

UNIVERSITÉ Du Québec à MONTRÉAL

PROJET DE RECHERCHE

Une considération holistique du diagnostic cognitif dans les EIAHs:  
questions, implications, propositions et formalisation

**Joséphine-Muriel P.-Tchétagni**

Département d'informatique

Essai présenté dans le cadre du cours DIC9410

**Programme du doctorat en informatique cognitive**

**Membres du jury:**

Bourdeau Jacqueline, co-directrice de recherche

Nkambou Roger, directeur de recherche

Janvier 2004

# Sommaire

Dans les systèmes éducatifs informatisés qui le nécessitent, le diagnostic des erreurs de l'étudiant devrait aussi – si ce n'est avant tout – être une activité pédagogique formative. En ce sens, l'implication majeure est une considération formelle des facteurs déterminants dans un suivi formatif de ce processus: les caractéristiques de la technique de diagnostic utilisée, les exigences du *designer pédagogique* dans un contexte particulier, les activités pédagogiques qui supportent ou relaient le diagnostic, l'influence de la théorie d'instruction utilisée dans l'environnement d'apprentissage, etc. Dans ce projet, *la thèse principale* est celle d'une approche "pan-pédagogique" du diagnostic cognitif, tout en préservant une nature pragmatique au niveau des EIAHs: on parlera de "processus de diagnostic cognitif" dans cette intégration de la technique de diagnostic en tant que telle avec les autres variables contextuelles et environnementales dans son déploiement. *Notre but* est d'analyser la pertinence d'une telle conception dans les EIAHs en général et dans les STIs en particulier.

La perspective "pan-pédagogique" implique une approche d'étude du processus de diagnostic cognitif qu'on pourrait qualifier d'"holistique": quelles sont ces variables contextuelles et environnementales qui lui confèreraient ce statut d'activité pédagogique formative? comment s'intègrent-elles dans la mécanique de diagnostic proprement dite? *Dans le registre cognitif, notre objectif* sera d'effectuer une analyse qui permettrait de répondre à ces questions et de traduire ces réponses dans une spécification formelle des caractéristiques, fonctions et exigences du diagnostic dans les EIAHs en général et dans les STIs en particulier.

La perspective pragmatique implique une analyse de l'utilité d'une telle formalisation: à quoi et à qui sert-elle? *Dans le registre informatique, notre objectif* sera d'illustrer comment un tel entendement du diagnostic se réaliserait dans: (1) un *Profiler* fournisseur des services propres au modèle de l'apprenant dans un STI (en l'occurrence le service de diagnostic); (2) un prototype de système d'assistance à un *designer pédagogique* d'EIAH en ce qui concerne la spécification de la tâche pédagogique de diagnostic.

# Table des matières

Sommaire . . . . .	i
Table des matières . . . . .	ii
Liste des tableaux . . . . .	iv
Liste des figures . . . . .	v
Liste des annexes . . . . .	vi
Introduction . . . . .	1
<b>1 Les rôles du modèle de l'apprenant et la modélisation du processus de diagnostic cognitif: solutions dans les communautés IAED et IA . . . . .</b>	<b>5</b>
1.1 Élaboration des objectifs . . . . .	5
1.2 Le module apprenant: quels services? . . . . .	6
1.2.1 Les systèmes modulaires . . . . .	7
1.2.2 Les systèmes multi-agents . . . . .	7
1.2.3 Les <i>shell</i> de modélisation de l'apprenant . . . . .	10
1.3 Les systèmes auteurs et le diagnostic cognitif . . . . .	12
1.3.1 Les systèmes auteurs orientés vers la pédagogie . . . . .	12
1.3.2 Les systèmes auteurs orientés vers la performance . . . . .	13
1.3.3 Les systèmes auteurs: Hybrides . . . . .	14
1.4 Conclusion . . . . .	14
<b>2 Un <i>Profilier</i> pour la gestion des services du module apprenant et une perspective pédagogique du diagnostic cognitif: Motivation et Approche . . . . .</b>	<b>16</b>
2.1 Le <i>Profilier</i> . . . . .	16
2.1.1 Les natures du module apprenant: information, implications . . . . .	17
2.1.2 Pertinence de la problématique . . . . .	19

2.1.3	Difficultés et retombées reliées à la spécification des services du <i>Profiler</i>	22
2.2	Modélisation du processus de diagnostic cognitif dans les STIs	23
2.2.1	Qu'est-ce que le diagnostic cognitif?	23
2.2.2	Intérêt de la problématique: Pourquoi considérer les aspects pédagogiques?	25
2.2.3	Difficulté de la problématique et questions de recherche	30
2.2.4	Originalité de la contribution envisagée	32
<b>3</b>	<b>Méthodologie</b>	<b>34</b>
3.1	Analyse	34
3.2	Implantation	35
3.3	Évaluation et Synthèse	36
<b>4</b>	<b>État d'avancement des travaux</b>	<b>37</b>
4.1	Organisation du système	37
4.1.1	Création d'un cours	37
4.1.2	Les composantes du système	37
4.2	Modélisation cognitive en Programmation Logique	39
4.3	Diagnostic et Remédiation dans <i>PLEARN</i>	42
4.4	Conclusion	46

# Liste des tableaux

2.1	Exemple de spécification générique du processus de diagnostic cognitif . . . . .	26
4.1	Modèles épistémologiques de quelques éléments dans la grammaire de la LP . .	40
A.1	Position par rapport à l'état de l'art: synthèse (1) . . . . .	B
A.2	Position par rapport à l'état de l'art: synthèse (2) . . . . .	C

# Liste des figures

2.1	Modélisation cognitive de l'apprenant dans un STI . . . . .	18
2.2	Modélisation du processus de diagnostic cognitif . . . . .	30
4.1	Architecture du système et interactions . . . . .	38
4.2	Modèle causal d'un exercice pour l'identification d'un terme . . . . .	39
4.3	Modèle causal pour l'identification d'un terme composé . . . . .	41
4.4	Modèle causal pour l'exécution d'une UNIFICATION . . . . .	41
4.5	Sélection de TERMES COMPOSÉS . . . . .	43
4.6	UNIFICATION de TERMES COMPOSÉS . . . . .	44
4.7	L'opération de diagnostic (Algorithme) . . . . .	45

# Liste des annexes

A	Tableaux récapitulatifs . . . . .	A
---	-----------------------------------	---

# Introduction

## *Enseignement et apprentissage basés sur la machine intelligente: historique*

L'usage des machines intelligentes à des fins d'enseignement remonte à 1926 avec par exemple la machine de Pressey[58] basée sur les conditions de Thorndike[74]. Un demi-siècle plus tard, le développement de l'informatique a favorisé l'émergence de l'enseignement assisté par ordinateur (*Computer Assisted Instruction* ou CAI) et par la suite des systèmes tuteurs intelligents (STI) (Carbonel[17], Sleeman et Brown[69], Livergood[39]) et des environnements d'apprentissage interactifs (*Interactive learning environments* ou ILEs) (Solloway et Bielaczyc[70]). Les débuts des STIs coïncident avec l'hégémonie de l'explication cognitive de la cognition humaine (Pylyshyn[59]) et la conception des STIs s'en est trouvée fortement influencée. Plus tard, le constructivisme, la théorie de l'activité (Leontiev[38]) et le "constructionisme" (Papert[55]) ont influencé la naissance des environnements interactifs d'apprentissage humains (EIAH) tel que LOGO, où la priorité résidait dorénavant dans la "manière" et l'historique du processus d'apprentissage: l'apprenant est modélisé en fonction du processus et des actions posées dans une situation donnée d'apprentissage (Self[64]).

## *Les STI: le I ou l'importance des modèles*

L'intelligence dans les STIs tient surtout de leur adaptabilité à chaque apprenant. Certains travaux s'inspirent du monde réel où l'on assume souvent que: un bon enseignant doit s'adapter par ses connaissances et son expérience aux réactions cognitives, émotionnelles et aux attitudes de l'apprenant (Shute et al.[66], GrandBastien[31]). Dans le monde des ordinateurs l'*adaptabilité* n'est possible que si l'on peut exprimer dans un langage qui leur est compréhensible, l'expertise générale d'enseignement du tuteur, pour enseigner un domaine de connaissances particulier et en fonction d'un apprenant particulier. Il faut donc trouver un moyen d'exprimer l'état cognitif de l'apprenant, ainsi que le modèle de connaissances associé au domaine particulier ciblé. Mais auparavant, il faut s'assurer que cette expertise d'enseignement, cet état cognitif de l'apprenant et ce modèle de connaissances ont une certaine validité en sciences de l'éducation, psychologique/cognitive et épistémologique, d'où l'importance de l'utilisation des *modèles (dérivés de théories)*. Pour bien comprendre cette implication, rappelons les trois rôles fondamentaux des modèles dans la recherche en intelligence artificielle (IA) et en éducation (IAED) selon Baker[7].

*Premièrement*, la modélisation est avant tout un outil scientifique. Toute science doit s'implanter à travers des artefacts et le rôle d'un modèle ou d'une théorie est de justifier



toute validation empirique d'un artéfact. En ce qui concerne les STIs dans la recherche IAED, le premier rôle du modèle en tant qu'outil scientifique est donc de justifier le choix d'une approche d'implantation des artéfacts éducationnels que sont le tuteur, l'apprenant ou le modèle du domaine. *Deuxièmement*, la modélisation peut aussi faire partie d'un artéfact. Le modèle en tant que composante d'un artéfact éducatif peut y jouer le même rôle qu'un modèle en tant qu'outil scientifique ou encore y jouer un rôle fonctionnel: le modèle du tuteur par exemple est une fonction d'un STI. La pertinence de la modélisation est alors appréciée lors de la comparaison de deux artéfacts remplissant la même fonction dans le STI. En effet, des critères stables et formels doivent guider cette comparaison, afin de justifier la primauté d'un modèle sur un autre. *Troisièmement*, un modèle permet de guider le *design* et la conception d'un module dans un système éducatif. On sait que les modèles nécessitent souvent des abstractions de la réalité. Lorsqu'on les utilise pour l'implantation d'un artéfact, ils peuvent en dicter les composantes, la structure, la dynamique interne sans que la version réelle corresponde exactement au modèle. Par exemple, le modèle cognitiviste de la cognition a inspiré l'implantation de l'artéfact communément appelé *Problem Solver* en IA (Newell et Simon[47]) sans qu'il corresponde exactement pour autant à cette théorie cognitive.

Ces rôles des modèles interpellent la rigueur méthodologique et la fiabilité à long terme dont doit se réclamer toute approche scientifique. et c'est en cela que les STIs doivent se distinguer des CAI, en théorie. Classiquement, les STIs intègrent 4 composantes: l'objet de l'instruction ou le domaine de connaissances à apprendre, le communicateur de la connaissance ou le tuteur, l'interface de communication entre le tuteur et l'apprenant ou l'environnement, et enfin le receveur de l'instruction qu'on appellera l'étudiant ou l'apprenant dans ce document. Idéalement l'implantation de ces composantes dans les STIs devrait être basée sur des modèles, selon un ou plusieurs des 3 rôles de Baker. Dans les STIs, la priorité est de programmer toute connaissance relative ou nécessaire à l'instruction plutôt que de programmer l'instruction elle-même. Dans les CAI par exemple, les actions du tuteur sont "fondues" à l'avance dans le programme du système, elles sont déterministes et ne tiennent donc pas compte (à court terme du moins) des changements du côté de l'apprenant. Dans les STIs, les actions du tuteur sont basées sur *un modèle* pédagogique.

#### *Problématique du projet: analyse et modélisation du processus de diagnostic cognitif*

La modélisation de l'apprenant est souvent – confusément – appelée diagnostic cognitif dans le domaine des STIs. Il réfère à l'interprétation des erreurs de l'apprenant lors de la résolution

d'un exercice pratique (Ohlsson[51]). Ce processus est devenu le sujet le plus étudié dans la recherche reliée à la modélisation de l'apprenant car il s'agit d'un support majeur pour l'adaptabilité du module tuteur. La majorité de ces travaux s'est concentrée sur l'application d'une technique particulière de diagnostic des erreurs de l'apprenant et les chercheurs ont rarement fait état de l'importance des modèles dans les STIs en ce qui concerne le processus de diagnostic en tant qu'activité pédagogique du module tuteur. Nous pensons que le diagnostic est une tâche pédagogique à part entière et que dans un STI, elle en appelle à beaucoup plus que l'application d'une technique d'IA appropriée pour un tâche particulière dans un domaine de connaissances particulier. La principale question de ce projet concerne l'importance et l'implication du processus de diagnostic cognitif en tant que activité pédagogique complexe qui nécessite la gestion de plusieurs sources d'information et un certain suivi par le biais du déclenchement d'autres activités pédagogiques. Le diagnostic est une tâche intrinsèquement complexe et dans les domaines qu'on peut qualifier de "Sciences Pures" (médecine, physique mécanique, physique des matériaux, etc.), on *peut* se limiter à l'application d'une technique particulière selon le problème à diagnostiquer et donc, il est important de reconnaître l'importance de l'aspect technique ou algorithmique qui soutient ce processus. Toutefois, le problème de diagnostic se pose également dans les domaines de "Sciences fondamentales" (éducation, psychologie, etc.) et dans ce cas, le diagnostic ne peut plus se limiter à l'application d'une technique, étant donné des observations. Il faut tenir compte de plusieurs facteurs (environnement, contexte, passé, collaboration, etc.) qui peuvent nuancer ou enrichir le résultat de l'application d'une technique. Nous sommes arrivés à la conclusion que dans le domaine particulier des usages de l'informatique en éducation, le diagnostic cognitif doit être *intégral*, les acteurs doivent *formellement définir* l'influence et les relations du contexte pédagogique dans cette tâche: nous nous intéressons à la *formalisation d'une vision holistique du diagnostic cognitif dans les STIs et les EIAHs*. L'aspect le plus important de cette recherche est la modélisation à proprement parler puisque l'implantation peut aisément en découler à travers un système auteur ou toute autre application de soutien à un concepteur ou tuteur humain (Baker[7]). Pour y arriver, il importe de répondre à un certain nombre de questions: outre la technique et l'inspiration psychologique qui permettent sa mise en oeuvre, quels sont les éléments clés du diagnostic cognitif (caractéristiques, exigences et activités pédagogiques qui devraient accompagner ou suivre ce processus dans une certaine mesure)? Les réponses à ces questions – entre autres – guideront nos choix au cours de l'élaboration d'un modèle de ce processus dans un STI. Notre contribution vise à

montrer l'utilité d'un tel modèle dans le design pédagogique et dans l'assistance fournie aux étudiants dans un STI: (1) ce modèle serait la base de conception d'un outil d'assistance au concepteur pédagogique dans la spécification d'un processus de diagnostic cognitif; (2) cet outil favoriserait pour le programmeur de ces systèmes une implantation plus formelle du diagnostic cognitif. Plus particulièrement, il s'agit de l'implantation et de l'harmonisation des activités pédagogiques d'assistance – appropriées, voir nécessaires dans un contexte de diagnostic (remédiation, exploitation d'un erreur productive, etc.) – avec le processus de diagnostic cognitif.

### *Cadre de déploiement du diagnostic*

La modélisation de l'apprenant permet de représenter certaines informations associées à l'utilisateur d'un STI. Ces informations peuvent couvrir plusieurs aspects: cognitif, émotionnel et affectif ([24, 80]). Cependant selon la priorité de *design* dans un système particulier, seul un sous-ensemble de ces aspects sera généralement considéré. Nous considérons uniquement la perspective cognitive dans ce processus tout en reconnaissant qu'en réalité, la cognition humaine émerge d'une intégration interactive de ces 3 pôles (Varela[77], Damasio[22]). Dans la modélisation cognitive de l'apprenant, la finalité est l'inférence des connaissances de l'apprenant à partir de ses performances lors des activités proposées par le système. Le diagnostic est souvent attribué à une tâche du modèle de l'apprenant mais la distinction entre le rôle qu'y jouent effectivement d'une part le module tuteur et d'autre part le module apprenant n'est pas claire. Cet état de fait n'est pas une question majeure mais, nous nous sommes quand même penchés sur la problématique des rôles du module apprenant dans un STI: quelles sont les contributions contemporaines sur la question et quelles sont les améliorations possibles. Nous avons deux intérêts: (1) analyser, justifier et spécifier formellement les rôles du modèle de l'apprenant dans un STI à travers un *General Agent Profiler* (GAP); (2) cadrer le déploiement d'une instance de diagnostic cognitif dans ce GAP en mettant clairement en relief la place du module apprenant dans ce processus. Nos résultats sur cette question seront une contribution mineure à travers un travail d'analyse et de formalisation.

### *Organisation de la proposition*

Après avoir situé nos problématiques et justifié leur pertinence, nous proposerons premièrement un projet de modèle pour le GAP, deuxièmement un projet de modélisation du processus de DC en tant qu'activité pédagogique dans un STI, et finalement nous présenterons l'état d'avancement de nos travaux à travers un prototype illustratif.

# Chapitre 1

## Les rôles du modèle de l'apprenant et la modélisation du processus de diagnostic cognitif: solutions dans les communautés IAED et IA

### 1.1 Élaboration des objectifs

L'expertise d'enseignement comporte deux facettes principales: l'expertise reliée au domaine enseigné et l'expertise indépendante de ce dernier ([31]). Dans le premier cas, il s'agit de modéliser le domaine à enseigner en termes d'objectifs d'apprentissage, de ressources didactiques appropriées en fonction de ces objectifs et de la séquence d'instruction de ceux-ci. Le deuxième cas concerne les règles générales reliées à l'engagement pédagogique du module tuteur. Dans les STIs, les activités qu'implique cet engagement sont associées au processus de modélisation de l'apprenant. Il s'agit donc d'un aspect important puisqu'il permet de faire la différence entre un expert du domaine et un enseignant dans un domaine (qui sera généralement *au moins expert* dans ce domaine). Grâce à la modélisation de l'apprenant, les stratégies d'enseignement, le contenu didactique et la planification de l'instruction peuvent être adaptés aux états cognitif, affectif ou émotif de l'apprenant. Ces constats n'ont pas été ignorés dans les STIs (Wasson[78]: planification adaptative; Van Marcke[76]: ajustement des stratégies d'instruction; Nkambou et Kabanza[49]: planification multi-niveaux; ) mais, peu de travaux ont été consacrés à la *spécification générale et la formalisation des services* par lesquels la modélisation de l'apprenant permet cette adaptation de l'enseignement.

