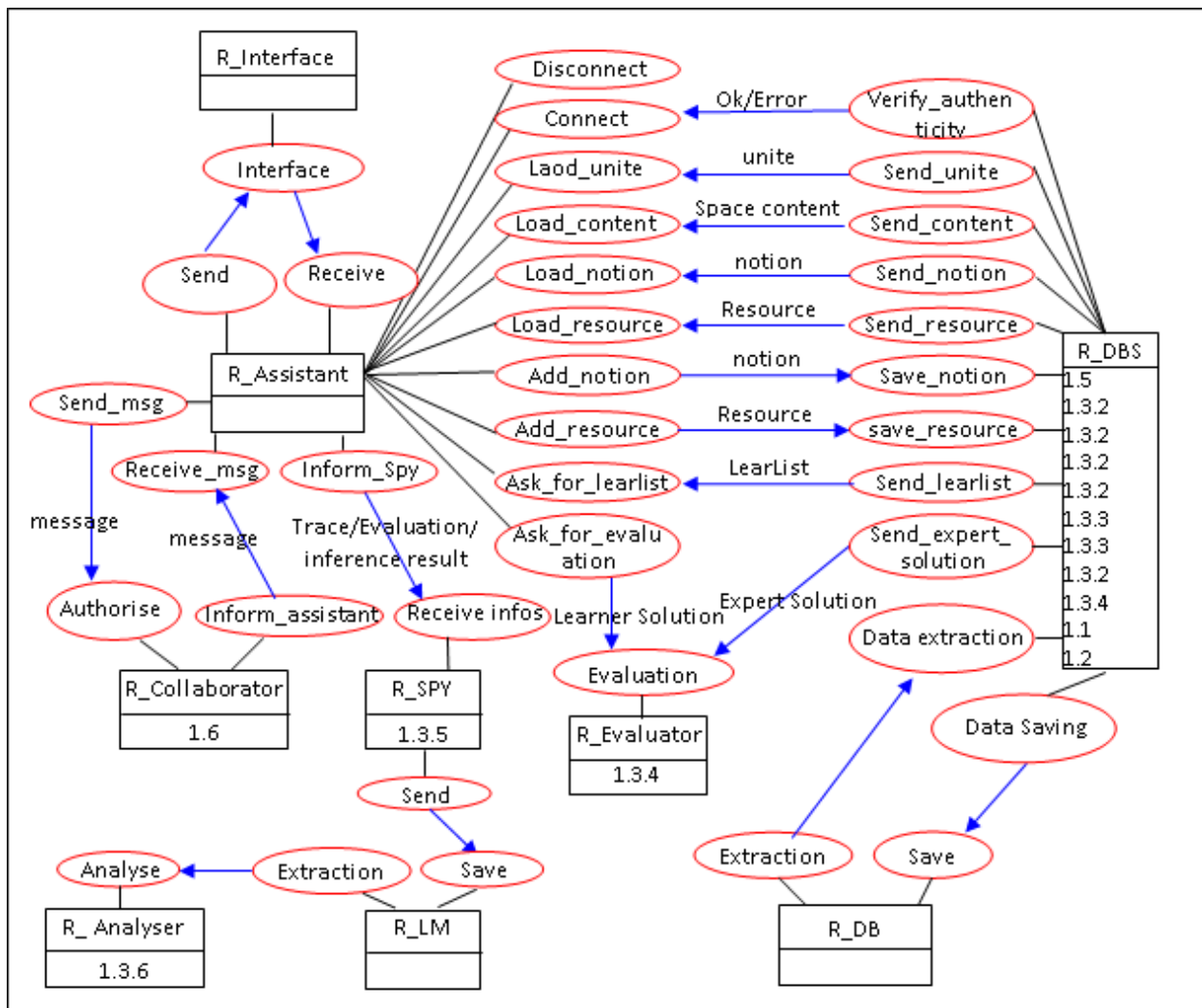


Figure 32. Diagramme des rôles et des tâches

5.3.5 La description des tâches concurrentes

Le but de cette étape est de spécifier les différentes tâches et les transitions des rôles dans la réalisation d'une tâche. Les automates à états finis sont utilisés pour illustrer le fonctionnement d'une demande de service formulée par un rôle émetteur de cette demande et la manière dont le rôle récepteur de la demande l'exécute. Nous décrivons dans ce qui suit (figures 33 et 34) deux automates à états finis : l'une pour un rôle émetteur d'une demande et l'autre pour un rôle traitant cette demande.

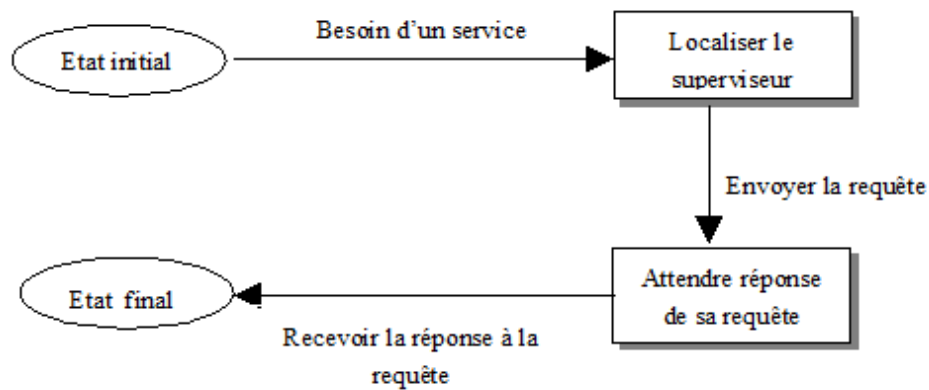


Figure 33. États et transitions du rôle demandeur d'un service.

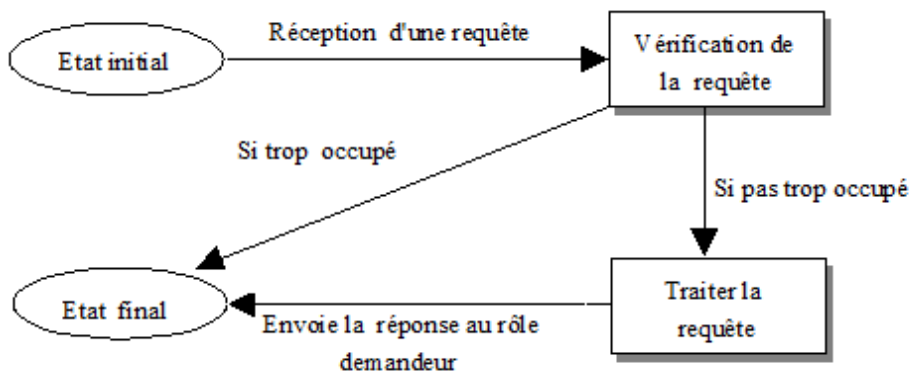


Figure 34. États et transitions du rôle récepteur de la demande.

Dans les figures suivantes, nous présentons les diagrammes d'états et de transitions de quelques tâches concurrentes:

5.3.5.1 L'accès à l'unité d'enseignement

Ce protocole de communication décrit la façon dont l'agent d'interface de l'apprenant (R_Interface) demande d'accéder à l'unité d'enseignement choisie et obtient l'unité et sa description de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre les rôles R_Interface et R_SBD avec leurs tâches concurrentes respectivement, *Laod_unite* et *Send_unite*. L'automate à états finis représentant les états et les transitions de R_Interface est présenté sur la figure 35, celui de R_SBD est présenté sur la figure 36.

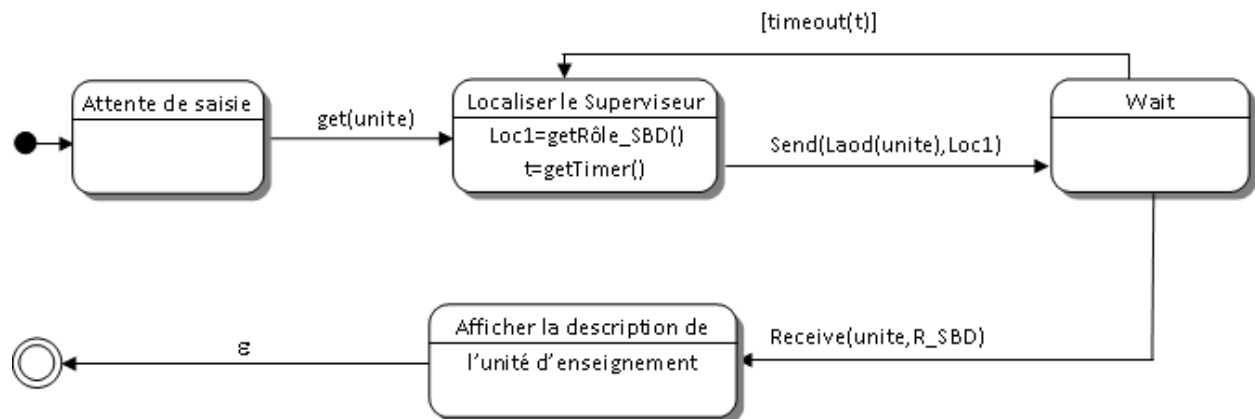


Figure 35. états et transitions du rôle R_Assistant demandant d'accéder à une unité de formation

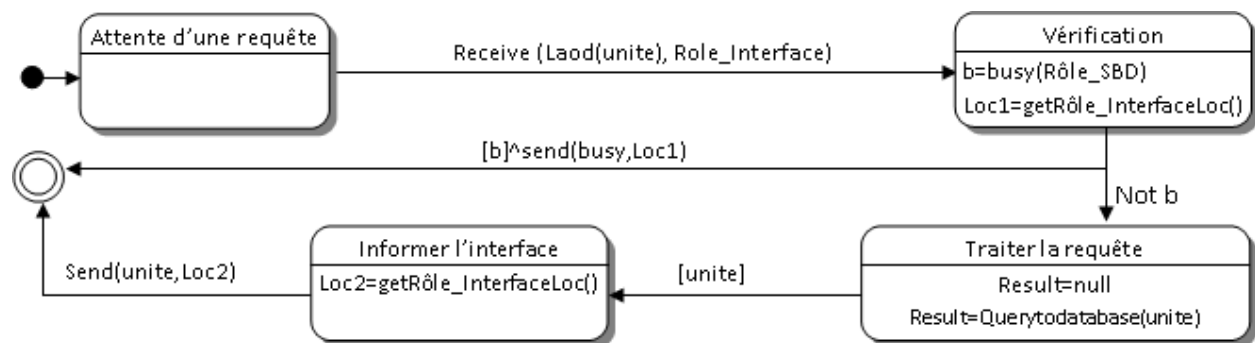


Figure 36. états et transitions du R_SBD envoyant l'unité demandée

Le concept *busy* est utilisé pour indiquer qu'un agent (un rôle à cette étape de l'analyse) est très occupé et qu'il ne peut pas répondre à une requête reçue. La syntaxe est :

Boolean=busy(agent) ;

Le concept *getTimer()* est utilisé pour Renvoyer le nombre de millisecondes qui se sont écoulées depuis le lancement.

5.3.5.2 La connexion de l'utilisateur

Ce protocole de communication décrit la façon dont l'agent d'interface de l'apprenant (R_Interface) demande et obtient l'autorisation de la connexion des utilisateurs de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre les rôles R_Interface et R_SBD avec leurs tâches concurrentes respectivement, *Connect* et *verify_authenticity*. L'automate à états finis représentant les états et les transitions de R_Interface est présenté sur la figure 37, celui de R_SBD est présenté sur la figure 38.

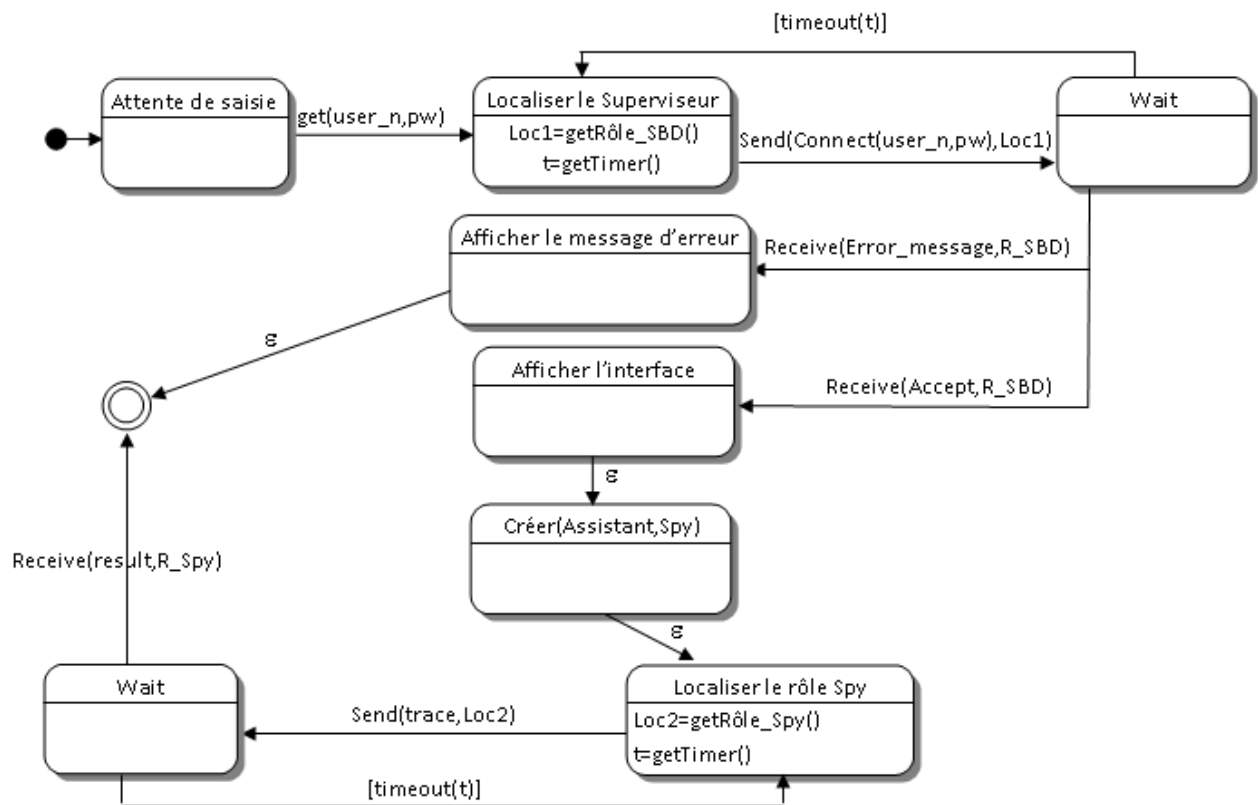


Figure 37. états et transitions de l'Interface émettant une demande de connexion d'un apprenant

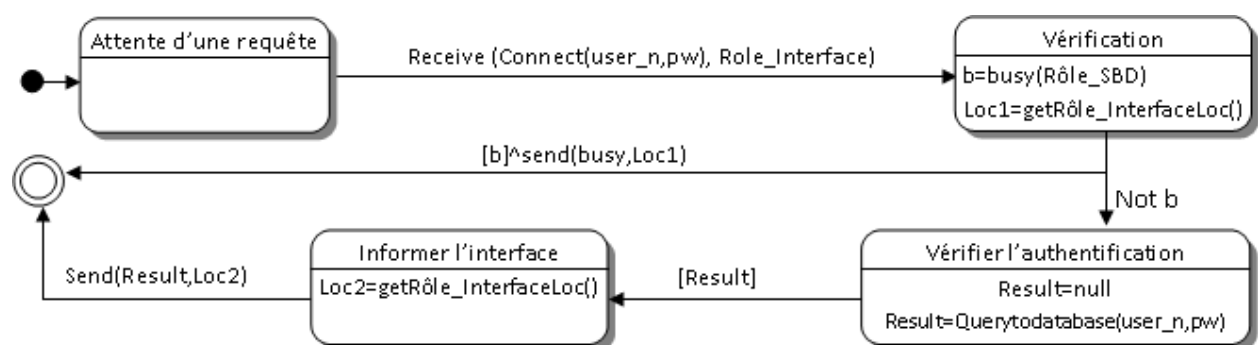


Figure 38. états et transitions du R_SBD envoyant le résultat de la connexion de l'apprenant

5.3.5.3 L'accès à un espace de travail

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande et obtient le contenu de l'espace de travail choisi (espace de formation, privé ou du groupe) de la part de l'agent superviseur de la base de données (R_SBD). Ce contenu est représenté par une ontologie d'application générée par l'agent assistant. Ce protocole met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_SBD avec leurs tâches concurrentes respectivement, *Load_content* et *send_content*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 39, celui de R_SBD est présenté sur la figure 40.

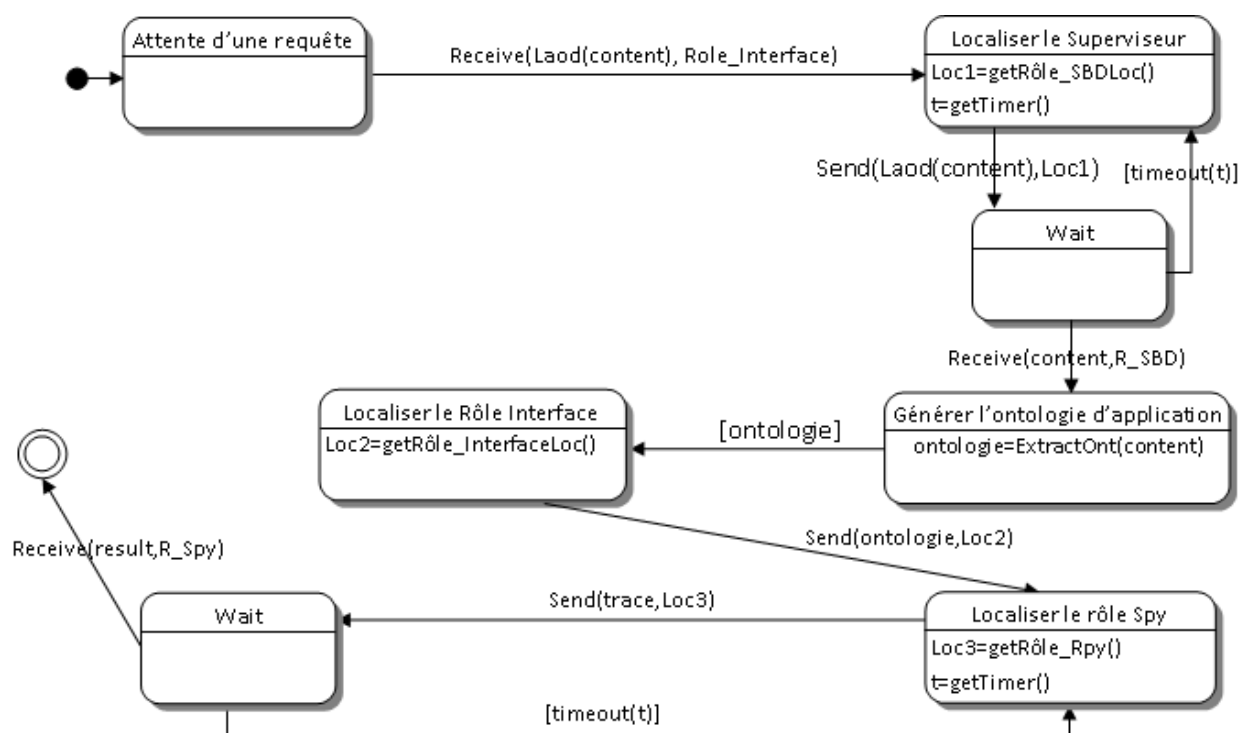


Figure 39. états et transitions du rôle R_Assistant demandant le contenu d'un espace de travail

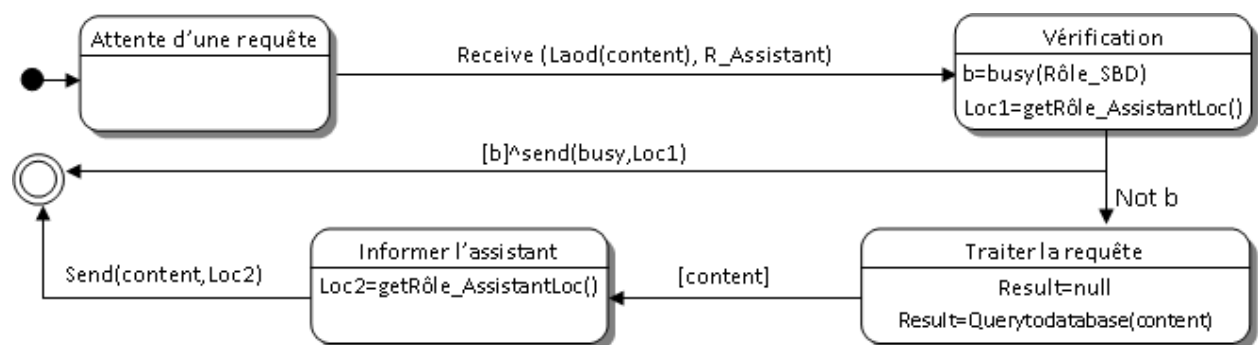


Figure 40. états et transitions du R_SBD envoyant le contenu d'un espace de travail

5.3.5.4 La consultation d'un concept (notion)

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande et obtient une notion de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_SBD avec leurs tâches concurrentes respectivement, *Load_notion* et *send_notion*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 41, celui de R_SBD est présenté sur la figure 42.

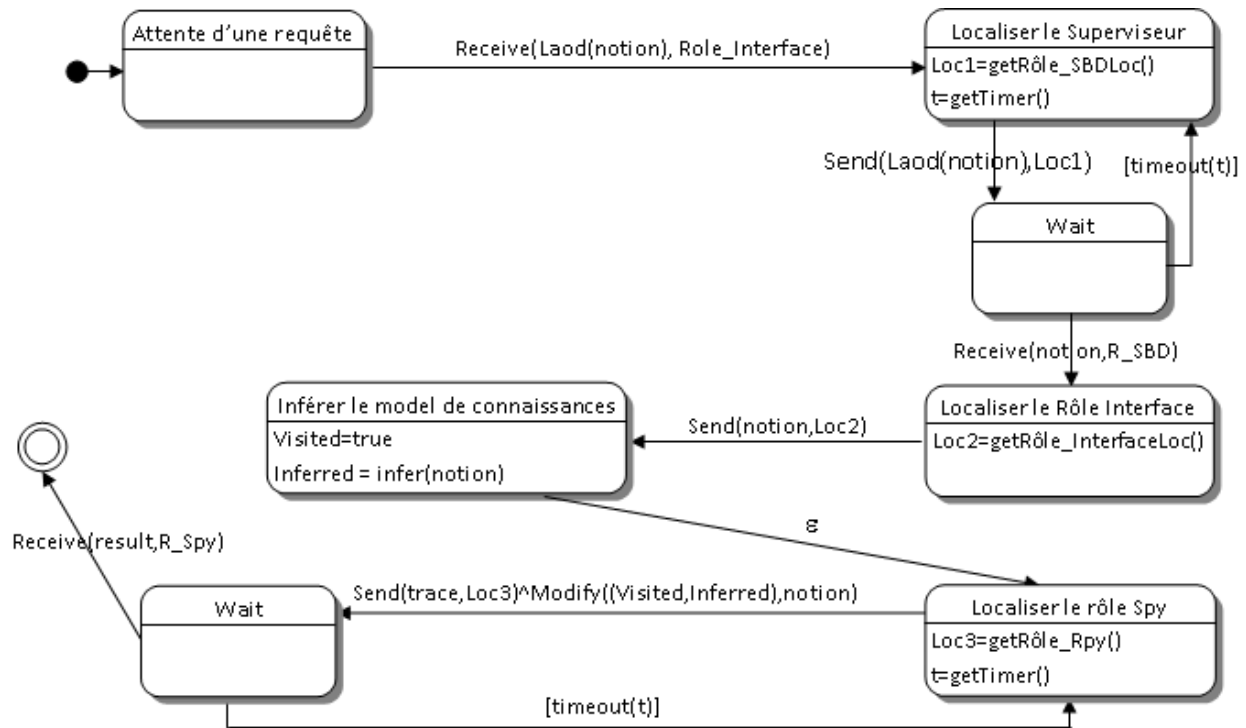


Figure 41. états et transitions du R_Assistant demandant la consultation d'un concept (notion)

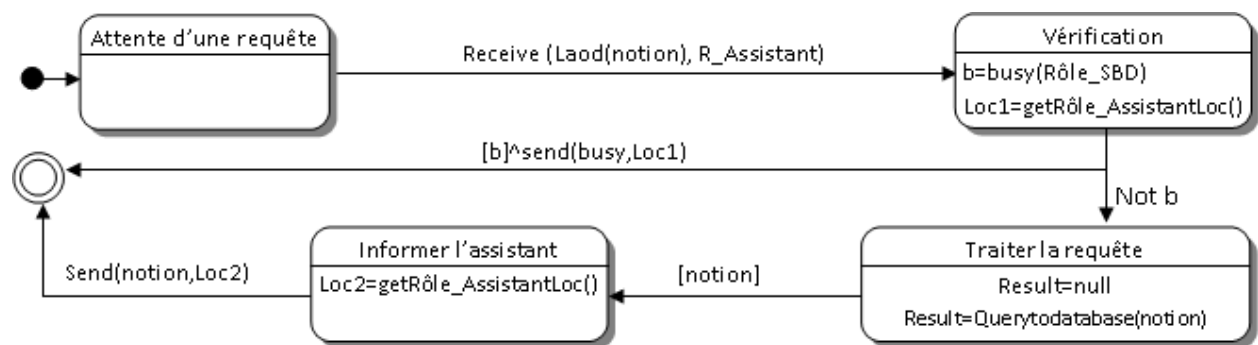


Figure 42. états et transitions du R_SBD envoyant le concept courant au R_Assistant

5.3.5.5 Le chargement d'une ressource pédagogique

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande et obtient une ressource pédagogique de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_SBD avec leurs tâches concurrentes respectivement, *Load_resource* et *send_resource*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 43, celui de R_SBD est présenté sur la figure 44.

