

UN SYSTEME MULTI-AGENTS POUR L'E.I.A.O

OKBA KAZAR

Département d'Informatique , Faculté des sciences et sciences de l'ingénieur
Université de Biskra
07000, ALGERIA, o.kazar@caramail.com

RESUME

Le travail présenté dans ce papier a pour but de réaliser un tuteur intelligent (système d'enseignement intelligent assisté par ordinateur) en utilisant une approche multi-agents. Cette approche, issue à partir des recherches abordées en intelligence artificielle distribuées (IAD), vise à répartir un problème sur un ensemble d'entités appelées agents qui communiquent entre eux pour coordonner leurs comportements intelligents et coopérer afin de résoudre ce problème. Le problème à résoudre dans cet article est celui de l'enseignement des principaux concepts de base de données où les différents types de connaissances sont issus de trois sources : l'expert pédagogique, l'expert du domaine à enseigner et l'étudiant. Pour cela nous avons proposé un système multi-agents dont nous avons distribué les différents types de connaissances sur un ensemble d'agents communicants par partage d'information via le mécanisme de tableau noir.

1. INTRODUCTION GENERALE

Le jour où n'importe quel humain pourra converser de façon naturelle avec une machine intelligente sera le jour béni et tant attendu des informaticiens, et de tous les scientifiques [WAF 95]. Pour aboutir à ce but, les chercheurs ont créés une discipline particulière dite *intelligence artificielle* (I.A) qui vise à comprendre la nature de l'intelligence en construisant des programmes d'ordinateurs imitant l'intelligence humaine [NIL 80]. L'approche classique de l'intelligence artificielle, qui est relative à la conception de système à base de connaissances, ne convenait pas aux applications progressives qui exigent une taille importante des connaissances, diversité des connaissances [BOU 94] [BOU 97].

De même dans la réalité, les individus travaillent généralement en groupe [WOO 94]. Cela a conduit à l'apparition d'une nouvelle approche de l'intelligence artificielle appelée *intelligence artificielle distribuée* (I.A.D). Elle repose sur le principe de la distribution de l'intelligence entre un ensemble d'entités qui coopèrent [SEA 72] [SEA 90] pour atteindre un objectif global que chacune entre elles ne peut le réaliser individuellement [BRI 91]. Une extension des systèmes d'I.A.D est proposée où les composants, qui possèdent une certaine autonomie [GOL 77], doivent être dotés des capacités de perception et d'action sur leur environnement, on parle alors d'agent [JEN 93] [JEN 95] et par conséquent les systèmes multi-agents [SHO 90]. Ces systèmes deviennent indispensables dans plusieurs domaines d'applications, dont l'enseignement intelligent

assisté par ordinateur (E.I.A.O).

Un système d'E.I.A.O ou système de tuteur intelligent autrement dit (S.T.I), est un logiciel qui utilise les techniques d'I.A et destiner à des fonctions d'éducation, ses différents types de connaissances, issues de moins de trois sources : l'expert pédagogique, l'expert du domaine à enseigner et l'étudiant, sont naturellement hétérogènes.

Le résultat est que les S.T.I sont complexe ce qui rend difficile le développement total de tel système, pour cela on a besoins d'une approche conceptuelle qui facilite la distribution de connaissances d'une part, et respecte les principes du génie logiciel d'autre part [DAH 99]. Face à une telle situation, l'approche multi-agent semble très prometteuse pour développer un S.T.I [BUR 91].

2. ENSEIGNEMENT ASSISTE PAR ORDINATEUR

Dans n'importe quel domaine, l'enseignement est l'utilisation des méthodes et de quelques techniques afin de transférer les différents types de connaissances d'un enseignant à un étudiant. Dans le but d'automatiser cette approche, c'est à dire simuler cet enseignant en un programme qui tourne sur un ordinateur, les chercheurs ont donné naissance à un nouveau domaine de recherche en informatique nommé «Enseignement Assisté par Ordinateur» (E.A.O).

Au fils du temps, et par l'apparition de l'intelligence

artificielle, les chercheurs ont tenté d'améliorer cette nouvelle approche (EAO) en introduisant les techniques de l'intelligence artificielle, ce qui a conduit à l'apparition de nouvelle approche d'E.A.O qui est l'Enseignement Intelligent Assisté par Ordinateur (E.I.A.O) ou aussi «Tuteurs Intelligents».

Les systèmes d'E.I.A.O ont pris plusieurs formes mais, par rapport à l'E.A.O, on a séparé les principales composantes du logiciel de telle sorte que cela donne à l'élève et au système une flexibilité de l'environnement qui s'approche davantage de ce qui se passe quand un étudiant et un enseignant s'assoient face à face et tente d'apprendre et d'enseigner ensemble.

La figure 1 [BRU 97] donne un aperçu des principales étapes qui ont marqué l'histoire de l'informatique en éducation : les années cinquante et l'enseignement programmé puis l'E.A.O, l'apport des techniques d'intelligence artificielle au début des années soixante-dix, d'un côté dans le prolongement des recherches issues de l'enseignement programmé (E.A.O et I.A) et de l'autre côté, dans le développement de LOGO et des micro-mondes.

La poursuite des approches précédentes avec respectivement les tuteurs intelligents et les environnements d'apprentissage ouverts dans les années quatre-vingt, ainsi que l'émergence des outils généraux accompagnant les diffusions des micro-ordinateurs. Enfin les tentatives de synthèse dans les environnements interactifs contrôlés avec l'apport des techniques d'hypertextes et la diffusion du multimédia jusqu'à l'explosion récente des réseaux comme Internet et l'arrivée des interfaces multimodèles et de la réalité virtuelle [BRU 97].

Dans ce schéma, la première colonne réfère à des courants qui ont privilégié l'enseignement, c'est à dire pour lesquels la machine est principalement amenée à jouer le rôle du maître (de l'enseignement programmé aux tuteurs intelligents).

La troisième colonne rend compte des recherches dans lesquelles l'ordinateur est un moyen permettant aux apprenants d'effectuer construction, exploration et découvertes. La colonne centrale correspond aux tentatives de synthèse de ces deux approches, essayant de concilier un certain guidage par la machine dans des environnements largement contrôlés par les apprenants.