En ce qui concerne la formalisation des services offerts par le module apprenant, 3 créneaux de recherche ont été choisis pour conduire un compendium: les STIs conçus par modules, les "shells" de modélisation de l'apprenant et les systèmes multi-agents. Les STIs par modules devraient favoriser la ré-utilisation des diverses composantes et dans ce cas, nous considérons que la spécification formelle – voir standardisée – de chaque composante (en l'occurrence le module apprenant) en favorise la portabilité et donc la ré-utilisation, d'où l'intérêt d'étudier ce domaine; les *shells* de modélisation de l'apprenant sont une source d'exploration importante de par leur nature générique qui implique une certaine abstraction des processus qui sous-tendent la modélisation de l'apprenant; les systèmes multi-agents peuvent être vus comme des cas particuliers de systèmes modulaires, et la pertinence de les aborder découle donc des mêmes arguments.

En ce qui concerne la problématique principale de ce projet, notons que la recherche sur l'expertise pédagogique dans les STIs a surtout contribué à la conception de modèles décrivant l'expertise d'enseignement reliée au domaine enseigné et reliée aux stratégies d'enseignement et à leur adaptation en fonction du contexte ([31]). Qu'en est-il de l'expertise pédagogique reliée au diagnostic des erreurs de l'apprenant? La justification de cette problématique découle de l'argument selon lequel la mise en oeuvre de tout phénomène est améliorée, plus complète si cette dernière est formellement décrite. Les rôles de Baker[7] évoqués dans l'introduction de ce document en faveur de l'importance des modèles dans le domaine des STIs renforcent la pertinence d'étudier cette question. Nous nous posons principalement les questions suivantes: dans quelle mesure les travaux de modélisation de l'expertise du tuteur dans un STI considèrent-ils *formellement et explicitement* les caractéristiques et exigences *pédagogiques* du DC? Quels sont les facteurs qui influencent cette formalisation? Quelles améliorations peuvent être apportées en fonction de ces facteurs?

L'objectif de notre travail est de proposer des réponses à ces questions, au moins en partie. Mais auparavant, nous ferons le point des travaux antérieurs et nous situerons la pertinence de nos idées par rapport aux solutions déjà proposées. Nous proposons en ce sens une analyse des systèmes auteurs pour les questions concernant la modélisation du processus de diagnostic cognitif. Ce choix est justifié car les systèmes auteurs sont l'approche par excellence pour la modélisation formelle de l'expertise ou des choix pédagogiques propres à un concepteur pédagogique.

## 1.2 Le module apprenant: quels services?

Pour répondre à cette question, il faut déjà reconnaître la pertinence de la notion de services offerts par le module apprenant. Selon Ohlsson[51] (corroboré par Self[63], Self et Dillenbourg[24]), le modèle de l'apprenant est le moteur de l'adaptabilité. Lorsque conçue *on-line*, la modélisation cognitive de l'apprenant consiste à interpréter les actions de l'apprenant et à en déduire des axiomes en ce qui concerne son état cognitif. Sans le modèle de l'apprenant, les STIs réagiraient de la même façon face à tous les usagers et ils perdraient par conséquent leur capacité d'adaptation individuelle. L'adaptation concerne principalement l'ajustement de l'instruction en fonction des connaissances, attitudes ou émotions inférées grâce à la modélisation de l'apprenant. L'ajustement de l'instruction est un terme vague car elle peut se manifester de diverses manières.

### 1.2.1 Les systèmes modulaires

L'architecture modulaire dans les STIs a été introduite par Ritter et Koedinger [62]. Elle préconise une vision des STIs non pas comme un tout avec une dynamique interne propre, mais comme un ensemble orchestré de logiciels (les systèmes multi-agents par exemple). Idéalement ces composantes logicielles seraient construites indépendamment les unes des autres. Le but d'une telle vision est de réduire les coûts d'implantation et les complications potentielles lors des mises à jour. Ces auteurs ont donc introduit la notion de tuteurs connectables (*plugin tutor*), philosophie où les STIs sont conçus par composante et non comme un tout. La substance de la vision des STIs par modules se résume à un développement indépendant de plusieurs composantes spécialisées pour les intégrer ensuite au moyen d'un support d'intercommunications. L'importance du développement de standards de communication est impliquée et donc reconnue par ces auteurs. Ces standards permettraient aux composantes développées indépendamment d'être assemblées dans un système afin d'y interagir. Un premier pas dans cette direction s'est matérialisé dans la spécification d'un protocole de communication entre l'interface ou l'environnement de l'apprenant et module tuteur du système dans [62]: traducteur supporte des requêtes de l'environnement vers le système (prochaine étape d'un problème, évaluation de l'action de l'apprenant, etc.). Par ailleurs, Ritter[61] applique les résultats de [62] dans un contexte d'apprentissage où l'étudiant peut être assisté par plusieurs tuteurs. Il s'agit donc d'une extension de l'architecture précédente ou on s'efforce de gérer les messages à provenance multiples ainsi que les conflits d'opinions entre les tuteurs. *En conclusion*, ce type d'architecture accompagne la standardisation des moyens de communication entre les composantes d'un STI modulaire. Cependant, les informations du modèle de l'apprenant sont interprétées uniquement par le tuteur et ce aux fins d'une application particulière. Aucune spécification formelle de la nature des informations n'est donnée: les travaux de [62, 61] concernent surtout l'illustration de l'idée "modulaire" à travers un exemple précis.

### 1.2.2 Les systèmes multi-agents

Les STIs construits à partir des systèmes multi-agents (SMA) peuvent être apparentés aux STIs modulaires mais dans ce cas chaque composante ou fonction n'est pas toujours assumée par un agent. En fait, 3 perspectives distinguent les STIs basés sur ce type d'architecture: certaines recherches concernent la conception d'un système multi-agents où l'accent est mis sur un type particulier d'agent (Machado et al.[40] proposent un agent-serveur de modèles de l'apprenant pour leur partage entre plusieurs applications); d'autres travaux utilisent des

agents pédagogiques (Johnson et al.[37]) et/ou des compagnons (Chan[19]); enfin, certains systèmes sont centrés autour d'un agent tuteur ([61]) ou d'un agent de planification ([49]) et dans ce cas les questions de la modélisation de l'apprenant et des choix pédagogiques sont les principales préoccupations de cet agent. Nous allons examiner ce dernier cas en nous demandant quelles sont les fonctions explicites de cet agent tuteur.

### 1.2.2.1 Généralités

Giraffa et Vicari[29] distinguent deux types d'agents dans un STI: les agents utilitaires (solicités lors d'une action pédagogique, ils exécutent des tâches qui leur ont été commandées par les *agents proactifs*) et les agents proactifs (agents indépendants qui décident et choisissent un comportement donné en fonction de leurs buts). *Les agents tuteur qui nous intéressent sont des agents proactifs*, ce qui illustre la tendance modulaire des STIs basés sur les SMAs: une "*société d'esprits*" où chaque agent proactif a un but réalisable par l'interaction avec les agents utilitaires. Les SMAs sont surtout orientés vers la communication de messages entre agents et notre discussion concernera surtout la question de savoir si les contenus des messages échangés avec l'agent de modélisation de l'apprenant correspondent à un ensemble (ou à des types) de services *formellement définis, assez concis pour: capturer tous les besoins des autres composantes; profiter de toutes les capacités du module apprenant.*

L'objectif du système GIA (Cheikes[20]) est de promouvoir la réutilisation. Il utilise une "confédération d'agents" regroupés dans 3 classes: les agents gestionnaires (gèrent la communication des messages entre les agents de la fédération dont ils sont responsables et ceux des autres fédérations), les agents du noyau pédagogique (gèrent les fonctions propres à un STI) et les agents de l'interface avec l'apprenant. La contribution visée était de définir un langage de communication entre des agents basé sur KQML, ce qui conférerait une sémantique explicite aux messages échangés (Finin et al.[25]). Cependant, ces messages ne révèlent pas formellement les fonctionnalités (ou services qu'on pourrait espérer) de l'agent de modélisation de l'apprenant: bien que basées sur une syntaxe et une sémantique formelles (KQML), il s'agit de requêtes informelles et spécifiques à l'application GIA. Tout en admettant l'importance de cette contribution qui illustre les progrès dans la formalisation de la communication entre agents dans les SMAs, soulignons que le consensus sur les *types de messages* constitue également un thème important. L'expression "Types de messages" réfère à la signification *idéelle* des messages, par exemple: les messages où l'*idée* est d'envoyer une information au planificateur *pour des fins de*

*révision de plan pédagogique*, les messages où l'idée est d'envoyer une information au tuteur *sur l'état d'acquisition d'une connaissance*, etc.

ABITS (Capuano et al.[16]), un STI basé sur une architecture SMA a pour objectif d'améliorer un système de gestion de cours dans l'apprentissage à distance, ce en gérant simultanément plusieurs requêtes, entre autres sur le modèle d'un apprenant particulier. Les auteurs ont clairement identifié les services offerts par ABITS et en ont déduit les agents nécessaires pour les assurer: un agent pour l'évaluation de l'état cognitif d'un apprenant, un agent pour l'évaluation des préférences d'un apprenant, un agent pédagogique pour l'évaluation et la planification du curriculum et un agent central pour régir la communication entre les autres agents et le système de gestion du cours. Malgré la spécificité caractéristique des composantes de cette architecture, les auteurs n'y définissent pas clairement les services offerts par l'agent de modélisation de l'apprenant. On comprend bien implicitement que le modèle de l'apprenant livre – au moins - des informations sur l'état cognitif de l'apprenant. Toutefois, ceci demeure général: aucune définition spécifique des services du modèle de l'apprenant n'est prescrite, encore moins leur utilité par rapport aux fonctionnalités et/ou services offerts par les autres agents.

Silveira et al.[68] souhaitent démontrer comment on peut exploiter les concepts de l'intelligence artificielle distribuée pour améliorer un environnement d'apprentissage qui doit réagir à des évènements de nature fondamentalement différente. En effet, la gestion des évènements qui ont lieu au niveau de l'interface avec l'apprenant est complètement différente de la gestion des événements pédagogiques qu'un tuteur intelligent aimerait générer. Selon les auteurs, la meilleure façon d'appréhender cela est de concevoir le système d'apprentissage comme un ensemble d'agents réactifs (utilitaires) et cognitifs (proactifs). L'agent cognitif *Guilly* fournit un support aux apprenants en envoyant des messages d'aide à l'interface. Cette aide se base sur le modèle de connaissances de l'apprenant, sa réponse actuelle et la stratégie d'enseignement adoptée. Encore un fois, les auteurs ne précisent pas comment et quelles informations *Guilly* demande au modèle de l'apprenant pour faire son travail d'assistance.

### 1.2.2.2 Applications

*I-help system* de Greer et al.[32] procure une assistance en ligne dans un contexte *e-learning*. Un apprenant peut recevoir de l'aide lors d'une discussion privée avec un autre apprenant et cette discussion est établie en utilisant les modèles *personnels* des 2 participants (qui gèrent les informations reliées à leur propriétaire: disponibilité sur le réseau à un moment donné,



état de connaissances dans le domaine étudié). Lorsqu'un apprenant demande de l'aide en formulant une requête à son agent personnel, celui-ci essaie de trouver les apprenants dont le modèle révèle une maîtrise de la connaissance mentionnée dans la requête en renvoyant la requête aux agents personnels des autres étudiants; celui dont l'agent personnel est le premier à répondre est mis en contact avec le demandeur d'aide. Bien que cette application et l'utilisation du modèle de l'apprenant semblent spécialement destinées à la collaboration entre apprenants (plutôt que la pédagogie d'un tuteur) remarquons que même dans ce cas, les auteurs n'expliquent pas comment le modèle de l'apprenant est interrogé et quelles informations il peut fournir.

Il existe de nombreuses autres applications qui utilisent les SMAs comme base de conception de leur architecture, mais à notre connaissance, le rôle de l'agent du modèle de l'apprenant n'y est pas formellement défini (Weber et al.[79] pour le diagnostic, Hospers et al.[33] pour la mise à jour, Nkambou et Kabanza[49] pour la planification dynamique des activités pédagogiques).

### 1.2.3 Les *shell* de modélisation de l'apprenant

Un *shell* est une implantation générique d'une composante d'un logiciel et il doit satisfaire les besoins de représentation d'un maximum de systèmes parmi ceux dont il permet de construire une instance (Pohl[57]). Un *shell* pour la modélisation de l'apprenant doit donc permettre la représentation des connaissances de l'apprenant, l'inférence de nouvelles informations sur l'apprenant et finalement la soumission de requêtes au modèle de l'apprenant pour assurer les différents services qu'il offre.

BGP-MS[57] est un *shell* de modélisation de l'utilisateur basé sur le formalisme AsTRA (*Assumption type representation framework*). La représentation du contenu avec AsTRA exige que la base de connaissances résultante du modèle de l'apprenant permettent les fonctions d'accès apparentées aux utilités des systèmes basés sur le raisonnement logique: la requête (vérifier la véracité ou la non véracité d'une proposition sur l'utilisateur) et l'assignation (assertion d'une proposition sur l'utilisateur). Dans le contexte d'un STI, la fonction de requête peut servir à plusieurs autres fonctions dans les autres modules et elle est donc trop générale pour être assimilable à un service particulier, idem pour la fonction d'assertion.

GUMS (Finin[26]) est un outil générique pour la modélisation de l'utilisateur qui utilise une arborescence de stéréotypes pour grouper les utilisateurs selon des traits communs. Les feuilles de l'arborescence correspondent à des modèles individuels d'utilisateurs elles sont essentiellement constituées d'assertions sur ces derniers. GUMS offre une interface de 6 commandes

dont 4 de nature essentiellement administrative. Les 2 autres commandes permettent respectivement de questionner un modèle individuel et d'y ajouter une assertion. Dans le contexte d'un STI, l'ajout d'une assertion peut être assimilé à une mise à jour du modèle de l'apprenant et l'interrogation sur une assertion peut être assimilée à une demande générale d'information venant d'une autre composante du système. GUMS demeure encore trop général pour répondre aux besoins d'un STI particulier, surtout en ce qui concerne les défis de la modélisation de l'apprenant. Il ne définit pas la nature pédagogique des requêtes sinon l'ajout et l'interrogation du modèle.

TAGUS (Paiva et Self[54]) est un *shell* pour la modélisation de l'apprenant dans un système d'apprentissage. Il utilise un formalisme logique pour représenter les connaissances, le raisonnement et les stratégies cognitives de l'apprenant (selon un formalisme dû à [24], basé sur la notion de *croiances*). TAGUS fournit une interface riche en fonctionnalités sous forme de services: 2 services pour s'enquérir d'une caractéristique ou d'une connaissance et 4 services pour la fonction de mise à jour du modèle de l'apprenant, leur particularité variant selon le fait que la cohérence du modèle doit être vérifiée ou pas, avant d'y ajouter une nouvelle information. Malgré cet effort de définition des services au niveau du modèle de l'apprenant, TAGUS ne spécifie pas: l'ensemble des services que devrait fournir un module apprenant (car on peut trouver plus que les 2 services sus-mentionnés, c.f. Chapitre2); quelle composante pourrait avoir besoin de ces services?

FITS (Ikeda et Mizoguchi[34]) propose une architecture générique à travers laquelle on peut créer une instance d'un STI. Les auteurs adoptent une approche où chacune des tâches principales est conçue comme une *tâche générique*, assurant ainsi l'indépendance de leur outil par rapport au domaine d'apprentissage. FITS est basé sur trois modules: le module de l'apprenant, le module d'expertise et le module du tuteur. Le module de l'apprenant est défini comme ayant trois fonctions principales: la construction du modèle, l'identification des erreurs de l'apprenant et enfin l'analyse des erreurs de l'apprenant. FITS est donc fortement orienté vers le diagnostic pour ce qui est du module apprenant. Cependant, ce module peut fournir des informations à *d'autres composantes du système* et même traiter des informations à l'interne (mise à jour du modèle), aspect qui n'est pas abordé dans FITS. En ce sens, FITS traite d'un aspect particulier du problème: le diagnostic et ses conséquences dans l'approche pédagogique.

### 1.3 Les systèmes auteurs et le diagnostic cognitif

Un système auteur fournit à un tuteur le moyen de modéliser son expertise pour faciliter la création d'un STI. Murray[46] catégorise les systèmes auteurs selon le type de STI qu'ils produisent dans 2 grandes classes: les systèmes orientés vers la pédagogie et les systèmes orientés vers la performance. Dans le premier cas, on doit surveiller l'importance accordée au processus de diagnostic cognitif et à ses implications pédagogiques car son résultat peut diriger la séquence du contenu et de l'instruction. Dans le deuxième cas, on doit observer la dimension pédagogique dans la modélisation du processus de diagnostic cognitif car il sert de base au support et à l'assistance fournis aux étudiants lors de l'exécution d'activités pratiques. Examinons la position de certains systèmes auteurs par rapport à ces considérations.