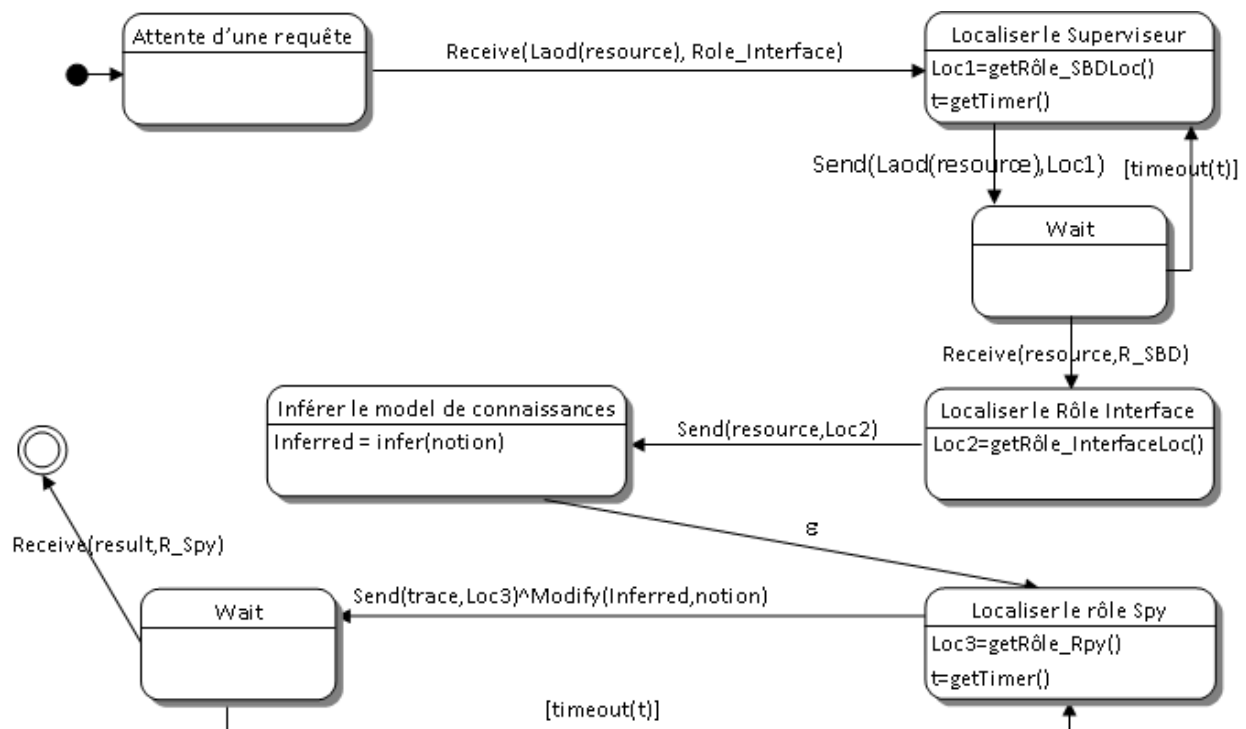


Figure 43. états et transitions du R_Assistant demandant le chargement d'une ressource pédagogique

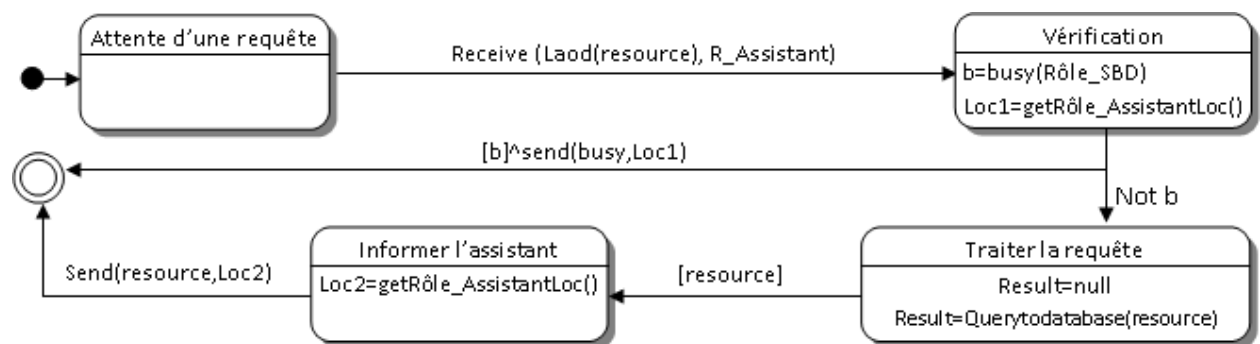


Figure 44. états et transitions du R_SBD envoyant la ressource pédagogique au R_Assistant

5.3.5.6 L'ajout d'un point d'accès dans l'espace privé ou l'espace du groupe

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande l'ajout d'un point d'accès dans son espace privé ou l'espace de son groupe et obtient ce service de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_SBD avec leurs tâches concurrentes respectivement, *Add_notion* et *Save_notion*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 45, celui de R_SBD est présenté sur la figure 46.

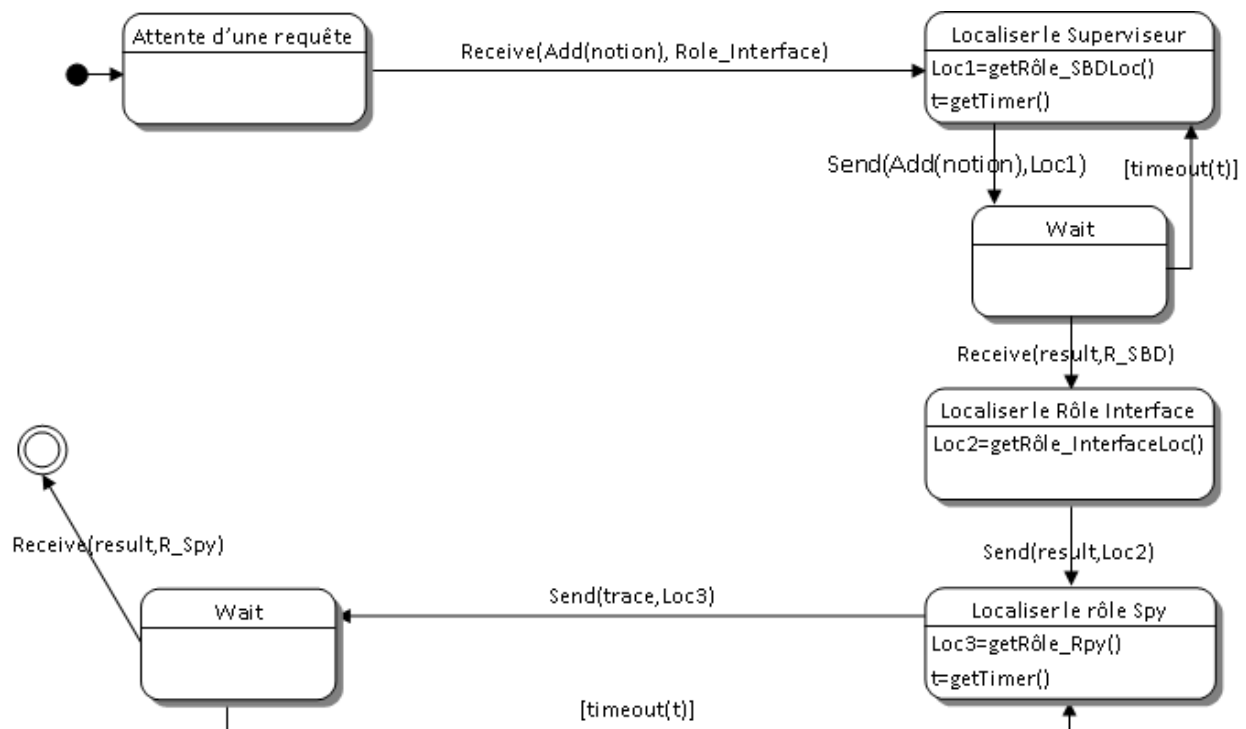


Figure 45. états et transitions du R_Assistant demandant l'ajout d'un point d'accès au R_SBD

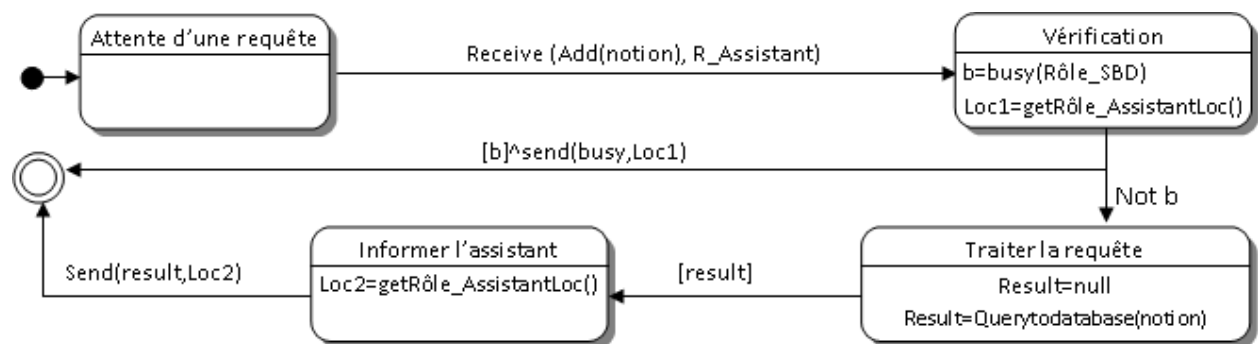


Figure 46. états et transitions du R_SBD sauvegardant un point d'accès dans la base de données

5.3.5.7 L'ajout d'une ressource dans l'espace privé ou l'espace du groupe

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande l'ajout d'une ressource et obtient ce service de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_SBD avec leurs tâches concurrentes respectivement, *Add_resource* et *Save_resource*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 47, celui de R_SBD est présenté sur la figure 48.

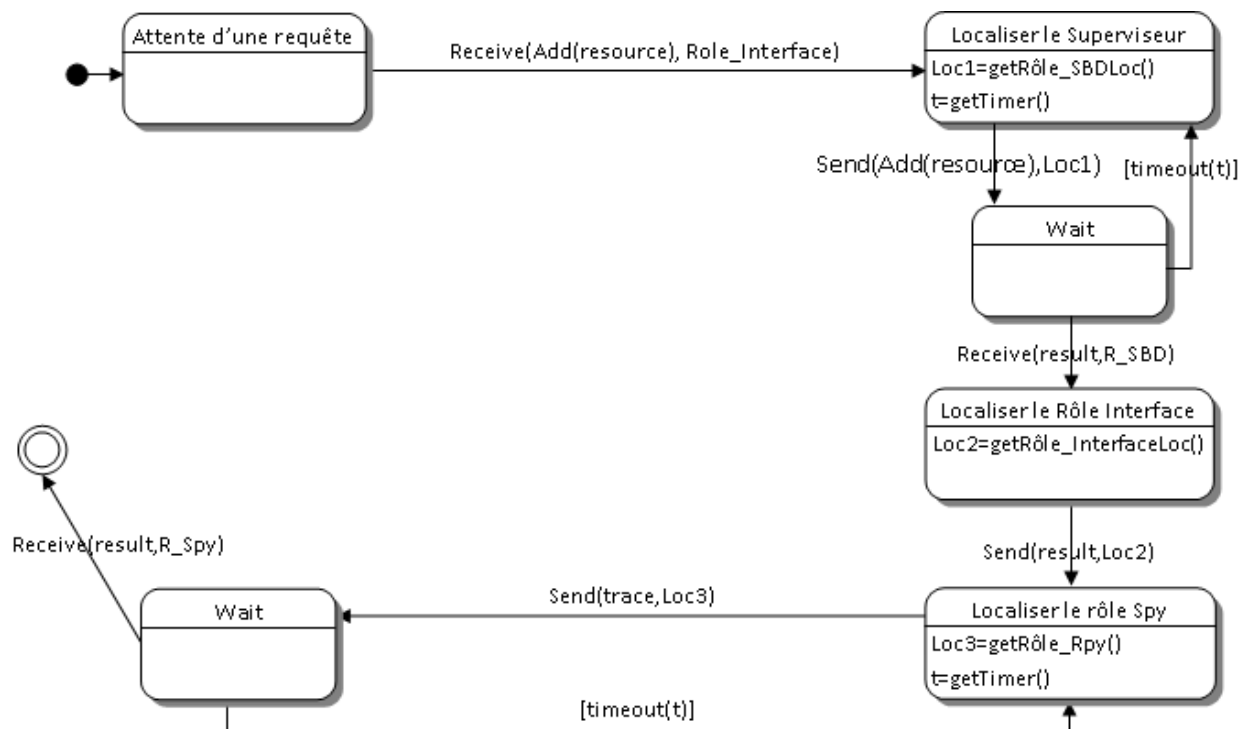


Figure 47. états et transitions du R_Assistant demandant l'ajout d'une ressource au R_SBD

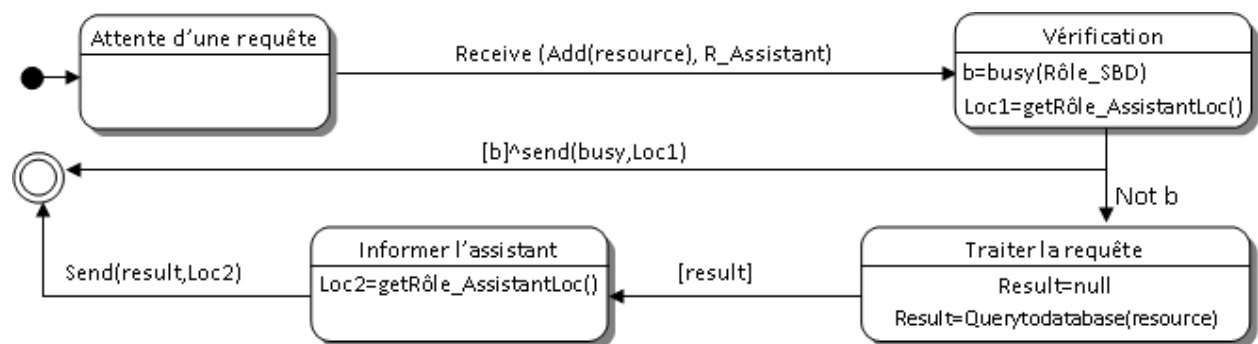


Figure 48. états et transitions du R_SBD sauvegardant une ressource dans la base de données

5.3.5.8 L'évaluation d'un exercice

Ce protocole de communication décrit la façon dont l'agent assistant demande et obtient l'évaluation d'un exercice de la part de l'agent Evaluator (R_Evaluator). Il met en œuvre le rôle R_Interface qui initie la demande, le rôle R_SBD qui satisfait les requêtes de la base de données ainsi que les rôles R_Assistant et R_Evaluator avec leurs tâches concurrentes respectivement, *Ask_for_evaluation* et *Evaluate*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 49, celui de R_Evaluator est présenté sur la figure 50.

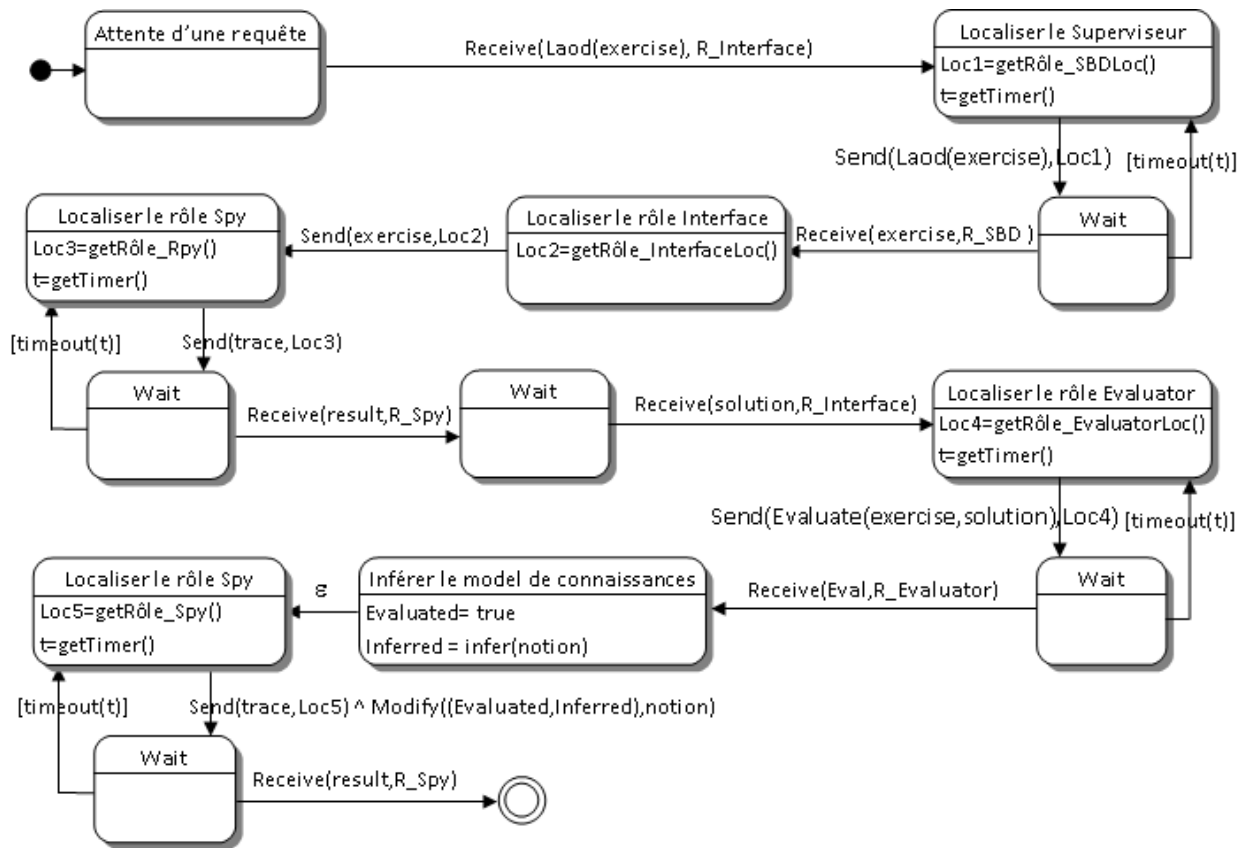


Figure 49. états et transitions du R_Assistant demandant l'évaluation d'un exercice

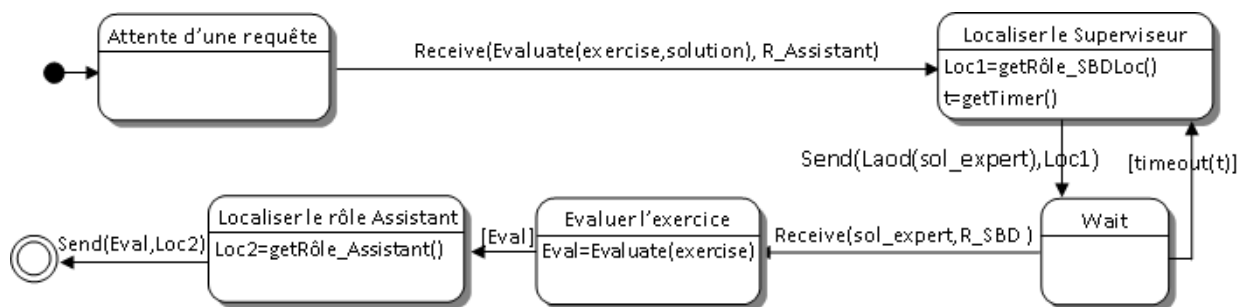


Figure 50. états et transitions du R_Evaluator évaluant un exercice

5.3.5.9 Le chargement de la liste des utilisateurs

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande et obtient la liste des apprenants de la part de l'agent superviseur de la base de données (R_SBD). Il met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_SBD avec leurs tâches concurrentes respectivement, *Load_learlist* et *send_learlist*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 51, celui de R_SBD est présenté sur la figure 52.

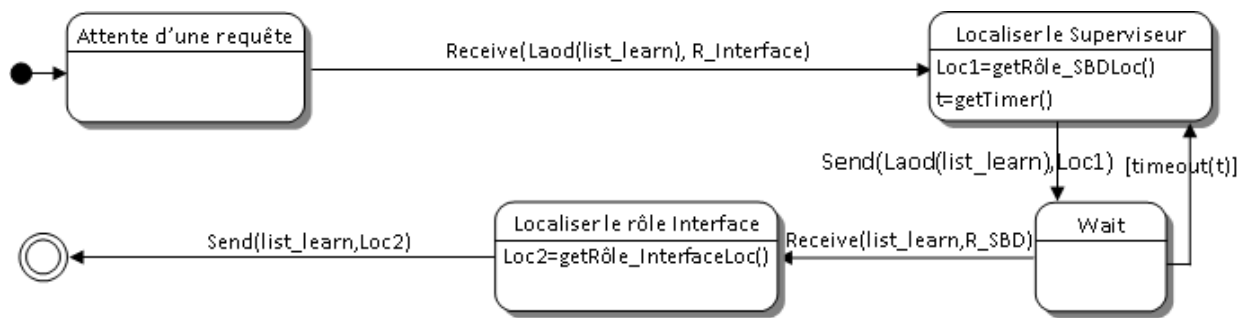


Figure 51. états et transitions du R_Assistant demandant la liste des utilisateurs

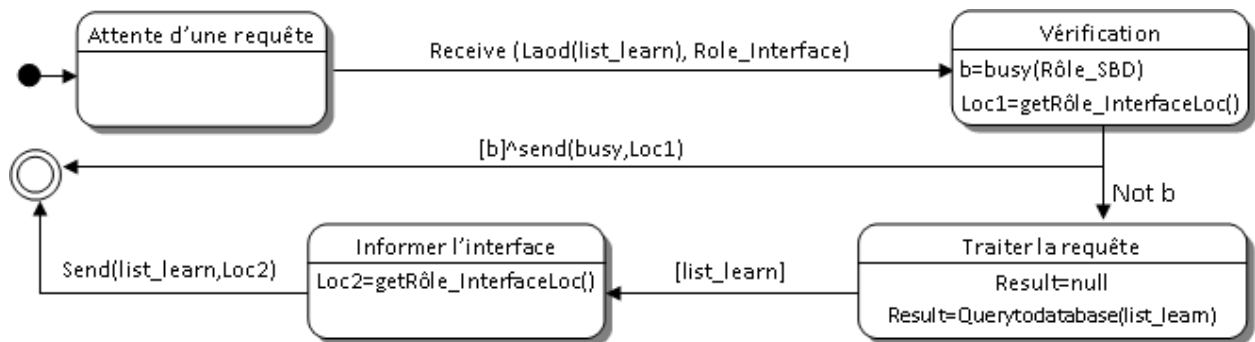


Figure 52. états et transitions du R_SBD envoyant la liste des utilisateurs au R_Assistant

5.3.5.10 L'envoi d'un message à un destinataire

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) demande de communiquer avec un autre membre de son groupe et obtient l'autorisation de communiquer de la part de l'agent collaborateur (R_Collaborator). Il met en œuvre le rôle R_Interface qui initie la demande ainsi que les rôles R_Assistant et R_Collaborator avec leurs tâches concurrentes respectivement, *Send_message* et

Authorize. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 53, celui de R_Collaborator est présenté sur la figure 54.