Bien entendu, un tel découpage est schématique et réducteur et ne correspond que très imparfaitement à ce qui s'est passé, les idées ayant traversé les différents courants [BRU 97].

L'enseignement assisté par ordinateur (E.A.O) apparaît véritablement comme le mariage de l'enseignement programmé fondé sur des concepts fondamentaux de la cybernétique et au développement de la psychologie du comportement, et de l'informatique.

Il apparaît aussi comme une technologie éducative

basée sur les trois composantes principales de l'action éducative : fixation des objectifs, sélection des moyens et des méthodes, et évolution des résultats [DEP 87]. On aboutit au model classique d'une session d'E.A.O. Elle se déroule de manière cyclique :

- Présentation d'information et d'une question
- Réponse de l'apprenant
- Analyse de la réponse
- Continuation ou branchement à une autre partie de cours

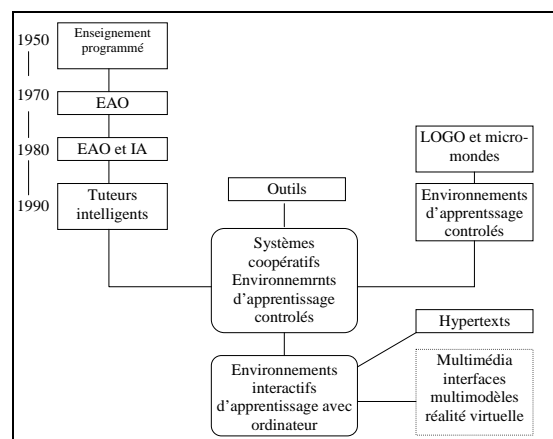


Figure 1 : Principaux courants en informatique et éducation

L'EAO présente deux inconvénients majeurs :

- La préparation du cours requiert un temps considérable et les résultats obtenus et très flexibles (difficilement modifiables).
- Changer un ensemble de questions ou des réponses implique un gros travail de réécriture du programme.

L'E.I.A.O est une tentative de développer l'E.A.O par l'application des techniques d'intelligence artificielle. Il a évolué à partir du domaine d'informatique et non à partir de la psychologie ou de la technologie éducative.

La réalisation d'un système E.I.A.O est en effet faire réaliser un tuteur intelligent qui est une tentative pour concevoir un programme tournant sur un ordinateur capable de se conduire d'une façon quelle serait jugée comme un bon enseignement si elle était faite par une personne humaine.

Pour cela il faut concevoir des systèmes ayant une triple expertise, celle du domaine à enseigner, celle de l'enseignement et celle des compétences et connaissances correctes et erronées des élèves.

Généralement, le programme SCHOLAR [CAR 70] et considéré comme le premier tuteur intelligent conçu pour l'enseignement de connaissances factuelles sur la géographie de l'Amérique du sud, son originalité réside dans le type de dialogue instauré avec l'apprenant.

3. LES COMPOSANTES D'UN SYSTEME D'E.I.A.O

Comme dans n'importe quel autre système éducatif, les composantes sont : la connaissance devant être enseigné, l'approche pédagogique et un mécanisme permettant de comprendre ce que l'élève sait et ce qu'il ne sait pas. Dans un système d'E.I.A.O, on appelle ces composantes module de résolution de problème ou d'expertise, modèle de l'étudiant et modèle tutorial [GIL 90].

La conception dominante organise le système d'E.I.A.O (tuteurs intelligents) autour de trois modules : le domaine, le modèle de l'étudiant et l'expert pédagogique.

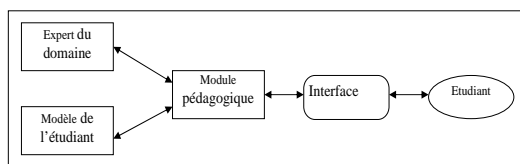


Figure 2 : Modèle général d'un système d'E.I.A.O

3.1 Module d'expertise

Un module d'expertise «ou de résolution de problème» consiste en une base de connaissance que le système tente d'enseigner à l'étudiant. Parce que le module d'expertise est utilisé pour générer un contenu éducatif et pour évaluer la performance de l'étudiant, le domaine de connaissance devra être organisé à l'intérieur de la structure d'un programme d'ordinateur, afin d'assurer une manipulation flexible des données durant le processus d'enseignement et d'apprentissage. Les méthodes représentatives de l'intelligence artificielle utilisées pour organiser la base de connaissance dans le module d'expertise incluent le développement des réseaux sémantiques, l'application des systèmes de règles de production, les représentations procédurales et la construction des scripts ou de «frame» [GIL 90].

3.2 Le modèle de l'étudiant

Le modèle de l'étudiant est utilisé pour évaluer l'état de connaissance de l'étudiant qu'il utilise pour arriver à son état actuel de connaissances.

Dans ce contexte, un modèle de l'étudiant est une structure de données, au sens informatique, qui caractérise, pour les systèmes d'enseignement, l'état d'un sous-ensemble des connaissances de cet étudiant. Il va se définir par l'écart entre la propre connaissance (supposées) de l'étudiant et les connaissances ciblées, enjeu de l'apprentissage, telles qu'elles sont représentées dans le système. La façon de concevoir cet écart conduit à distinguer deux grandes classes de modèle :

- *Les modèles d'expertise partielle ou de superposition* (overlay model, [GOL 77]) : dans lesquels les connaissances de l'élève n'est qu'un sous-ensemble de la connaissance cible. L'idée sous-jacente à ce type de modèle est que l'élève présente des lacunes ou des connaissances encore mal assurées, en quelque sorte des faiblesses, qu'il s'agit d'identifier pour lui permettre de progresser. L'objectif des systèmes d'E.I.A.O est alors de compléter les connaissances de l'élève pour qu'il acquière l'ensemble de connaissance précisée.
- *Les modèles différentiels* : cette approche veut représenter les sous tâches mal apprises par l'étudiant, sous tâche qu'ils ne sont pas des sous-ensembles fondamentaux de la connaissance de l'expert, mais plutôt des variantes de cette connaissance. Ce modèle est proposé par BROWN et BURTON en 1978.