#### 1.3.1 Les systèmes auteurs orientés vers la pédagogie

Avec IDE (Merrill[42]), il est question d'utiliser des objets de connaissance pour gérer l'instruction dans un environnement d'apprentissage par simulation. Un objet de connaissance décrit les entités (objets, actions, processus de l'environnement de simulation) que les apprenants vont manipuler. Les objets de connaissances servent de support à une instruction préfixée à l'avance et le diagnostic est uniquement basé sur ces objets et non à l'état propre de l'apprenant. Dans ce cas, un modèle de diagnostic aurait permis d'ajuster ou de choisir les actions pédagogiques pour l'apprenant, en considérant aussi son état de connaissances.

GTE (Van Marcke[76]) permet la modélisation de l'expertise de l'enseignement ou du tuteur d'un point de vue épistémologique: quelles sont les connaissances nécessaires pour effectuer un enseignement efficace? L'expertise d'enseignement est basée sur la notion de tâche: une description générique de ce qu'un tuteur fait en instruisant. Dans GTE, le diagnostic est vu comme une tâche du tuteur dont les retombées supportent la sélection des stratégies et du matériel d'instruction. La question pour nous est de savoir si ce processus est explicitement et formellement spécifié en tenant compte des considérations pédagogiques qui s'y imposent. Dans GTE, le processus de diagnostic n'est pas considéré comme une composante centrale du tuteur qui mériterait modélisation, il est plutôt mentionné comme une étape dans un processus de vérification de réponse. En fait, ce système ne le conçoit pas comme une tâche générique avec un minimum de contraintes et exigences pédagogiques, comme c'est le cas avec d'autres aspects de la compétence d'enseignement qui y sont considérés.

Cream-Tools (Nkambou et al.[48]) est un système orienté vers la planification des objectifs d'un cours, sur la base d'un modèle pédagogique appelé CKTN. CKTN est une com-

binaison d'un modèle d'objectifs, de capacités et de ressources préalablement définis. REDEEM (Ainsworth et al.[1]) est centré sur la description des connaissances reliées à l'expertise d'enseignement, notamment en ce qui concerne le choix d'une stratégie particulière pour un étudiant particulier. REDEEM utilise une phase d'interview d'enseignants où ils spécifient dans quelles conditions une stratégie pédagogique prédéfinie doit être sélectionnée ou modifiée. L'approche de modification de stratégie d'enseignement dans REDEEM tient compte des informations statiques du modèle de l'apprenant (état des connaissances, historiques des ressources consultées, etc). EON (Murray[45]) intègre un ensemble d'instruments pour supporter la conception d'un STI selon l'approche classique: le modèle du domaine, le modèle des stratégies d'instruction, le modèle de l'apprenant et le modèle de l'environnement. EON fournit entre autres un outil auteur pour la définition et le choix des stratégies d'instruction en fonction des informations du modèle de l'apprenant. Dans ces 3 derniers systèmes, il est question de modéliser des activités pédagogiques qui auraient pu bénéficier ou même faire partie d'un processus de diagnostic cognitif si on avait envisagé d'en établir un modèle formel. Cela n'est pas le cas et on verra dans le chapitre 2 les avantages que pourrait occasionner une telle approche.

### 1.3.2 Les systèmes auteurs orientés vers la performance

DEMONSTR8 (Blessing[9]) fait partie d'une famille de systèmes auteurs concernée par la conception des tuteurs dont l'approche de suivi et de diagnostic cognitif utilise la technique du "*model tracing*", basée sur le modèle psychologique ACT\* (Anderson [2]). DEMONSTR8 permet seulement la construction des éléments nécessaires à *une technique de diagnostic particulière* et dans le contexte ACT\*, il s'agit des connaissances déclaratives de la mémoire de travail et des règles de production de la mémoire à long terme. Cependant, il ne modélise pas les fonctions, exigences et dynamiques pédagogiques qui confèreraient un aspect plus formatif à ce processus, ce qui est notre intérêt.

DIAG (Towne[75]) est un environnement de simulation d'appareils où le but est de proposer un support à l'entraînement pour la réparation des appareils défectueux. Dans ce système, la pédagogie est implicite et il n'y a pas de modèle de l'apprenant et donc pas de diagnostic approfondi. La seule information que le tuteur exploite est la performance de l'apprenant lors de son dernier exercice: cette information est utilisée pour ajuster le niveau de difficulté du prochain exercice. Notons toutefois que ce système peut être d'une toute autre utilité: on peut s'en inspirer puisqu'il illustre quelques principes et autres connaissances mis en oeuvre dans le

processus général de diagnostic (spécification des relations de causes à effets, le classement des causes probables ou localisation de la cause et le raffinement de ce classement, ou dégagement la cause la plus probable).

### 1.3.3 Les systèmes auteurs: Hybrides

Nous avons informellement identifié un groupe de systèmes auteurs où l'essence réfère tant à la pédagogie adoptée côté tuteur qu'à la surveillance de la performance ou du comportement côté apprenant. Voici une brève analyse des contributions des systèmes qui peuvent être associés à ces deux classes.

IRIS (Arruarte et al.[5]) se base sur une théorie de l'instruction héritée du modèle CLAI (Arruarte et al.[4]) qui intègre les processus cognitifs de l'apprenant, les événements et les actions d'instruction, pour générer automatiquement un STI. Chacun de ces aspects est explicitement défini dans le système auteur, notamment le diagnostic cognitif qui est une forme d'action d'instruction. La composante de diagnostic dans IRIS est basée sur deux éléments principaux: la création des exercices en fonction des objectifs poursuivis et l'association de chaque exercice à un processus de diagnostic générique fourni par l'outil DETECTIVE (Ferrero et al.[6]). Malgré cet abord explicite de la question de diagnostic, DETECTIVE semble définir ce processus en terme *d'algorithme appliqué* ou *de technique d'IA à utiliser selon la nature de la tâche à diagnostiquer*, plutôt que comme un processus pédagogique. Notre but est de modéliser l'expertise pédagogique inhérente au diagnostic cognitif dans un STI. Cette différence est importante car le processus de diagnostic qui découle d'un modèle d'inspiration pédagogique est plus générique et donc potentiellement exportable dans d'autres systèmes d'apprentissage qui ne sont pas nécessairement des STIs (voir Chapitre 2). En résumé, bien que IRIS et DETECTIVE se penchent sur les problématiques que nous soulevons, notre approche se démarque en ce qu'elle attaque le problème de modélisation du processus de diagnostic cognitif sous une perspective *pan-pédagogique* et *supra-technique*. Nous pensons qu'un système auteur devrait aussi permettre à son utilisateur d'exprimer ses préférences et le contexte pédagogiques qu'il s'est fixés, à travers des paramètres suffisamment expressifs (c.f. Chapitre 2).

## 1.4 Conclusion

Ce chapitre nous a permis de situer et de justifier la pertinence des questions soulevées dans notre projet (Tableaux A.1 et A.2 dans l'annexe A). Nous pouvons conclure qu'au mieux, les réponses proposées jusqu'ici restent générales ou informelles car elles s'intègrent souvent

dans le cadre d'une autre problématique. Les *shells* de modélisation de l'utilisateur proposent des services trop généraux au niveau de ce module pour être applicables dans les STI. Les systèmes auteurs quant à eux se sont surtout penchés sur la modélisation des STI en général et traitent rarement la problématique "d'*authoriser*" le processus de diagnostic cognitif en tenant compte de ses exigences, fonctions et implications pédagogiques. Jean[35, 36] a également étudié la question de la modélisation du diagnostic et de son implantation dans une application. Notre contribution se dégage de ces initiatives dans 2 mesures. D'abord, nous souhaitons matérialiser le rôle d'un modèle en tant qu'outil scientifique en proposant un modèle formel du processus de diagnostic cognitif où la perspective et l'environnement pédagogiques revêtent une certaine importance. Ensuite, nous souhaitons matérialiser le rôle d'un modèle en tant que composante d'un artefact éducatif ou encore en tant que base dans sa conception en proposant une implantation de ce modèle – sous forme d'outil auteur ou d'application – qui permettrait aux concepteurs pédagogiques de spécifier leurs préférences par rapport au processus de diagnostic dans un STI dont ils seraient responsables. Au delà de l'importance fondamentale de ce résultat potentiel due au rôle central que la modélisation y tient (Baker[7]), cet outil serait une contribution d'ordre pratique dans la mesure où il permettrait la réutilisation d'un module de diagnostic si celui-ci est conçu à partir d'un modèle formel et assez général pour aborder tous les aspects essentiels du processus.

## Chapitre 2

### Un *Profiler* pour la gestion des services du module apprenant et une perspective pédagogique du diagnostic cognitif: Motivation et Approche

Le premier chapitre de ce document aura permis de mettre en exergue la ligne directrice de ce projet de thèse. Notre principal *objectif* est d'établir un modèle formel du diagnostic cognitif dans les STIs en tenant compte de *l'intégration* de la technique utilisée, des exigences pédagogiques associées au diagnostic cognitif (par exemple vérifier la cohérence du résultat du diagnostic et interroger l'apprenant en cas d'ambiguïté) et des activités tutorielles qui lui confèrent une propriété pédagogique et formative (la remédiation, l'exploitation des erreurs productives[11]). Notre *but* à travers cet objectif est d'illustrer le déploiement de cette conception du diagnostic cognitif sur les créneaux du *design* pédagogique (utiliser le modèle de diagnostic pour concevoir un outil d'assistance à un concepteur pédagogique) et des STIs (intégrer une instance du modèle de diagnostic – par exemple une technique particulière de diagnostic *intégrant* une activité de remédiation – comme service dans un agent *Profiler*; illustrer son déroulement dans un STI et argumenter sur les gains de performance et/ou d'apprentissage obtenus du côté de l'apprenant). Malgré son statut secondaire dans ce projet, nous abordons également la problématique de la formulation des services essentiels ou souhaitables dans un module apprenant; de la spécification des mécanismes associés à ces services dans un agent *Profiler*, ne serait ce que des points de vue analytique et illustratif.

Après avoir examiné les points saillants – i.e. les points susceptibles d'influencer les choix et décisions qui orienteront notre contribution – sur les thèmes de la modélisation de l'apprenant et du diagnostic cognitif, nous dégageons les contributions potentielles de nos idées.

#### 2.1 Le *Profiler*

Nous explorons d'abord la nature et l'inspiration de cette problématique: à quoi sert la communication des informations provenant du module apprenant? quelles sont les natures de ces informations en fonction du type de modélisation utilisé dans ce module? Notre analyse en tant que telle commence par une discussion sur la pertinence de l'idée d'un consensus sur les services offerts par le module apprenant.

### 2.1.1 Les natures du module apprenant: information, implications

*Le module apprenant sert-il aux autres composantes du système?*

Outre la capacité de modéliser l'expertise d'enseignement plutôt que de la programmer dans le système, les pionniers des STIs ont souvent mentionné le modèle de l'apprenant comme la clé de leur adaptabilité. Cela va de soi dans la mesure où cette propriété est tributaire des états affectif, cognitif et émotionnel de l'étudiant observé. Dépendamment du type et de la richesse des informations qu'il renferme, le modèle de l'apprenant fournira des informations au tuteur et aux autres modules, qui les utiliseront pour ajuster le déroulement de l'instruction et de l'apprentissage dans un STI; il utilisera aussi ses informations pour procéder à des traitements internes comme la mise à jour des connaissances. Avant de se demander quelles sont ces informations, nous devons déjà savoir quels types d'information renferme un modèle particulier en fonction de *l'approche de modélisation sur laquelle il se base*: quelles sont les approches classiques de modélisation de l'apprenant dans les STIs et quels types d'information y correspondent?

#### *2.1.1.1 Les natures du modèle de l'apprenant: information*

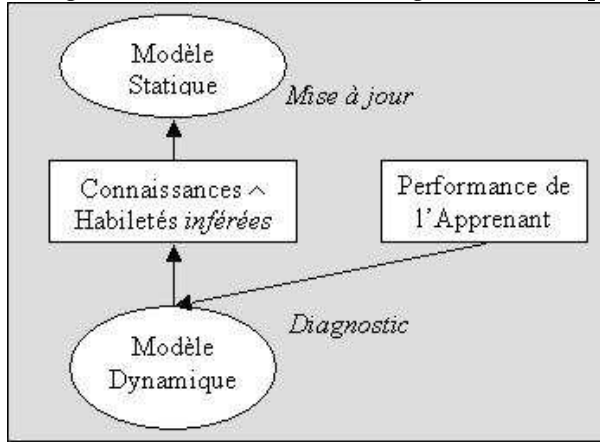
La modélisation cognitive de l'apprenant dans un STI est l'inférence des croyances sur l'état de ses connaissances à partir d'une analyse de sa performance dans des situations de résolution de problèmes. Le module apprenant peut être perçu en 2 dimensions: le modèle statique qui concerne l'état des connaissances de l'apprenant à long terme et le modèle dynamique qui supporte les mécanismes d'analyse cognitive de l'apprenant (inférence de nouvelles connaissances, diagnostic cognitif, etc.). La communication des informations entre le modèle de l'apprenant et les autres composantes d'un STI doit rester indépendante du traitement interne de ce modèle (type de représentation, mécanismes d'inférence, etc.) et le rôle du *Profiler* est de maintenir cette transparence. Nous allons d'abord survoler les principales approches de modélisation de l'apprenant.

#### *Les modèles de type overlay*

Dans l'approche originelle de type *overlay* (Carr et Goldstein[18]), le modèle de l'étudiant correspond à un sous-ensemble des connaissances de l'expert. Plusieurs variantes de cette approche existent et dans la majorité des cas, le modèle de l'apprenant comprend toutes les unités auxquelles on assigne un niveau ou une probabilité d'acquisition. Le problème avec les *overlay* est que l'absence d'un élément (ou une probabilité de maîtrise inférieure à 1) dans le



Figure 2.1: Modélisation cognitive de l'apprenant dans un STI



modèle implique que cet élément est aussi absent ou partiellement absent de l'état cognitif de l'apprenant. Dans certains cas toutefois, l'échec à un exercice peut être dû à un concept *mal acquis* (*misconception*) ou à une règle *mal appliquée* (*malrule* ou *buggy-rule*). Les modèles de type *bibliothèque d'erreurs* (*Bug library*) ont été proposés pour pallier ce problème.

#### *Les bibliothèques d'erreurs*

Les bibliothèques d'erreurs consignent les différentes manières par lesquelles les étudiants commettent des erreurs dans une tâche particulière. Chaque erreur est reliée à un concept ou à une règle mal comprise, et elle permet d'expliquer les difficultés rencontrées par les apprenants et par conséquent, de supporter le diagnostic de manière plus précise. Les bibliothèques d'erreurs ont été utilisées avec succès dans les systèmes tels que DEBUGGY (Brow et Burton[12], Burton[15]) où l'on tentait d'associer chaque solution incorrecte d'un problème de soustraction à une procédure fautive ou mal appliquée. L'établissement d'une bibliothèque d'erreurs est exhaustif car il nécessite un processus élaboré d'observation des apprenants, à travers des protocoles qui peuvent souvent être complexes à réaliser et à analyser.

#### *Les modèles de type "model tracing" ou Reconnaissance des plans*

Ces modèles dérivent de la technique de protocole d'analyse de Newell et Simon[47] qui a donné lieu au *General Problem Solver* (GPS). Étant donné un problème et les différentes règles de jeu, on utilise ces dernières pour en générer les solutions (bonnes ou mauvaises) possibles. Le chemin menant à une solution correspond à l'application séquentielle de ces règles et indique donc la démarche empruntée pour résoudre le problème. Sur la base de ces idées, Ohlsson et Langley[53] ont élaboré une approche de reconnaissance de plan appelée le *Diagnostic Path Finder* (DPF), dont le but est d'inférer la démarche (plan d'action et connaissances

menant à ce plan d'action) qui aurait pu mener à la solution finale donnée par un étudiant. L'approche *model tracing* (Anderson et Reiser[3]) est une autre technique de reconnaissance de plan mais dans ce cas, on essaie d'identifier la démarche adoptée par l'apprenant d'une manière incrémentale plutôt que d'inférer son plan global à la fin de son processus de résolution. Au terme de cette démarche, on obtient idéalement un chemin correspondant à l'ensemble des règles qui expliquent les actions de l'apprenant.

### *Les modèles à base de contraintes*

Tout en rappelant que l'objectif de la modélisation de l'apprenant est avant tout de guider et d'adapter les choix pédagogiques en fonction des besoins de l'apprenant, Ohlsson[52] soutient que seuls certains maillons de sa démarche sont réellement importants pour la compréhension et le suivi de son comportement. Le domaine peut être modélisé en terme de contraintes où chaque contrainte permet de définir les classes d'actions de l'apprenant correspondant à un certain type d'erreur. Dans ce cas, le module apprenant, de manière analogue à (mais non identiquement) aux modèles de type *overlay* correspond à l'ensemble des contraintes qu'il viole.

Pour conclure, essayons de situer ces différentes approches par rapport à la vision bi-dimensionnelle du module apprenant. L'approche de type *overlay* et l'approche à base de contraintes peuvent être apparentées à la modélisation statique de l'apprenant: elles définissent clairement ce qui est modélisé et quelle information on peut trouver en interrogeant le modèle. La consultation d'une bibliothèque d'erreurs ou la tentative de reconnaître le plan de l'apprenant rime plus avec la modélisation dynamique: elles permettent d'expliquer le comportement d'un apprenant.

Bien que certains auteurs aient proposé d'autres cadres de modélisation de l'apprenant, nous pensons qu'ils s'inscrivent dans une intégration, une variation ou une abstraction des modèles fondamentaux qui ont été décrits (Dillenbourg et Self[24], Brusilovsky et Weber[14], etc).