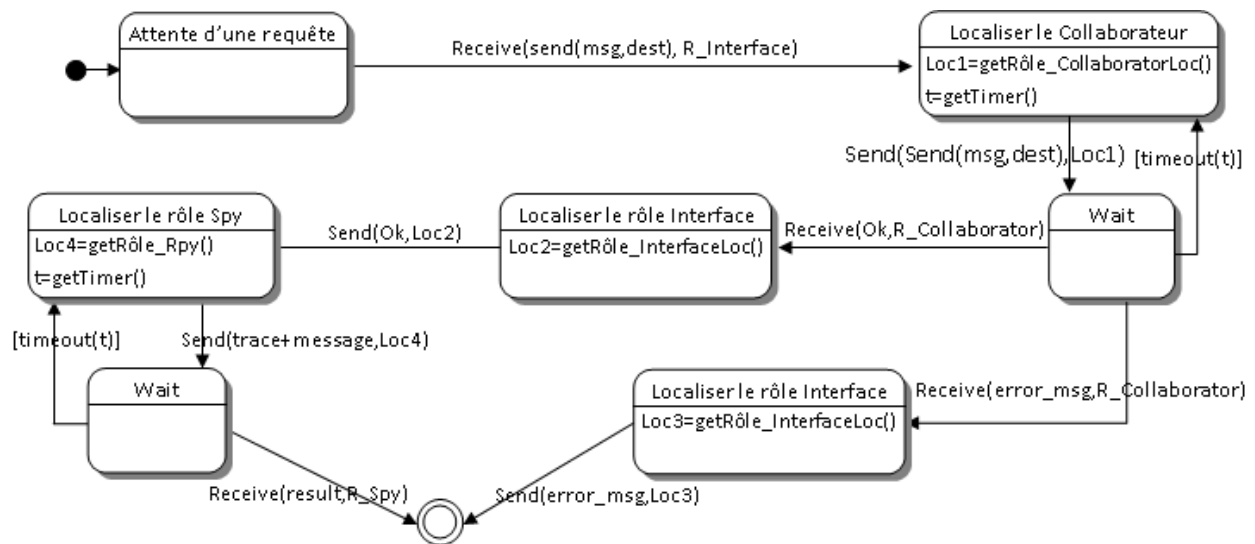


Figure 53. états et transitions du R_Assistant envoyant un message à son destinataire

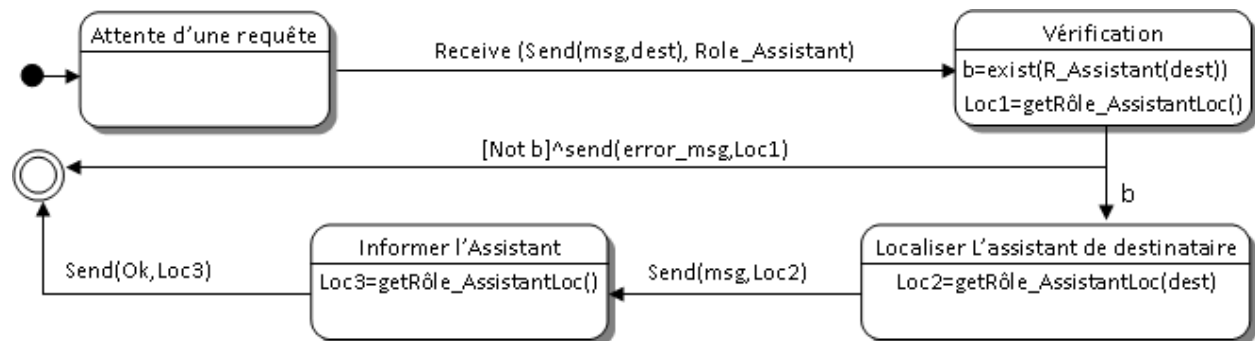


Figure 54. états et transitions du R_Collaborator transmettant le message au destinataire

5.3.5.11 La réception d'un message

Ce protocole de communication décrit la façon dont l'agent assistant (R_Assistant) reçoit un message emmenant d'un autre membre de son groupe. Il met en œuvre les rôles R_Assistant et R_Collaborator avec leurs tâches concurrentes respectivement, *Receive_message* et *Inform_assistant*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 55, celui de R_Collaborator est présenté sur la figure 56.

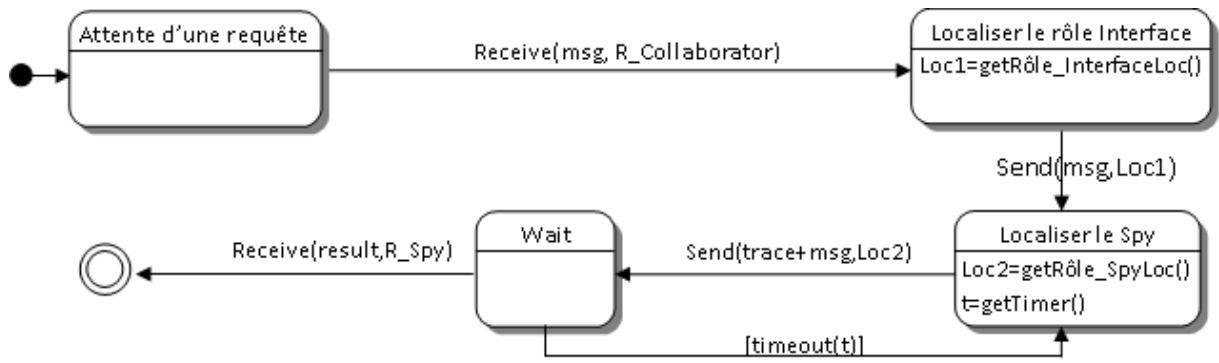


Figure 55. états et transitions du R_Assistant recevant un message

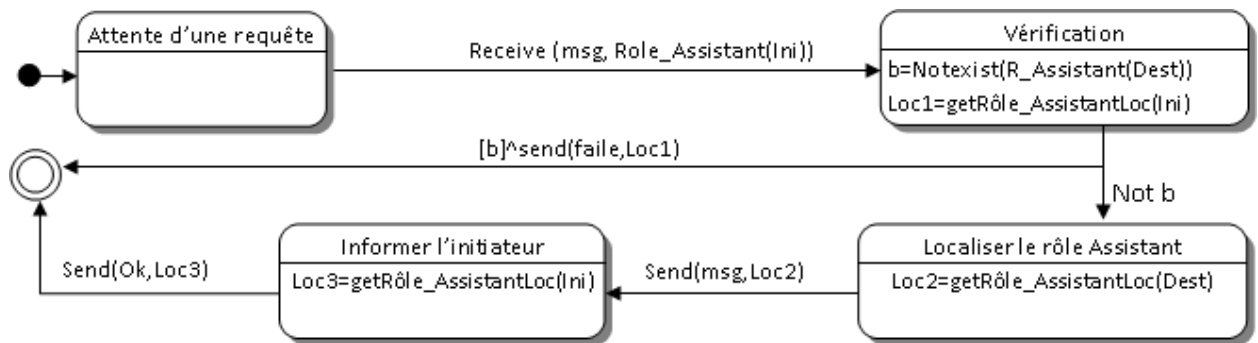


Figure 56. états et transitions du R_Collaborator transmettant un message

5.3.5.12 La déconnexion de l'utilisateur

Ce protocole de communication décrit la façon dont l'agent d'Assistant (R_Assistant) déconnecte du système. Il met en œuvre le rôle R_Interface qui initie la demande ainsi que le rôle R_assistant avec sa tâche concurrente *Disconnect*. L'automate à états finis représentant les états et les transitions de R_Assistant est présenté sur la figure 57.

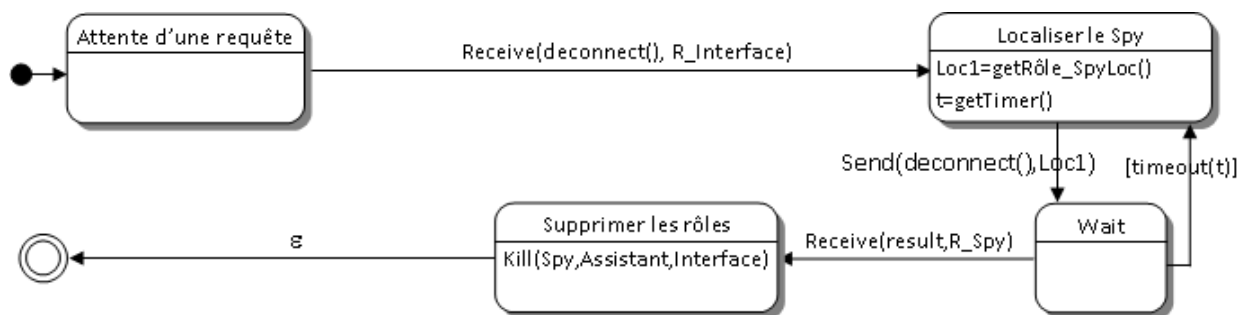


Figure 57. états et transitions de l'Interface émettant une demande de déconnexion d'un apprenant

5.4 La phase de conception

Le passage entre la phase d'analyse et la phase de conception se fait comme suit :

- Le concepteur crée les classes d'agents et assigne chaque rôle à au moins une classe d'agent.
- Les communications entre les tâches déterminent les conversations entre les agents.

5.4.1 Le diagramme des classes d'agents

En ce qui concerne notre environnement, nous avons transformé directement nos rôles en agents sauf pour les agents Assistant et Inter_DB qui remplissent ici plusieurs rôles, c'est-à-dire que pour les autres, nous avons affecté chacun des rôles de l'étape précédente à un agent différent. La méthodologie recommande de noter dans chaque rectangle représentant une classe d'agent les rôles qu'elle remplit.

Pour notre environnement d'apprentissage E-PACAD, nous avons donc sept classes d'agents:

L'agent Assistant : (qui remplit les rôles R_assistant, R_Collector et R_Spy) est l'agent de l'apprenant (un agent par apprenant). Il assiste son propriétaire en lui fournissant le contenu pédagogique et en sauvegardant les données concernant ses préférences et ses interactions.

L'agent Evaluator : (qui remplit le rôle R_evaluator) est l'agent qui évalue automatiquement les exercices de type QCM de l'apprenant.

L'agent collaborator : (qui remplit le rôle R_collaborator) est l'agent qui gère les communications entre les utilisateurs du système.

L'agent Analyser : (qui remplit le rôle R_Analyser) est l'agent chargé d'analyser les données d'interaction de l'apprenant.

L'agent Supervisor : (qui remplit le rôle R_DBS) est chargé de fournir les données de la base de données à tous les autres agents. Il assure l'interfaçage de la base de données avec tout le reste de l'environnement E-PACAD.

L'agent Inter_DB : (qui remplit les rôles R_DB et R_LM) encapsule l'interface du système avec la base de données représentant le contenu pédagogique et le modèle de l'apprenant en même temps.

L'agent Interface : (qui remplit le rôle R_interface) encapsule l'interface entre l'environnement et l'apprenant.

Le diagramme des classes d'agents de notre environnement E-PACAD est représenté sur la figure 58.

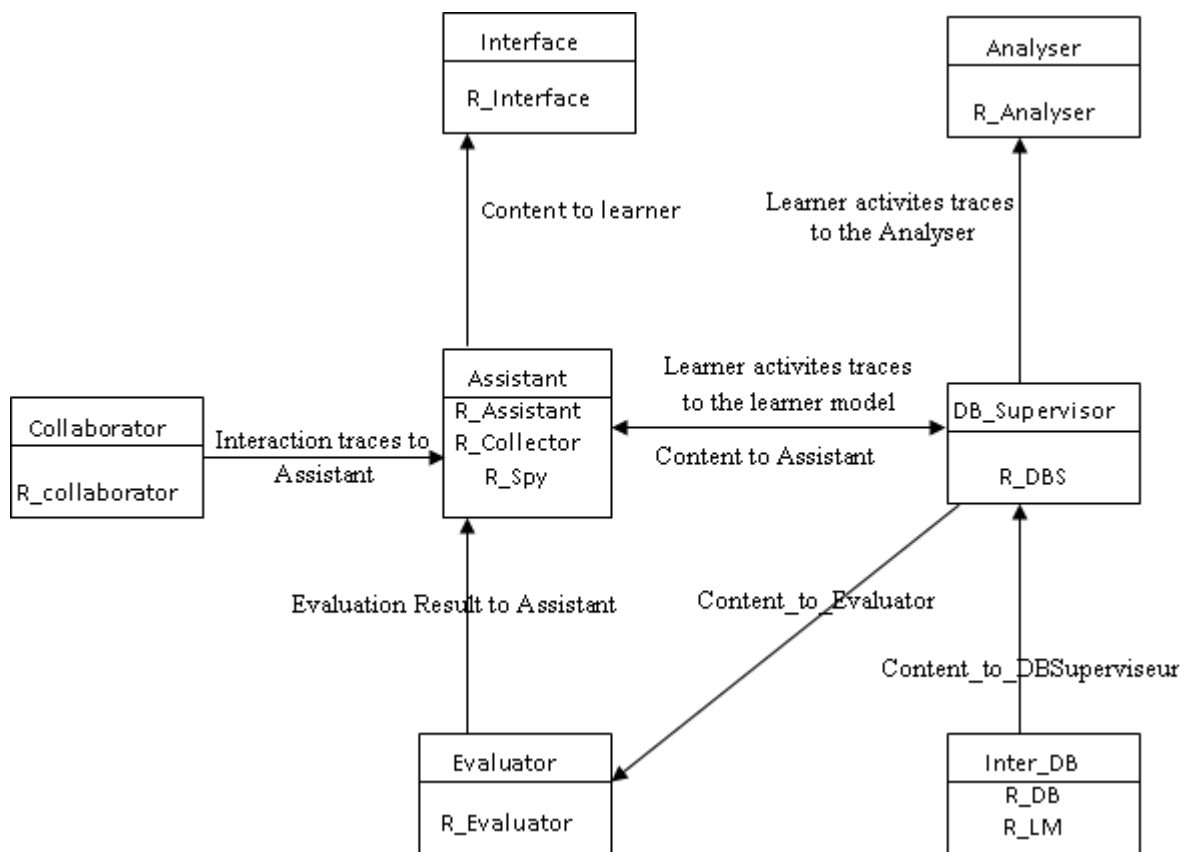


Figure 58. *Diagramme des classes d'agents*

5.4.2 La construction des communications inter-agents

Cette étape a pour but de construire le détail des interactions entre les agents. Ces interactions sont spécifiées grâce à des diagrammes d'automates à états finis selon le même principe que la spécification des tâches concurrentes de la troisième étape. Pour chaque protocole de communication entre agents, le concepteur doit définir deux diagrammes de communication de classes : l'un pour l'initiateur et l'autre pour le destinataire.

5.4.3 Implémentation des agents

La méthodologie de conception multi-agents MaSE ne définit et n'impose aucune architecture interne des agents mis en œuvre. Cette tâche est laissée au concepteur. Pour la construction de l'intérieur des agents (« Assembler les agents » de la phase de conception de MaSE), étant donné que nous avons implémenté nos agents dans la plateforme JADE, nous avons adopté l'architecture des agents de JADE comme architecture interne de nos agents.

5.4.4 Diagramme de déploiement

Cette dernière étape de MaSE nous permet d'instancier les classes d'agents construites en définissant les agents du système. Le résultat est un diagramme de déploiement qui montre le nombre, le type et la localisation des agents du système. La figure 59 montre le diagramme de déploiement de l'environnement E-PACAD. Nous avons au total 6 classes d'agents. Si on a au total n apprenants, nous aurons $2*n$ agents d'apprenants (Interface+Assistant). Le nombre d'agents total de E-PACAD est donc de $2*n+4$. Le nombre 4 correspond à l'agent collaborateur, l'agent évaluateur, l'agent analyseur, et l'agent superviseur de la base de données.

Les agents collaborateur, évaluateur, analyseur et superviseur de la base de données s'exécutent sur le serveur. Les autres agents s'exécutent sur les postes clients des apprenants.

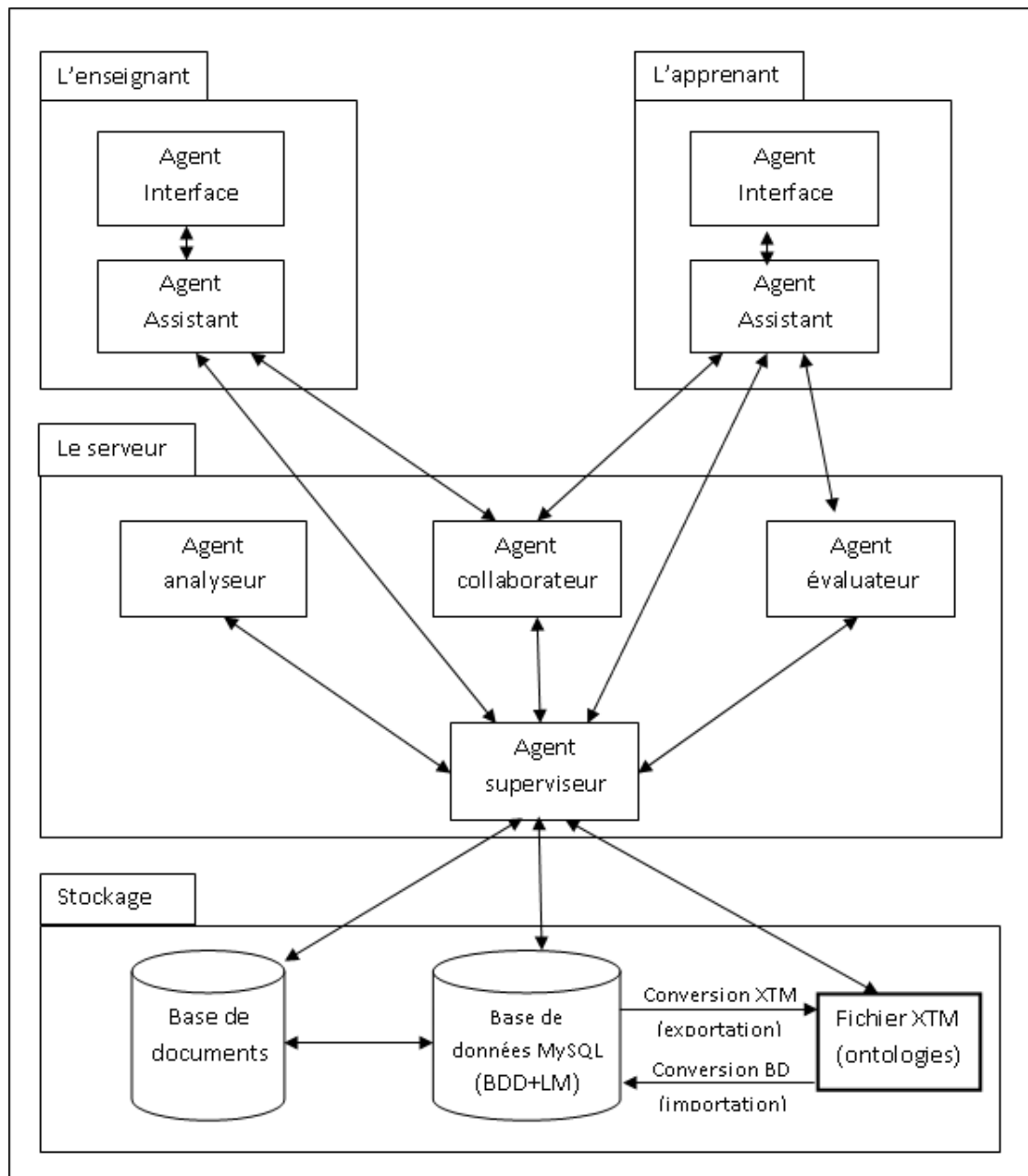


Figure 59. Diagramme de déploiement

5.5 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle vision dans la conception des environnements collaboratifs. Une conception qui prend en compte l'ensemble des éléments qui interviennent dans l'architecture de l'environnement. Nous avons eu recours au paradigme d'agent pour proposer et implémenter une architecture à base d'agents pour la modélisation des interactions de l'apprenant qui se trouve dans une situation d'apprentissage collaboratif à distance. Nous avons également recensé

l'importance de l'application d'une méthode pour développer un système multi-agents. Cependant, notre système a été analysé, conçu et spécifié suivant une méthodologie de développement du système multi-agents MaSE. La méthode MASE nous a énormément aidés à modéliser notre système grâce à la simplicité de ses concepts et l'utilisation de certains diagrammes d'UML. Partant d'un ensemble d'objectifs à réaliser, nous sommes arrivés à identifier plusieurs types d'agents :

- L'agent Interface : qui se charge de l'interface entre l'apprenant et l'environnement.
- L'agent Assistant : c'est un compagnon de l'utilisateur qui peut jouer plusieurs rôles. Il observe les actions de l'utilisateur, il scrute l'utilisateur et il lui facilite l'accès et l'usage de l'environnement. Pour réaliser ses tâches, l'assistant gère une structure permettant de garder toutes les informations concernant l'utilisateur plus particulièrement l'apprenant. Cette structure est appelée le modèle de l'utilisateur.
- L'agent Evalueur : il se charge de l'évaluation automatique des exercices du type QCM.
- L'agent collaborateur: il se charge de la communication entre l'apprenant et les autres acteurs de l'environnement (apprenants, enseignants, etc.) via les outils de communication et spécialement le chat.
- L'agent Analyse: il effectue une analyse des traces enregistrées dans le modèle de l'apprenant afin de fournir à l'agent demandeur l'état de l'apprenant.
- L'agent superviseur de la base de données : il assure l'interface de l'environnement avec la base de données qui contient le contenu pédagogique et le modèle de l'apprenant et aussi la base des documents pédagogiques.

L'ensemble de ces agents se coopère pour assurer la gestion du modèle de l'apprenant.

Dans le chapitre suivant, nous fournissons quelques résultats issus de l'application d'E-PACAD et nous illustrons l'utilisation concrète qui pourrait être faite de notre prototype au cours d'une session d'apprentissage collaboratif à distance.

Chapitre 6 : Mise en œuvre et expérimentation d'usage

6.1 Introduction

Afin de tester la fiabilité de notre approche, nous avons développé un environnement d'apprentissage collaboratif à base d'agents qui met l'accent sur la modélisation de l'utilisateur apprenant pour lui aider dans son apprentissage et lui faciliter l'accès au contenu de l'environnement.