3.3 Le module tutorial

Un module tutorial est l'ensemble de spécification concernant la matière à présenter par le système ainsi que la façon et le moment où il devrait la présenter. Dans les systèmes existants d'E.I.A.O, les stratégies pédagogiques pour spécifier la présentation de matériel éducatif sont basées sur deux méthodes : La méthode SOCRATIQUE est la méthode de l'entraîneur [GIL 90]. La méthode SOCRATIQUE fournit aux étudiants une série de questions les guidant à travers le processus de «debugging» de leur propre conception erronée. Durant ce processus on suppose que les étudiants résonnent à partir de ce qu'ils savent et de ce qu'ils ne savent pas et qu'ils modifient ainsi leurs conceptions. La méthode de l'entraîneur place l'étudiant dans un environnement où il s'engagera dans des activités comme les jeux informatisés pour maîtriser des habilités associées et des capacités générales de résolution de problèmes.

4. LA DIFFERENCE ENTRE L'E.A.O ET L'E.I.A.O

L'E.A.O traditionnel a été développé par les chercheurs en éducation et par les formateurs afin de résoudre leurs problèmes courants par l'utilisation de la technologie informatique. Par contraste, l'E.I.A.O a été utilisé au début surtout par les informaticiens pour explorer les capacités des techniques d'IA dans les processus d'apprentissage et d'enseignement, c'est pourquoi les projets d'E.I.A.O se sont concentrés surtout sur les aspects techniques du système (représentation de connaissance, dialogue en langage naturel, mécanisme d'inférence, etc...) plutôt que la pédagogie ou de la matière [GIL 90]. Le tableau

suivant résume les principales différences entre l'E.A.O et l'E.I.A.O.

Tableau 1 : Comparaison entre l'E.A.O et les tuteurs intelligents (E.I.A.O)

Sujet	E.A.O	E.I.A.O
Buts des concepteurs	Développer des systèmes pédagogiques efficaces	Exploration des possibilités des techniques d'IA en pédagogie
Bases théoriques	Théories de l'apprentissage et principes pédagogiques	Sciences cognitives.
Principes pédagogiques	Divers : exposées, exercices, exploration et découverte.	Approche par la découverte et l'activité de l'élève.
Modèle de l'étudiant	Jugements binaires des réponses procédures prescrites, méthodes quantitatives	Etudiant comme système à traiter les connaissances Evaluation qualitative
Formats de présentation	Divers : présentateur, exercices, jeux, Simulateur.	Surtout des présentateurs interrogeables, et des jeux
Matériels et logiciels	Micro, mini et gros ordinateurs et langages généraux.	Station de travail d'IA, langages LISP et PROLOG.

5. UN MODELE D'ENSEIGNEMENT ADAPTATIF

Dans ce qui suit, on présente un modèle conceptuel de l'enseignement adaptatif. Ce modèle fut à l'origine présentée par SEIDEL et des associés. Il illustre les composantes qu'un système d'E.A.O ou d'E.I.A.O devrait posséder et de quelle manière ces composantes devraient être reliées dans le déroulement du processus pédagogique. Le modèle montre aussi quelle capacité d'auto-amélioration ou de mise à jour le système devrait pouvoir disposer.

Comme le montre la figure 3, le modèle sépare le processus pédagogique en trois étapes : l'entrée, les échanges et le résultat. L'analyse des caractéristiques des données de l'étudiant constitue l'étape de l'entrée. Ces caractéristiques contiennent non seulement «l'histoire» de ses réponses, mais aussi des caractéristiques antérieures, comme des informations sur les aptitudes de l'étudiant (mesures d'aptitudes intellectuelles, style cognitif et style d'apprentissage, stage de développement cognitif, connaissances antérieures). Les caractéristiques des données de l'étudiant, particulièrement celles qui ont trait à la progression à l'intérieur de la leçon, seront mises à jour à partir des informations d'évaluation de la performance (les mesures de résultat) [GIL 90]. L'étape des échanges est celle au cours de laquelle ont lieu les interactions entre le système et la

machine. A partir des caractéristiques des données, le système choisit des problèmes ou des explications à présenter. Le système évalue la réponse de l'étudiant au problème donné. L'évaluation de la réponse fournit de l'information pour diagnostiquer les besoins d'apprentissage de l'étudiant et pour mesurer le niveau de performance globale sur la tâche à accomplir.

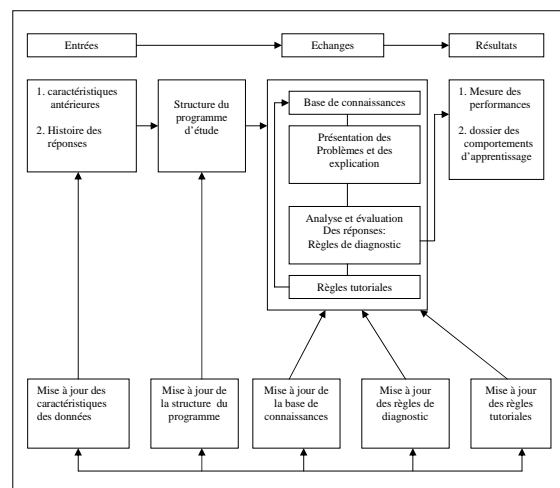


Figure 3 : Un modèle d'enseignement adaptatif (d'après SEIDEL)

Les besoins d'apprentissage sont alors inférés à partir des règles de diagnostic. Finalement, le système choisit de nouveaux affichages (présentation et question) pour l'étudiant en fonction de règles tutorielles.

L'étape de résultat consiste principalement en une évaluation de la performance. Celle-ci peut inclure non seulement le niveau de performance globale face à la tâche à accomplir. Mais aussi l'analyse de tous les comportements d'apprentissage noté par le système. D'après l'évaluation de la performance, les composantes pédagogiques seront modifiées ou mises à jour. Celle-ci devant être mise à jour peut comprendre les contenus de la base de connaissances, les stratégies pédagogiques, les règles de diagnostic et les règles tutorielles, la structure du cours ou du programme et les caractéristiques des données.