## **2.1.2 Pertinence de la problématique**

### *2.1.2.1 Quelle composante utilise et à quelles fins utilise-t-on le modèle de l'apprenant dans les STIs?*

La description bi-dimensionnelle du modèle de l'apprenant telle que présentée fait ressortir 2 types d'information: les informations globales reliées à l'état cognitif général ou de long terme de l'apprenant et les informations locales qui sont contextuelles et qui supportent les dispositions pédagogiques de court terme (par exemple, s'informer de l'état mental de

l'apprenant pendant qu'il résout un problème). Dans ses approches formelles de modélisation de l'apprenant, Self[63] relève que la modélisation de l'apprenant est justifiée si l'on considère que sa raison d'être est avant tout de supporter un contexte ou une activité pédagogique. Nous pensons que cette activité pédagogique détermine la nature de la modélisation: le choix de l'approche de modélisation et donc les besoins auxquels le modèle résultant devra répondre. Par exemple, pour une tâche de révision du plan d'enseignement, il est plus probable que le tuteur ou le planificateur ait besoin de savoir le niveau d'acquisition de l'apprenant par rapport à un ou plusieurs objectifs (Nkambou et al.[48]). Dans ce cas précis, le modèle le plus approprié est celui de type *overlay*. Selon Self, le modèle de l'apprenant est utilisé dans 3 pôles majeurs du fonctionnement d'un STI: la prédiction et la planification, le diagnostic et la remédiation et dans une moindre mesure (que nous n'aborderons pas ici) l'interaction et la communication apprenant/système. Concernant la partie statique du module apprenant, on peut ajouter la maintenance de l'intégrité du modèle, la gestion de l'épistémologie des connaissances dans le modèle, la considération des facteurs temporels tels que l'oubli. Ces aspects sont aussi importants car ils jouent un rôle important *selon le contexte* d'enseignement et d'apprentissage du système.

#### *Pour la planification dynamique*

Un STI devrait pouvoir planifier dynamiquement c'est à dire, créer ou réviser "*en-ligne*" (en cours de session d'apprentissage) des plans d'instruction afin de s'adapter aux besoins actuels de l'apprenant. Certains auteurs ont étudié les aspects classiques de cette activité en ce qu'elle consiste à ajuster la séquence des objectifs d'enseignement (Brusilovsky et Vassileva[13]). D'autres auteurs ont attribué à la planification un statut plus large en l'associant à de nombreuses tâches tutorielles: planification globale des objectifs du cours, planification des leçons pour une session d'apprentissage, planification des séquences d'activités de l'apprenant, planification de l'interaction avec l'apprenant au cours d'une activité (Nkambou et Kabanza[49]). Dans les deux cas, la contribution des informations sur les états affectif, cognitif et émotionnel de l'apprenant enrichira et raffinerà toujours le processus.

La planification dynamique se base surtout sur les informations globales du modèle de l'apprenant et dans ce sens, le modèle statique sera sollicité abondamment: les approches de type *overlay* et à base de contraintes sont donc les plus indiquées pour remplir ce rôle. Dans la réalité la planification est une tâche de l'enseignant humain qui réajuste sans cesse ses priorités d'instruction en fonction des besoins des élèves. Dans les STIs, les contraintes com-

putationnelles rendent cette tâche suffisamment complexe pour qu'une composante spécifique s'en occupe: le planificateur. On dira donc que le planificateur ou le tuteur sont des composantes qui d'une manière ou d'une autre auront besoin des services du module apprenant.

#### *Pour le diagnostic et la remédiation*

Le diagnostic cognitif supporte plusieurs activités pédagogiques (la remédiation par exemple). Le diagnostic opère généralement à court terme, lors de la résolution d'un problème par l'apprenant; il est "observé" par le tuteur qui tente d'interpréter chacune de ses actions en termes des connaissances: qu'il utilise correctement (diagnostic positif entraînant la mise à jour du modèle statique sans nécessité de remédiation); qu'il utilise incorrectement (diagnostic négatif sur un faux concept ou une fausse règle suivi d'une remédiation et d'une mise à jour du modèle appropriées); qu'il n'utilise pas alors qu'il le devrait (idem au cas précédent). L'interprétation immédiate des actions de l'apprenant se base souvent sur des techniques de modélisation comme les bibliothèques d'erreurs, la reconnaissance de plans et dans certains cas les modèles à base de contraintes. Dans ce sens, le diagnostic cognitif sera probablement basé sur la partie dynamique du module apprenant et la majorité des STIs spécialisés dans le suivi "en ligne" des apprenants confirment ce fait (POLA[21], ANDES[28]). Le module tuteur qui assure l'adaptation des interactions pédagogiques, le choix de la stratégie pédagogique la mieux appropriée, etc., est probablement la composante du système qui bénéficierait le plus des informations nécessaires au diagnostic et à la remédiation.

#### *Pour le maintien de l'intégrité du modèle statique*

Le module apprenant est basé sur un formalisme de représentation de connaissances (pour supporter la modélisation statique), avec les mécanismes d'inférence associés (pour supporter la modélisation dynamique). Par exemple, dans un réseau bayésien, la représentation de l'apprenant consiste à attribuer une probabilité d'acquisition à chaque noeud correspondant à une connaissance ciblée dans le curriculum. Les mécanismes d'inférence: *belief updating* et *belief revision* permettent de supporter la mise à jour des connaissances et le diagnostic des échecs de l'apprenant. Dans un système stable et utilisé à grande échelle, la maintenance de l'intégrité peut devenir une question importante: les ajouts manuels d'informations dans le modèle statique ou dans le modèle dynamique doivent s'opérer sans risque par rapport à l'intégrité syntaxique du module; les inférences successives et répétées peuvent aussi engendrer une incohérence sémantique dans le modèle statique. Il faut donc prévoir chez le *Profiler* un

service pour traiter ces problèmes et dans ce cas, il s'agira d'une manipulation interne: aucune composante autre que le module apprenant n'entre en jeu.

#### *Pour la gestion de l'épistémologie du modèle statique*

La manière de représenter une connaissance dans un domaine dépend de l'épistémologie adoptée pour la modélisation cognitive de ce domaine. Dans la modélisation statique de l'apprenant, la gestion de l'épistémologie des connaissances consiste à utiliser les modèles épistémologiques des connaissances du domaine. Si l'on considère l'épistémologie de Gagné[27] par exemple, la manière de formuler l'état des connaissances de l'apprenant par rapport à un concept ne sera pas la même que par rapport à une procédure. Ce service peut être particulièrement utile au module tuteur dans une activité pédagogique de remédiation épistémologique (Tchétagni et Nkambou [72]): on exploite la nature d'une connaissance pour aider l'apprenant à comprendre et à corriger une erreur due à cette connaissance.

#### *Pour la prise en compte des facteurs temporels*

La question des facteurs temporels est aussi importante, bien que sa pertinence doive s'inscrire dans un contexte précis. La considération d'un facteur comme *l'oubli* ne peut être indiquée que dans un contexte de long terme, où les informations sont suffisantes et établies de longue haleine, sur la base d'interactions soutenues. De plus, les apprenants devraient utiliser le système *assez longtemps* pour que la question de l'oubli se pose. Dans ces conditions, les mécanismes de gestion de l'oubli deviennent importants dans l'interprétation des erreurs de l'apprenant, surtout si elles sont inattendues d'après les informations de son modèle statique. La gestion des facteurs temporels est de nature interne au *Profiler* mais elle peut supporter les activités des autres composantes: le module tuteur (quelle connaissance devrait être rappelée ou révisée?) ou le planificateur (quel objectif apparemment acquis devrait être prévu à titre de rappel?).

### **2.1.3 Difficultés et retombées reliées à la spécification des services du *Profiler***

D'un point de vue conceptuel, spécifier les rôles d'un module aussi important que le module apprenant nécessite une réflexion mûrie pour les dégager et surtout les justifier. Dans ce cas, il faudra respecter en même temps des critères de nécessité, d'utilité et d'opportunité dans un STI. D'un point de vue pratique, la mise en oeuvre des services spécifiés comporte une certaine difficulté. Au delà de la complexité technique – qui va de soi –, la coordination *formelle* (et non ad-hoc, pour les buts d'une application particulière) entre le *Profiler*, le module apprenant et

les autres composantes du système peut devenir difficile à réaliser dans un contexte en temps réel. Une contribution potentielle mais mineure consisterait en une spécification explicite du *type* de services livrables par le module apprenant, ce qui en *formaliserait* l'implantation et engendrerait des retombées, par exemple, la possibilité de comparer formellement 2 modules apprenants. Pour notre part, nous nous limiterons justement à une illustration de l'agent *Profiler* en montrant son utilisation dans une application spécifique, centrée autour du service de diagnostic cognitif. Notre travail sur ce sujet constituera donc une contribution mineure, voir simplement technique.

## 2.2 Modélisation du processus de diagnostic cognitif dans les STIs

Le terme *diagnostic cognitif* est confusément utilisé pour référer à diverses fonctions qui, bien que “consanguines” peuvent s'avérer fondamentalement différentes sous un angle pédagogique. Malgré la similarité évidente de ces fonctions (inférer la cause de l'erreur de l'apprenant, inférer l'état des connaissances de l'apprenant, etc.), une considération holistique puis, une étude formelle de ce processus (qui constituent nos objectifs), en nécessitent une définition claire et nuancée.

### 2.2.1 Qu'est-ce que le diagnostic cognitif?

#### *Fondements*

Issue du couplage grec *dia* (à travers) + *gnôsis* (connaissance), l'étymologie du terme *diagnostic* donne lieu à l'interprétation suivante: *la connaissance à travers les signes*. Cette entrée de jeu facilite la compréhension de l'extension *diagnostic cognitif* dans le contexte général des EIAH et particulier des STIs: connaître l'état cognitif de l'apprenant à travers ses signes. Précisons que nous parlons de diagnostic cognitif dans un contexte d'éducation individuelle. En d'autres termes, les environnements d'apprentissage considérés concernent l'individu qui apprend tout seul à travers l'assistance fournie par le module tuteur du système (constructivisme ou instructivisme), et non à travers des interactions sociales avec ses pairs (socio-constructivisme).

#### *Les aspects du diagnostic cognitif*

L'état de l'art sur les STIs nous a permis de distinguer le “diagnostic cognitif immédiat” et le “diagnostic cognitif *a posteriori*”. Dans les deux cas, l'objectif est d'inférer le niveau d'acquisition d'un ensemble de connaissances par un étudiant (la cause d'une erreur indique le niveau d'acquisition de la connaissance correspondante). La différence entre les deux types

de diagnostic vient surtout de leur forme car dans les deux cas, ils permettent le support pédagogique.

Le “diagnostic cognitif immédiat” intervient pendant ou au terme d’une activité d’apprentissage pratique et son objectif pédagogique est de *comprendre* les erreurs de l’étudiant, d’aider l’étudiant à comprendre ses erreurs et à les corriger. Les erreurs sont donc considérées comme des éléments indicatifs dans la résolution d’un problème et non comme des faiblesses. Le diagnostic cognitif immédiat peut donc être associé à l’évaluation formative (de LandeSherre[23]). Dans l’évaluation formative, on exploite ce que certains auteurs ont appelé *erreurs productives* (Bourdeau et al.[11], Winer et al.[81]). Ces erreurs peuvent être associées à toute action pédagogique de nature formative (par exemple les discussions qui découlent d’une telle erreur [11]). On peut les caractériser comme des erreurs révélatrices d’un état cognitif particulier qui par conséquent, facilitent l’orientation des actions pédagogiques subséquentes. Le “diagnostic cognitif *a posteriori*” se soucie d’établir l’état des connaissances de l’apprenant (“*cognitive assessment*”) non pas à court terme, pendant ou au bout d’une activité pratique d’apprentissage, mais à partir d’un ensemble de tests qui servent effectivement à évaluer l’état cognitif d’un groupe d’apprenants pour: ajuster un programme d’apprentissage, graduer les étudiants, etc. Ce type de diagnostic se rapproche davantage de l’évaluation sommative tout en n’y étant pas entièrement assimilable. En effet, le diagnostic *a posteriori* établit des scores quantitatifs sur les tests passés par les étudiants. Les erreurs commises par ces derniers ne sont pas utilisées comme indicateur de la démarche pédagogique à suivre (on peut parler d’*erreurs non productives ou non utiles* dans ce cas).

Le diagnostic cognitif dans un STI peut être de l’un des deux types, tout dépend des choix de *design* du concepteur pédagogique comme on le verra plus loin. Ce qui est important, c’est qu’en cas de diagnostic immédiat, il faudra absolument spécifier *et* différencier les erreurs productives des erreurs non-productives associées à une tâche particulière. Ce point est trop spécifique pour être considéré au niveau conceptuel de la modélisation du processus de diagnostic cognitif. Au mieux, cette modélisation mentionnera les alternatives en ce qui concerne les types d’erreur à considérer dans une tâche particulière. Dans ce cas, il sera du ressort du concepteur pédagogique ou du programmeur de définir en quoi consiste chaque type d’erreur dans une tâche particulière, dans un domaine particulier.

Lorsqu'on parle de diagnostic cognitif, une analyse cognitive des tâches d'apprentissage est toujours nécessaire dans les STIs, pour dégager les processus cognitifs qui sont utilisés par les apprenants pour les résoudre. Pour le diagnostic immédiat, cette analyse mène à un modèle cognitif de la tâche, programmable afin de permettre au module tuteur du système de suivre l'étudiant. Pour le diagnostic *a posteriori*, l'analyse cognitive mène à la conception de tests d'évaluation valides (i.e. ils permettent effectivement de stimuler les connaissances et habiletés dont on désire mesurer le niveau d'acquisition), éventuellement présentables sous forme informatisée. L'analyse cognitive est généralement assurée par les psychologues en éducation. Dans leurs travaux, Glaser et ses collègues ([30, 8]) ainsi que Shute[65], ont examiné l'analyse cognitive de la performance des apprenants dans les domaines scientifiques; Mislevy et al.[43, 44] ont entrepris des études pour concevoir des tests dont l'interprétation est basée sur des théories probabilistes (par exemple le modèle ECD).

Les autres tâches reliées au diagnostic cognitif sont de diverses natures et Wenger[80] en identifie 3 principales: les inférences de l'état cognitif de l'apprenant, l'interprétation de ses erreurs et la catégorisation de son comportement. Le but des inférences est de reconstruire le processus mental ou le plan qui aurait mené un étudiant à poser l'action ou le comportement observé. L'interprétation consiste à rationaliser, expliquer la démarche attribuée à l'étudiant. La catégorisation quant à elle peut être vue comme une généralisation de l'évaluation: il s'agit de distinguer les différents comportements ou actions de l'apprenant en fonction des attentes fixées dans une situation donnée.

Le diagnostic est donc bel et bien une question vaste et d'autant plus difficile lorsqu'il s'agit de le considérer dans un contexte où le principal acteur n'est pas humain. Nous avons décidé de limiter notre intérêt au diagnostic cognitif immédiat dans le cadre des STIs, c'est-à-dire au diagnostic dont le but est d'aider l'étudiant dans le cours de son apprentissage.

### **2.2.2 Intérêt de la problématique: Pourquoi considérer les aspects pédagogiques?**

Le but du diagnostic cognitif est avant tout de déterminer les besoins de l'apprenant. Comprendre la nature et la source d'une erreur productive, inférer l'impact d'une erreur non-productive, sont des exemples d'activités qui supporteraient des mesures pédagogiques importantes (par exemple, décider de corriger directement l'erreur, de l'exploiter dans la correction; décider de réviser le plan des objectifs, etc.). Il ne s'agit donc pas seulement de "comment utiliser une technique donnée pour inférer la source d'une erreur".



Tableau 2.1: Exemple de spécification générique du processus de diagnostic cognitif

Diagnostic			
Caractéristiques		Exigences	Activités pédagogiques
Type	<u>Exemple</u> Béhavioral, Épistémique	Gestion de la cohérence du modèle statique par rapport au diagnostic	Remédiation. <u>Exemples</u> Analogies, Rappel,          Articulation, démonstration, exploitation de l'erreur productive
Aspects Traités	<u>Exemple</u> Observation, In- terprétation, Évaluation		
Technique de diagnos- tic	<u>Exemple</u> Bibliothèque d'erreurs,      reconnaissance de plan, explication la plus probable		

*2.2.2.1 Motivation: le diagnostic cognitif en tant que technique mais aussi processus pédagogique*

La majorité des travaux liés au diagnostic cognitif dans le domaine des STIs correspond à l'application des techniques de diagnostic cognitif (bibliothèques d'erreurs, reconnaissance des plans). Dans les domaines complexes (domaines où la résolution de problème est l'un des principaux objectifs d'enseignement), l'analyse cognitive des tâches et leur modélisation basée sur une technique particulière sont inévitables pour assurer un support pédagogique effectif. Ainsi dans certains cas, les apprenants ne pourront être aidés que lorsqu'on applique à leur modèle une reconnaissance de plan combiné à l'utilisation d'une bibliothèque d'erreurs (Conati et Van Lehn[21]).

Nous sommes dirigés par un intérêt d'optimisation qualitative dans la construction des STIs. La modélisation du diagnostic cognitif permettrait d'en dégager une "théorie" du processus, non pas au sens pur du terme – nous n'avons pas l'intention de faire une étude fondamentale de ce processus –, mais au sens qu'on en définirait les caractéristiques et exigences clés tant sur le plan conceptuel que sur le plan pratique. Ces éléments devront avoir cette particularité d'être communs et essentiels à tous les contextes de diagnostic, quel que soit le domaine étudié. Une telle spécification formelle augmenterait donc la qualité du processus en autant qu'elle en capture tous les côtés nécessaires. Le tableau 2.1 est un exemple de spécification du processus de diagnostic.

### 2.2.2.2 *Retombées potentielles*

Nous avons identifié trois retombées potentielles qu'engendreraient cette vision pédagogique du processus de diagnostic cognitif: dans la conception du processus, dans la conception ou l'implantation informatique des STIs ou des EIAHs (contexte pratique), et enfin dans la collaboration entre concepteurs pédagogiques et programmeurs de systèmes.