Ce chapitre traite des aspects techniques liés à l'implémentation et la mise en œuvre de notre environnement. Nous y présenterons d'abord nos choix des technologies et outils adoptés pour le développement de l'environnement. Nous aborderons ensuite les principales fonctionnalités offertes par notre environnement aux usagers.

Les prises d'écrans et les explications assez approfondies sont à même de résumer le fonctionnement global de notre environnement E-PACAD.

6.2 Choix des outils de développement utilisés

6.2.1 Choix du Java comme langage de programmation

L'environnement de développement utilisé pour l'implémentation de notre application doit assurer la portabilité de l'application finale d'une plateforme à une autre sans être obligé de la recompiler.

Nous avons opté pour Java comme langage de développement, non seulement à cause de la contrainte liée à la portabilité, mais aussi pour ses nombreuses caractéristiques qui répondent aux exigences de notre application dont on peut citer :

- Java est un langage de programmation orienté objet. L'objet contient des données et du code permettant de les manipuler. Cela permet de développer des applications plus simples à gérer et faciles à maintenir et offre un moyen de développer des objets génériques réutilisables.
- Java est une plateforme d'exécution. Sa machine virtuelle Java permet de garantir sa portabilité. C'est un programme appartenant à la plateforme d'exécution qui interprète le code Java et le convertit en code natif qui sera ensuite exécuté par le système d'exploitation.
- Java est le langage le plus utilisé dans le domaine du développement des services. De plus, la plupart des API disponibles sur Internet sont des API Java.
- Java est un langage multithread. Il exécute en apparence simultanée plusieurs processus. Cela se réalise en accordant de façon séquentielle un peu du temps processeur à chaque processus.
- Java est un langage sécurisé. Ses programmes peuvent être protégés contre l'introduction des virus et la destruction des données. Les informations transmises à travers le réseau peuvent être cryptées et décryptées par Java.

6.2.2 Choix de JADE comme plateforme de développement

Notons que la meilleure façon pour construire un système multi-agents (SMA) est d'utiliser une plate-forme multi-agents. Cette plateforme englobe un ensemble d'outils nécessaire à la construction et à la mise en service d'agents au sein d'un environnement spécifique. Nous avons choisi la plate-forme JADE (*Java Agent Development Framework*) pour implémenter notre environnement.

JADE est une plate-forme créée par le laboratoire TILAB et décrite par (Bellifemine et al., 2000). JADE permet de construire des systèmes multi-agents (SMA). Elle est entièrement implémentée en JAVA, et répond aux spécifications FIPA (Foundation for Intelligent Physical Agents) pour l'interopérabilité des systèmes multi-agents. JADE fournit un grand nombre de classes qui implémente le comportement des agents qu'elle crée et possède trois modules principaux:

- Agent Management System (AMS) : cet agent est responsable de contrôler l'accès à la plateforme et aussi de l'authentification et de l'enregistrement des agents participants.
- Directory Facilitator (DF) : cet agent fournit un service de « pages jaunes» à la plate-forme. A partir de ces pages jaunes, un agent peut trouver d'autres agents pour lui fournir les services dont il a besoin afin de réaliser ses buts.
- Agent Communication Channel (ACC) : cet agent gère la communication entre les agents.

JADE est la plate-forme qui se rapproche plus de nos critères. Le concept d'agent est vu par JADE comme un processus autonome et indépendant qui a une identité, qui requiert la communication (collaboration, compétition, etc.) avec les autres agents dans le but de remplir ses missions. Les agents communiquent à travers des messages représentés en FIPA-ACL. De plus JADE est dotée d'une interface graphique utilisateur permettant de gérer et contrôler la société d'agents du système (figure 60).

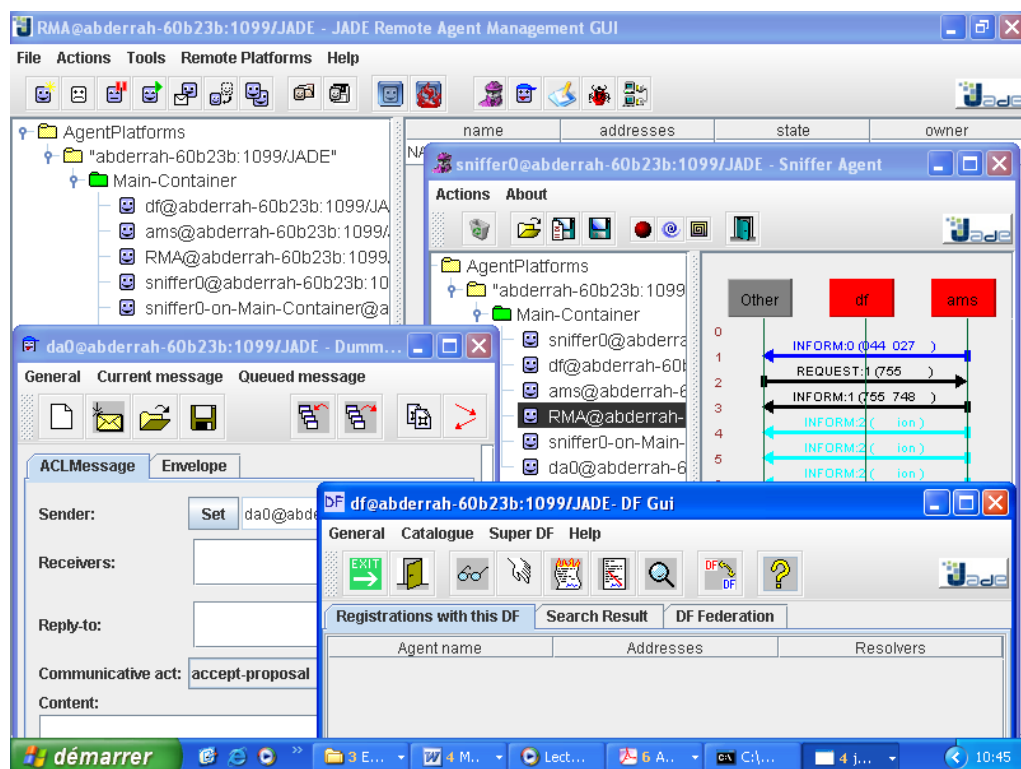


Figure 60. Interfaces graphiques de JADE

6.2.3 Les outils web

La simplicité de la manipulation de l'interface d'accès est considérée comme l'un des critères décisifs pour le succès d'un environnement, c'est la raison pour laquelle le choix des outils de programmation et surtout les outils du développement web doit être bien étudié.

6.2.3.1 *Les servlets*

L'une des étapes primordiales de notre environnement d'apprentissage développé est de pouvoir échanger des données avec le serveur. L'application qui va être installée sur le serveur doit être en mesure de récupérer ces données, et le plus important c'est de pouvoir récupérer simultanément des données de plusieurs utilisateurs.

L'une des solutions les plus efficaces qui répond à ces conditions est les servlets². Ce sont des programmes Java qui s'intègrent dans un serveur web pour fournir le traitement côté serveur des requêtes issues d'un navigateur client. Les servlets sont des programmes multithreads permettant de récupérer des données de plusieurs clients en même temps, cela est tiré du fait que le servlet crée une instance ou un thread à chaque requête du client pour servir ce client.

La technologie des servlets a pour avantage par rapport à d'autres, la vitesse d'exécution, cela est expliqué par le fait qu'un servlet est chargé une seule fois en mémoire et s'exécute depuis cette mémoire après son chargement initial.

Les servlets s'exécutent dynamiquement sur le serveur Web, permettant ainsi l'extension des fonctions de ce dernier, typiquement : accès à des bases de données, transactions, communication avec d'autres serveurs, etc.

6.2.3.2 *Les Java Server Pages (JSP)*

L'interface utilisateur de notre environnement d'apprentissage est composée de plusieurs pages web qui vont être générées dynamiquement selon les requêtes de

² <http://www.java2s.com>, Java Source and Support,

l'utilisateur. Pour la conception de notre interface, nous avons eu recours aux *Java Server Pages* (JSP). C'est une technologie basée sur Java qui permet aux développeurs de générer dynamiquement du code HTML, XML ou tout autre type de page Web.

Les JSP ont pour avantages par rapport à d'autres scripts, qui s'exécutent du côté serveur, le temps du traitement qui est moins important, cela est dû au fait que ces scripts sont compilés une seule fois sans être obligés de les recompiler à chaque accès.

Les JSP ont le même principe de fonctionnement des servlets, mais ils sont différents dans le fait qu'on peut les intégrer dans des pages HTML et de définir des liens entre eux.

6.2.4 Les serveurs

Les JSP et les servlets nécessitent un serveur pour fonctionner nommé souvent moteur de servlets ou moteur de JSP. Nous avons opté pour le serveur Tomcat. C'est un serveur Open Source qui agit comme un conteneur de servlets. Il fait partie du projet Jakarta, au sein de la fondation Apache.

6.2.5 Choix de MySQL comme un système de gestion de base de données

MySQL est un système de gestion de bases de données (SGBD) relationnelles SQL.

Nous avons opté pour MySQL comme SGBD pour l'implémentation de la base de données de notre environnement E-PACAD pour les raisons suivantes :

- C'est un SGBD libre, gratuit, très utilisé et très populaire.
- MySQL est fonctionnel sur des systèmes d'exploitation différents (UNIX, Windows, MacOS X).
- MySQL est très rapide, multithread et multiutilisateur permettant ainsi de gérer simultanément plusieurs transactions.
- MySQL dispose d'une vaste bibliothèque de fonctions et d'API. Et aussi d'une haute capacité de stockage (peut supporter des dizaines de millions d'enregistrements).

6.3 L'environnement E-PACAD

Notre environnement partagé E-PACAD permet à plusieurs utilisateurs d'interagir simultanément avec l'environnement pour accéder aux notions d'un cours donné et aux ressources indexées par ces notions. Cet environnement a pour objectifs d'aider l'apprenant à apprendre les notions de cours et de faciliter les échanges et les transferts des connaissances au sein d'une communauté d'apprenants.

6.4 Didacticiel E-PACAD

Nous présentons dans ce qui suit, une vue de l'application à travers des interfaces utilisateurs qui ont été réalisées.

6.4.1 Accès à la formation recherchée

La page suivante sera affichée au démarrage de notre site. Elle permet, à partir de cet écran aux utilisateurs (apprenant, enseignant, administrateur), de choisir la formation qui leurs intéresse (partie entourée de rouge).

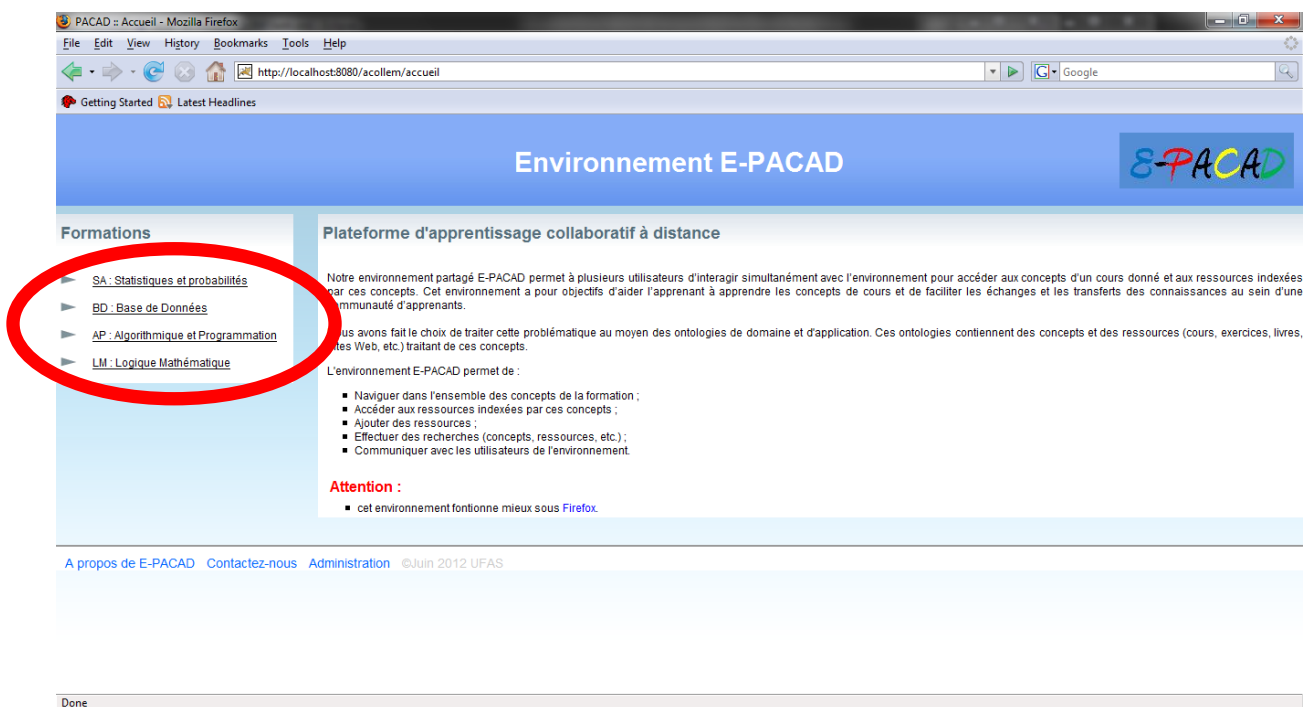


Figure 61. L'accès à la formation

Pour la suite, nous prendrons nos exemples dans la formation de l’algorithmique et programmation.

6.4.2 La connexion

Une fois la formation choisie, nous arrivons à cet écran :

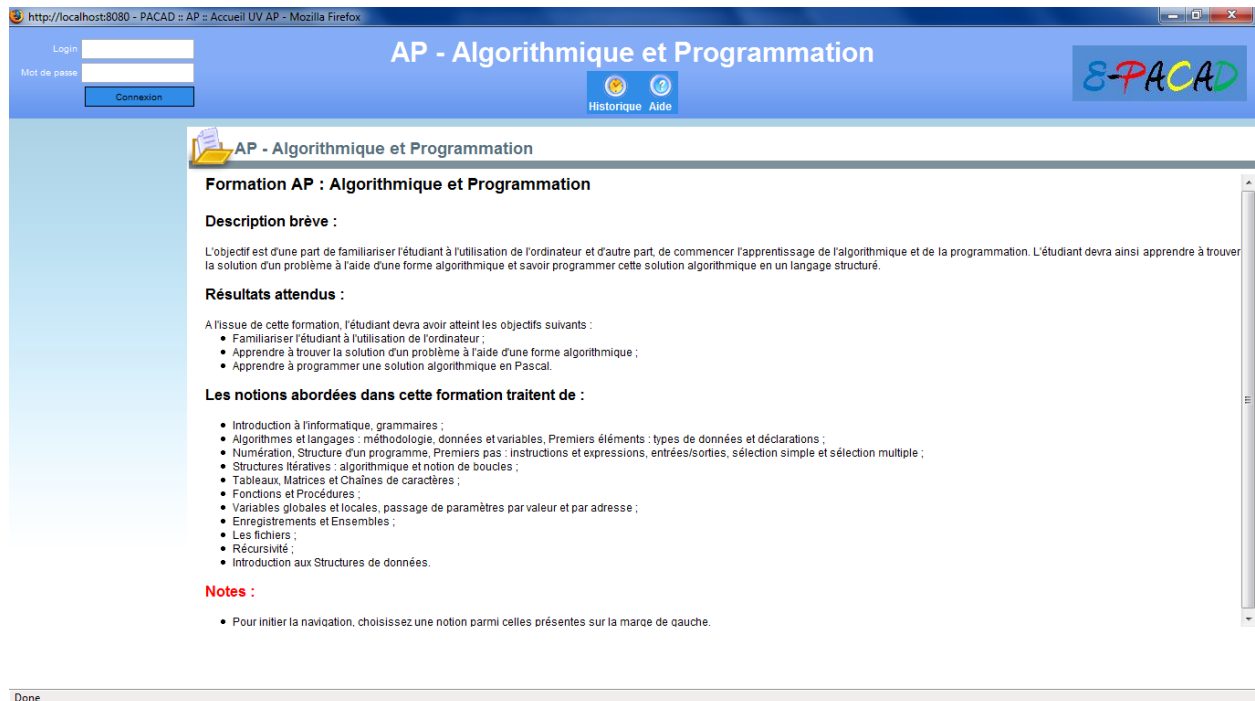


Figure 62. *La connexion des utilisateurs*

Ce dernier permet aux utilisateurs de connecter à l’environnement grâce à un login et un mot de passe.

6.4.3 Le lancement des agents de l’environnement

L’agent RMA de la plateforme JADE permet la création de l’agent et aussi sa manipulation d’une manière plus commode.

Une application Java peut être répartie sur plusieurs hôtes. Une seule machine virtuelle Java est exécutée sur chaque hôte. Chaque machine virtuelle Java est un conteneur d’agents permettant l’exécution de plusieurs agents concurrents sur le même hôte.

La figure suivante montre l’exécution des agents de notre environnement sur les différents conteneurs.

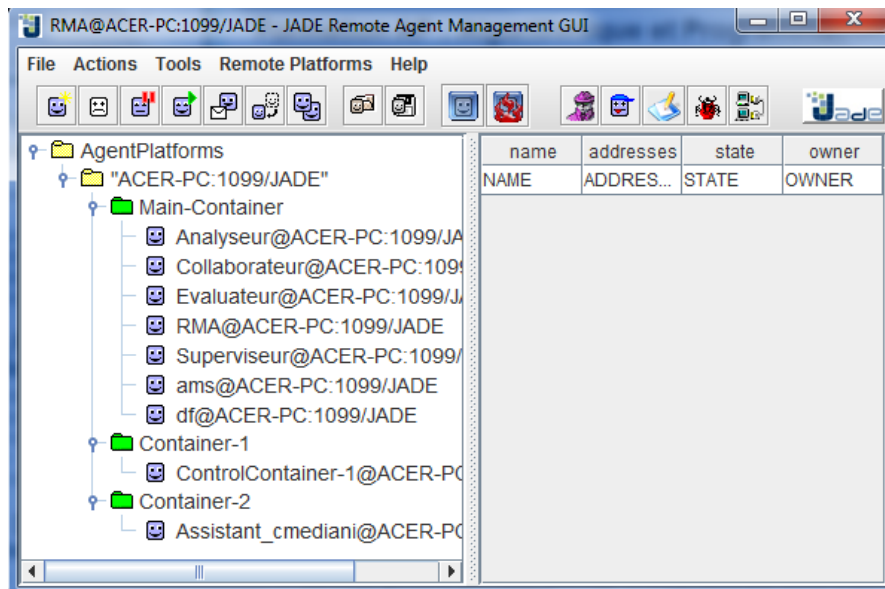


Figure 63. L'agent RMA de la plate-forme JADE

6.4.4 L'interface de l'environnement E-PACAD

Une fois qu'un utilisateur de type apprenant a saisi correctement ses coordonnées, la page suivante s'affiche, elle contient une liste des espaces de travail située sur le coin de la partie gauche de l'écran et permettant d'initialiser la navigation au sein de l'espace de travail choisi, ainsi qu'une description de l'unité de formation choisie située sur la partie droite de l'écran.

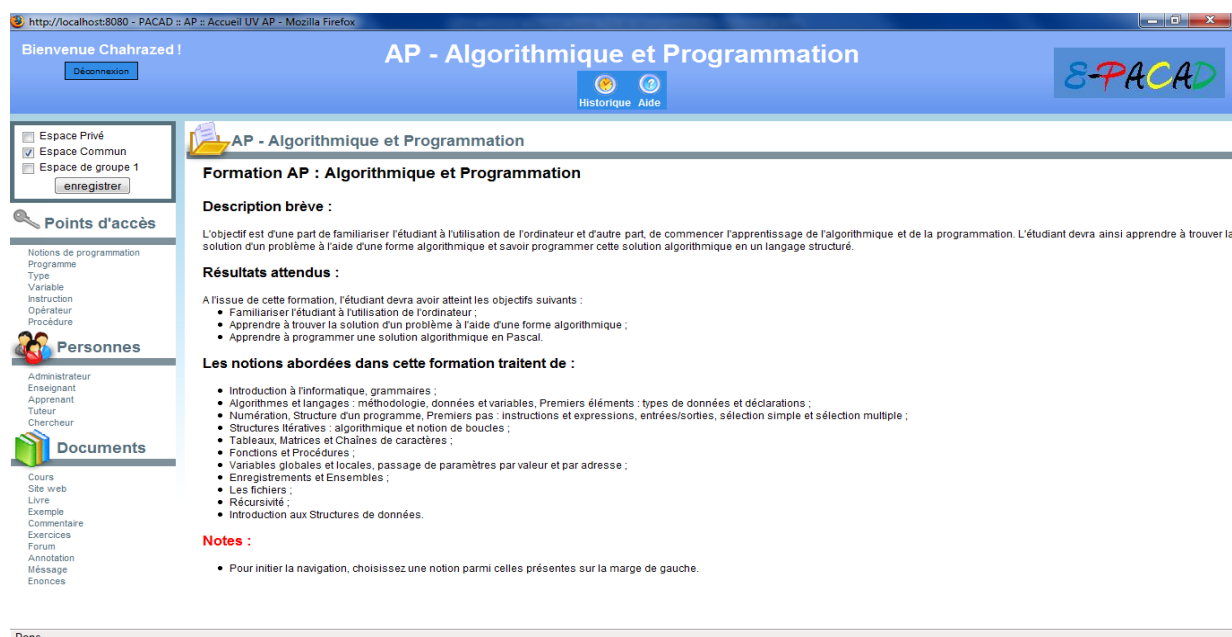


Figure 64. Le choix d'un espace de travail

6.4.5 L'accès aux différents espaces de l'environnement

L'environnement permet à chaque utilisateur de choisir d'accéder à différents espaces : que ce soit l'espace commun destiné à la formation, l'espace privé, ou l'espace des groupes auxquels il appartient.

- L'espace de formation : c'est l'espace commun qui sert à représenter la vue des enseignants sur la formation. Ce n'est que le responsable de la formation qui a le droit de modifier cet espace.
- L'espace privé : c'est l'espace où chaque utilisateur peut mettre ses propres ressources et aussi ses points d'accès qui lui aident dans son apprentissage. Le contenu de cet espace n'est accessible que par cet utilisateur.
- Les espaces du groupe : cet espace n'est accessible que par les membres du groupe. Et dans lequel, ils partagent et échangent des ressources et des points d'accès communs.

6.4.6 Le détail des pages de l'environnement

Les pages de notre environnement sont généralement composées des parties suivantes :

- La barre de menu permettant à l'utilisateur d'accéder à différentes fonctions comme : Consulter, Rechercher, Historique, Aide, etc.
- Une liste de points d'accès située en partie gauche de l'écran et permettant d'initialiser la navigation à partir d'un concept choisi au sein de l'ontologie d'application. Un point d'accès est un concept de l'ontologie.
- Une liste de ressources liées au concept en cours qui est située en bas de l'écran. Ces ressources sont organisées selon leur type (cours, exercices, livres, sites web, forum, chat, etc.) et permettent de présenter la notion courante du cours. L'apprenant peut choisir de consulter une ressource en cliquant la-dessus.
- L'historique des interactions de l'apprenant : il permet de visualiser le parcours réalisé par l'apprenant jusque-là. Cet historique est situé en partie droite de l'écran.

- Une partie de l'ontologie présentée au milieu de l'écran en circonscrivant la notion courante pointée par l'utilisateur.