6. PRESENTATION DU MODELE

La phase la plus importante dans n'importe quel processus de développement d'un logiciel est la conception qui est définie comme «le processus d'application de divers principes et techniques pour définir un système dans les détails suffisantes afin de permettre sa réalisation physique» [KAZ 01c]. Les systèmes multi-agents [FRA 99] [GAS 90] visent à résoudre un problème donné d'une manière intelligente par la coopération d'un groupe d'agents

[KAZ 00c][KAZ 01a][AUS 66]. Pour cela nous allons présenter une architecture adéquate de notre SMA à communication par partage d'information afin d'atteindre le but global du système qui est l'E.I.A.O de quelques concepts des bases de données [EAR 96]. La figure 4 présente notre système en utilisant le paradigme multi-agents [FER 88] [FER 95] et qui fait distribuer l'expertise sur un ensemble d'agents qui modélisent les différents modules de l'E.I.A.O (modèle pédagogique, module du domaine et le modèle de l'étudiant) [BEC 60]. Notre système est constitué de :

- une interface pour la communication entre l'apprenant et le système.
- un tableau noir qui est une structure de données pour assurer la communication entre les différents agents du système.
- un ensemble d'agent qui modélisent les différents modèles de l'E.I.A.O :
 - Le modèle pédagogique est modélisé par un agent tutorial.
 - Le modèle d'expert est modélisé par deux agents :
 - Un agent qui présente des informations à l'apprenant (informateur).
 - Un agent qui évalue l'apprenant à partir de sa solution d'un exercice (évaluateur).
- une structure de données qui modélise l'étudiant, elle contient l'état actuel de
- connaissances de l'apprenant.
- un contrôleur qui assure la coopération entre les agents.
- une base de cours partagé entre l'informateur et le tuteur.

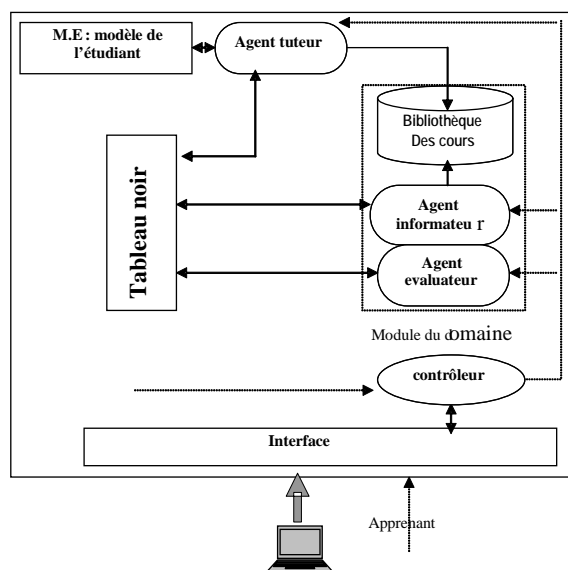


Figure 4 : Architecture globale du système

7. FONCTIONNEMENT DES COMPOSANTS DU SYSTEME

7.1 L'interface

L'interface est un outil important dans n'importe quel système informatique, elle joue le rôle d'un intermédiaire dans la communication homme / machine. Les qualités souhaitées de l'interface sont : la souplesse, l'efficacité et la commodité d'emploi. Dans les systèmes d'enseignement l'interface doit disposer d'un mécanisme de dialogue interactif avec l'apprenant pour permettre à celui-ci de bien comprendre ce qu'il est entraîné d'apprendre.

7.2 Les agents du système

7.2.1 Agent pédagogique

Le modèle pédagogique est le mécanisme qui va transmettre de manière efficace les connaissances de l'enseignant [SPE 86] (dans notre cas le système) à l'apprenant. Il existe deux méthodes d'apprentissage : la méthode SOCRATIQUE et la méthode de l'entraînement [BRU 97]. Notre modèle pédagogique place l'étudiant dans un environnement d'apprentissage. Ainsi, pour un tel système nous avons pris en considération les points suivants :

◆ Les tâches à accomplir

- Présenter un cours.
- Déterminer le point suivant à étudier.
- Consulter et augmenter l'état de connaissances de l'étudiant.

◆ Fonctionnement et architecture interne

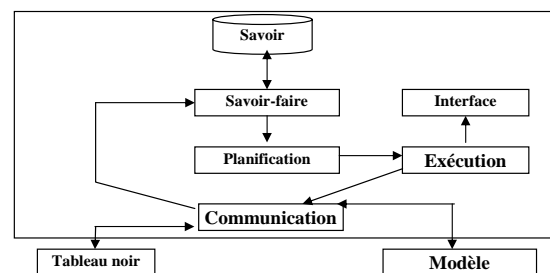


Figure 5 : Fonctionnement et architecture interne de l'agent pédagogique

◆ Le savoir

Les connaissances de cet agent sont présentées sous formes de règles de production qui sont :

- **Si** l'étudiant est un nouveau apprenant **alors** présenter le premier cours.
- **Si** l'étudiant est déjà suivi un cours **alors** {Déterminer le point d'arrêt dans la dernière session de cour , Présenter le point suivant à étudier. }
- **Si** l'étudiant a terminé un paragraphe **alors** augmenter son état de connaissance.

- Si l'étudiant a terminé un cours **alors** lancer une requête d'évaluation.

7.2.2 Agents du domaine

Le modèle du domaine va présenter les différentes expertises du domaine à enseigner, ce modèle est composé de deux agents :

- **Agent informateur**
 - ◆ **Tache à réaliser**
 - Présenter des informations sur un concept donné.
 - Donner des rappels cours.
 - ◆ **Le savoir**

Puisque cet agent joue le rôle d'un informateur, c'est à dire présenter une information sur l'un des concepts appartenant à un paragraphe qui lui-même appartient à un cours, il est préférable de représenter cette hiérarchie par un arbre qu'est un graphe dans lequel les nœuds figurent les objets (cours ou paragraphe) ou des concepts. Par conséquent la représentation du savoir de cet agent est illustré dans la figure 6.