*Contexte conceptuel: permettre à un concepteur pédagogique de spécifier ses préférences*

La spécification formelle des caractéristiques et exigences du processus de diagnostic cognitif constitue avant tout un travail d'analyse. Pour que cette analyse soit utile, cette spécification pourrait prendre la forme d'une application informatique aux bénéfices des concepteurs pédagogiques: un système auteur ou un système conseiller. Cette application aurait deux objectifs principaux: d'abord amener le concepteur pédagogique à prendre conscience de tous les aspects importants du processus de diagnostic cognitif et ensuite lui permettre d'exprimer ses choix et préférences par rapport à ces aspects. Par exemple, une caractéristique optionnelle du diagnostic concerne la considération des étapes suivantes: observation, catégorisation et interprétation. Si un concepteur pédagogique est explicitement conscient de ces étapes et de leur caractère facultatif, il peut décider de ne pas faire d'interprétation et de se limiter à reconnaître les erreurs de l'apprenant et à les corriger de manière magistrale. Cette décision pourrait être exprimée à travers une action explicite à l'interface du système conseiller ou du système auteur. Par contre, certains aspects de l'instruction pourraient être interprétés comme une exigence pour assurer la continuité pédagogique du processus de diagnostic cognitif, par exemple la rémédiation des erreurs. Dans ce cas, l'application pourrait offrir au concepteur pédagogique différentes techniques de remédiation et aller jusqu'à lui suggérer les choix appropriés selon la stratégie d'instruction qu'il a adoptée. En effet, le diagnostic cognitif doit suivre la même logique que celle qui a été considérée depuis le début du processus de *design pédagogique*: les choix du concepteur pédagogique par rapport à sa mise en oeuvre doivent être cohérents avec les théories d'instruction et d'apprentissage qui ont soutenu la conception du reste du système. Nous reconnaissons l'importance de ce fait, mais notre recherche actuelle ne s'approfondira pas sur les liens entre les choix au niveau du diagnostic et les choix au niveau du design pédagogique. Il s'agit cependant d'un aspect clé qui devra inexorablement être étudié, modélisé et idéalement implanté, sûrement à travers une recherche future.

Ces exemples illustrent comment le diagnostic cognitif a des objectifs qui vont au-delà de

l'établissement de la cause des erreurs de l'étudiant. La prise en compte de ces objectifs est inévitable dans un projet de modélisation et par conséquent, la contribution d'un modèle conceptuel consisterait à contraindre tout système qui l'adopte à considérer non seulement les aspects techniques mais aussi les aspects moins évidents, d'ordre plus fondamental et proprement pédagogique: différenciation des erreurs productives et non productives, spécification des actions pédagogiques à prendre pour chaque type d'erreur, etc.

*Contexte pratique: spécifier ses exigences à un programmeur*

L'application qui utilise le modèle conceptuel d'un processus de diagnostic permettrait donc à un concepteur pédagogique de spécifier ses préférences par rapport aux alternatives correspondant à chaque caractéristique et exigence. Cette spécification doit à son tour être d'une certaine utilité, par exemple, comme base de travail d'un programmeur des systèmes d'apprentissage correspondants. C'est pour cette raison que nous parlons d'optimisation qualitative dans l'implantation des STIs ou des EIAHs car dans ce cas, elle aura des bases formelles puisque s'inspirant d'une spécification elle-même établie selon un modèle.

#### Exemple

Supposons que nous sommes dans le domaine de la programmation logique et que la tâche à accomplir est une simulation du processus de RÉSOLUTION dans une BASE DE CONNAISSANCES donnée; le concepteur pédagogique a choisi d'effectuer: un diagnostic de *type behavioral* avec observation et évaluation des erreurs, sur la base de la technique de reconnaissance de plan; une remédiation des erreurs basée sur l'articulation de la connaissance associée à cette erreur ([72]).

Le programmeur doit donc: implanter la reconnaissance de plan *pour* la tâche de RÉSOLUTION dans le module apprenant (partie dynamique); implanter l'observation et l'évaluation des erreurs; implanter une instance de diagnostic comportemental pour la tâche particulière de résolution en utilisant le modèle de reconnaissance de plan; implanter la remédiation des erreurs basée sur l'articulation du comportement diagnostiqué (par exemple, un mauvais plan de résolution).

Il faut tout simplement retenir que cette application de la spécification formelle du diagnostic cognitif est une illustration de l'importance des modèles dans la recherche en IAED (Baker[7], comme mentionné dans notre introduction).

*Contexte collaboratif: le partage des modules de diagnostic*

On peut considérer l'interopérabilité comme une *retombée* de la spécification formelle du diagnostic cognitif. Cependant, les hypothèses qu'implique cette retombée devancent l'état de

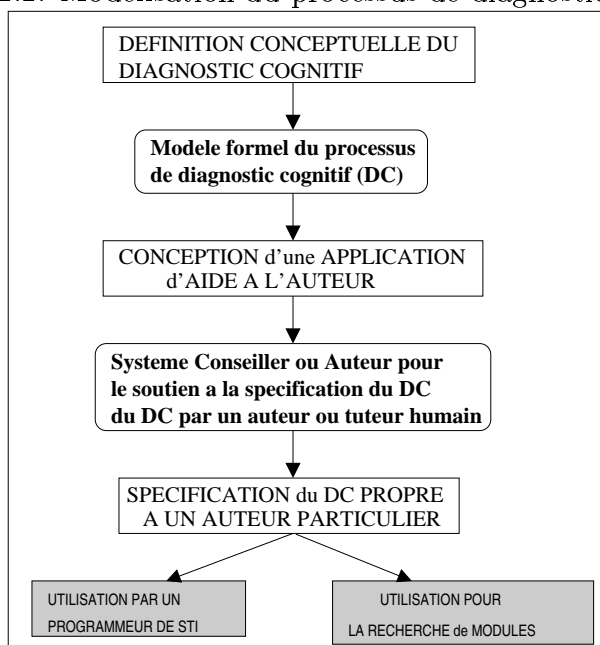
l'art en ce qui concerne la réutilisation des modules des STIs. Pour cette raison, cette retombée est mentionnée uniquement à titre de potentiel de recherche et nous ne considérerons pas cet aspect dans nos contributions envisagées.

Si nous assumons que les modules de diagnostic sont implantés ou spécifiés sur la base du même modèle formel, alors cela implique qu'ils sont comparables et qu'on peut les repérer dans une recherche basée sur des critères formels. Par exemple, si l'on se base sur l'illustration du tableau 2.1, on peut comparer deux spécifications de diagnostic sur la base de leur type: *behavioral* ou épistémique. On peut utiliser la spécification d'un processus de diagnostic cognitif pour retrouver dans une bibliothèque de modules de STIs les modules les plus proches de cette spécification. Dans ce sens, la spécification d'un processus de diagnostic, sachant qu'elle est basée sur des caractéristiques formelles et communément reconnues, pourrait servir de critère de recherche dans une librairie de modules de STIs dans le but de recueillir les modules satisfaisant ce critère et de les ré-utiliser dans d'autres STIs. Vue ainsi, la modélisation favorise l'interopérabilité du module de diagnostic dans un STI: il s'agit de la partie dynamique du module apprenant et des activités du module tuteur visant à supporter la dimension pédagogique de ce processus. Malgré l'importance de ce bénéfice, il est basé sur des hypothèses risquées et qui n'ont pas encore été envisagées dans l'état de l'art. Sur le plan fondamental, un changement au niveau de la conception architecturale des STIs s'impose (si l'on veut pouvoir utiliser la partie du module tuteur qui gère les activités reliées au diagnostic cognitif ainsi que la partie du module apprenant correspondant au modèle dynamique, il faut les implanter de manière modulaire pour qu'ils puissent facilement être exportés vers d'autres systèmes sans interférer avec les autres éléments). Par exemple au sein du module tuteur, on devrait envisager un sous-module spécialisé dans les activités reliées à la pédagogie du diagnostic cognitif: révision du plan pédagogique; dialogues interactifs pour la discrimination de diverses causes, l'interprétation et la compréhension de l'apprenant, la remédiation, l'exploitation des erreurs productives, etc. Ainsi, ce module pourrait facilement être exporté vers d'autres systèmes sans interférer avec les autres aspects du module tuteur. Sur le plan pragmatique, la mise en place d'une bibliothèque de ressources ou de modules provenant des STIs sera nécessaire.

### 2.2.2.3 Objectifs de recherche

La figure 2.2 synthétise les aspects de la problématique d'une description formelle du diagnostic cognitif, telle que décrite ci-dessus et dont résulte de nombreuses questions. Dans ce projet de thèse, notre *objectif conceptuel* est de *formaliser* le processus de diagnostic cognitif.

Figure 2.2: Modélisation du processus de diagnostic cognitif



Nous considérons une perspective où sont associées des exigences et activités pédagogiques au diagnostic cognitif et où le but est (ainsi que leur dynamique ou leur orchestrage dans le processus entier) de les intégrer à une technique particulière (par exemple *model tracing*). La formalisation devra donc établir et justifier ces éléments du diagnostic cognitif, avant de décrire formellement comment et dans quelles circonstances ils s’y incorporent. *Notre objectif pratique* est d’illustrer l’utilité de ce modèle ou de cette formalisation dans 2 contextes. Du côté du concepteur pédagogique, nous souhaitons aboutir à une application informatique qui assiste dans la spécification du processus de diagnostic désigné à un STI ou un EIAH donné, dans un contexte donné. Dans la continuité, on aimerait montrer que l’utilisation de cette spécification par un programmeur de STI ou de EIAH permettrait aux étudiants de bénéficier d’une assistance pédagogique centrée autour du diagnostic, lors de la performance de leurs diverses tâches.

### 2.2.3 Difficulté de la problématique et questions de recherche

Décrite dans une optique de spécification et d’application informatique, la modélisation du processus de diagnostic cognitif soulève des questions d’ordres fondamental et pratique.

#### 2.2.3.1 Questions fondamentales: les exigences propres à un processus de diagnostic cognitif

Sur le plan fondamental, la définition formelle du processus de diagnostic cognitif nous imposera d’en analyser les caractéristiques et implications sur les plans cognitif et pédagogique.

Toutefois, énumérer ou identifier les caractéristiques du diagnostic cognitif ne sera pas suffisant; il faudra également distinguer les caractéristiques assez générales pour embrasser toute instance de réalisation de ce processus, tout en demeurant significative pour en préserver l'essence. Cette problématique ouvre le débat sur un thème intéressant: pour chaque aspect du diagnostic et pour chaque instance possible de cet aspect (par exemple l'aspect correspondant au *type de diagnostic*), quel est le choix approprié selon la philosophie de *design de l'instruction* et de *modélisation cognitive* adoptée ou assumée dans un STI? Bien que nous ne traitons pas ce thème dans le projet actuel, il faut reconnaître qu'il s'agit d'une considération clé pour le concepteur pédagogique qui utiliserait le système que nous visons. En effet pour préserver une certaine cohérence, le concepteur pédagogique devra faire des choix judicieux dans l'application d'aide à la spécification du processus de diagnostic cognitif, en fonction de la philosophie de *design pédagogique* qu'il se sera fixée pour la conception de son système.

#### 2.2.3.2 Questions pratiques

La réalisation informatique d'une spécification de diagnostic cognitif soulève de nombreuses questions de recherche notamment: peut-on concevoir des modèles génériques pour le processus correspondant à chaque exigence quitte à l'appliquer à un contexte particulier (par exemple, le *diagnostic behaviorial* pour une tâche particulière avec un modèle de tâche particulier dans un domaine de connaissances particulier)? Quelles sont les différentes actions pédagogiques qu'on peut rattacher à chaque exigence (par exemple comment effectuer la vérification de la cohérence du modèle de l'apprenant avec un diagnostic établi)? Peut-on modéliser ces actions pédagogiques de manière générique comme dans GTE ([76])? Quelles que soient les réponses à ces questions, chacun des cas devra être traité comme un thème à part entière. L'envergure de la recherche impliquée par les questions liées au couplage "choix du concepteur pédagogique/implantation du programmeur" de système indique l'ampleur et la complexité de notre problématique.

#### 2.2.3.3 Autres questions

Relevons que la formalisation de toute connaissance impose d'en clarifier les fondements, les éléments, la sémantique de ces éléments et la dynamique interne entre ces éléments. Un travail d'analyse supplémentaire permettrait de traduire notre spécification dans une ontologie, ce qui renforcerait l'objectif de partage des modules de diagnostic entre divers systèmes.

#### 2.2.4 Originalité de la contribution envisagée

Les questions et difficultés sus-mentionnées révèlent un certain nombre de contributions potentielles. Sur le plan fondamental, nous offrons une vision du diagnostic cognitif dans les STIs comme un processus qui va au-delà de l'application des *technique de diagnostic* – bien que nous reconnaissons l'importance de ces derniers dans la dimension computationnelle de ce processus –. Notre problématique est motivée par le fait que les activités pédagogiques formatives inhérentes au diagnostic sont rarement explicitées comme telles, bien qu'on les considère implicitement dans certaines applications (*coaching*, présentation d'exemples, etc.). L'originalité de la spécification formelle des exigences et fonctions du diagnostic cognitif dans un contexte d'apprentissage assisté par l'ordinateur, vient du fait qu'elle contraint les applications informatiques à tenir compte de ces aspects pédagogiques et ce de manière formelle et en les orchestrant dans un ensemble. Dans le registre applications et pratiques, la modélisation de tout processus en supporte la conception *artéfactuelle*[7]. De nombreux artéfacts éducatifs émergent de nos jours, tant dans le domaine des STIs que dans celui de l'éducation à distance. Au niveau des STIs, considérons la conception d'un artéfact pour la spécification d'un processus de diagnostic cognitif, basée sur un modèle de diagnostic cognitif. La contribution de ce type d'artéfact (basé sur un modèle formel et complet) découle de sa généralité, de sa polyvalence et du partage des ressources qu'il pourrait entraîner dépendamment de son contexte d'utilisation. Pour ce qui est de la généralité, rappelons nous de l'argument de Baker[7] qui soutient qu'un modèle tient lieu de théorie dans la mesure où il explique formellement toutes les instances et nuances d'un phénomène. Dans ce cas un artéfact pour l'aide à la spécification du diagnostic dans un STI, basé sur un modèle formel de ce processus offrirait la possibilité au concepteur pédagogique d'en exprimer une instance sans se soucier du domaine particulier, ni de la tâche particulière dont il est question. Concernant la polyvalence, le terme réfère ici à la capacité d'offrir plusieurs fonctions. Il ne s'agit pas d'offrir simplement plusieurs fonctions, mais d'offrir des fonctions clés et pertinentes pour le processus que l'on tente de définir. Tout concepteur pédagogique utilisant un artéfact éducatif pour la spécification d'une activité pédagogique devrait avoir la possibilité de paramétrer son approche particulière selon ses préférences et selon le contexte pédagogique qu'il s'est donné. Par préférences, nous n'entendons pas une vision arbitraire et personnelle du processus; elles réfèrent aux valeurs particulières que prendront l'ensemble des caractéristiques et exigences du processus de diagnostic cognitif dans la vision particulière de ce concepteur pédagogique. Par exemple, la caractéristique "type de diagnostic" pourrait prendre la valeur "diagnostic comportemental" pour

indiquer le fait que l'instance actuelle de diagnostic a surtout pour objectif d'inférer le plan ou la démarche adoptée par l'apprenant observé, plutôt que l'état des connaissances qui a mené à cette observation. En résumé, concevoir une application pour soutenir la spécification d'un processus de diagnostic cognitif basée sur un modèle permettrait au concepteur pédagogique d'exprimer sa propre vision du mécanisme dans les limites de la nature de ce mécanisme. Dans le chapitre suivant, nous présentons la méthodologie que nous souhaitons adopter pour atteindre les objectifs décrits dans ce chapitre.



# Chapitre 3

## Méthodologie

Nous envisageons une méthodologie en trois étapes: l'analyse qui nous permettra d'établir les modèles nécessaires, l'implantation qui nous permettra de les mettre en valeur à travers une application informatique et finalement l'évaluation qui nous permettra de juger leur utilité par rapport aux usagers appropriés.

### 3.1 Analyse

Nous proposons deux phases parallèles d'analyse: (1) l'analyse des éléments du diagnostic cognitif et (2) l'analyse des fonctions et services du modèle de l'apprenant.

#### A. Analyse des éléments du diagnostic cognitif

Nous devons analyser et caractériser les éléments suivants, en justifiant nos choix:

1. La dynamique générale du diagnostic cognitif (algorithme);
2. Les caractéristiques et exigences clés et/ou optionnelles;
3. Les activités pédagogiques associées ainsi que la manière dont elles s'inscrivent dans la dynamique du diagnostic;

Au terme de l'analyse, proposition d'une définition/expression *formelle* des éléments obtenus.

#### B. Analyse des services et fonctions du module de l'apprenant

Nous allons approfondir notre proposition du chapitre 2 en:

1. Justifiant la pertinence de chacune d'elles dans un contexte STI;
2. Proposant une définition formelle de chaque fonction en termes d'intrant, de processus interne et de résultat.

Note Il convient de mentionner que nous nous limiterons à cette analyse, le *Profiler* demeure un cadre de déploiement du diagnostic.

## 3.2 Implantation

Dans l'implantation, nous visons deux illustrations informatiques. La première pourrait être associée à un système d'assistance à un concepteur pédagogique et la deuxième correspond davantage à une illustration du déploiement d'un processus de diagnostic cognitif *de concert avec* une ou plusieurs des activités pédagogiques que son modèle recommande.