6.4.7 L'accès aux concepts de la formation

6.4.7.1 Les points d'accès

Sous les cases à cocher permettant l'accès aux différents espaces de travail se trouvent les «Points d'accès». Il existe deux types de point d'accès :

- Ceux de la formation qui sont des concepts définis par le responsable de l'unité de formation pour permettre d'initialiser la navigation au sein de l'ontologie d'application. Ces concepts sont jugés importants et c'est pour cela qu'ils sont donc mis en avant.
- Ceux qu'on a ajoutés dans les espaces individuels ou les espaces du groupe permettant, par exemple, dans un espace individuel, de pointer un concept non compris ou qu'on doit approfondir, ou dans un espace du groupe, d'informer les autres membres du groupe que ce concept est important et qu'on doit l'acquérir.

Ces points d'accès permettent donc l'initialisation de la navigation au sein de l'arborescence de l'ontologie d'application. Un clic sur l'un d'eux permet de le mettre au centre de la vue arborescente, par exemple, un clic sur le concept « variable » :

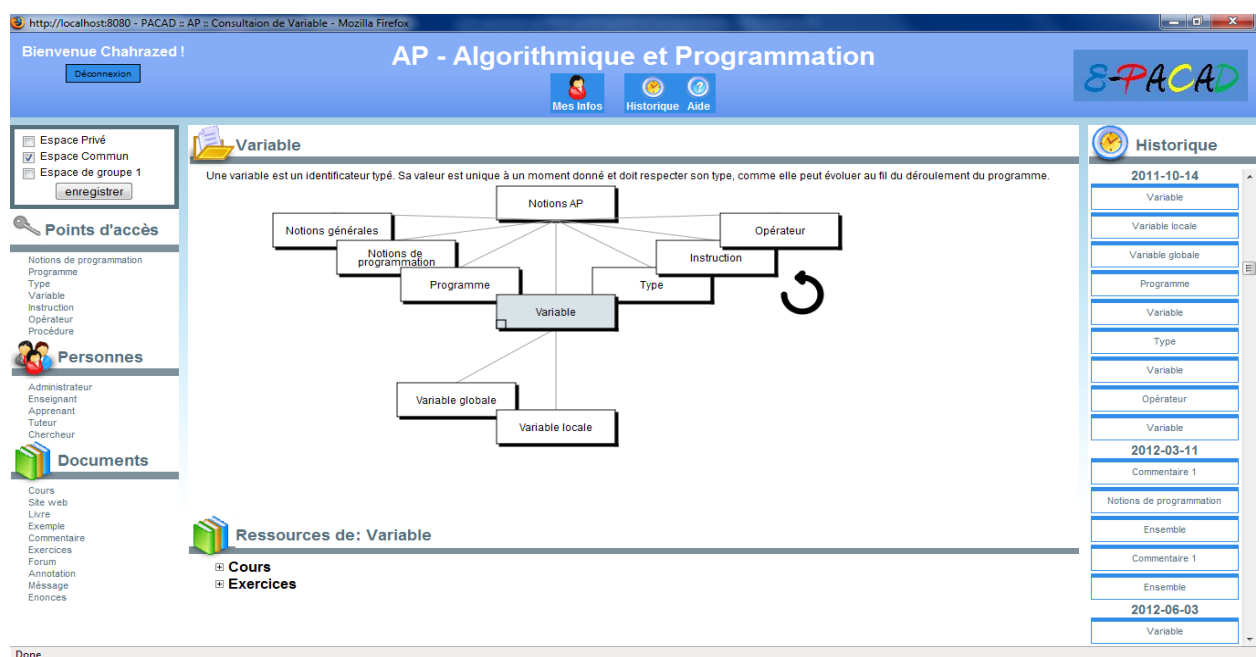


Figure 65. La navigation au sein de l'ontologie d'application

Sur cet écran, on est dans l'espace commun de la formation. La vue est centrée sur le concept «variable». En bas se trouvent les différentes ressources indexées par ce concept et rangées par type (cours, exercice, livre, ...). Pour visualiser ou non les ressources d'un certain type, on doit cliquer sur le + ou le - se trouvant à gauche du nom du type.

6.4.7.2 Navigation dans l'environnement

L'environnement permet de naviguer dans la hiérarchie de concept pour accéder aux concepts de l'application et aux ressources indexées par ces concepts. Chaque concept pouvant être spécialisé en plusieurs autres. La figure suivante présente un zoom sur la partie navigation dans l'environnement :

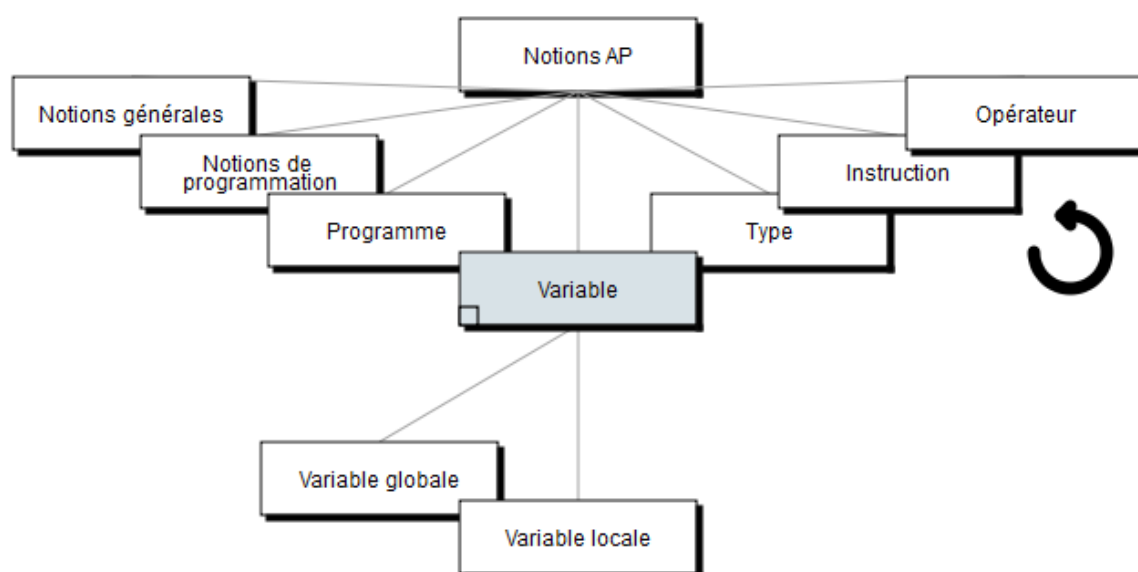


Figure 66. Visualisation d'une partie de l'ontologie d'application (AP)

Dans cet exemple, la vue est centrée sur le concept «variable». Par exemple, le lien entre «variable» et «variable globale» signifie : « variable globale » est un sous concept ou bien une spécialisation de « variable ». Dans cette vue, on peut accéder au «père» du concept courant, à ces « frères » et à ces « fils ». Le père est un concept qui représente une notion plus générique, les fils sont des sous concepts plus spécialisés et les frères sont des concepts ayant le même père que le concept central.

Pour placer un concept au centre de la vue, il faut cliquer là-dessus.

6.4.8 Les ressources indexées

6.4.8.1 Accès aux ressources

Les concepts permettent d'indexer un ensemble de ressources se trouvant rangées par type dans le cadre inférieur de l'écran :

The screenshot displays the PACAD web application interface. The top header includes the title 'AP - Algorithmique et Programmation' and the PACAD logo. The left sidebar contains navigation links for 'Espace Privé', 'Espace Commun', 'Espace de groupe 1', 'Points d'accès', 'Personnes', and 'Documents'. The main content area is titled 'Variable' and contains a diagram illustrating the concept of a variable. The diagram shows 'Notions AP' at the top, branching into 'Notions générales', 'Notions de programmation', 'Programme', 'Type', 'Opérateur', and 'Instruction'. 'Notions de programmation' further branches into 'Variable', which then branches into 'Variable globale' and 'Variable locale'. A red circle highlights the 'Ressources de: Variable' section at the bottom, which lists 'Cours' and 'Exercices'. The right sidebar shows a 'Historique' section with a list of recent activities.

Figure 67. Les ressources indexées par le concept « variable »

Un clic sur la ressource permet d'ouvrir une fenêtre contenant plus d'informations sur la ressource :

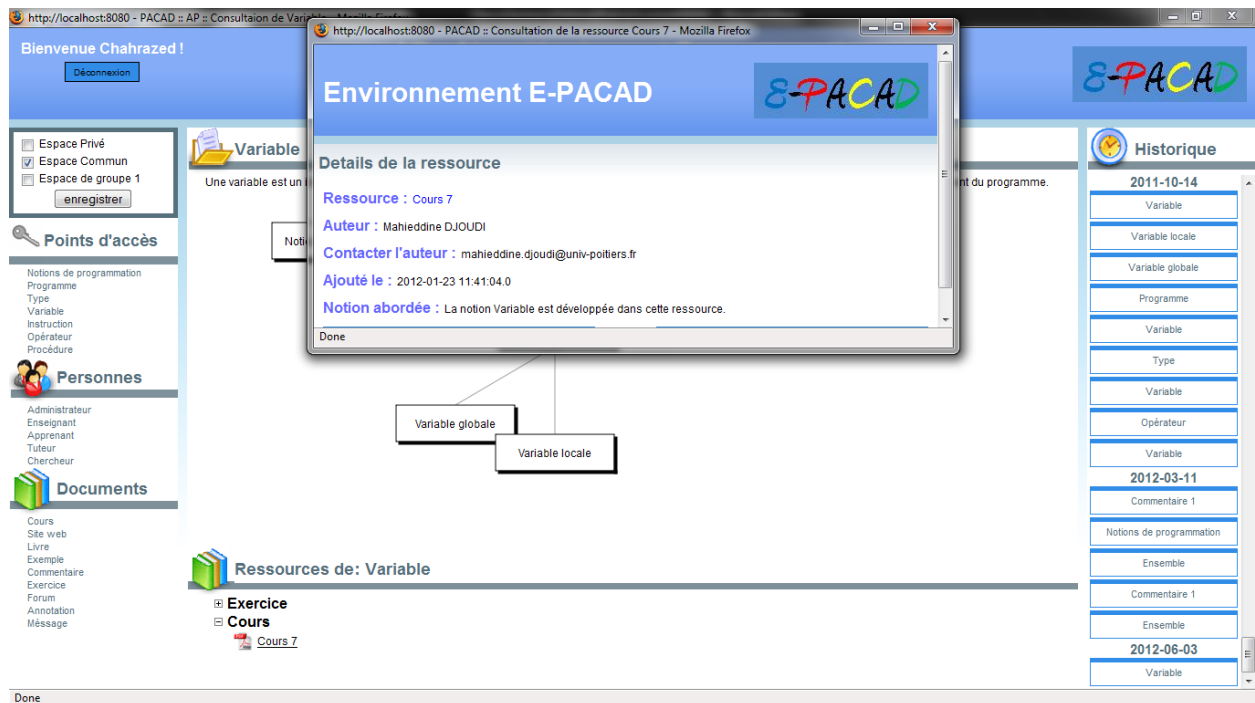


Figure 68. *L'accès à une ressource*

Cette fenêtre permet de télécharger la ressource s'il s'agit d'un fichier enregistré sur le serveur, ou d'accéder à un site ou une ressource de communication s'il s'agit d'une ressource internet.

6.4.8.2 Ajout d'une ressource

L'ajout d'une ressource par l'apprenant ne peut s'effectuer que dans l'espace individuel ou l'espace du groupe. Une fois ce choix fait, on place le concept sur lequel on veut ajouter la ressource au centre de la vue arborescente et on clic sur le bouton ajout de ressource pour un concept (entouré sur la figure ci-dessous) :

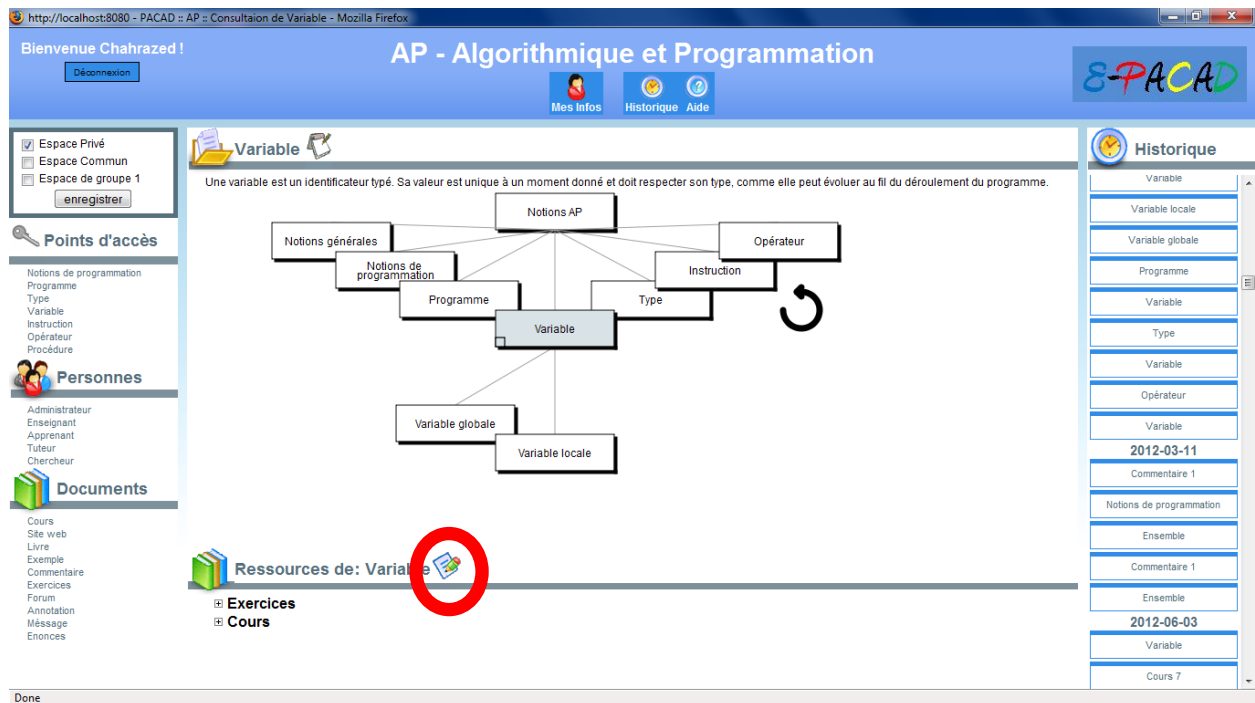


Figure 69. L'ajout d'une ressource

On peut alors remplir les différents champs qui se trouvent dans la page d'ajout d'une ressource indexée par le concept courant. Après la validation, la ressource apparaîtra dans l'espace où l'on se trouve.

6.4.9 Les ressources de différents espaces

Nous présentons dans les trois figures suivantes les ressources indexées par la notion « variable » correspondant respectivement aux trois espaces : l'espace commun, l'espace privé et l'espace du groupe 1 à lequel appartient l'étudiant choisi. Dans l'espace commun de la formation, un seul cours nommé "cours 2" est indexé par la notion « variable ». Par contre dans l'espace privé, deux ressources de type cours sont indexées par la même notion, il s'agit du cours « cours 2" et aussi du cours « cours 3 ». Ce dernier cours a été ajouté par l'apprenant dans son espace privé et n'est visible que dans cet espace.

Dans l'espace du groupe 1, un site Web ainsi qu'une ressource de type forum ont été ajoutés par les membres du groupe dans cet espace. Ces deux ressources ne sont visibles que par les membres de ce groupe.



Figure 70. *Visualisation des ressources de l'espace commun.*



Figure 71. *Visualisation des ressources de l'espace privé.*



Figure 72. *Visualisation des ressources de l'espace du groupe 1.*

6.4.10 L'historique

Notre environnement permet de sauvegarder les historiques de navigation pour que des observateurs puissent avoir pour consigne de noter tout ce que faisaient les étudiants sans jamais intervenir. L'historique du parcours réalisé permet à l'apprenant de connaître ce que fut son parcours jusque-là. Après chaque action de l'apprenant, l'historique de navigation est mis à jour.

6.4.11 Le modèle cognitif de l'apprenant

Notre environnement est doté d'un modèle cognitif de type expertise partielle représenté par un réseau bayésien qui permet d'associer, à chaque concept du domaine, un ensemble de valeurs mesurant le niveau de connaissance de l'apprenant dans ce concept. Un clic sur le coin inférieur gauche du concept courant permet de visualiser ces valeurs :

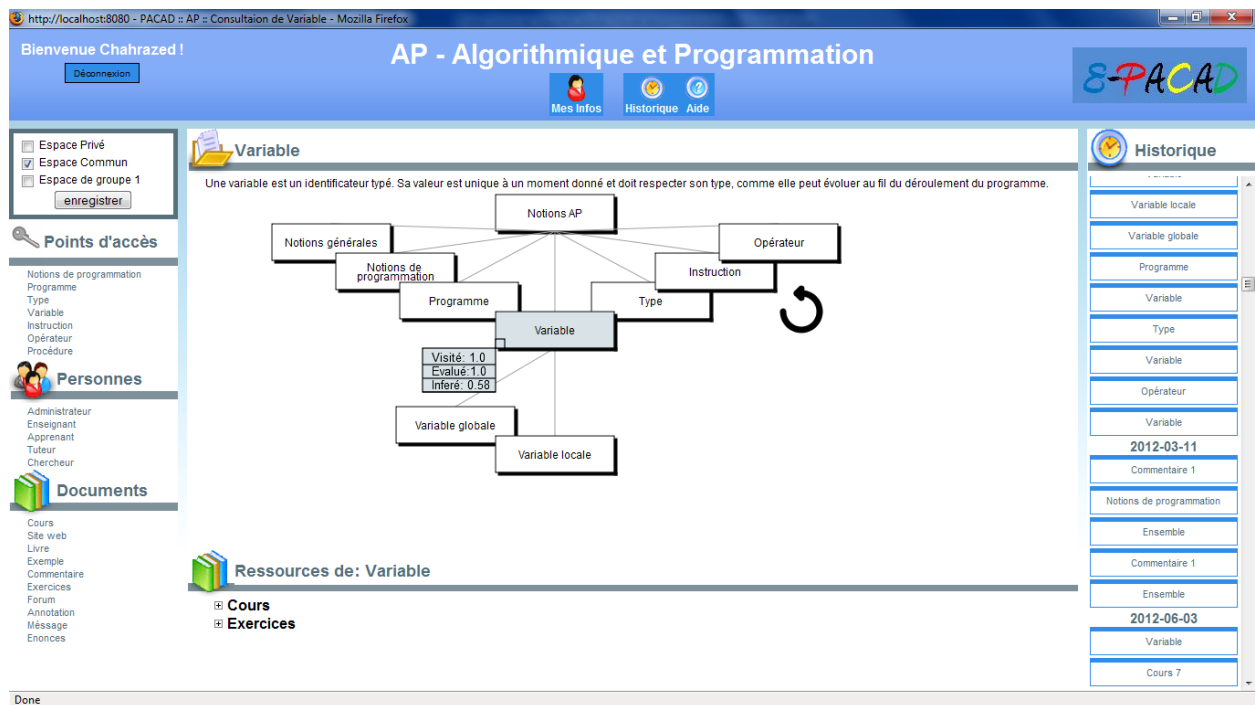


Figure 73. Le modèle cognitif de l'apprenant

Dans cet exemple, l'apprenant a consulté le concept « variable ». Il a obtenu 1 comme note d'évaluation dans les ressources correspondant à ce concept. La valeur d'inférence calculée en fonction de ses connaissances est de 0.58.

6.4.12 La communication entre les agents

Le paradigme de communication adopté par JADE est l'envoi des messages asynchrones. Chaque agent a une boîte aux lettres contenant les messages reçus par l'agent. Les messages reçus sur cette boîte sont notifiés afin d'être traités par l'agent et ils sont du type FIPA ACL. La figure suivante montre l'échange de messages entre quelques agents de notre environnement.

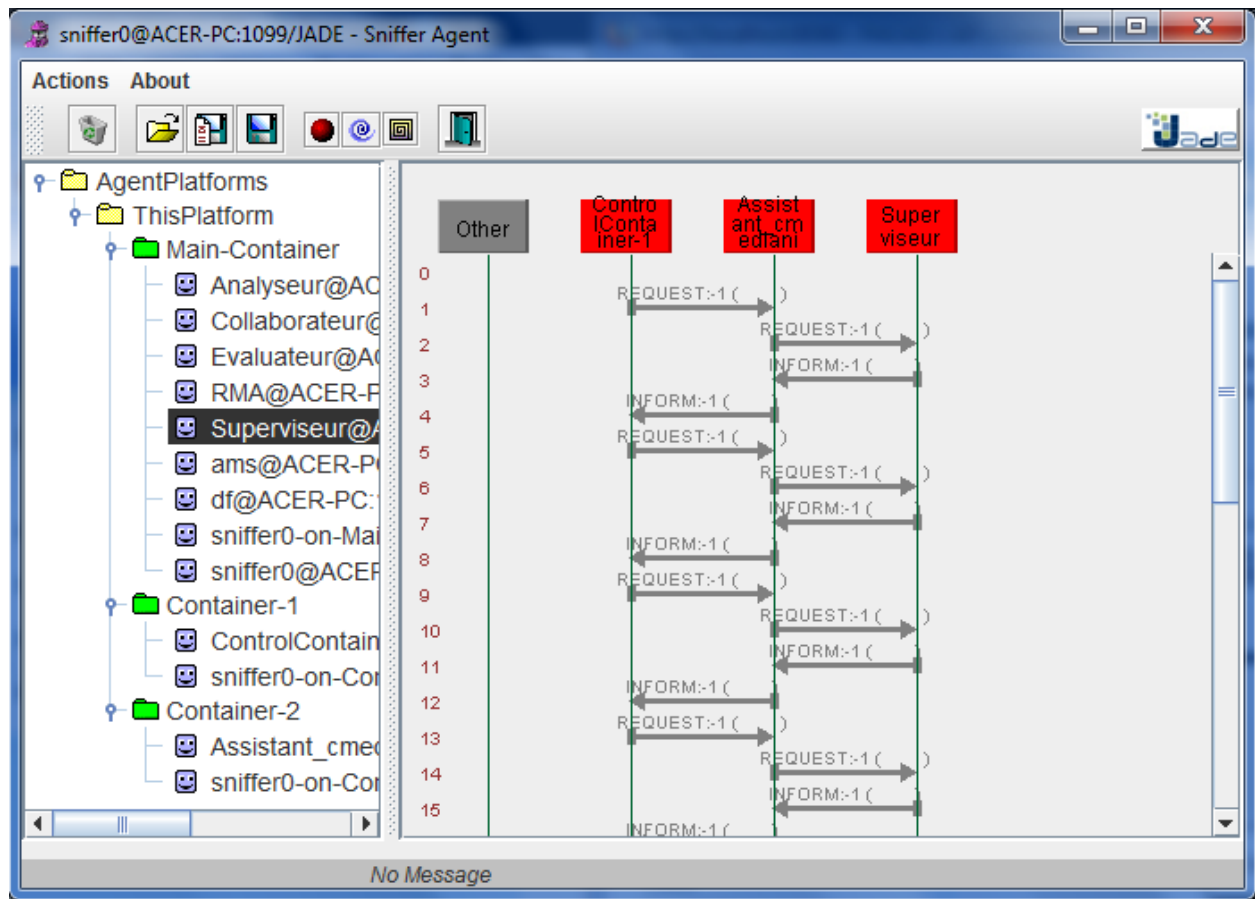


Figure 74. *L'agent Sniffer de JADE*

6.5 Expérimentation d'usage

L'objectif principal de cette étape d'expérimentation de l'usage est d'évaluer la manière dont une communauté d'apprenants utilise les fonctionnalités de notre environnement pour faciliter leurs échanges de connaissances et favoriser l'apprentissage collectif. Plus précisément, nous souhaitons que les apprenants puissent :

- Naviguer dans les différents espaces (l'espace de formation, l'espace privé et les espaces de leurs groupes),
- Ajouter de nouvelles ressources dans leur espace privé ou leur espace du groupe,
- Utiliser un moyen de communication (dans notre cas le forum) pour poser des questions ou répondre aux questions déjà posées,
- Résoudre un problème en groupe,

- Evaluer et inférer leurs niveaux de connaissance.