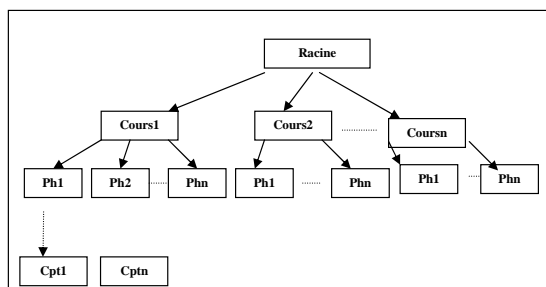


Figure 6 : Représentation de connaissances de l'informateur

◆ Fonctionnement et Architecture interne

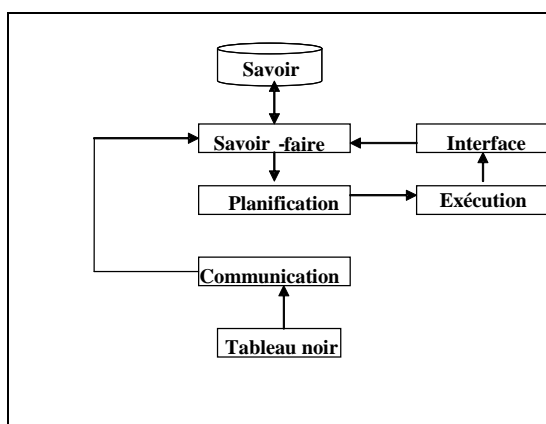


Figure 7 : Fonctionnement et architecture interne de l'agent informateur

Puisque les connaissances de cet agent sont représentées sous forme d'arbre, son module de raisonnement doit être un navigateur qui parcourt le

graphe en cherchant les informations possibles concernant un concept donné.

➤ Agent évaluateur

L'objectif de cet agent est d'évaluer l'étudiant à partir de la solution de ce dernier sur l'exercice posé par cet agent [SAL 97]. Cet agent se constitue de trois agents :

- un agent annonceur d'exercices.
- un agent correcteur.
- un agent explicateur.
- ◆ **Taches à réaliser**
 - Énoncer un exercice.
 - Présenter une solution de cet exercice.
 - Évaluer la solution de l'étudiant.
 - Explication d'évaluation.
- ◆ **Le savoir d'agent annonceur d'exercices**

Il dispose d'une librairie d'exercices avec leurs solutions et une représentation interne de chaque solution d'exercice. Il peut aussi donner une représentation interne de la solution introduite par l'étudiant.

◆ Le savoir d'agent correcteur

La correction de la solution de l'étudiant sera faite en le donnant une note, et pour arriver à une évaluation proche à un enseignant réel, nous avons proposé trois types d'exercices à évaluer :

Première forme : l'exercice se compose d'un ou plusieurs questions dont la réponse sera présenter sous forme de choix.

Exemple : Quelle est la formule qui s'adapte le mieux à DBASE ?

C'est un bon SGF.

C'est un bon SGBD.

C'est une bonne base de données.

Deuxième forme : dans ce type d'exercice, la solution sera un ensemble de mots qui représente soit une entité, un attribut, une association ou une relation.

Exemple : à partir du texte suivant, on vous demande de lister les entités et les associations qui peuvent exister entre elles :

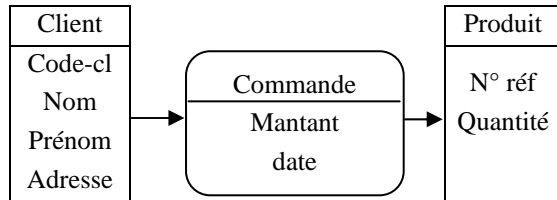
Le fichier « PASSAGERS » contient des informations sur le nom, prénoms, adresse et nationalités des passagers.

Le fichier « VOLS » est associé aux numéros de vols, heures du départ et d'arrivée de chaque vol, ainsi que le nom d'aéroport de départ et d'arrivée.

Enfin, dans le fichier « AVION » on stocke le type de chaque avion, son année de mise en service et les éventuelles réparations effectuées sur celui-ci.

Troisième forme : dans ce cas, la solution sera sous forme de schéma relationnel.

Exemple : à partir de la modélisation conceptuelle suivante, donner les schémas relationnels.



Les connaissances de cet agent seront présenter sous forme de règles de production, et nous avons les trois méta-règles suivantes :

1. **Si** exercice est de 1^{er} forme **alors** activer les règles de 1^{er} forme.
2. **Si** exercice est de 2^{eme} forme **alors**
 - Initialiser la note (N) à 0.
 - Calculer la note ($N_c=20/NB$ entités) .
 - Activer les règles de 2^{eme} forme. }
3. **Si** exercice est de 3^{eme} forme **alors**
 - Initialiser la note (N) à 0.
 - Calculer la note ($N_c=20/NB$ de relation).
 - Analyse syntaxique de chaque schéma relationnel.
 - Activer les règles de 3^{eme} forme. }

Les règle de 1^{er} forme :

1. **Si** NB de choix \geq NB de choix correct **alors**
 - Initialiser la note (N) à 20.
 - Calculer la note (Ns) à soustraire.
 - Pour chaque choix incorrect faire $N=N-Ns$.}
2. **Si** NB de choix $<$ NB de choix corrects **alors**
 - Initialiser la note (N) à 0.
 - Calculer la note à additionner (Na).
 - Pour chaque choix correct faire $N=N+Na$.}
3. **Si** NB de choix = NB de choix donnés **alors** la note (N) est 0.

Les règles de 2^{eme} forme :

1. **Si** entité est correcte **alors** { $N=N+N_c/2$. pour chaque attribut correct de l'entité faire $N=N+N_c/(2*NB$ attributs). }

les règles de 3^{eme} forme :

1. **Si** syntaxe des schémas relationnels est correcte **alors** {pour chaque relation correcte faire $N=N+NC/2$.
 - Pour chaque attribut correct de la relation faire $N=N+N_c/(2*NB$ attributs).}

◆ **le savoir d'agent explicateur**

A chaque règle du correcteur est associée

une explication.

◆ **Fonctionnement et architecture interne de l'évaluateur**

Comme il est illustré dans la figure si dessus, l'évaluateur se constitue de trois sous agents, l'un à pour but de fournir l'exercice et sa solution a l'apprenant d'une part, d'autre part, il donne la représentation interne de la solution de l'étudiant et celle du système au sous agent correcteur pour que ce dernier puisse les comparer.