### C. Migration du modèle de diagnostic dans une application informatique

Cette application informatique offrirait des facilités au concepteur pédagogique dans les buts de:

1. Permettre aux concepteurs pédagogiques d'exprimer leur vision personnelle du diagnostic cognitif,
2. Dans les limites des caractéristiques, exigences, activités pédagogiques propres à ce processus (ainsi que de leur dynamique respective au sein du diagnostic).
3. En permettant au concepteur pédagogique de *comprendre la signification, les implications et surtout les propriétés* de chaque élément du processus, les *éléments optionnels et obligatoires*, ainsi que leur bien fondé.

### D. Illustration d'un processus de diagnostic et des activités pédagogiques qui l'accompagnent

Il faut donc définir clairement le contexte cognitif et pédagogique de ce STI:

1. Établir le contexte du STI illustratif: objectifs d'apprentissage, théorie de *design* pédagogique utilisée et modélisation cognitive du domaine en fonction d'une théorie d'apprentissage choisie; architecture et organisation du STI;
2. Définir les tâches associées aux objectifs; proposer des modèles de tâches et établir le modèle de diagnostic associé à chaque tâche;
4. Implanter chaque élément souhaité dans l'instance de processus de diagnostic cognitif qu'on souhaite illustrer.
5. Intégrer le tout dans une instance de *Profiler* qui fournirait particulièrement le service de diagnostic.

### 3.3 Évaluation et Synthèse

Dans l'évaluation des implantations, notre but est de *tester une hypothèse d'utilité*.

#### E. Préparation de l'évaluation

Pour préparer la validation, nous utilisons un procédé analogue aux 4 premières étapes de [67]. Dans ce qui suit, *A* réfère au prototype du STI et *B* réfère au prototype de l'outil d'assistance:

1. **Établir les objectifs de l'application et Définir les buts de l'évaluation de cette application pour *A* et *B*.**
2. **Sélectionner le *design* approprié pour atteindre ces buts.**
3. **Préparer la logistique de l'étude:** Il s'agira de choisir le nombre de sujets approprié, déterminer les groupes de contrôle pour chaque expérience, déterminer les mesures qu'on désire effectuer.

#### F. Évaluation et synthèse

Il s'agit de définir comment l'étude et l'évaluation se déploieront avec la participation des sujets. Bien que [67] suggère des *tests pilotes* ([41]) avant de procéder à l'évaluation en tant que telle, notre recherche se limite à l'établissement d'un prototype, voire d'une illustration de nos idées. Le but n'est donc pas d'établir un système sophistiqué qui fonctionnerait à grande échelle. Dans ce cas, la prudence associée à l'exécution des tests pilotes ne s'applique pas.

4.0 **Évaluation de *B*** Il s'agit d'une évaluation de type *interne*[67] en ce sens qu'on évalue une composante ou un système par rapport à lui-même.

- a. Soumettre au moins un concepteur pédagogique à l'utilisation de l'outil d'assistance;
- b. Soumettre le concepteur à un questionnaire. Ce questionnaire devra se concentrer sur les questions de l'utilité et de la pertinence des propositions de l'outil (Bostock[10], Richards et Fukuzawa[60]).

4.1 **Évaluation de *A*** Nous avons choisi l'approche de recherche expérimentale[41] qui vise à *moduler les variables indépendantes* et à mesurer la valeur des variables dépendantes.

5 **Analyse des résultats:** Elle devrait aller de concert avec l'évaluation.

6 **Synthèse:** Elle consistera à tirer les conclusions et leçons de notre analyse.

# Chapitre 4

## État d'avancement des travaux

Ce chapitre présente un prototype conçu pour *illustrer* le déploiement du processus de diagnostic cognitif avec comme activité pédagogique clé la *remédiation*. Il s'agit d'un début de la partie (D) de la méthodologie exposée au chapitre 3. Nous avons atteint les étapes suivantes: établissement des objectifs d'apprentissage, définition des tâches associées à ces objectifs, modélisation cognitive dans le domaine étudié (limitée aux connaissances intervenant dans les tâches choisies), implantation de la boucle diagnostic et remédiation. Cette illustration est une plate-forme pour l'apprentissage des notions de base en Programmation Logique (PL) et nous la dénommerons *PLEARN* à des fins de référence.

### 4.1 Organisation du système

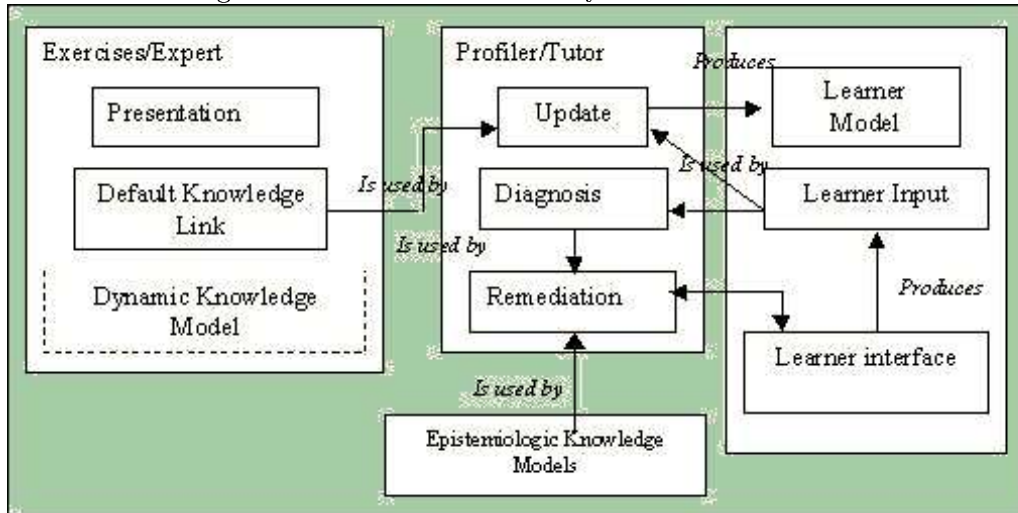
#### 4.1.1 Création d'un cours

En nous inspirant de CKTN[48], nous avons organisé le cours sur les notions de base en PL en deux objectifs d'apprentissage (qui peuvent être subdivisés en sous-objectifs): la grammaire des termes (savoir identifier les termes utilisés en PL) et les sujets relatifs au traitement logique, par exemple l'UNIFICATION. L'ensemble est regroupé dans une hiérarchie d'objectifs. Chaque objectif correspond à un objectif d'apprentissage de Gagné (par exemple apprendre à *identifier une VARIABLE, soit acquérir le concept de VARIABLE*) et possède des activités d'apprentissage: soit une activité d'enseignement (l'apprenant est exposé à un contenu instructif et dans notre cas, il s'agit de fichiers HTML), soit une activité de pratique (l'apprenant doit résoudre un exercice et dans notre cas, il s'agit de petits programmes JAVA). Cette vision est similaire à la séquence globale des événements de Gagné: on enseigne la matière à l'apprenant et par la suite, on stimule une démonstration de performance afin d'interagir avec lui pour l'aider à apprendre.

#### 4.1.2 Les composantes du système

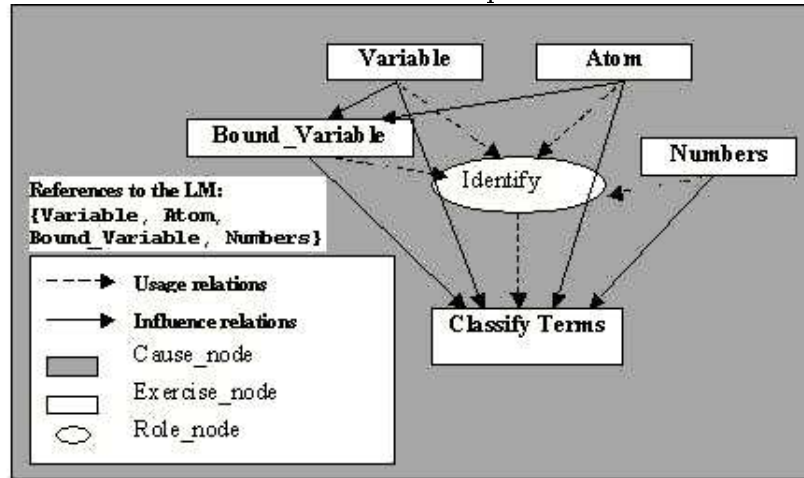
La figure 4.1 présente l'architecture générale de l'application. Le *modèle statique de l'apprenant* est un *overlay* de la hiérarchie d'objectifs et dans ce cas particulier, il est interprété comme réseau bayésien notamment pour supporter les fonctionnalités de mise à jour (Tchetagni et Nkambou[71]). Les exercices sont de 2 types selon le nombre de connaissances qu'ils exigent:

Figure 4.1: Architecture du système et interactions



les exercices de niveau 0 correspondant à l'acquisition d'un concept (par exemple, pouvoir identifier une variable dans le langage PROLOG); les exercices de niveau 1 nécessitant plusieurs connaissances (appliquer une règle, exécuter une procédure, l'UNIFICATION par exemple). Chaque exercice a par défaut une liste de références aux objectifs correspondant aux connaissances nécessaires pour le réussir; ces références permettent aussi de faire un lien entre cet exercice et le modèle statique de l'apprenant. Les exercices de niveau 1 ont en outre soit un modèle dynamique, soit un *modèle causal* si aucun modèle dynamique n'a pu être établi et ces modèles serviront de support à la *modélisation dynamique de l'apprenant*. Pourquoi imaginer un modèle causal associé à un exercice? Le modèle dynamique (ou modèle de tâche) n'est pas toujours disponible, d'autant plus qu'un bon modèle dynamique nécessitera souvent une analyse poussée de la tâche à résoudre à travers des protocoles d'observation des apprenants lorsqu'ils y travaillent. De plus, une tâche peut faire appel à l'utilisation de plusieurs connaissances sans pour autant qu'il soit possible ou facile de l'exprimer en termes de plan ou de comportement planifié et observable (par exemple, identifier un TERME COMPOSÉ). Dans ces cas, on peut utiliser un modèle causal pour exprimer l'influence de chaque connaissance nécessaire pour cet exercice dans la réussite de ce dernier. Dans notre cas particulier, le modèle causal correspond à un réseau Bayésien avec 2 types de noeuds: les "noeuds-cause" (*nc*) qui correspondent aux connaissances impliquées et le "noeud-exercice" (*ne*) qui représente l'exercice lui-même (Figure 4.2). Les liens de chaque *nc* vers un *ne* expriment la probabilité conditionnelle qu'un exercice soit réussi *sachant* que la connaissance associée à ce *nc* est acquise. Ces liens peuvent donc permettre de: prédire la réussite à un exercice, diagnostiquer

Figure 4.2: Modèle causal d'un exercice pour l'identification d'un terme



l'échec à un exercice (comme on le verra plus loin, il s'agit d'une façon alternative lorsqu'on ne dispose pas d'un modèle de tâche détaillé<sup>1</sup>). Pour enrichir la sémantique du modèle causal, nous y avons ajouté un troisième type de noeud pour exprimer *comment* chaque connaissance doit être employée dans l'exercice: les "noeuds-role" (*nr*) qui serviront surtout à la remédiation, pour indiquer ou suggérer à l'apprenant comment employer une connaissance impliquée dans l'exercice. Ces noeuds n'interviennent pas dans le traitement Bayésien du modèle causal qu'on expliquera plus tard. Enfin, chaque exercice a un *modèle épistémologique* qui permet de soutenir la remédiation épistémologique (Tchétagne et Nkambou[72]) comme activité pédagogique associée au diagnostic cognitif de cette illustration. Le modèle épistémologique est un modèle de connaissances associé à chaque connaissance intervenant dans l'exercice. Par exemple, l'énoncé "Identifier les LISTES ayant une syntaxe correcte" est un exercice qui fait appel au concept de "LISTE" entre autres. Le modèle épistémologique inclura le modèle de connaissances associé au concept de LISTE: un modèle exprimant les caractéristiques et attributs d'une liste dans PROLOG.

## 4.2 Modélisation cognitive en Programmation Logique

Plutôt que de se lancer dans la modélisation cognitive de l'ensemble de la PL, nous avons d'abord identifié 3 tâches de base et nous avons effectué une analyse cognitive par rapport aux connaissances impliquées dans ces tâches. Par rapport à la méthodologie annoncée au chapitre précédent, cette étape est une démonstration de l'implantation d'un diagnostic cognitif *pan-*

<sup>1</sup>Nous utilisons les termes "modèle dynamique de l'apprenant", "modèle de tâche", "graphe de tâche" pour référer à la même entité à savoir un modèle comportemental et cognitif de la tâche à résoudre

*pédagogique* dans un STI. Pour l'évaluation, il nous reste à concevoir 2 tâches un peu plus élaborées qui seront utilisées dans des tests en contexte réel. Les objectifs d'apprentissage suivent l'épistémologie de Gagné et al.[27]. On vise donc l'acquisition de connaissances en tant que concepts, règles, procédures et capacités de résoudre des problèmes. La modélisation de ces connaissances se fera donc en fonction de leurs caractéristiques épistémologiques.

L'interface des 2 premières tâches permet à l'apprenant de sélectionner les termes qu'il juge appropriés pour répondre à la question. L'interface de la troisième tâche permet à l'apprenant de poser des actions en plusieurs étapes.

*Première tâche: L'identification des variables*

Malgré leur simplicité, l'*identification* et l'*utilisation* d'une VARIABLE sont des notions clés dans l'acquisition de la grammaire en PL puisqu'elles seront mises à contribution plus tard. Les deux objectifs sont donc: l'acquisition du **concept** de VARIABLE à travers l'*identification* de ses instances et l'acquisition de la **règle** de VARIABLE LIÉE à travers *son application adéquate*. Dans les exercices correspondant à cette tâche, une seule connaissance est en jeu, soit le concept de VARIABLE, soit la règle de VARIABLE LIÉE. Dans ces cas, nous avons décidé d'utiliser comme modèle dynamique de ces tâches le modèle épistémologique (modèle du concept ou de la règle) associé à ces connaissances. En effet, le diagnostic dans ce cas se souciera d'analyser les éléments de la connaissance unique qui intervient dans cet exercice d'apprentissage.

Tableau 4.1: Modèles épistémologiques de quelques éléments dans la grammaire de la LP

CONCEPT	ATTRIBUT	CONTRAİNTE
VARIABLE	1 <sup>ère</sup> lettre	Doit être une majuscule
TERME COM- POSÉ	foncteur  arguments	Doit être un atome;Doit définir un prédicat; Doit définir une relation; Doivent être des termes valides

*Deuxième tâche: L'identification du TERME COMPOSÉ*

La figure 4.3 illustre le modèle causal associé à un exercice sur l'identification d'un TERME COMPOSÉ (TC). Le terme composé est également modélisé comme un *concept* et comme on

Figure 4.3: Modèle causal pour l'identification d'un terme composé

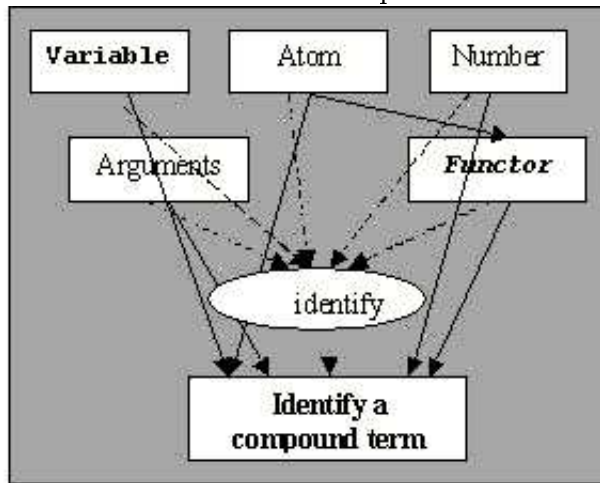
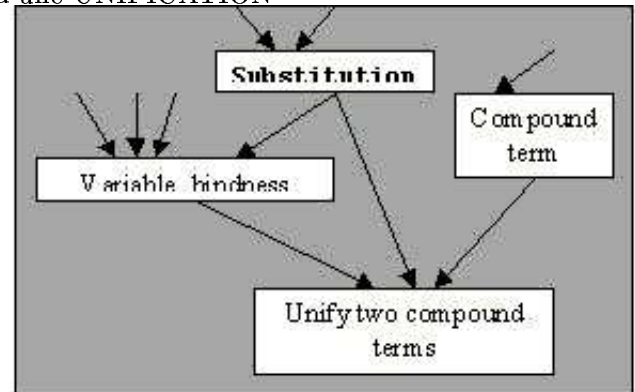


Figure 4.4: Modèle causal pour l'exécution d'une UNIFICATION



le voit au tableau 4.1, le modèle de connaissances associé contient ses attributs (par exemple le *FONCTEUR*) et les contraintes sur ces attributs (par exemple “*doit être un ATOME*”). Nous considérons donc que pour réussir cette tâche, l'apprenant doit: (1) s'assurer que les attributs d'un TC figurent dans l'expression présentée; (2) vérifier les valeurs de ces attributs en s'assurant qu'elles respectent les contraintes qui y sont associées. Plus d'une connaissance entre en jeu dans cette tâche, mais *a priori* elle n'est pas de nature procédurale. Pour en établir un modèle dynamique nous avons distingué 2 cas: un modèle à base de contraintes car on peut introduire des “fausses-bonnes réponses” exprès qui correspondent à une *misconception* ou à la violation d'une règle dans le domaine de la PL; un modèle causal pour traiter les sélections complètement fausses (pas de *misconception*).

*Troisième tâche: Exécution d'une unification impliquant un TERME COMPOSÉ*

La figure 4.4 illustre le modèle causal de cette tâche. L'apprenant doit utiliser 3 connaissances pour réussir cet exercice: identifier la structure du TC, substituer une variable et surtout comprendre l'implication de cette substitution, au moyen de la règle de la VARIABLE LIÉE dans certains cas (relié au concept de VARIABLE LIÉE). Cependant, nous avons choisi de modéliser l'unification comme une *procédure* dans l'épistémologie de [27] car il s'agit d'un algorithme bien défini dans le domaine de la PL. On pourra utiliser cette procédure pour suivre l'étudiant lors de sa performance.