6.5.1 Evaluations

Afin d'évaluer les fonctionnalités de notre approche et les choix que nous avons faits, nous avons choisi de tester notre environnement auprès de 20 étudiants inscrits en première année licence académique pour la filière Maths Informatique (MI) à l'Université Ferhat Abbas de Sétif (UFAS) et qui suivent l'unité d'enseignement algorithmique et programmation (AP). Ce test a été mené au mois de mai 2012. Dans ce qui suit, nous présentons le protocole d'évaluation utilisé puis les résultats obtenus.

6.5.1.1 *Protocol d'évaluation*

Nous avons regroupé les étudiants par binôme et affecté à chaque binôme un problème à résoudre. Nous avons attribué à chaque étudiant un login et un mot de passe lui permettant d'avoir accès à l'espace de la formation, à son espace privé et à l'espace de son groupe qui n'était accessible que par lui et son binôme.

Pour chaque espace du groupe, nous avons ajouté un point d'accès permettant d'accéder au concept qui représente le problème. Ce concept a été relié à d'autres concepts de la formation et qui ont été jugés nécessaires pour résoudre le problème en question.

Pour lire l'énoncé du problème, chaque étudiant doit accéder à l'espace de son groupe puis choisir le point d'accès qui mène au concept qui représente le problème et enfin à la ressource qui contient l'énoncé du problème. (Ces étapes ont été expliquées aux étudiants par l'enseignant).

Cette évaluation n'a duré que trois semaines, c'est-à-dire jusqu'à la fin de l'année universitaire. Durée qui n'a pas été jugée significative. A la fin de l'évaluation, un questionnaire a été mis aux étudiants et leurs réponses ont été récoltées.

6.5.1.2 *Questionnaire*

Ce questionnaire de post évaluation a pour objectif de récolter quelques informations concernant les étudiants ainsi que leurs appréciations sur l'environnement.

Pour commencer le questionnaire, l'apprenant doit remplir les informations lui concernant comme, par exemple, son profil, sa fréquence d'utilisation d'internet, etc. Ensuite, nous avons posé des questions sur l'environnement et son utilisation. La plupart des questions sont de la forme « Pour l'affirmation suivante, dites si vous êtes d'accord ou non ». L'étudiant pouvait choisir entre les trois réponses suivantes : pas d'accord, un peu d'accord et d'accord.

Le questionnaire qui se trouve en annexe 3 est décomposé en plusieurs parties :

- Le profil de l'étudiant et l'utilisation d'internet : cette partie contient les informations concernant l'apprenant ainsi que ses habitudes d'utilisation d'Internet (son navigateur préféré, le nombre de connexions par semaine, etc.).
- La prise en main de l'environnement : cette partie concerne les appréciations des étudiants sur l'utilisation générale de l'environnement (la facilité d'accès, les points d'accès, etc.).
- La présentation de l'information : cette partie concerne la présentation de l'environnement et des différentes parties qui le composent (la navigation au sein de l'ontologie, les concepts et les liens entre concepts, les ressources, l'historique, l'utilisation des trois espaces de travail, l'utilisation des outils de communication comme le forum ainsi que les appréciations des étudiants sur le modèle d'inférence).
- La dernière partie est la partie avis général qui est réservée aux avis et suggestions des étudiants pour améliorer l'environnement.

6.5.2 Retour d'expérience

En consultant les historiques de navigation des étudiants, nous avons pu montrer que tous les étudiants ont accédé aux trois espaces de travail (de formation, privé et de groupe) et ont téléchargé l'énoncé du problème.

12 étudiants parmi les 20 étudiants ont réellement utilisé l'environnement. Les autres se sont limités uniquement à télécharger l'énoncé du problème. La quasi-totalité des étudiants ont exploité le lien qui existe entre le concept du problème et les autres

concepts de la formation qui ont relation avec le problème et aussi ils ont testé le fonctionnement du forum mais ils ne l'ont pas exploité (ni questions, ni réponses). Un seul binôme a ajouté une ressource dans l'espace du groupe tandis que quatre étudiants ont ajouté des ressources dans leurs espaces privés. Les problèmes n'étaient pas notés, un seul binôme a remis un résultat qui a été jugé correct par l'enseignant responsable.

Maintenant, nous focalisons sur les résultats qui concernent l'utilisation de l'historique et l'intégration d'un modèle cognitif et qui ont relation avec notre problématique. Nous donnons les résultats suivants :

6.5.2.1 L'historique

Le tableau suivant montre les résultats obtenus et concernant les appréciations des étudiants sur l'utilisation de l'historique.

		Nombre	%
L'utilisation de l'historique	d'accord	15	75
	un peu d'accord	01	05
	pas d'accord	04	20
La visualisation de l'historique	d'accord	16	80
	un peu d'accord	02	10
	pas d'accord	02	10

Table 19. *L'utilisation de l'historique*

Plus de 70% des étudiants ont apprécié l'utilisation de l'historique de navigation et plus de 80 % ont apprécié sa visualisation.

6.5.2.2 Le modèle cognitif

Le tableau suivant montre les résultats obtenus et concernant les appréciations des étudiants sur l'utilisation d'un modèle d'inférence.

		Nombre	%
L'utilisation d'un modèle cognitif	d'accord	16	80
	un peu d'accord	02	10
	pas d'accord	02	10
La visualisation du modèle cognitif par les étudiants	d'accord	16	80
	un peu d'accord	03	15
	pas d'accord	01	05
Visité?	d'accord	16	80
	un peu d'accord	02	10
	pas d'accord	02	10
Évalué?	d'accord	08	40
	un peu d'accord	02	10
	pas d'accord	10	50
Inféré?	d'accord	00	00
	un peu d'accord	00	00
	pas d'accord	20	100

Table 20. *L'utilisation du modèle cognitif*

Plus de 80% des étudiants ont apprécié l'utilisation d'un modèle cognitif et plus de 80 % ont apprécié sa visualisation.

Plus de 80% ont compris l'utilité du concept **Visité**, 40% des étudiants ont compris l'utilité du concept **Évalué**. Tandis qu'aucun étudiant n'a compris l'utilité du concept **Inféré**.

On peut conclure que malgré que la durée de cette première expérimentation n'ait pas été jugée significative, les résultats obtenus sont encourageants. Les étudiants semblent avoir apprécié l'intérêt d'un environnement tel qu'E-PACAD. Plus de 60% disent qu'ils vont conseiller l'environnement à leurs amis, et environ 50% disent qu'ils vont continuer à utiliser l'environnement et qu'ils vont l'utiliser pour la révision de leur examen.

6.6 Conclusion

Dans ce chapitre, nous avons présenté quelques idées d'implémentation suivies de quelques interfaces de l'environnement E-PACAD implémenté pour la mise en œuvre et la validation de notre démarche de modélisation de l'apprenant. Cette représentation devrait guider les pas de réalisation des concepts présentés au cours des chapitres précédents.

En résumé, Nous nous sommes intéressés dans l'environnement E-PACAD à la mise en place d'un apprentissage collaboratif et à la modélisation de certains aspects cognitifs de l'apprenant au cours de cet apprentissage. Cet environnement permet à un groupe d'apprenants :

- D'accéder à son espace de travail,
- De lire l'énoncé du problème qu'ils doivent résoudre,
- De naviguer dans l'ontologie d'application et d'accéder aux concepts de la formation qui peuvent leur aider dans la résolution du problème,
- D'ajouter des ressources dans leurs espaces de travail,

- De partager des connaissances avec les membres du groupe,
- De communiquer via les outils de communication,
- De déposer leurs solutions afin d'être évalué,
- Et enfin, de tester leurs niveaux de connaissances grâce au modèle cognitif mis en œuvre.

Malgré les difficultés rencontrées dans la mise en place d'une évaluation pertinente et qui sont liées à la contrainte du temps (c'était l'achèvement de l'année universitaire), nous avons pu constater que notre approche est intéressante. Les étudiants ont pu réaliser quelques-uns de nos objectifs comme, par exemple, la navigation dans l'environnement, le téléchargement des ressources pédagogiques, l'utilisation des espaces de partage et d'échange, etc.

Pour bien montrer l'intérêt de notre modèle cognitif mis en œuvre dans une situation d'apprentissage, nous prévoyons une deuxième évaluation auprès des étudiants de l'UFAS avec d'autres situations pédagogiques.

Conclusion générale et perspectives

Bilan des travaux et apports de la thèse

Dans le cadre de cette thèse, nous avons étudié la modélisation de l'apprenant dans un environnement d'apprentissage coopératif et proposé une architecture à base d'agents pour la gestion du modèle de l'apprenant. Nous présentons dans cette section un bilan et les apports de nos travaux de thèse. Dans les sections suivantes, nous dégagons les intérêts du projet. Nous concluons par la suite et nous terminons par les perspectives futures.

L'objectif de notre travail était de modéliser les attitudes cognitives et affectives de l'apprenant dans un environnement d'apprentissage coopératif à base d'agents. Et comme le modèle de l'apprenant n'est pas un composant autonome et isolé dans un environnement d'apprentissage, ceci nous a amené à proposer une architecture modulaire pour l'environnement d'apprentissage et la doter de plusieurs modules assurant la gestion du modèle de l'apprenant.

Pour des considérations liées à la complexité de la tâche, nous avons eu recours au paradigme d'agent pour proposer un environnement multi-agents assurant les fonctionnalités annoncées. Cet environnement multi-agents permet, d'un côté, d'enregistrer toutes les traces d'interaction de l'apprenant avec l'environnement d'apprentissage et avec les autres acteurs du système (les apprenants de son groupe, les enseignants, etc.). De l'autre côté, de fournir principalement aux enseignants une appréciation de l'état de l'apprenant et de l'état du groupe, et ce, en analysant ces traces d'interaction. Cette analyse nous a permis de définir quatre types de traces : les traces informatives (identification), les traces liées à l'exploitation d'une ressource, les traces associées à l'activité d'apprentissage et les traces attachées à l'activité de communication.

Pour la gestion du modèle de l'apprenant, un ensemble d'agents a été identifié :

- L'agent Interface : a en charge l'interface entre l'apprenant et l'environnement. C'est lui le responsable des interactions avec l'apprenant lui présentant le contenu pédagogique. Il transfère également les actions/réponses de l'apprenant aux autres agents de l'environnement.
- L'agent Assistant : il assiste l'apprenant durant toutes ses interactions avec l'environnement. Il joue plusieurs rôles : le rôle de collecteur des informations informatives concernant l'apprenant, le rôle assistant qui adresse des requêtes relatives au savoir (charger une ressource pédagogique, ajouter une ressource, etc.). Et par la suite il va assister l'apprenant dans son apprentissage, et enfin le rôle espion qui sauvegarde toutes les traces concernant les activités et les communications de l'apprenant dans le modèle de l'apprenant.
- L'agent Evalueur : Il effectue l'évaluation automatique des exercices de type QCM en comparant les connaissances de l'apprenant à celles de l'expert. Les résultats de cette évaluation sont enregistrés dans le modèle de l'apprenant et servent dans la mise à jour des connaissances de l'apprenant
- L'agent collaborateur: a en charge la communication entre l'apprenant et les autres acteurs de l'environnement (apprenants, enseignants, etc.) via les outils de communication.
- L'agent Analyse: Il analyse les traces enregistrées dans le modèle de l'apprenant afin de fournir à l'agent demandeur l'état de l'apprenant (nombre de messages reçus, nombre de messages envoyés, nombre de messages lus, le type de message, l'émetteur du message, nombre d'accès à une ressource pédagogique, pourcentage de lecture d'une ressource pédagogique, durée de consultation, les activités de l'apprenant, etc.) et l'état du groupe (apprenants présents, absents, dormants, actifs ou dynamiques).

Et d'autres agents assurant l'interface de l'environnement avec les ressources externes (bases de données, base de documents, etc.)

Comme nous l'avons vu au cours de cette thèse, cette architecture est également facilement évolutive en ajoutant de nouveaux agents permettant d'intégrer de nouvelles informations au modèle.

D'un point de vue du modèle de l'apprenant, nous avons adopté une nouvelle approche qui n'avait pas été explorée dans ce sens : représenter le modèle de l'apprenant lui-même comme un système multi-agents. Ainsi, le modèle de l'apprenant est géré par plusieurs agents où chacun a en charge une petite partie du modèle. Le modèle est donc passé d'une perspective statique avec les approches existantes à une perspective dynamique où le modèle de l'apprenant se maintient tout seul à jour.

Malheureusement, la complexité du domaine abordé et l'ampleur de la tâche nous ont conduit à nous limiter à quelques informations du modèle de l'apprenant : les données informatives (l'identification, les préférences, etc.), quelques aspects cognitifs, quelques aspects comportementaux.

De plus, ce travail a nécessité de nombreuses réflexions sur de nombreux domaines comme l'apprentissage de la didactique informatique, la modélisation de l'apprenant, les ontologies, ou bien encore les agents, et ceci, n'a pas toujours été évident.

Dans les EIAH, plusieurs recherches utilisent une approche par composants dans le développement des prototypes afin de tirer profit des propriétés de réutilisabilité pour en faciliter la coopération et l'interopérabilité. Nous avons positionné par rapport à cette approche et montrer que le paradigme d'agent était le mieux à répondre à nos préoccupations.

Nous avons eu recours au paradigme d'agent pour identifier des outils de développement et leur mise en œuvre. Nous nous sommes rendu compte que les recherches scientifiques faites dans le domaine des agents sont pleines de bonnes intentions, mais restent assez discrètes sur les outils et méthodologies de mise en œuvre des systèmes à base d'agents. Durant nos travaux, une méthodologie d'analyse, de conception et de spécification de systèmes multi-agents a été utilisée. Le point fort de cette méthodologie réside dans l'identification et la spécification des interactions entre les agents du SMA. Cette méthodologie ne définit et n'impose aucune architecture

interne des agents mis en œuvre. Cette tâche est laissée au concepteur. Afin de construire l'intérieur de nos agents, nous avons identifié un modèle d'architecture interne pour nos agents et utilisé les mécanismes de communication implémentés dans cette architecture, il s'agit de l'architecture JADE implémenté sous java.

Le fait que de nombreux travaux soient encore en cours aussi bien en ce qui concerne la conception et la réalisation des autres agents de l'environnement qu'en ce qui concerne la réalisation d'autres situations didactiques, nous a considérablement limités dans le développement d'un prototype. Nous avons dû essentiellement nous concentrer sur le choix des différents outils ainsi que sur la pertinence et la souplesse d'utilisation des outils retenus. De fait, il a été tout simplement impossible d'envisager de faire une évaluation pertinente de notre prototype autre que de juger les différents outils retenus pour le réaliser. Néanmoins, le prototype que nous avons réalisé a permis de poser les bases pour de futurs développements.

Intérêt du projet

Notre travail s'inscrit dans le champ de recherches sur les environnements informatiques pour l'apprentissage humain (EIAH), Il s'agit alors d'environnements informatiques conçus dans le but de favoriser un apprentissage personnalisé et adaptatif. Nous nous intéressons plus particulièrement au traitement des interactions entre les apprenants et les environnements d'apprentissage pour envisager une personnalisation de l'apprentissage.

Nous avons montré dans cette thèse que les données d'interaction enregistrées dans les environnements d'apprentissage peuvent être utilisées pour construire des indicateurs sur l'état de l'apprenant et sur la progression de ses connaissances au sein du groupe dans une session d'apprentissage. Nous avons montré que la complexité de la prise en compte de l'apprenant n'apparaît pas toujours évidente. Ceci constituait pour nous la justification majeure pour la mise au point d'un environnement informatique intégrant des outils permettant de prendre en compte les aspects cognitifs et affectifs de l'apprenant afin d'envisager un apprentissage adaptatif à ses besoins et ses préférences.

Conclusion

La présente thèse est le résultat de plusieurs années de recherches. Partant d'une problématique : comment prendre en compte les aspects affectifs et cognitifs d'un apprenant dans une situation apprentissage coopératif à distance?, nous avons eu recours au paradigme d'agents comme solution informatique à une telle problématique (Mediani et Djoudi, 2007). Nous avons proposé une architecture multi-agents pour notre environnement de modélisation de l'apprenant. Cette architecture est basée sur un élément très essentiel pour notre travail qui est le modèle de l'apprenant. Puis nous avons procédé à la définition des données qui peuvent être incluses dans le modèle de l'apprenant. Nous avons pu montrer que ces données peuvent être de nature statique ou dynamique et donc une taxonomie pour ces données a été proposée. Pour le recueil et l'organisation des données d'interaction, il a fallu utiliser des méthodes de collecte. Nous avons utilisé les formulaires de saisie et les questionnaires pour les données statiques. Par contre, pour les données dynamiques, nous avons mis en œuvre un outil à base d'agents recueillant toutes les données d'interaction de l'apprenant avec l'environnement. Une base de données contenant ces informations a été construite. Il a aussi fallu montrer que les données d'interaction enregistrées pouvaient permettre de construire des indicateurs de l'état des apprenants, de l'état des groupes, et de la progression de la session de formation. L'absence de tels indicateurs dans les environnements d'apprentissage et d'enseignement existants nous a permis de justifier de devoir proposer ces nouvelles fonctionnalités. Pour tirer ces indicateurs, un agent analyseur des traces a été proposé et il est en cours de mise en œuvre. Nous avons choisi d'implémenter ces fonctionnalités sous la forme d'un système multi-agents. Nous avons choisi d'aller jusqu'au bout de l'implémentation de notre système multi-agents. Très vite, nous nous sommes aperçus de la rareté des outils de développement appropriés. Il nous a fallu donc utiliser une méthodologie pour l'analyse, la conception et la spécification des systèmes multi-agents.

Nous l'avons fait et avons appliqué toutes les étapes de cette méthodologie. L'objectif de la construction concrète des agents nous a amené à nous intéresser à un modèle d'architecture interne des agents. Cette architecture interne nous conduit vers le

langage Java, dont il a fallu saisir toutes les subtilités pour programmer notre système multi-agents.

Et comme la formation à distance étant dès le départ un champ pluridisciplinaire, ces années de recherches nous auront permis de nous intéresser à plusieurs domaines de recherche allant des sciences humaines et sociales à l'informatique.

Perspectives

Nous pouvons envisager plusieurs perspectives possibles pour étendre ce travail.

Prise en compte d'autres informations dans le modèle de l'apprenant

Comme nous l'avons expliqué, notre modèle de l'apprenant a pris en compte quelques informations concernant l'apprenant (des informations informatives concernant l'identification, les préférences, quelques aspects cognitifs, quelques aspects comportementaux). Dans la suite de nos travaux, il faudrait prendre en compte de manière plus précise les autres entrées du modèle de l'apprenant. Pour cela, il conviendra de travailler encore avec les experts en apprentissage de la didactique d'informatique afin de détailler davantage les situations didactiques existant sur les autres entrées de notre modèle : d'autres aspects affectifs, cognitifs et comportementaux.

D'un point de vue informatique, cela reviendra à ajouter de nombreuses autres règles d'analyse. Celles-ci devront bien entendu être intégrées aux agents d'analyse déjà existant pour que ceux-ci puissent émettre des hypothèses sur les autres entrées du modèle.

Développement d'autres situations didactiques

Toutes les situations didactiques de notre environnement n'ont pas été réalisées. Ceci en raison du travail qui est encore en cours et de la complexité de la tâche. Les situations didactiques réalisées concernent quelques modules pour l'apprentissage de l'informatique. Il conviendrait donc de poursuivre le travail de réalisation des autres situations didactiques du système.

D'un point de vue informatique, cela reviendra à concevoir de nouvelles ontologies d'applications capables de satisfaire ces situations didactiques. Pour cela, il faudra, toujours en collaboration avec les experts en apprentissage, recueillir les différentes règles d'analyse permettant d'émettre des hypothèses sur chaque situation didactique et les intégrer aux ontologies.

Travail avec plusieurs apprenants

Le fait de travailler avec plusieurs apprenants pourrait être utile et apporter de nouvelles connaissances. En effet, les modèles de l'apprenant étant stockés dans une base de données, on pourrait imaginer exploiter cette base pour découvrir, par exemple, de nouvelles règles d'analyse. A cet effet, nous pourrions envisager d'utiliser des techniques de datamining ou d'apprentissage automatique.

L'élargissement de l'architecture de notre environnement

Au cours de nos réflexions, il nous est apparu que notre architecture pourrait être élargie. Celle-ci pourrait être une architecture composée d'agents auxquels on intègre une intelligence plus importante que ceux de l'architecture proposée. Il faudrait intégrer au sein de chaque agent des capacités d'analyse des comptes-rendus en ce qui concerne la connaissance gérée. Ceci serait envisageable à partir du moment où l'on disposerait de l'ensemble des situations didactiques de l'environnement.

Exploration de la collaboration

Nous continuons notre exploration de la collaboration en essayant d'évaluer l'environnement auprès des étudiants de l'UFAS et de prendre en compte les observations et les critiques des utilisateurs pour faire évoluer notre environnement E-PACAD.

Bibliographie

- Abel, M.H., Lenne, D., Leblanc, A. (2007). "An organizational memory-based environment as support for organizational learning", Dans European Symposium on Innovative Management Practices, ERIMA'07, pages 15-16.
- Abel, M.H. (2008). "Apport des Mémoires Organisationnelles dans un contexte d'apprentissage", mémoire d'habilitation à diriger des recherches, université de technologie de Compiègne.
- Akhras, F., Self, J. (1996). "From the process of instruction to the process of learning constructivist implications for the design of intelligent learning environment", Proceedings of the Euro. AIED, Lisbon, Portugal, 9-15.
- Alani H., Kim, S., Millard D.E., Weal, M.J., Hall, W., Lewis, P.H., Shadbolt, N.R. (2003). "Automatic Ontology based knowledge extraction from Web documents", IEEE computer society, IEEE intelligent system.
- Alesso, P.H., Smith, C.F. (2004). "Developing Semantic Web Services", A K Peters ltd, Wellesey MA, Canada, 165-272.
- Anderson, J.R., Boyle, C.F., Corbett, A., Lewis, M.W. (1990). "Cognitive modelling and intelligent tutoring", Artificial Intelligence, 42, 7-49.
- Anderson, J.R., Corbett, A.T., Koedinger, K., Pelletier, R. (1995). "Cognitive tutors: Lessons learned", The Journal of Learning Sciences, 4, 167-207.
- Anderson, J.R. (1983). "The Architecture of Cognition", Harvard University Press, Cambridge, Mass.
- Anderson, J.R., Reiser, B.J. (1985). "The Lisp tutor", Byte, vol. 10, n° 4, pp. 159-175.
- Avouris, N., Komis, V., Fiotakis, G., Margaritis, M., Voyiatzaki, E. (2005). "Logging of fingertip actions is not enough for analysis of learning activities", Workshop AIED'05, Amsterdam, pp. 1-8.
- Bannon, L., Schmidt, K. (1991). "Cscw : four character in search of a context", Studies in Computer Supported Collaborative Work : theory, practice and design.
- Barr, A., Feigenbaum, E.A. (1981). "The handbook of artificial intelligence", New-York: Addison-Wesley, Vol. 1.
- Barros, B., Verdejo, M. F. (2000). "Analysing student interaction processes in order to improve collaboration. The DEGREE approach", International Journal of Artificial Intelligence in Education.