Le troisième sous agent explique la manière d'évaluation. Les trois sous agents doivent coopérer pour accomplir les tâches de l'agent père (évaluateur).

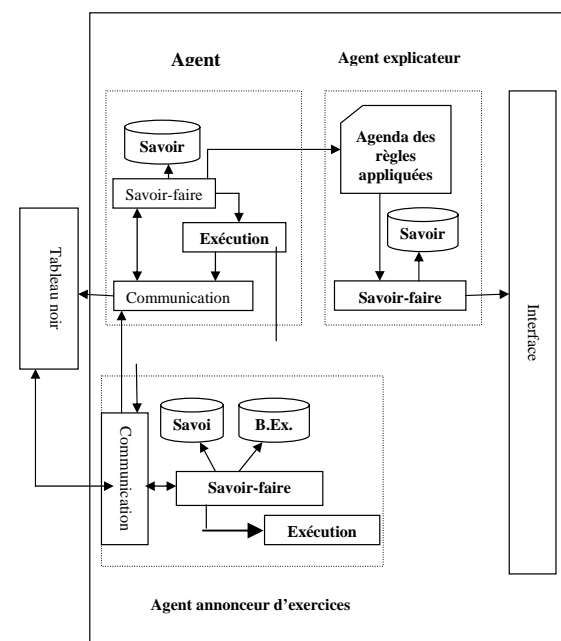


Figure 8 : Fonctionnement et architecture interne de l'agent évaluateur

7.3 Modèle de l'étudiant

Pour concevoir le modèle de l'étudiant qui contient l'état global de connaissance de l'étudiant et son historique dans le processus d'apprentissage, nous avons utilisé un fichier qui contient la liste de tous les étudiants qui ont déjà suivi des sessions de cours et leur état de connaissances.

7.4 Tableau noir

Le tableau noir est l'un des composantes principales dans les systèmes multi-agents à communication par partage d'information, il a pour but d'assurer les différentes communications et échanges entre les agents du système. Dans notre système, le tableau noir est une structure de données qui représente les

8.2.2 Apprentissage

Il Permet d'enseigner à un apprenant les principaux concepts de base de données, la fiche correspondante est illustré comme suit:

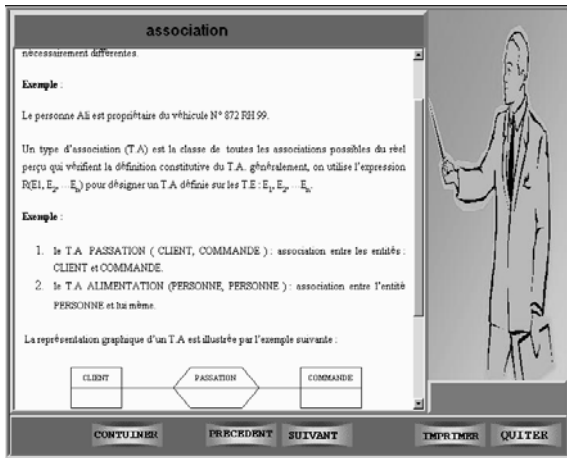


Figure 11 : Apprentissage

Cette fiche possède des caractéristiques d'une page *web* tel que le parcours dans les paragraphes du cours à présenter, et pour que le logiciel devienne hautement interactif, nous avons essayé d'intégrer des techniques multimédia tel que le *son*.

8.2.3 Consultation

Permet de chercher le code d'un apprenant en introduisant son nom comme une clé de recherche. La fiche correspondante est comme suit:

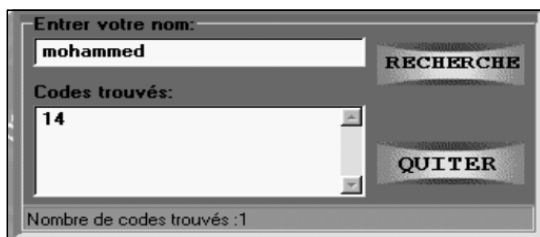


Figure 12 : Consultation

8.3 Agent informateur

8.3.1 Information

Cela consiste à donner des informations à un apprenant d'un concept donné [WEG 87]. L'utilisateur a le choix entre chercher des informations par contenus où il sélectionne l'un des trois cours :

- introduction générale aux bases de données et le SGBD
- le modèle entité-association
- le modèle relationnel.

Si le cours est choisi, tous les paragraphes appartenant à ce cours sont disponibles, puis il sélectionne un concept à chercher.

Il peut chercher les informations par index où il suffit d'écrire le concept ou le localiser dans la liste de tous les concepts du module. La figure 13 illustre le procédé de recherche d'information.

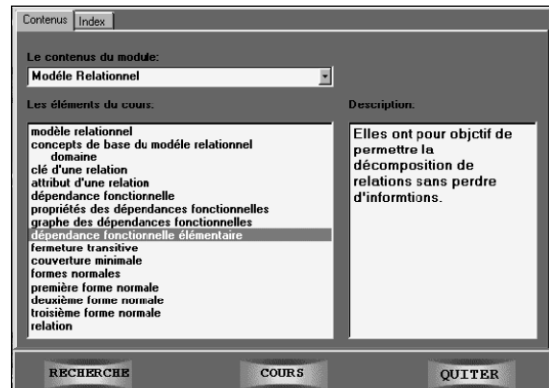


Figure 13 : Information

8.3.2 Rappel

Il permet de faire des rappels de cours à un apprenant en lui donnant le choix des paragraphes à rappeler qu'il a déjà suivi (fig.10).