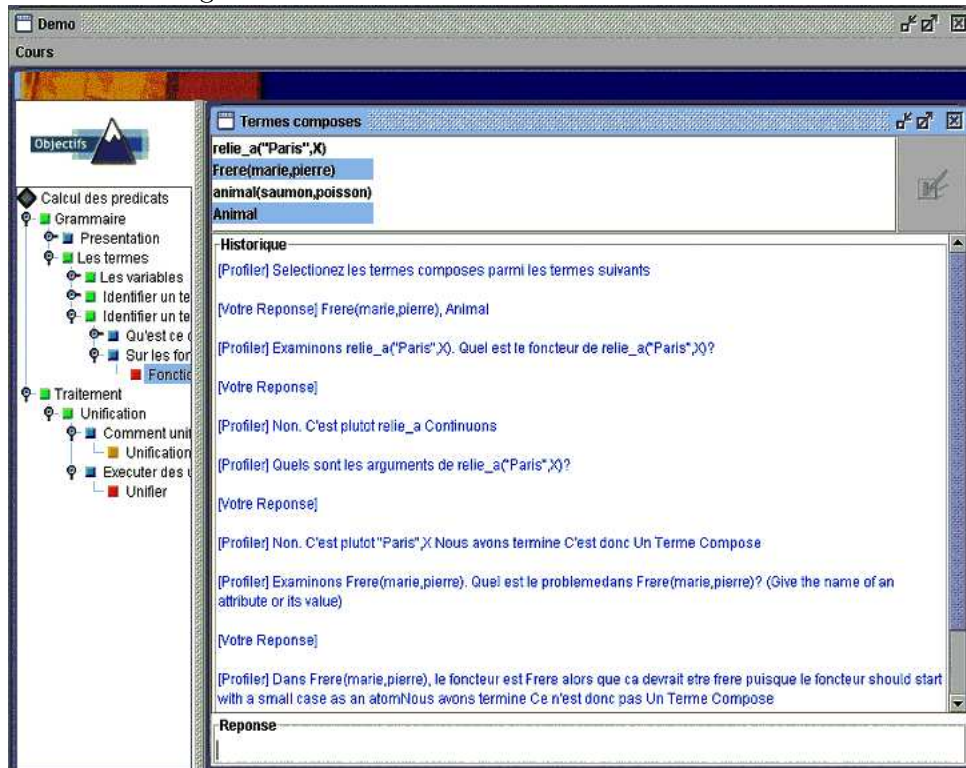
### 4.3 Diagnostic et Remédiation dans PLEARN

Ce que nous voulons dégager à travers cette *illustration* dans l'état de nos travaux, c'est que le processus de diagnostic s'accompagne d'autres activités pédagogiques dont l'absence dans la suite du diagnostic cognitif le rendrait pratiquement obsolète. Pour cette raison, nous allons brièvement présenter la technique de diagnostic que nous utilisons, mais bien que faisant elle-même partie des questions importantes de ce projet, elle ne pourra être présentée en détail pour le moment (Tchetagni et Nkambou[72], Tchetagni et al.[73], Nkambou et Tchetagni[50]). Le même argument est valable quant aux techniques de remédiation que nous utilisons: nous ne sommes pas encore au stade de juger l'opportunité pédagogique d'une approche particulière par rapport à une autre, dans un contexte donné d'instruction (théorie d'instruction assumée, théorie d'apprentissage assumée, domaine d'application, type de tâche particulière, modèle conatif de l'apprenant, etc.).

#### *Le diagnostic*

Nous utilisons un modèle de diagnostic épistémologique qui se concentre sur les connaissances manquantes ou mal utilisées par l'étudiant. Premièrement, rappelons que chaque exercice comporte soit un modèle de tâche, soit un modèle causal. Si on dispose d'un modèle de tâche, on peut utiliser une technique comme la reconnaissance de plan (*model tracing*) ou une bibliothèque d'erreurs pour inférer les connaissances manquantes lorsqu'un étudiant n'arrive pas à résoudre une tâche, tout dépend des informations disponibles. Si on ne dispose pas d'un modèle de tâche, on peut utiliser le modèle causal. Pour la première tâche, le diagnostic se fait sur la base de la détection d'une réponse absente dans la liste des réponses de l'apprenant ou encore d'une réponse présente mais qui ne le devrait pas. Puisque l'exercice ne fait appel qu'à une connaissance, cette dernière est immédiatement diagnostiquée. Dans la deuxième tâche, l'exercice fait appel à plusieurs connaissances. Si l'erreur provient d'une '*misconception*', le diagnostic est facilité puisque cette erreur a été expressément caractérisée. Par exemple à la figure 4.5, le fait de choisir le terme "Frere(pierre, marie)" comme un terme composé traduit une *misconception*: la structure de l'expression est effectivement celle d'un terme composé mais la restriction sur l'attribut "FONCTEUR" n'est pas respectée. Pour tout autre type d'erreur (par exemple la sélection de "Animal" qui est une mauvaise réponse sans particularité), le diagnostic: 1) génère les causes possibles de cette erreur et 2) trouve les causes les plus probables. Les deux étapes se font en utilisant le modèle causal de la tâche. Pour la troisième tâche, le diagnostic d'une erreur à une étape donnée correspond aux connaissances qui sont

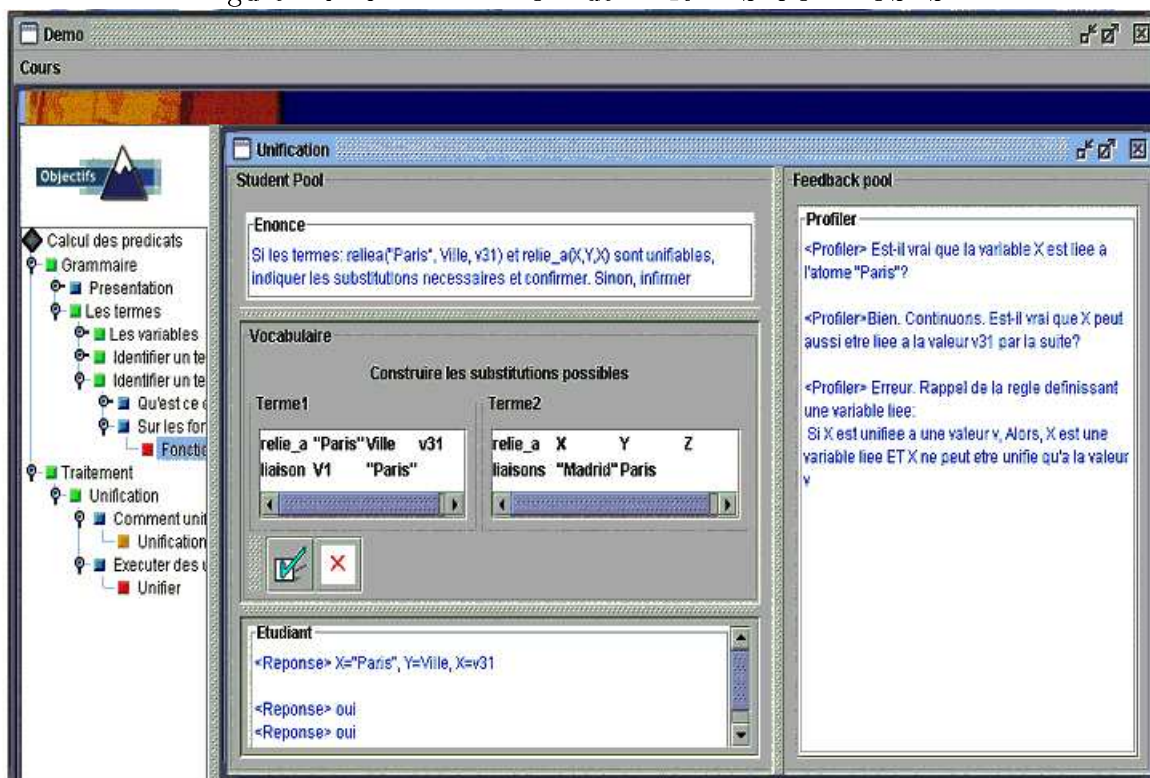
Figure 4.5: Sélection de TERMES COMPOSÉS



nécessaires à cette étape. Si plusieurs connaissances sont nécessaires, l’opération de diagnostic se déroule en deux temps, de la même façon que celle décrite dans le cas de la deuxième tâche.

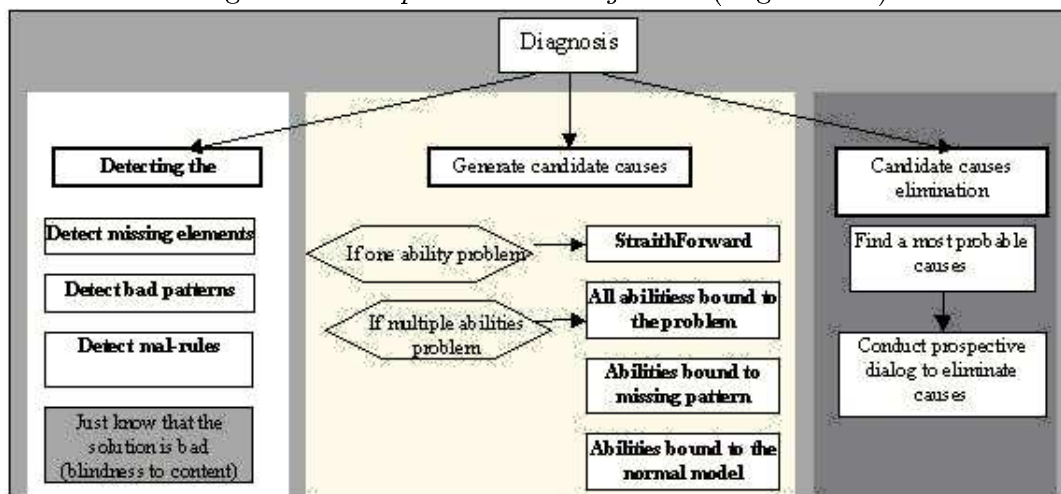
L’opération de diagnostic est résumée à la figure 4.7. Examinons les étapes de génération et d’élimination des causes. On peut déjà noter que ces étapes sont très générales et peuvent s’appliquer à tout processus de diagnostic, cognitif ou pas. Dans la tâche 1 par exemple, la génération et l’élimination ne font qu’un car il n’y a qu’une seule cause possible. Dans les 2 autres tâches, la génération de causes consiste juste à générer toutes les connaissances associées à la tâche en question (deuxième tâche) ou à une étape particulière de la tâche en question (troisième tâche) et ce en utilisant la liste des connaissances associées à chaque exercice ou étape. Par exemple dans la deuxième tâche, si la réponse de l’apprenant contient un terme qui n’est pas un TC sans pour autant correspondre à une “fausse-bonne réponse”, les causes potentielles qui sont générées seront:  $\{identification\ du\ foncteur,\ identification\ des\ arguments,\ validation\ du\ foncteur,\ validation\ des\ arguments\}$ . Mais ce résultat doit être d’une utilité pédagogique et dans ce sens, il faut éliminer les causes les moins probables pour concentrer le support pédagogique sur les besoins les plus urgents de l’apprenant. Pour cela, on utilise le modèle causal de chaque exercice et l’inférence Bayésienne par induction. L’induction bayésienne aussi appelée *Most Probable Explanation* (MPE, Pearl[56]) consiste à déterminer

Figure 4.6: UNIFICATION de TERMES COMPOSÉS



l'état des noeuds d'un réseau Bayésien qui maximise la probabilité d'une observation. Considérons que la sémantique de l'état de chaque noeud du réseau causal associé à un exercice est la suivante: si un  $nc$  a la valeur 0, alors on dit que la connaissance correspondante n'est pas acquise et s'il a la valeur 1, on considère qu'elle est acquise; si un  $ne$  a la valeur 0, alors l'exercice correspondant n'a pas été réussi, sinon, c'est le contraire. Dans la liste des causes {*identification du foncteur, identification des arguments, validation du foncteur, validation des arguments*}, on a par exemple déterminé que c'est lorsque le noeud correspondant à "*identification du foncteur*" a la valeur 0 que la probabilité d'échec dans cet exercice est maximale. Cette conclusion dépend évidemment des probabilités conditionnelles qui relient chaque "noeud-cause" au "noeud exercice" dans le modèle causal. Le résultat de la MPE donne donc une configuration du réseau qui explique le mieux l'échec à un exercice ([72, 50]). L'analyse de ce résultat permet de retenir uniquement les causes les plus probables comme diagnostic final. Notons que la technique de la MPE a une validité théorique et une analogie avec le raisonnement humain([56]) et c'est pour cette raison que nous l'avons choisie. Cependant, toute autre méthode aurait pu être appliquée, dépendamment du choix de modélisation des influences de chaque connaissance impliquée dans l'exercice, dans la réussite ou l'échec à

Figure 4.7: L'opération de diagnostic (Algorithme)



ce dernier. L'avantage de la MPE est qu'elle permet un diagnostic cognitif dans les problèmes qui font appel à un grand nombre de connaissances, sans qu'un modèle de tâche complet soit disponible. Le désavantage de cette technique est qu'elle tient seulement compte des facteurs reliés au problème (on peut corriger cette lacune en ajoutant d'autres noeuds): dans l'exemple précédent, seuls les facteurs reliés au concept de TC sont inclus dans le modèle causal de l'exercice.

### La remédiation

Une fois qu'un diagnostic a été posé, nous avons conçu le module tuteur de *PLEARN* de manière à ce qu'un processus de remédiation soit déclenché immédiatement. Ceci correspond au genre de choix que nous aimerions considérer dans la dimension pédagogique du diagnostic cognitif, à savoir la possibilité de commencer une remédiation qui non seulement pourrait aider l'étudiant à comprendre et corriger ses erreurs, mais pourrait aussi aider le tuteur à comprendre l'étudiant. Ceci est facilité lorsque le processus de remédiation permet à l'apprenant de s'exprimer, par exemple, à travers un dialogue avec le module tuteur ou encore à travers la résolution d'un exercice faisant appel aux connaissances diagnostiquées, et spécialement configuré comme exercice "thérapeutique" (En effet, un exercice ayant pour but de remédier une erreur ou de clarifier/préciser un diagnostic ne sera ni présenté, ni analysé de la même façon qu'un exercice de performance).

Nous avons élaboré une méthode de remédiation particulière appelée *Articulation de la connaissance diagnostiquée* ([72]) qui peut s'apparenter à une version interactive des *tactiques reliées aux erreurs de l'apprenant* dans [51]. Le but de cette technique est de "décortiquer

l'exercice" avec l'étudiant et de l'amener à constater lui-même son erreur ou à trouver la solution lui-même. En utilisant un dialogue socratique, on pose à l'étudiant des questions qui *exploitent la nature de la connaissance diagnostiquée*. À la figure 4.5, on sait que l'erreur provient de la sélection de la "fausse-bonne reponse" *Frere(pierre, marie)*: le problème se trouve au niveau de l'*attribut FONCTEUR* qui dans ce cas ne respecte pas la **restriction d'être un ATOME**. La série de questions sur la connaissance "TERME COMPOSÉ" exploitera la nature de ce *concept* et plus particulièrement, les restrictions reliées à son attribut "FONCTEUR". On parlera de remédiation épistémologique ([72, 73]) car elle exploite les caractéristiques épistémologiques d'une connaissance pour en favoriser l'acquisition (voir le dialogue de la figure 4.5: l'essentiel de l'interaction entre le module tuteur et l'apprenant se base sur la nature – la structure et le contenu – de la connaissance "TERME COMPOSÉ" en tant que concept).

#### 4.4 Conclusion

Le but de ce chapitre était d'illustrer comment le processus de diagnostic cognitif pourrait être déployé tout en y incorporant des activités pédagogiques qui mettent en valeur son rôle constructif, formatif et pédagogique. Dans notre cas particulier, nous avons illustré comment le processus de remédiation pouvait suivre celui de diagnostic en continu. Nous avons utilisé les notions de base en Programmation Logique comme domaine d'intérêt et nous avons brièvement décrit l'organisation architecturale et fonctionnelle du système. Notre objectif était de sensibiliser au fait que le processus de diagnostic incorpore ou fait nécessairement appel à des activités pédagogiques formatives (remédiation, révision du plan) ou des activités pédagogiques d'analyse (vérification du modèle de l'apprenant en fonction de son résultat, quête d'informations supplémentaires auprès de l'étudiant pour raffiner le diagnostic, etc.) dont le but est de mieux comprendre l'apprenant pour s'adapter à ses besoins réels ou plus urgents. L'objectif de notre projet est d'exprimer formellement ces aspects du diagnostic afin qu'un programmeur de STI puisse en tenir compte selon les souhaits d'un concepteur pédagogique. L'illustration de ce chapitre devait dégager le résultat ou les retombées de cette vision alternative du diagnostic.