- Beauvisage, T. (2004). "Sémantique des parcours des utilisateurs sur le Web", Thèse de doctorat, Université de Paris X : 361p.
- Becker A., Naïm P., (1999). "Les réseaux bayésiens, modèles graphiques de connaissances", Eyrolles, Paris.
- Benayache, A., (2005). "Construction d'une mémoire organisationnelle de formation et évaluation dans un contexte elearning : le projet Memora", thèse de doctorat de l'université de technologie de Compiègne.
- Bermudez, L., Piasecki, M., (2006). "Metadata Community Profiles for the Semantic Web", *Geoinformatica*, Springer 10:159-176.
- Billsus, D., Pazzani, M. (1999). "An hybrid user model for news story classification", In: *User Modeling, Proceedings of the Seventh International Conference*, Banff, Canada, pp. 99-108.
- Blanchard, J., Petitjean, B., Artières, T., Gallinari, P. (2005). "Un système d'aide à la navigation dans des hypermédias", Actes de la conférence Extraction et Gestion des Connaissances, EGC'05, Paris, France.
- Boukelif, A., (2008). "Apprentissage collaboratif par résolution de problèmes et réalisation de projets pédagogiques : la méthode app", Dans *TICEMED*.
- Bourigault, D. and Aussenac-Gilles N., (2003). "Construction d'ontologies à partir de textes", In *Proceedings of : 10ème conférence annuelle sur le Traitement Automatique des Langues (TALN 2003)*, pages 27–50, Batz-sur-Mer, Juin.
- Brecht B.J., MacCalla G.I., Greer J.E., Jones M., (1989). "Planning the Content of Instruction", *proceedings of Artificial Intelligence and Education*, pp 32-41, 1989.
- Broisin J., Vidal P., Sibilla M. (2006). "A management framework for tracking user activities in a web-based learning environment based on a model driven approach", dans: *educational multimedia, hypermedia & telecommunications (EDMEDIA 2006)*, Orlando, AACE, juin.
- Brooke, A., Wei, D. (2005). "Architecture for annotation and Ontology based querying of semantic Web Resources", *IEEE/WIC/ACM international Conference on Web Intelligence*.
- Brost, W.N., (1997). "Construction of engineering ontologies", University of Twente, Enschede, Center for Telematica and Information Technology.
- Bruillard, Éric, (1997). "Les machines à enseigner", Éditions Hermès, Paris, 320 p.
- Brusilovsky P., Vassileva J. (2003a). "Course sequencing techniques for large-scale web-based education", *International Journal of Continuing Engineering Education and Lifelong Learning*, 13 (1-2), 75-94.
- Brusilovsky, P. (2003b). "Adaptive navigation support in educational hypermedia: The role of student knowledge level and the case for meta-adaptation", *British Journal of Educational Technology*, 34 (4), pp.487-497.
- Brusilovsky, P. (2003c). "Authoring Tools for Advanced Technology Learning Environments", Chapter *Developing Adaptive Educational Hypermedia Systems: from Design Models to Authoring Tools*, pages 377-409. Kluwer Academic Publishers, 2003d.

- Brusilovsky, P. (2004c). "KnowledgeTree: A Distributed Architecture for Adaptive E-Learning", WWW 2004, New York, New York, USA. ACM 1-58113-912-8/04/0005.
- Brusilovsky, P. Pesin, L. (1994). "ISIS-Tutor: An adaptive hypertext learning environment", JCKBSE'94, Japanese-CIS Symposium on knowledge-based software engineering. Pereslavl-Zalesski, Russia, 83-87.
- Brusilovsky, P., Anderson, J. (1998a). "ACT-R electronic bookshelf: An adaptive system for learning cognitive psychology on the Web", In H. Maurer, & R. G. Olson (Eds.), Proceedings of WebNet'98, World Conference of the WWW, Internet, and Intranet, November 7-12, 1998. Orlando, FL, AACE. - pp. 92-97.
- Brusilovsky, P., Eklund, J., Schwarz, E. (1998b). "Web-based education for all: A tool for developing adaptive courseware", Computer Networks and ISDN Systems, 30 (1-7), 291-300.
- Burton R. and Brown J. (1978). "A tutoring and student modelling paradigm for gaming environments", ACM SIGCSE bulletin 8(1) : 236-246.
- Burton R.R., Brown J.S. (1982). "An investigation of computer coaching for informal learning activities", dans Sleeman D., Brown J.S. (dir.), Intelligent Tutoring Systems, Academic Press, Londres, pp 79-98.
- Carberry, S., (2001). "Techniques for Plan Recognition", User Modeling and User-Adapted Interaction, 11(1 – 2), 31 – 48.
- Carr B., Goldstein I. P. (1977). "Overlay: a theory of modelling for computer aided instruction", AI Memo 406. Cambridge USA: Massachusetts Institute of Technology.
- Castillo G., Gamma J., Breda A. (2003). "Adaptive bayes for a student modelling prediction task based on learning styles", user modelling 2003, 9th International Conference, Springer Verlag.
- Champin, P.-A., Mille, A., Prié, Y. (2003), "MUSSETTE: Modelling USEs and Tasks for Tracing Experience", in Workshop From structured cases to unstructured problem solving episodes- WS 5 of ICCBR'03, Trondheim (NO), B. Fuchs and A. Mille, Editors. 2003, NTNU.
- Champin, P.-A., Prié, Y., Mille, A. (2004). "MUSSETTE : a framework for Knowledgefrom Experience", EGC'04, RNTI-E-2, Cepadues Edition, Clermont-Ferrand. France, pp. 129-134.
- Clancy, W. (1983). "GUIDON", Journal of Computer Based Instruction, 10, 8-15.
- Conati C., Gertner A., VanLehn K., (2002). "Using Bayesian networks to manage uncertainty in student modeling", Journal of User Modeling and User-Adapted Interaction, Vol. 12, pp 371-417.
- Conati C., VanLehn K., (1996). "POLA: a Student Modeling Framework for Probabilistic On-Line Assessment of Problem Solving Performance", In Proceedings of UM-96, Fifth International Conference on User Modeling , Kailua-Kona, Hawaii.
- Cottier, P., Choquet, C. (2005). "De l'utilisateur construit à l'utilisateur participant", EIAH 2005, Montpellier, pp. 449-454.

- Coutaz, J., (1990). "Interfaces homme-ordinateur : conception et réalisation", Dunod edition.
- Cram, D., Jouvin, D., Mille, A. (2007). "Visualisation interactive de traces et réflexivité : application à l'EIAH collaboratif synchrone eMédiathèque", STICEF, volume 14.
- Crawley, R. (1997). "What is cscl ?", Collaborative Computing Research Group, Joint Information Systems Committee.
- Cristea, A., Garzotto, F. (2004a). "ADAPT Major Design Dimensions for Educational Adaptive Hypermedia", ED-MEDIA'04, AACE.
- Cristea, A. (2004b). "Evaluating Adaptive Hypermedia Authoring while Teaching Adaptive Systems", SAC'04, ACM, Nicosia, Cyprus.
- Cristea, A., Kinshuk, (2003). "Considerations on LAOS, LAG and their Integration in MOT", Proceedings of ED-MEDIA'03, Honolulu, Hawaii, AACE.
- Cristea, A., Verschoor, M. (2004c). "The LAG Grammar for Authoring the Adaptive Web", International Conference on Information Technology: Coding and Computing (ITCC'04) Volume 1, pp. 382.
- De Bra, P. M. E. (1996). "Teaching Hypertext and Hypermedia through the Web", Journal of Universal computer science, 2(12), 797-804.
- De Bra, P., Aerts, A., Berden, B., De Lange, B., Rousseau, B., Santic, T., Smits, D., Stash, N. (2003). "AHA! the adaptive hypermedia architecture", In HT'03: Proc. of the 14th ACM Conf. on Hypertext and Hypermedia, pages 81-84. ACM Press.
- De Bra, P., Ruiter, J.-P. (2001). "AHA! Adaptive hypermedia for all", In W. Fowler, & J.Hasebrook (Eds.), Proceedings of WebNet'2001, World Conferen.
- DeLoach, S.A., Wood, M.F., Sparkman, C.H. (2001). "Multiagent Systems Engineering", International Journal on Software Engineering and Knowledge Engineering, Vol. 11, N° 3, pp: 231-258.
- Demazeau, Y. (1995). "From Interactions to Collective Behaviour in Agent-Based Systems", In: Proceedings of the 1st European Conference on Cognitive Science. Saint-Malo, France, Avril 1995.
- Dillenbergh, P., Baker, M., Blaye, A., O.Malley, C. (1996). "The evolution of research on collaborative learning", In E.Spada and P.Reiman (Eds) Learning in humans and machine . Towards an interdisciplinary learning science. (pp. 189-211). Oxford: Elsevier.
- Dillenbourg, P., Self, J.A. (1992). "A framework for learner modelling. Interactive Learning Environments", vol. 2, n°2, pp. 111-137.
- Dillenbourg, P. (1999). "What do we mean by collaborative learning?", In Dillenbourg P. (Ed) Collaborative-learning Cognitive and computational Approaches. (pp.1-19). Oxford: Elsevier.
- Dimitrova, V. (2001). "Interactive open learner modeling", PhD thesis, University of Leeds.
- Fansler, K., Riegel, R. (2004). "A Model of Online Instructional Design Analytics", 20th Annual Conference on Distance Teaching and Learning, Madison, Etats-Unis.

- Ferber, J., (1995). "Les systèmes multi-agents : vers une intelligence collective", Interditions.
- Ferraris, C., Vignollet, A. L., David, J. (2005). "Modélisation de scénarios d'apprentissage collaboratif pour la classe", Environnements Informatiques pour l'Apprentissage Humain EIAH'05, Montpellier, 26 et 27 Mai, 2005.
- Finin, T., Fritzson, R., McKay, D., McEntire, R. (1994). "KQML as an Agent Communication Language", Third International Conference on Information and Knowledge Management (CIKM-94), ACM Press.
- Foell, S., Boessling, P., Linner, D., Radush, I., Steglich, S. (2007). "Self organizing pervasive online communities". IEEE International symposium on autonomous decentralized system.
- Fournier, J.P., (1994). "Vers la programmation dociles", Revue Techniques et Sciences Informatique, Volume 13 - n° 1, pp. 79-104.
- Frasincar F., Barna P., Houben G., Fiala Z. (2004). "Adaptation and Reuse in Designing Web Information Systems", International Conference on Information Technology: Coding and Computing (ITCC'04) Volume 1, pp. 387.
- Gamboa, H., and Fred A. (2001). "Designing intelligent tutoring systems : a bayesian approach", ICEIS Artificial Intelligence and Decision Support Systems, 452-458.
- Gandon, F. (2006). "Le Web sémantique n'est pas antisocial", Semaine de la connaissance. Nantes.
- George, S. (2001). "Apprentissage collectif à distance. SPLACH : un environnement informatique support d'une pédagogie de projet", Thèse de doctorat, Université du Maine.
- George, S. (2004). "Analyse automatique de conversations textuelles synchrones d'apprenants pour la détermination de comportements sociaux", Revue Sciences et technologies de l'information et de la communication pour l'éducation et la formation (STICEF) Numéro spécial : technologies et formation à distance, pp. 165-193.
- Georgeon, O., Mille, A., Bellet, T. (2006). "Abstract : un outil et une méthodologie pour analyser une activité humaine médiée par un artefact technique complexe", Ingénierie des Connaissances IC 2006, Nantes.
- Grandbastien, M., Labat, J.M., (2006). "Environnements informatiques pour l'apprentissage humain", Cognition et traitement de l'information, Hermes Lavoisier, ISBN 2-7462-1171-8, mars.
- Greer, J. E., McCalla, G. I. (1994). "Student Modelling: The Key to Individualized Knowledge-Based Instruction", Berlin, Springer-Verlag.
- Grigoriadou, M., Papanikolaou, K., Kornilakis, H., & Magoulas, G. (2001). "INSPIRE: An INtelligent System for Personalized Instruction in a Remote Environment", In P. D. Bra, P. Brusilovsky, & A. Kobsa (Eds.), Proceedings of Third workshop on Adaptive Hypertext and Hypermedia, Sonthofen, Germany, Technical University Eindhoven. - pp. 13-24.
- Grüber, T., (1993). "A Translation Approach to Portable Ontology Specifications", Knowledge Acquisition 5, pp. 199-220.
- Gutknecht, O., Ferber J. (1999). "Vers une méthodologie organisationnelle des systèmes multiagents", Actes des JFIADSMA'99, pp. 81-92, Hermès, Paris, France.

- Haton J.P., Bouzid N., Charpillet F., Haton M., Léasri H., Marquis P., Mondot T., Napoli A. (1991). "Le raisonnement en intelligence artificielle", Modèles, techniques et architectures pour les systèmes à bases de connaissances, InterEditions.
- Henri, F., Lundgren-Cayrol, K., (1998). "Apprentissage collaboratif et nouvelles technologies".
- Heraud J.-M., Marty J.-C., France L., Carron T. (2005). "Helping the Interpretation of Web Logs : Application to Learning Scenario Improvement", Workshop AIED'05, Amsterdam.
- Heraud, J.-M., France, L., Mille, A. (2004). "Pixed: An ITS that guides students with the help of learners' interaction logs", In Proceedings of the 7th International Conference on Intelligent Tutoring Systems. Maceio, Alagoas, Brazil.
- Hibou, M., Labat, J.M. (2004). "Embedded Bayesian network student models", conference ITHET04, Istanbul.
- Holt, P., Dubs S., Jones M. and Greer J. (1994). "The state of student modeling", Student modelling: the Key to individualized knowledge-based instruction, 3-38.
- Ikeda, M., Mizoguchi, R., Kakusho, O. (1989). "Student Model Description Language SMDL and Student Model Inference System SMIS", IEICE, pp 112-120.
- Ikeda, M., Seta K., Mizogushi R., (2002). "Ontospec : une méthode de spécification semi-informelle d'ontologies", dans IC'2002.
- Iksal, S., Choquet, C. (2005). "An open architecture for usage analysis in a e-learning context", The 5th IEEE International Conference on Advanced Learning Technologies, Kaohsiung, Taiwan, pp. 177-181.
- Jennings, N.R., Sycara, K., Wooldridge, M., (1998). "A Roadmap of Agent Research and Development", Autonomous Agents and Multi-Agent Systems, 1, pp. 275-306. Kluwer Academic Publishers, Boston.
- Jermann, P., Soller, A., Muehlenbrock, M. (2001). "From Mirroring to Guiding: A Review State of the Art Technology for supporting Collaborative Learning", Proceedings of the First European Conference on Computer-Supported Collaborative Learning.
- Kass, R. (1987). "The Role of User Modelling in Intelligent Tutoring System", MooreSchool, Université de Pennsylvanie.
- Kay J. (2000a). "Stereotypes, Student models and Scrutability", Proceedings of fifth International Conference on Intelligent Tutoring Systems, Springer-Verlag, 19-30.
- Kay, J. (2000b). "Accretion representation for scrutable student modeling", Proceedings of fifth International Conference on Intelligent Tutoring Systems, Springer-Verlag, 292 – 303.
- Kay, J. (2001). "Learner control", User Modeling and User-Adapted Interaction 11(1-2), 111-127.
- Klyne, G., Reynolds, F., Woodrow, C., Hidetaka, O., Hjelm, J., Butler, M.H., Tran, L. (2003). "Composite capability/preference profiles (cc/pp): Structure and vocabularies", W3C Working Draft 25 March.

- Kostelnik R. and Bielikova M. (2003). "Web-based environment using adapted sequences of programming exercises", In M. Beneš, editor, Proc. of Int. Conf. On Information Systems Implementation and Modelling, pages 33-40.
- Kristofic, A., Bielikov, M. (2005). "Improving Adaptation in WebBased Educational Hypermedia by means of Knowledge Discovery", HT'05 ACM 1595931686/05/0009, Salzburg, Austria September 6-9.
- Lehtinen, E., Hakkarainen, K., Lipponen, L., Rahikainen, M. et Muukkonen, H. (1999). "Computer supported collaborative learning: A review", Rapport. <http://etu.utu./papers/clnet/clnetreport.html>.
- Loghin, G.-C. (2006). "Aide à la compréhension du comportement de l'utilisateur par la transformation des traces collectées", 1ère Rencontre de Jeunes Chercheurs en EIAH'2006.
- Loghin, G.-C. (2008). "Observer un Environnement Numérique de Travail pour réguler les activités qui s'y déroulent", Thèse de Doctorat. Cotutelle entre l'Université de Savoie et l'Université Technique de Cluj-Napoca.
- Maes, P. (1997). "Agents that Reduce Work and Information Overload". In Software Agents, ed. J.M. Bradshaw. Menlo Park, Californie, AAAI Press.
- Manber, U., Patel, A., Robison, J. (2000). "Experience with Personalization on Yahoo!", Zeitschrift Communications - The ACM, Volume 43, Nummer 8, August.
- Martin, B. (1999). "Constraint-Based Modeling: Representing Student Knowledge", New Zealand Journal of Computing, Vol 7(2), November 1999, pp 30-38.
- Mayo, M., Mitrovic, A. (2001). "Optimising ITS Behaviour with Bayesian Networks and decision Theory", International Journal of Artificial Intelligence in Education, 12, pp.124-153.
- Mediani, C., Djoudi, M. (2007). "An Agent-Based Design for the Learner Behaviour Modelling", Information Technology Journal (ITJ), ISSN: 1812-5638, Vol. 6, No. 5, 681-689.
- Michel, C., Prié, Y., Le Graet, L. (2005). "Construction d'une base de connaissance pour l'évaluation de l'usage d'un environnement STIC", 17eme Conférence Internationale Francophone sur l'Interaction Homme- Machine, Toulouse, France, pp. 199-202.
- Millan, E., Pérez-de-la-Cruz, J.L., Suarez, E. (2000). "Adaptive Bayesian networks for multilevel student modeling", Proceedings of fifth International Conference on Intelligent Tutoring Systems, Springer-Verlag, 534 - 543.
- Mille, A., Prié Y. (2006). "Une théorie de la trace informatique pour faciliter l'adaptation dans la confrontation logique d'utilisation/logique de conception", 13ème Journées de Rochebrune -Traces, Enigmes, Problèmes : Emergence et construction du sens - Rencontres interdisciplinaires sur les systèmes complexes naturels et artificiels. Rochebrune, France:12 p.
- Mislevy, R., Gitomer D., (1996). "The role of probability-based inference in an intelligent tutoring system", User Modeling and User-Adapted Interaction, Vol. 5, pp 253-282.

- Mitrovic, A. (1998). "Experiences in Implementing Constraint-Based Modeling in SQL-Tutor", *Intelligent Tutoring Systems 1998*: 414-423.
- Moore, A.W. (2004). "Bayesian Networks", Tutorial Slides (<http://www.autonlab.org/tutorials/bayesnet.html>).
- Moulin, B., Chaib-draa, B. (1996). "An Overview of Distributed Artificial Intelligence in Foundations of Distributed Artificial Intelligence", Eds O'Hare and Jennings. New-York : Wiley. pp. 3-55.
- Najjar, J., Ternier, S., Duval, E. (2004). "User Behavior in Learning Object Repositories: An Empirical Analysis", *Educational Multimedia, Hypermedia & Telecommunications - EDMEDIA04*, Lugano, Suisse, pp. 4373-4379.
- Nguyen, L., Phung, D. (2009). "Combination of bayesian network and overlay model in user modeling", *iJET*, 4(4): 41-45.
- Nicaud, J., (1987). "Aplusix : un système expert en résolution pédagogique d'exercices d'algèbre", PhD thesis, Thèse de l'Université Paris XI-Orsay.
- Nicaud, J.F., Vivet, M., (1988). "Les tuteurs intelligents, réalisations et tendances de recherche", *Revue Technique et Science Informatiques*.
- Nwana, H. S. (1996). "Software Agents : An Overview", *Knowledge Engineering Review*, Vol. 11, N° 3, pp. 1-40.
- Occello, M., Koning, J-L., Baeijs, C. (2001). "Conception de systèmes multi-agents : quelques éléments de réflexion méthodologique", TSI 2001.
- Ohlsson S. (1994). "Constraint-Based Student Modelling, Student Modelling : the Key to Individualized Knowledge-based Instruction", J. E. Greer, G. McCalla (Eds.), Berlin, Springer, p. 167-189.
- Ollagnier-Beldame, M., et Mille, A. (2007). *Faciliter l'appropriation des EIAH par les apprenants via les traces informatiques d'interactions*.
- Papanikolaou, K., Grigoriadou, M., Kornilakis, H., Magoulas, G. (2003) "Personalizing the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE", *User Modeling and User-Adapted Interaction*, 13(3), 213 – 267.
- Pearl J., (1988). "Probabilistic Reasoning in Intelligent System", Morgan Kauffman.
- Pérez A.G. and Benjamins V.R., (1999). "Overview of Knowledge Sharing and Reuse Components : Ontologies and problem-Solving Methods", *Proceeding og the IJCAI-99 workshop on Ontologies and problem-Solving Methods (KRR5)*, Stockholm (Suède), pp. 1.1-1.15.
- Pernin, J.-P. (2005). "CSE, un modèle de traitement de traces", CLIPS-IMAG.
- Piombo, C. (2007). "Modélisation probabiliste du style d'apprentissage et application à l'adaptation de contenus pédagogiques indexés par une ontologie", thèse de doctorat de l'université de Toulouse.

- Psyché, V., Mendes, O., Bourdeau, J., (2003). "Apport de l'ingénierie ontologique aux environnements de formation à distance", Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, Revue sticef.org, Vol. 10. 2003.
- Ragnemalm E.L. (1996). "Student diagnosis in practice; Bridging a gap", User Modelling and User Adapted Interaction, Vol. 5, No. 2, pp 93-116.
- Ritter, F. E., Baxter, G. D., Jones, G., & Young, R. M. (2000). "Supporting cognitive models as users". ACM Transactions on Computer-Human Interaction, 7(2), 1-33. pp. 141-173.
- Rodden, T. (1991). "A survey of cscw systems", Interacting with Computer, 3.
- Rodden, T. (1994). "Computer-supported cooperative work : Its history and participation", IEEE Computer, 27.
- Rodriguez, M.A., Steinbock, D.J., Watkins, J.H., Gershenson, C., Bollen, J., Grey, V., Degraf, B. (2007). "Smartocracy: Social networks for collective decision making", Hawaii international conference on system sciences.
- Roschelle, J., Teasley, S., (1995). "The construction of shared knowledge in collaborative problem solving", In C.E. O'Malley (Ed) Computersupported collaborative learning. Heidelberg: Springer-Verlag.
- Salber, D., Coutaz, J., Decouchant, D. et Riveill, M. (1995). "De l'observabilité et de l'honnêteté : le cas du contrôle d'accès dans la communication homme-homme médiatisé", Dans CEPAD, IHM'95.
- Sanderson, P.-M., et Fisher, C.-A. (1994). "Exploratory sequential data analysis: foundations", Human-Computer Interaction, 9(3): 251-317.
- Schiaffino S., Amandi A., (2000). "User profiling with Case-Based Reasoning and Bayesian Networks", international joint conference IBERAMIA-SBIA 2000 – Atibaia, Brazil, 12-21.
- Schuyten, G., Dekeyser, H.M. (2007). "Preference for textual information and acting on support devices in multiple representations in a computer based learning environment for statistics", Computers in human behaviour 23, 2285-2301, Elsevier ScienceDirect.
- Séjourné, A., Baker, M., Lund, K., Molinari, G. (2004). "Schématisation argumentative et co-élaboration de connaissances : le cas des interactions médiatisées par ordinateur", In E.A. Théodile Lille 3 (Ed.), Actes du colloque international "Faut-il parler pour apprendre ?".
- Self J. (1994). "The role of student models in learning environments", Transactions of the Institute of Electronics, Information and Communication Engineers.
- Self, J. (1988). "Bypassing the intractable problem of student modeling", Proceedings of Intelligent Tutoring Systems Conference, Montreal, 18-24.
- Settouti, L., Prié, Y., Mille, A., Marty, J-C. (2006). "Système à base de traces pour l'apprentissage humain", Colloque international TICE 2006, Technologies de l'Information et de la Communication dans l'Enseignement Supérieur et l'Entreprise.
- Shen, B., Yao, M., Wu, Z., Zhang, Y., Yi, W. (2006). "Ontology based association rules retrieval using protégé tools", IEEE international conference on Data mining workshop.