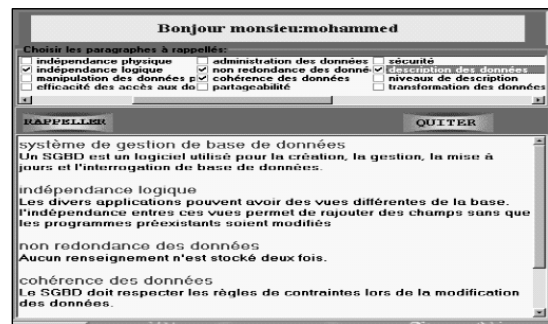


Figure 14 : Rappel

8.4 Agent évaluateur

8.4.1 Evaluation

Permet d'évaluer la solution d'un apprenant à un exercice choisi. Pour cela on a les trois fiches suivantes :

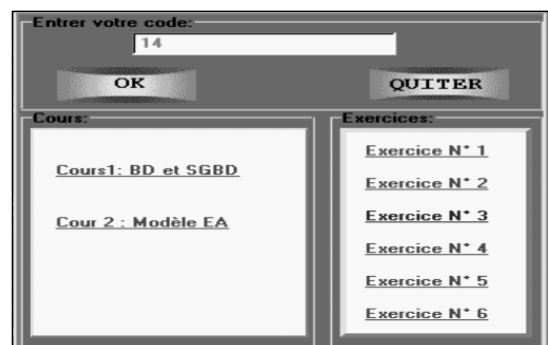


Figure 15 : Evaluation

Cette fiche a pour but de présenter une série d'exercices à l'étudiant selon le choix du cours. Un simple click sur l'un des exercices présentés, la fiche suivante apparaît :

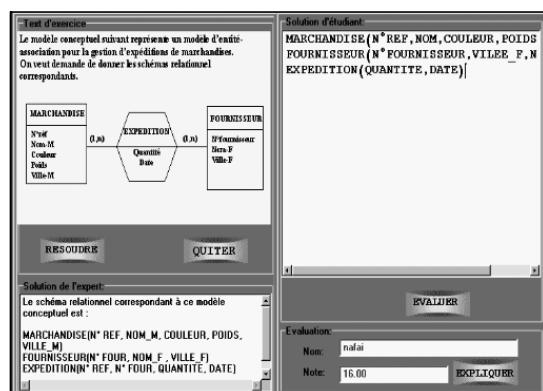


Figure 16 : Exercice

Cette fiche permet d'afficher le texte d'exercice, évaluer la solution introduite par l'utilisateur, expliquer la manière d'évaluation et résoudre l'exercice en cas de demande. La fiche d'explication est illustrée sur le schéma suivant :

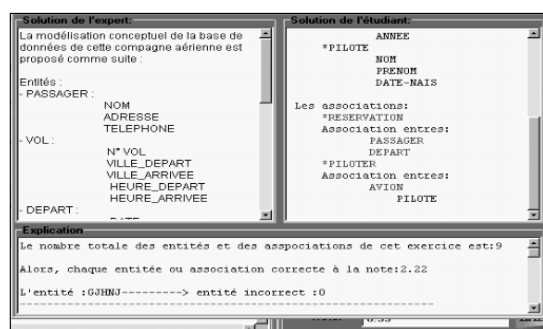


Figure 17 : Explication de la solution

8.5 Tableau noir

Le tableau noir est modélisé par la fiche suivante. Cette fiche permet de consulter soit l'état actuel ou soit l'historique du tableau noir.

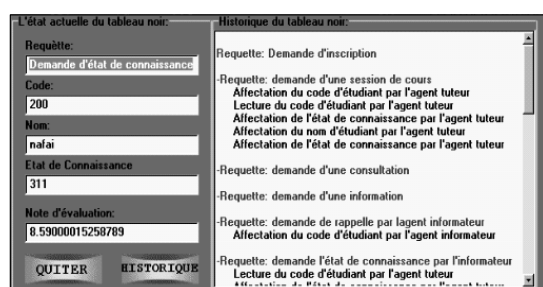


Figure 18 : Etat du tableau noir

9. CONCLUSION

Nous pouvons dire que nous avons touché deux domaines d'informatique très intéressants qui sont *l'enseignement intelligent assisté par ordinateur* et *les systèmes multi-agents*.

Ces derniers, qui essayent de prendre une bonne place dans presque tous les domaines de recherche actuelle, reposent sur le principe de la coopération et la coordination entre plusieurs agents ayant des langages de représentation des connaissances et des mécanismes d'inférence distincts.

L'approche multi-agents de notre tuteur intelligent nous a permis d'intégrer trois agents qui communiquent à l'aide de mécanisme de tableau noir et qui collaborent pour amener un étudiant à la compréhension des principaux concepts des bases de données.

Le premier agent est le *pédagogue* qui a pour but de présenter des sessions de cours à un étudiant à partir de son état de connaissance qui est enregistré dans le modèle de l'étudiant. Le deuxième agent est *l'informateur* qui essaye de donner des informations globales sur un concept donné et de faire des rappelles de cours. Enfin, le troisième agent est *l'évaluateur* qui est lui-même une société de trois autres agents (annonceur d'exercice, correcteur et explicateur) en interactions pour satisfaire les buts de l'agent évaluateur.

La réalisation de notre tuteur intelligent nous a montré que le paradigme multi-agents est une bonne approche pour la modélisation et la conception des systèmes ayant des expertises différentes et nécessitant la coopération de compétence spécialisée.

En perspective de notre travail, nous nous intéressons aux points suivants :

l'intégration d'un sous système capable de modifier ou d'enrichir les différents savoir des agents , par exemple l'ajout d'autre cours et éventuellement leurs concepts de bases et les associent des exercices et ses solutions.

doter l'agent pédagogue d'un outil pour diversifier ses stratégies d'enseignement.

Notre logiciel possède une ergonomie qui lui permet d'être exploité dans le réseau Internet.

A cet effet il suffit de lui créer un site héberger dans un ordinateur et lui affecter une adresse de site.

BIBLIOGRAPHIE

- [AUS 66] Austin J.L. « How to Do Things with Words », oxford University Press, 1962
- [BEC 60] Becker H. S. «Notes of Concept of Commitment », American Journal of Sociology 66, July 1960

- [BOU 94] Boury-Brisset A.-C, Moulin B., « Mise en oeuvre de raisonnements multiples dans un système multi-agent », actes des journées IAD SMA, Grenoble mai 1994, pp169-182
- [BOU 97] Boury-Brisset A.C, Moulin B., « Un modèle d'agent intégrant des mécanismes de raisonnement multiple », Revue d'Intelligence Artificielle, Vol. 11 n°1/1997, pp.73-107
- [BRI 91]: Brigitte A., Haton J.P, Bouzid N., Charpillet M.C, « Le raisonnement en intelligence artificielle. Modèles, techniques et architectures pour les systèmes à base de connaissances », Inter Edition 1991.
- [BRU 97]: Bruillard E. «Les machines à enseigner ». Enseignement sur mesure, Aspects : EAO, Tuteurs, micromondes et hypertextes http://www.terminal.ens.echan.fr/articles/65/multi_mediabaron.html. 1997.
- [BUR 91] Burmeister B. and Sundermeyer K. « Cooperative problem solving guided by intentions and perception», in Werner E. and Demazeau Y., editor, Decentralized AI 3-Proceedings of the Third European Workshop on Modeling Autonomous Agents and Multi-Agent Worlds (MAAMAW-91),pp.77-92, Elsevier Science Publishers B.V., Amsterdam, The Netherlands
- [DAH 99]: Dahmani F., Ouamer A. «The Expert Component of a Distributed Intelligent Tutoring System for Teaching Algorithmic». Mouloud Mammeri University. Computer Science Institute. Compus Hasnaoua B.P.531.RP. Tizi-Ouzou. Algeria. From The First Mediterranean International Conference on Computer Technologies, June 1-3.1999.
- [DEP 87]: Depover C., « L'ordinateur media d'enseignement ». Un cadre conceptuel. De Boek-Wesmael, Bruxelles, pp. 235, 1987.
- [DUR 87a] Durfee E. H., Gasser V. R. and Korkill D. D.,« Coherent Cooperation Among Communicating Problem Solvers», IEEE Transactions on Computers C-36, 1987
- [DUR 87b] Durfee E. H. and Lasser V. R., « Using Partial Global plans to Coordinate Distributed Problem Solving», In proceeding of an the 10th IJCAI, Milan, Italy, 1987.
- [EAR 96] Earl C. and Fibry R.« combined execution and monitoring for control of autonomous Agents », Proceeding of the First International Conference on Autonomous Agents, 1996
- [FER 88]: Ferber J. and Gallab M., « Problématiques des univers multi-agents intelligent », IN actes des journées nationales PRC.GRECO. Intelligence artificielle. Toulouse, mars 1988.
- [FER 95]: Ferber J., « les systèmes multi-agents, vers une intelligence collective », Inter Editions. 1995.
- [FRA 99]: Fransis M., Mc cannell S., Odell J., Ratickaine K., Staut K. et Thampson C. « Agent Technology (Green Paper) ». Agent Working Group OME Document ec/99 12.02 version 0.9. 24 december 1999.
- [GAS 90] Gasser L., et Hill R. W. « coordinated problem solvers », Annual Reviews of Computer Science. 4,1990, pp. 203-253
- [GOL 77] Goldstein I., Carr B.. « The Computer as a coach: an athletic paradigm for intellectual education ». Proc. Of 1977 ACM annual Conference, Seattle, P.227-233.
- [JEN 93] Jennings N. R.,« Specification and implementation of a belief desire joint-intention architecture for collaborative problem solving», Journal of Intelligent and Cooperative Information System, 1993, 2(3),pp.289-318
- [JEN 95] Jennings N. R.,« Controlling cooperative problem solving in industrial multi-agent systems using joint intentions», Artificial intelligence, 74(2), 1995
- [KAZ 00a] Kazar O., Sahnoun Z., Frécon L., « Dynamic Agent Reasoning », Agent Based Simulation I, Passau, Germany, May 2-3, pp. 175-180, 2000
- [KAZ 00b] Kazar O., Sahnoun Z., Frécon L., « Cognitive Agent Architecture For Medical Care», IC-AI'2000 Las Vegas, The 2000 International Conference on Artificial Intelligence, June 26-29, 2000 Monte Carlo Resort, Las Vegas, Nevada (USA)
- [KAZ 00c] Kazar O., Sahnoun Z., Frécon L., «Adaptable Agent Reasoning», CARI2000,16-19 Octobre 2000, pp. 273-279, Antananarivo, Madagascar
- [KAZ 01a] Kazar O., Sahnoun Z., Frécon L., «multi-Competence System of Multi-Paradigm », Agent Based Simulation II, Passau, Germany, April 2-4, pp. 125-129, 2001
- [KAZ 01b] Kazar O., Sahnoun Z., Frécon L., «Multi-agent systems for Aeronotic simulation », ICCTA 2001, International Conference on Computer Theory and Applications, Alexandria, Egypt, 28-30, pp. 5-12, August 2001
- [KAZ 01c] KAZAR O., BAARIR S., « Générateur de Plan pour un robot », CIP2001, international Conference on Productic, pp. 55-65, Algiers, 09-11, June 2001
- [NIL 80] Nilsson N. J., «Principles of Artificial Intelligence », Traduit en Français par Anne et Michel Manago , Cepadues Editions, pp. 150-180, 1980
- [SAL 97] Sallantin J., « Les agents intelligents, essai sur la rationalité des calculs », Edition Hermes,pp. 21-38, Paris, 1997.
- [SEA 72] Searle J. R., «Speech Acts, Cambridge University Press, 1969, Herman (en Français), 1972.
- [SEA 90] Searle J. R., «Intentions in communication, chapter 19 : collective Intentions and Actions », pp 401-415, MIT Press, London, 1990
- [SHO 90] Shoham Y., « Agent-oriented programming », Technical Report STAN-CS-1335-90, Computer Science Department, Stanford University, Stanford, CA 94305, 1990
- [SPE 86] Sperber D. and Wilson D., « Relevance Communication and Cognition », Edition de Minit en français,pp. 25-45, 1986

[WEG 87] Wegner P., «Dimensions of object-based language design », Department of computer science. Brown University Providence, RI 02919. OOPSCLA, Proceeding October 4-8, 1987

[WOO 94] Wooldridge M., Jennings N. R., « towards a

theory of cooperative problem solving », in pre-proceedings of the sixth European Workshop on modeling Autonomous Agents in a Multi-agent World, Demazeau Y., Müller J.P., Perram J., (eds), 1994