- Shoham, Y. (1997). "An overview of Agent-oriented Programming Languages", In *Software Agents*, ed J.M. Bradshaw. Menlo Park, Californie. AAAI Press.
- Shortliffe, E.H. (1976). "Computer-Based Medical Consultations: MYCIN", Elsevier/North Holland, New York.
- Siebra, S., Salgado, A.C. , Brézillon, P. , Tedesco, P. (2005). "A learning interaction memory using contextual information". The CONTEXT-05 Workshop on Context and Groupware. Paris, France.
- Skinner, B.F. (1954). "The science of learning and the art of teaching", *Harvard Educational Review*, 24(2), 86-97.
- Stahl, G. (2002). "Groupware goes to school", Dans *Groupware : Design, Implementation and Use CRIWG 2002*, 8th International Workshop on Groupware .
- Stermsek, G., Strembeck, M., et Neumann, G. (2007). "A User Profile Derivation Approach based on Log-File Analysis", *International Conference on Information and Knowledge Engineering*. Las Vegas, Etas-Unis.
- Tanimoto, S. (2005) "Dimensions of Transparency in Open Learner Models", *Proc. Int'l Workshop on Learner Modelling for Reflection, to Support Learner Control, Metacognition and Improved Communic. between Teachers and Learners*, in conj.with AIED, Amsterdam, July, 2005.
- Tchounikine Piere (2002). "Quelques éléments sur la conception et l'ingénierie des EIAH", In: *Actes du GDR I3*, p. 233-245, Cepadues Editions.
- Tepper, M. (2003). "The rise of social software". *netWorker*, 7(3):18-23.
- Trella, M., Conejo, R., Bueno, D., Guzmán, E. (2002). "An autonomous component architecture to develop WWW-ITS". In Brusilovsky, P., Henze, N., Milln, E. (eds.) *Proceedings of the Workshops on Adaptive Systems for Web-Based Education*. Malaga (this volume). <http://citeseer.ist.psu.edu/trella02autonomous.html>.
- VanLehn K. (1988). "Towards a theory of impasse-driven learning", *Learning Issues for Intelligent Tutoring Systems*, H. Mandl & A. Lesgold (Eds.), New York, NY: Springer, pp 19-41, 1988.
- Vassileva, J. J. Greer, G. McCalla, R. Deters, D. Zapata, C. Mudgal, S. Grant (1999). "A Multi-Agent Approach to the Design of Peer-Help Environments", in *Proceedings of AIED'99*, Le Mans, France, July, 1999, 38-45 IOS Press: Amsterdam.
- Vygotsky, L.S. (1978). "Mind and society: The development of higher psychological processes", Cambridge, MA: Harvard University Press.
- Webb, G., Pazzani, M., & Billsus, D. (2001). "Machine learning for user modeling", *User modeling and user-adapted interaction*, 11(19-20).
- Webber, C. (2003). "Modélisation informatique de l'apprenant, une approche basée sur le modèle cKc et la théorie de l'émergence", thèse de doctorat de l'université Joseph Fourier Grenoble 1.

- Weber, G., & Specht, M. (1997). "User modeling and adaptive navigation support in WWW based tutoring systems", In Jameson, A., Paris, C., & Tasso, C. (Eds.), *User Modeling*, Springer-Verlag, Wien 289-300.
- Weber, G., Brusilovsky, P. (2001). "ELM-ART: An adaptive versatile system for Web-based instruction", *International Journal of Artificial Intelligence in Education* 12 (4), Special Issue on Adaptive and Intelligent Web-based Educational Systems, (pp. 351-384).
- Wenger, E. (1987). "Artificial Intelligence and Tutoring Systems", CA: Morgan Kaufmann Publisher, Los Altos.
- Winter M. and McCalla G. (1999). "The emergence of student models from an analysis of ethical decision making in a scenario-based learning environment", proceedings of the seventh international conference on user modeling (UM99), Springer-Verlag, 265-274.
- Wooldridge, M. J., Jennings, N. R. (1995). "Agents Theories, Architectures and Languages : A Survey". In *Intelligent Agents: ECAI-94 Workshop on ATAL*, eds Wooldridge and Jennings. Springer- Verlag. Berlin. pp. 1-39.
- Wooldridge, M., Jennings, N. R., Kinny, D., (1999). "A methodology for agent-oriented analysis and design", In *Proceedings of the Third International Conference on Autonomous Agents*. pp. 69-76, Seattle, USA.
- XTM Authoring Group (2001). "XML Topic Maps (XTM) 1.0: www.TopicMaps.org Specification", 3 March 2001.

Liste des publications personnelles

Publications dans des revues internationales

1. Mediani, C., Djoudi, M. (2007). "An Agent-Based Design for the Learner Behaviour Modelling". *Information Technology Journal (ITJ)*, ISSN: 1812-5638, Vol. 6, No. 5, 681-689.

Communications dans des conférences internationales

2. Mediani C, Djoudi M., (2009) « Modélisation des interactions de l'apprenant à base d'agents ». International Conference On applied Informatics ICAI'09, Bordj Bou Arréridj, Algérie, novembre 15-17.
3. Mediani, C., Djoudi, M. (2007). "Application de la méthodologie MaSE (Multi agent System Engineering) dans la conception d'un outil de mémorisation des traces d'interaction de l'apprenant.". H2PTM'07 : Hypermedias Hypertexts, Products, Tools and Methods International Conference, Hammamet, Tunisie, 29-31 October.
4. Mediani, C., Djoudi, M. (2006). "Un outil de mémorisation des traces de navigation de l'apprenant à base d'agents en apprentissage à distance". 3ème Colloque Euro-Méditerranéen et Africain d'Approfondissement sur la Formation à Distance, CEMAFORAD 3 Sousse'2006, Tunisie, 13-15 novembre.
5. Mediani, C., Djoudi, M. (2006). "Une architecture à base d'agents intelligents pour la modélisation du comportement de l'apprenant dans un environnement d'apprentissage coopératif à distance". 23ème Congrès de l'Association Internationale de Pédagogie Universitaire (AIPU) Monastir 2006, 15-18 mai.
6. Mediani, C., Djoudi, M. (2006). "Conception et réalisation d'un outil de mémorisation des traces de l'apprenant à base d'agents intelligents dans un environnement numérique de travail". Conférence Internationale sur l'Informatique et ses Applications, CIIA'2006, Saida - Algérie, 15-16 mai.
7. Mediani, C., Harbouche, K., Djoudi, M. (2005). "Modélisation de l'apprenant dans un environnement d'apprentissage collaboratif". Congrès International en Informatique Appliquée, CIIA'05, Bordj Bou Arréridj, Algérie., 19-21 Novembre.
8. Harbouche, K., Mediani, C., Djoudi, M. (2005). "Environnement numérique d'apprentissage à distance a base d'agents". Congrès International en Informatique Appliquée, CIIA'05, Bordj Bou Arréridj, Algérie., 19-21 Novembre.

9. Harbouche, K., Mediani, C., (2005). « Environnement Numérique de travail pour un apprentissage coopératif médiatisé à distance », Colloque Euro Méditerranéen pour l'Approfondissement de la formation à distance, CEMAFORAD2, Université de Béjaia, Algérie, 12,13,14 Novembre.

Communications dans des congrès nationaux (avec comité de lecture).

10. Mediani, C., Djoudi, M. (2006). "Architecture à base d'agents pour la modélisation du comportement de l'apprenant en apprentissage à distance". Journées d'étude sur les TIC, JeTIC2006, Bechar, Algérie, 15-16 Avril.

Annexes

Annexe 1 : Partie du code source du système

Partie de l'agent Superviseur de la base de données

```
package pacad.agent;
/*****
Agent which answers all messages from other agents
*****/

import agent.Demande;
import agent.Serveur;
import agent.Reponse;

import jade.core.AID;
import jade.core.Agent;
import jade.core.behaviours.*;
import jade.lang.acl.*;

import jade.domain.DFService;
import jade.domain.FIPAAgentManagement.*;
import jade.domain.FIPAException;

import java.util.*;
import java.text.*;

public class AgentSBDD extends Agent
{
    Demande demande = new Demande("", "", "", "", "", "", "");
    Serveur serveur = new Serveur();
    String message = "";
    protected void setup()
    {
        // Supervisor behaviour
```

```

addBehaviour(new CyclicBehaviour(this)
{
public void action()
{
ACLMessage msg = receive();
String content= "";
if (msg!=null) {
    _analyser(msg);
}
else block();
}
});
}

protected void takeDown()
{
try { DFService.deregister(this); }
catch (Exception e) {}
}

protected void _send(String ss,String agent) {
    ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
    msg.addReceiver(new AID(agent, AID.ISLOCALNAME));
    msg.setLanguage("English");
    msg.setOntology("Weather-forecast-ontology");
    msg.setContent(ss);
    send(msg);
}

protected void _analyser(ACLMessage message1) {
    message = message1.getContent();
    demande = demande.toDemande(message);
    agent.Reponse reponse = new agent.Reponse("", "", "", "", "", "", "");
    if(demande.Drapeau.equals("UN")){
        agent.Reponse uv =serveur.selectOneUV(demande.Contenu2);
        demande.Contenu1 =uv.toString(uv);
        demande.Drapeau = "UN";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("CP")){
        agent.Reponse cpt =serveur.selectOneConcept(demande.Contenu5);

```

```

        demande.Contenu1 =cpt.toString(cpt);
        demande.Drapeau = "CP";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("FO")){
        agent.Reponse[] fonction = serveur.selectFonctions();
        demande.Contenu1 =reponse.toStringTabreponse(fonction);
        demande.Drapeau = "FO";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("DO")){
        agent.Reponse[] document = serveur.selectDocuments();
        demande.Contenu1 =reponse.toStringTabreponse(document);
        demande.Drapeau = "DO";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("GR")){
        agent.Reponse[] groupe=
serveur.selectPersonneGroupe(demande.Contenu2,demande.Contenu4);
        demande.Contenu1 =reponse.toStringTabReponse(groupe);
        demande.Drapeau = "GR";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("PO0")){
        agent.Reponse[] PointENT=
serveur.selectPointsEntree(demande.Contenu2,demande.Contenu3,"","");
        demande.Contenu1 =reponse.toStringTabReponse(PointENT);
        demande.Drapeau = "PO0";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("PO1")){
        agent.Reponse[] PointENT=
serveur.selectPointsEntree(demande.Contenu2,demande.Contenu3,demande.Contenu4,"");
        demande.Contenu1 =reponse.toStringTabReponse(PointENT);
        demande.Drapeau = "PO1";
        _send(demande.toString(demande),demande.Contenu6);
    }
    if(demande.Drapeau.equals("PO2")){
        agent.Reponse[] PointENT=
serveur.selectPointsEntree(demande.Contenu2,demande.Contenu3,"",demande.Contenu4);
        demande.Contenu1 =reponse.toStringTabReponse(PointENT);
    }

```

```

        demande.Drapeau = "PO2";
        _send(demande.toString(demande), demande.Contenu6);
    }
    if (demande.Drapeau.equals("TR0")) {
        agent.Reponse[] Types=
serveur.selectTypeRessourcesConcept (demande.Contenu5,demande.Contenu3,"",
");
        demande.Contenu1 =reponse.toStringTabreponse (Types);
        demande.Drapeau = "TR0";
        _send(demande.toString(demande), demande.Contenu6);
    }
    if (demande.Drapeau.equals("TR1")) {
        agent.Reponse[] Types=
serveur.selectTypeRessourcesConcept (demande.Contenu5,demande.Contenu3,dema
nde.Contenu4,"");
        demande.Contenu1 =reponse.toStringTabReponse (Types);
        demande.Drapeau = "TR1";
        _send(demande.toString(demande), demande.Contenu6);
    }
    if (demande.Drapeau.equals("TR2")) {
        agent.Reponse[] Types=
serveur.selectTypeRessourcesConcept (demande.Contenu5,demande.Contenu3,"",d
emande.Contenu4);
        demande.Contenu1 =reponse.toStringTabReponse (Types);
        demande.Drapeau = "TR2";
        _send(demande.toString(demande), demande.Contenu6);
    }
    if (demande.Drapeau.equals("HIS")) {
        agent.Reponse[] DateHisto =
serveur.selectDateHistorique (demande.Contenu4,demande.Contenu2);
        demande.Contenu1 =reponse.toStringTabReponse (DateHisto);
        demande.Drapeau = "HIS";
        _send(demande.toString(demande), demande.Contenu6);
    }
    if (demande.Drapeau.equals("HIC")) {
        DateFormat date1=new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");
        Date date = null;
        try{
            date = (Date)date1.parse(demande.Contenu3);
        } catch (Exception e) {System.out.println(e);}

        serveur.InsertHistoConcept (demande.Contenu4,demande.Contenu5,date1.for
mat(date));
        demande.Contenu1 ="";
    }

```

```

        demande.Drapeau = "HIC";
        _send(demande.toString(demande), demande.Contenu6);
    }
    if (demande.Drapeau.equals("HIR")) {
        DateFormat date1 = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");
        Date date = null;
        try {
            date = (Date) date1.parse(demande.Contenu3);
        } catch (Exception e) { System.out.println(e); }

        serveur.InsertHistoConcept(demande.Contenu4, demande.Contenu5, date1.format(date));

        demande.Contenu1 = "";
        demande.Drapeau = "HIR";
        _send(demande.toString(demande), demande.Contenu6);
    }
    //...
}
}

```

Un extrait montrant la déclaration du topic variable locale

```

<topic id="cpt_117">
  <baseName id="x208nrl8l5-40">
    <baseNameString>variable locale</baseNameString>
  </baseName>
  <occurrence id="x1ojr0j9c8-58a">
    <instanceOf>
      <topicRef xlink:href="#groupe"/>
    </instanceOf>
    <resourceData>0</resourceData>
  </occurrence>
  <occurrence id="x1ojr0j9c8-58b">
    <instanceOf>
      <topicRef xlink:href="#personne"/>
    </instanceOf>
    <resourceData>0</resourceData>
  </occurrence>
  <occurrence id="x1ojr0j9c8-58c">
    <instanceOf>
      <topicRef xlink:href="#unite"/>
    </instanceOf>
  </occurrence>
</topic>

```



```
    <resourceData>1</resourceData>
  </occurrence>
  <occurrence id="x1ojr0j9c8-58d">
    <instanceOf>
      <topicRef xlink:href="#description"/>
    </instanceOf>
    <resourceData>La variable locale est une variable.</resourceData>
  </occurrence>
</topic>
```

Annexe 2 : Questionnaires

Questionnaire pour identifier quelques caractéristiques et préférences de l'apprenant.

Consignes

Veuillez choisir une seule réponse pour chaque question.

Répondez le plus naturellement possible. Il n'y a pas de bonnes ou mauvaises réponses.

Présentation du cours :

1. Lorsque l'enseignant nous présente le sujet à étudier, il est important pour moi

(a) qu'il le présente étape par étape.

(b) qu'il donne une vue d'ensemble et la relie à d'autres sujets.

2. Lorsque l'enseignant nous présente le cours

(a) je préfère qu'il dessine des schémas ou des graphes.

(b) je préfère qu'il donne beaucoup d'explications verbales ou qu'il écrit un résumé.

3. Pour comprendre les différentes parties d'un nouveau sujet à étudier, est-ce la présentation du sujet par un plan :

(a) m'aide peu.

(b) m'aide beaucoup.

4. Je préfère les cours basés sur

(a) l'expérimentation et l'analyse des données.

(b) des activités intellectuelles.

5. En lisant une leçon

(a) dès que je comprends tous les paragraphes, je comprends le thème général.

(b) dès que je comprends le thème général, je comprends les relations entre les paragraphes.

6. En lisant un texte

(a) j'essaie de réfléchir aux intrigues et les regrouper pour comprendre le thème général.

(b) je termine d'abord la lecture pour comprendre le thème général, ensuite je retourne au texte pour retrouver les intrigues.

7. En lisant une bande dessinée, je me concentre sur

(a) l'image.

(b) le texte des bulles.

8. Si je pense à ce qu'on a fait en classe, je me souviens mieux

(a) des images.

(b) des commentaires de l'enseignant.

9. j'ai tendance à mieux me rappeler d'un conte

(a) si je le vois en vidéo.

(b) si on me le raconte.

10. Au temps libre, je préfère

(a) regarder la télévision.

(b) lire un livre.

11. Lorsqu'on m'explique l'itinéraire pour se rendre à un nouvel endroit, je préfère

(a) qu'on me fasse un plan.

(b) qu'on me donne des indications verbales ou écrites.

12. En lisant, j'aime les textes qui sont exprimés d'une façon

(a) simple, claire et nette.

(b) intéressante, créative et ingénieuse.

13. Lorsque je rédige un texte, j'ai tendance à

(a) commencer par l'introduction puis développer le reste des parties.

(b) développer les différentes parties du texte puis essayer de les assembler.

14. Lorsque je résous un problème de mathématique

(a) je vérifie soigneusement mes calculs en répétant toutes les étapes.

(b) je trouve ennuyeux de vérifier mes calculs et de répéter les étapes.

Travail du groupe :

15. Lorsque j'étudie une nouvelle leçon, je comprends mieux

(a) en parlant à mes camarades.

(b) en réfléchissant seul.

16. Lors d'un travail de groupe

(a) je participe activement et suggère des idées aux membres du groupe.

(b) j'écoute les idées proposées par les autres membres du groupe.

17. A l'école, j'ai l'habitude d'avoir

(a) beaucoup d'ami(e)s.

(b) rarement plus de 2 ami(e)s.

18. Lors de la préparation d'un travail de groupe, je préfère la faire

(a) avec les membres du groupe.

(b) tout(e) seul(e).

19. Lors de la préparation d'un travail en groupe, je préfère chacun

(a) apporte ses idées dès le départ.

(b) doit réfléchir seul(e) et ensuite on compare nos idées en groupe.

L'évaluation :

20. Après la réalisation d'un travail en groupe, je préfère que la note d'évaluation

(a) soit la même pour tous les membres du groupe.

(b) soit individuelle pour chaque membre du groupe.

Apprentissage par problème :

21. Lors de la résolution d'un problème

(a) j'élabore la solution étape par étape.

(b) j'éprouve des difficultés à trouver les étapes qui mèneraient à la solution.

Les caractéristiques :

22. Je considère comme une qualité d'être

(a) appliqué(e) dans mon travail.

(b) débrouillard(e) dans la façon de faire mon travail.

23. J'apprends

(a) à un rythme régulier.

(b) Je peux être embrouillé puis soudainement avoir un déclic.

24. Pour aller à un nouvel endroit

(a) l'heure venue, je pars et j'essaie de trouver le chemin.

(b) quelques minutes avant le départ, je réfléchis au chemin en utilisant un plan.

Annexe 3 : Fiches d'évaluation

Le profil de l'apprenant

1. Nom : Prénom :

2. Age :ans

3. Sexe : ☐ féminin ☐ masculin

4. Numéro d'étudiant :

5. Expérience post-bac :

6. Connectez-vous à l'environnement :

☐ toujours ☐ souvent ☐ régulièrement ☐ peu ☐ jamais

7. Votre niveau est-il :

☐ très bon ☐ bon ☐ moyen ☐ faible ☐ très faible

Utilisation d'Internet

8. Pensez-vous savoir utiliser Internet :

☐ très bien ☐ bien ☐ moyennement ☐ mal ☐ très mal

9. Depuis quand utilisez-vous Internet :

☐ moins d'un an ☐ entre un et deux ans ☐ plus de deux ans

10. A quelle fréquence utilisez-vous Internet :

☐ jamais ☐ moins de 3 jours par mois ☐ 1 jour par semaine

☐ 2 à 3 jours par semaine ☐ plus de 4 jours par semaine

11. Quel navigateur avez-vous l'habitude d'utiliser :

☐ Internet Explorer ☐ Firefox ☐ Netscape ☐ autre :

Pour toutes les affirmations suivantes, dites si vous êtes d'accord ou non.

Prise en main de l'environnement

Questions	pas d'accord	un peu d'accord	d'accord
12. Vous avez l'habitude de ce type d'environnement.	☐	☐	☐
13. L'utilisation générale de l'environnement est simple.	☐	☐	☐
14. Vous avez fait des erreurs de manipulation.	☐	☐	☐
15. Vous avez effectué une recherche sur des choses qui n'étaient pas à l'endroit attendu.	☐	☐	☐
16. L'utilisation des cours.	☐	☐	☐
17. L'utilisation des exercices.	☐	☐	☐
18. L'utilisation des liens pour accéder à d'autres sites.	☐	☐	☐
19. L'environnement e-pacad offre une bonne façon de mettre des ressources pédagogiques à la disposition d'apprenants.	☐	☐	☐

Présentation de l'information

Questions	pas d'accord	un peu d'accord	d'accord
La navigation dans l'environnement e-pacad			
20. Vous avez navigué dans l'arborescence pour satisfaire vos	☐	☐	☐

recherches.			
21. Vous avez utilisé les points d'accès pour satisfaire vos recherches.	☐	☐	☐
22. Vous avez apprécié l'utilisation d'une arborescence pour présenter les informations.	☐	☐	☐
23. Vous avez accéder aux relations autres que la relation de spécialisation.	☐	☐	☐
Les espaces de travail			
25. Vous avez apprécié d'avoir différents espaces de travail.	☐	☐	☐
26. Les différents espaces de travail offrent un bon moyen d'échange et de partage de ressources au sein d'un groupe.	☐	☐	☐
27. L'accès à un espace de travail se fait de façon simple.	☐	☐	☐
28. Vous avez accédé à l'espace de la formation.	☐	☐	☐
29. Vous avez accédé à votre espace privé.	☐	☐	☐
30. Vous avez accédé à votre espace du groupe.	☐	☐	☐
Approche par concepts			
31. Vous avez apprécié la décomposition de la formation en un ensemble de concepts.	☐	☐	☐
Points d'accès			
32. Vous avez utilisé les points d'accès.	☐	☐	☐
33. Vous avez trouvé que ces points d'accès satisfirent bien les recherches que vous avez effectuées.	☐	☐	☐
Les ressources pédagogiques			
34. Vous avez apprécié l'indexation des ressources par des concepts.	☐	☐	☐
35. L'ajout de ressource se fait de manière simple.	☐	☐	☐

36. Vous avez ajouté des ressources dans votre espace privé.	☐	☐	☐
37. Vous avez ajouté des ressources dans votre espace du groupe.	☐	☐	☐
L'historique			
38. Vous avez apprécié l'utilisation de l'historique pour sauvegarder vos actions.	☐	☐	☐
39. Vous avez trouvé que la visualisation de votre historique est très utile.	☐	☐	☐
Le modèle d'inférence			
40. Vous avez apprécié l'utilisation d'un modèle d'inférence pour mesurer votre niveau de connaissance pour chaque concept.	☐	☐	☐
41. Vous avez trouvé que la visualisation des valeurs mesurant votre niveau de connaissance pour chaque concept est très utile.	☐	☐	☐
42. Vous avez compris la signification du terme Visité.	☐	☐	☐
43. Vous avez compris la signification du terme Evalué.	☐	☐	☐
44. Vous avez compris la signification du terme Inféré.	☐	☐	☐
L'outil de communication (le chat)			
45. Vous avez trouvé que le chat est un bon moyen d'échange entre les membres du groupe.	☐	☐	☐
46. Le chat vous a aidé dans votre travail de groupe.	☐	☐	☐
47. Vous aurez plus utilisé le chat si vous ne pouviez pas rencontrer les membres de votre groupe.	☐	☐	☐

Général

Questions	pas d'accord	un peu d'accord	d'accord
48. Je vais continuer à utiliser l'environnement.	☐	☐	☐
49. Je vais conseiller l'environnement à d'autres amis.	☐	☐	☐
50. Je vais utiliser l'environnement pour préparer mes examens.	☐	☐	☐

Avis général

51. Donnez votre avis général sur l'environnement :

.....

.....

.....

.....

Suggestion

52. Si vous avez des suggestions afin d'améliorer l'environnement :

.....

.....

.....

.....