## Références

- [1] S. Ainsworth, N. Major, S. Grimshaw, M. Hayes, J. Underwood, B. Williams, et D. Wood. Redeem: Simple intelligent tutoring systems from usable tools. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 8. Kluwer Academic Publishers, Dordrecht, 2003.
- [2] R. Anderson, J. *The architecture of cognition*. MA: Harvard University Press, Cambridge, 1983.
- [3] R. Anderson, J. et J Reiser, B. The lisp tutor. *Byte*, 10:159–175, 1985.
- [4] A. Arruarte, I. Fernandez-Castro, et J. Greer. The clai model: A cognitive theory of instruction to guide its development. *Journal of artificial intelligence and education*, 7(3/4):277–314, 1996.
- [5] A. Arruarte, B. Ferrero, I. Fernandez-Castro, M. Urretavizcaya, A. lvarez, et J. Greer. The iris authoring tool. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 9. Kluwer Academic Publishers, Dordrecht, 2003.
- [6] B. B. Ferrero, I. Fernandez-Castro, et M. Urretavizcaya. Using detective, a generic diagnostic system for procedural domains. In S. Lajoie et .M Vivet, editors, *Proceedings of the 10th International Artificial Intelligence in Education*, pages 667–670, Le Mans, 1999. IOS Press.
- [7] J. Baker. The roles of models in artificial intelligence and education research: a prospective view. *International Journal of Artificial Intelligence in Education*, 11:122–143, 2000.
- [8] P. Baxter, G., D. Elder, A., et R. Glaser. Knowledge based cognition and performance assessment in the science classroom. *Educational psychologist*, 31(2):133–140, 1996.
- [9] S. Blessing. A programming by demonstration authoring tool for model-tracing tutors. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 4. Kluwer Academic Publishers, Dordrecht, 2003.
- [10] S. Bostock. Evaluation of authoring systems. URL:[www.keele.ac.uk/depts/cs/Stephen\\_Bostock/docs/authass.htm](http://www.keele.ac.uk/depts/cs/Stephen_Bostock/docs/authass.htm).
- [11] J. Bourdeau, L. Winer, M. Chomienne, J. Vazquez-Abad, et B. Wasson. The baker's dozen: Design principles for collaborative telelearning. In J. Bourdeau et R. Heller, editors, *World Conference on Educational Multimedia, Hypermedia, & Telecommunications*, Charlottesville, 2000. VA: Association for the Advancement of Computing in Education, VA: Association for the Advancement of Computing in Education.
- [12] S. Brown, J. et R. Burton, R. Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science*, 2:155–191, 1978.
- [13] P. Brusilovsky et J. Vassileva. Evaluation methodologies for intelligent tutoring systems. *Int. J. Cont. Engineering Education and Lifelong Learning*, 13(1/2):75–95, 2003.
- [14] P. Brusilovsky et G. Weber. Elm-art: An adaptative versatile system for web-based education. *International Journal of Artificial Intelligence in Education*, 12(4):351–384, 2001.
- [15] R. Burton, R. Diagnosing bugs in a simple procedural skill. In H. Sleeman, D. et S. Brown, J., editors, *Intelligent Tutoring Systems*, pages 157–183, London, 1982. Academic Press. London.
- [16] N. Capuano, M. Marsella, et S. Salerno. Abits: An agent based intelligent tutoring system for distance learning. In G. Gauthier, C. Frasson, et K. VanLehn, editors, *Intelligent Tutoring Systems, 5th International Conference, ITS2000, Montréal, Canada, June 19-23, 2000, Proceedings*, number 1839 in Lecture Notes in Computer Sciences, pages 31–38, Amsterdam, 2000. Springer Verlag.
- [17] R. Carnonell, J. Ai in cai: An artificial intelligence approach to computer-assisted instruction. *IEEE Transactions on Man-Machine Systems*, 11:199–202, 1970.
- [18] B. Carr et I. Goldstein. Overlays: A theory of modeling for computer aided instruction. Rapport technique, MIT, 1977.
- [19] W. Chan, T. Learning compaign systems. *Journal of Artificial Intelligence in Educa-*

- tion, 7(2):125–159, 1996.
- [20] Brant A. Cheikes. GIA: An agent-based architecture for intelligent tutoring systems. In Tim Finin et James Mayfield, editors, *Proceedings of the CIKM '95 Workshop on Intelligent Information Agents*, Baltimore, Maryland, 1995.
  - [21] C. Conati, S. Gertner, A., K. VanLehn, et J. Druzdzel, M. On-line student modeling for coached problem solving using Bayesian networks. In A. Jameson, C. Paris, et C. Carlo Tasso, editors, *User Modeling: Proceedings of the Sixth International Conference, UM97*, pages 231–242. Springer Wien New York, Vienna, New York, 1997.
  - [22] A. Damasio. *Descartes' Error: Emotion, Reason and the Human Brain. 2000.* uill, HarperCollins Publishers Inc., New York NY,, 2000.
  - [23] G. De LandSheere. *Dictionnaire de l'évaluation et de la recherche en éducation.* PUF, 1992.
  - [24] Dillenbourg et J. Self. A framework for learner modelling. Rapport technique, Lancaster Univ., 1992.
  - [25] T. Finin, R. Fritzson, D. McKay, et R. McEntire. Kqml as an agent communication language. In N. Adam, K. Bhargava, et Y. Yesha, editors, *3rd International Conference on Information and Knowledge Management*, Gaithersburg, Maryland, 1994. ACM Press, New York.
  - [26] W. Finin, T. Gums: A general user modeling shell. In A. Kobsa et Wahlster.W., editors, *User Models in Dialog Systems*, pages 411–430. Springer, Berlin, Heidelberg, 1989.
  - [27] R. Gagne, L. Briggs, et W. Wager. *Principles of Instructional Design.* Fort Worth: Harcourt Brace Jovanovich, 1992.
  - [28] A. Gertner et K. VanLehn. Andes: A coached problem solving environment for physics. In G. Gauthier, C. Frasson, et K. VanLehn, editors, *5th International Conference, ITS 2000, Montreal Canada, June*, pages 133–142. Springer Verlag, 2000.
  - [29] M. Giraffa, L. et M. Vicari, R. The use of agents techniques on intelligent tutoring systems. In *RIBIE 98, IV Congresso da Rede Iberoamericana de Informtica Educativa*, 1998.
  - [30] R. Glaser. The assessment of student achievement: the hundred years's war conflicts, engagements, skirmishes and attempts at peace. *Annual meeting of American Education Research Association*, 1999.
  - [31] J. GrandBastien. Teaching expertise is at the core of its research. *International Journal of Artificial Intelligence and education*, 10:335–349, 1999.
  - [32] J. Greer, G. McCalla, J. Vassileva, R. Deters, S. Bull, et L. Kettel. Lessons learned in deploying a multi-agent learning support system: The i-help experience. In J. Moore, L. Johnson, W., et L. Redfield, C., editors, *10th International Artificial Intelligence in Education, San Antonio, TX*, pages 410–421, Amsterdam, 2001. IOS Press.
  - [33] E. Hospers, M., A. Kroezen, R. Nijholt, et D. Heylen. Developing a generic agent-based intelligent tutoring system and applying it to nurse education. In *Proceedings IEEE International Conference on Advanced Language Technologies*, pages 443–444, Athens, Greece, 2003. IEEE Computer Society.
  - [34] M. Ikeda et R. Mizogouchi. Fits: A framework for its – a computational model of tutoring. *Journal of Artificial Intelligence in Education*, 5(3):319–348, 1994.
  - [35] S. Jean, E. Delozanne, P. Jacoboni, et B. Grugeon. Design and validation of a model based diagnosis system. In P. Brna, M. Baker, et K. Stenning, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software.* C-LEMMAS,Corsica, 1999.
  - [36] S. Jean, E. Delozanne, P. /Jacoboni, et B. Grugeon. A diagnosis based on a qualitative model of competence in elementary algebra. In S. Lajoie et .M Vivet, editors, *Proceedings of the 10th International Artificial Intelligence in Education*, pages 491–498, Le Mans, 1999. IOS Press Inc.
  - [37] L. Johnson, W., W. Rickel, J., et C. Lester, J. Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education*, 11:47–78, 2000.
  - [38] A. Leontieva. *Activité, conscience et personnalité.* Éditions du progrès, 1975.
  - [39] N. Livergood, L. From computer-assisted instruction to intelligent tutoring systems. *Journal of Artificial Intelligence in Education*, 2(3):39–50, 1991.

- [40] I. Machado, A. Martins, et A. Paiva. One for all and all for one: a learner modelling server in a multi-agent platform. In *Proceedings of the 7th International Conference on User Modeling*, pages 211–221, 1999.
- [41] A. Mark, M. et J. Greer. Evaluation methodologies for intelligent tutoring systems. *Journal of Artificial Intelligence in Education*, 4(2/3):129–153, 1993.
- [42] D. Merrill, M. Ide: Using knowledge objects to design instructional learning environments. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 7. Kluwer Academic Publishers, Dordrecht, 2003.
- [43] J. Mislevy, R., S. Steinberg, L., et . Almond, R. On the roles of task model variables in assessment design. In S. Irvine et P. Kyllonen, editors, *Generating items for cognitive tests: Theory and practice*, pages 97–128, London, 2002. Hillsdale, NJ: Erlbaum.
- [44] J. Mislevy, R., S. Steinberg, L., et G. Almond, R. On the structure of educational assessments. Rapport technique CSE-597, University of Maryland, 2003.
- [45] T. Murray. Eon: Authoring tools for content, instructional strategy, student model, and interface design. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 11. Kluwer Ac. Publishers, Dordrecht, 2003.
- [46] T. Murray. An overview of intelligent tutoring system authoring tools: Updated analysis of the state of the art. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 17. Kluwer A. Publishers, Dordrecht, 2003.
- [47] A. Newell et H.A. Simon. *Human Problem Solving*. Prentice Hall, Englewood Cliffs, 1972.
- [48] R. Nkambou, C. Frasson, et G. Gauthier. Cream-tools: An authoring environment for knowledge engineering in intelligent tutoring systems. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 10. Kluwer Academic Publishers, Dordrecht, 2003.
- [49] R. Nkambou et F. Kabanza. Designing intelligent tutoring systems: A multiagent planning approach. *ACM SIGCUE Outlook*, 27(2):46–60, 2001.
- [50] R. Nkambou et J. Tchétagni. Diagnosing student errors in e-learning environments using mpe theory. In Accepté, editor, *Web Based Education 2004*, 2004.
- [51] S. Ohlsson. Principles of intelligent tutoring. In W. Lawler R. et M. Yazdani, editors, *Artificial Intelligence and Education*, pages 203–238. Ablex Pub., Norwood, 1987.
- [52] S. Ohlsson. Constraint-based student modeling. *Journal of Artificial Intelligence in Education*, 3(4):429–447, 1994.
- [53] S. Ohlsson et P. Langley. Psychological evaluation of path hypothesis in cognitive diagnosis. In H. Mandl et A. Lesgold, editors, *User Models in Dialog Systems*, pages 42–62. Springer, New York, 1988.
- [54] A. Paiva et J. Self. Tagus – a user and learner modeling workbench. *User Modeling and User-Adapted Interaction*, 4(3):197–226, 1995.
- [55] S. Papert. *Logo Philosophy and Implementation*, chapter What is Logo? And who needs it? LCSI, 1999.
- [56] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.
- [57] W. Pohl. Logic-based representation and reasoning for user modeling shell systems. *User Modeling and User-Adapted Interaction*, 9(3):217–282, 1999.
- [58] L. Pressey, S. A simple apparatus which gives tests and scores and teaches. *School and Society*, 23(586):373–376, 1926.
- [59] Z. Pylyshyn. *Computing in Cognitive Sciences. Dans Foundations of Cognitive Sciences*. MIT Press, Cambridge, 1989.
- [60] T. Richards et J. Fukuzawa. A checklist for evaluation of courseware authoring systems. *Educational Technology*, 29(10):24–29, 1989.
- [61] S. Ritter. Communication, cooperation and competition among multiple tutor agents. In B. du Boulay et R. Mizoguchi, editors, *Artificial Intelligence in Education: Knowledge*



- and Media in Learning Systems*, number 39 in *Frontiers in Artificial Intelligence and Applications*, pages 31–38, Amsterdam, 1997. IOS Press.
- [62] S. Ritter et R. Koedinger, K. An architecture for plug-in tutor agents. *Journal of artificial intelligence and education*, 7(3/4):315–347, 1996.
- [63] J. Self. Formal approaches to student modelling. Rapport technique, Lancaster Univ., 1994.
- [64] J. Self. The defining characteristics of intelligent tutoring systems research: Itss care, precisely. *International Journal of Artificial Intelligence and education*, 10:350–364, 1999.
- [65] V. Shute. Smart: Student modeling approach for responsive tutoring. *User Modeling and User-Adapted Interaction*, 5:1–44, 1995.
- [66] V. Shute, S. Lajoie, et K. Gluck. *Individualized and group approaches to training*. In S. Tobias and J. D. Fletcher (Eds.), *Training and retrainng: A handbook for business, industry, government and the military*. New York: Macmillan, 2000.
- [67] V. Shute et W. Regian, J. Principles for evaluating intelligent tutoring systems. *Journal of Artificial Intelligence in Education*, 4(2/3):245–271, 1993.
- [68] M. Silveira-Netto, L. Dihl, L. Fraga, R. Woszezenki, L. Oliveira, J. Francisco, C. Machado, C. Nogueira, et M. Notargiacomo. Multi-agent systems applied to intelligent tutoring systems. In R. Arabnia, H. et Y. Mun, editors, *Proceedings of the International Conference on Artificial Intelligence*, pages 361–366, Las Vegas, Nevada, 2002. CSREA Press.
- [69] D. Sleeman et S. Brow, J. *Intelligent Tutoring Systems*. Academic Press, 1981.
- [70] E. Solloway et K. Bielaczyc. Interactive learning environments: Where they’ve come from and where they’re going. In *Proceedings of CHI95*, pages 347–348, 1995.
- [71] J. Tchétagni, P. et R. Nkambou. Hierarchical representation and evaluation of the student in an intelligent tutoring system. In A. Cerri, S., G. Gouardères, et F. Paraguau, editors, *Intelligent Tutoring Systems, 6th International Conference, ITS2002, Biarritz, France, June 2-7, 2002, Proceedings*, pages 708–718, Heidelberg, 2002. Springer Verlag.
- [72] J. Tchétagni, P. et R. Nkambou. A framework for diagnosis and epistemological remediation in learning logic programming. In À préciser, editor, *2nd International Conference on Computer Aided Learning in Engineering Education , CALIE04, Grenoble, France, February 18-22, 2004, Proceedings*, page À préciser. Accepté, 2004.
- [73] J. Tchétagni, P., R. Nkambou, et F. Kabanza. Epsitemological remediation in intelligent tutoring systems. In Accepte, editor, *The 17th International Conference on Industrial and Engineering Applications of Artificial Intelligence & Expert Systems*, 2004.
- [74] L. Thorndike, E. *Education: A first book*. New York: The MacMillan Company, 1912.
- [75] D. Towne. Automated knowledge acquisition for intelligent support of diagnostic reasoning. In T. Murray, S. Blessing, et S. Ainsworth, editors, *Authoring Tools for Advanced Technology Learning Environments: Toward cost-effective adaptive, interactive, and intelligent educational software*, chapter 5. Kluwer Academic Publishers, Dordrecht, 2003.
- [76] K. Van Marcke. Gte: An epistemological approach to instructional modelling. *Learning Sciences*, 26(3/4):147–191, 1998.
- [77] J. Varela, F., E. Thompson, et E. Rosch. *L’inscription corporelle de l’esprit*. Editions du Seuil, Paris, 1993.
- [78] B. Wasson. Facilitating dynamic pedagogical decision making: Pepe and gte. *Learning Sciences*, 26(3/4):299–316, 1998.
- [79] C. Webber, S. Pesty, et N. Balacheff. A multi agent and emergent approach to learner modelling. In F. van Harmelen, editor, *15th European Conference on Artificial Intelligence*, pages 98–102, Lyon, FRANCE, 2002. IOS Press, Amsterdam.
- [80] E. Wenger. *Artificial Intelligence and Tutoring Systems*. Morgan Kauffman, S. Mateo, CA, 1987.
- [81] L. Winer, M. Chomienne, et J. Vázquez-Abad. The electrical circuit simulator : Towards a distributed and collaborative science learning laboratory. *The American Journal of Distance Education*, 14(1):47–62, 2000.

## Annexe A

### Tableaux récapitulatifs

Tableau A.1: Position par rapport à l'état de l'art: synthèse (1)

DOMAINE DE RECHERCHE	RÉSULTATS	NOTRE POSITION
<b>Specification formelle des services du module apprenant</b>		
Systèmes modulaires	<i>Plug-in Tutors</i> [62, 61]: Communication Interface/Module apprenant Communication entre multiples tuteurs	Pas de specification formelle du rôle du module apprenant <i>via services</i> ; Illustration de la notion de <i>module</i> dans un contexte particulier
SMAs Agents proactifs	GIA[20]: Gestion des messages entre confédération d'agents; Contenu des messages explicite mais pas formel	Définir formellement la teneur des requêtes reçues par l'agent du module apprenant
	ABITS[16]: Agent de modélisation de l'apprenant livre des informations sur son état cognitif	La nature et le destinataire de ces informations ne sont pas formellement définies
	GUILLY[68]: Agent "coach": fournit de l'aide aux apprenants en utilisant leur modèle	La nature et la communication des requêtes de <i>GUILLY</i> n'est pas spécifié formellement
Applications	<i>I-HELP</i> [32]: Gestion de la collaboration entre apprenants dans un système d'apprentissage basé sur le Web par des agents: ils utilisent les informations sur l'état des connaissances de l'apprenant pour <i>associer</i> les interlocuteurs.	Seul l'état des connaissances de l'apprenant est utilisé
Les <i>SHELLs</i>	BGP-MS[57],GUMS[26]: <i>Shells</i> de modélisation de l'utilisateur dans un système général; Formalisme de représentation et de raisonnement basés sur la logique	Les services offerts restent trop généraux pour le contexte d'un STI: ajout et retrait d'une assertion de la base de connaissances représentant l'apprenant
	TAGUS[54]: <i>Shell</i> de modélisation de l'utilisateur dans un STI; offre 2 services principaux pour s'encquérir de la nature d'une connaissance et pour mettre à jour le module apprenant	Si on essaie d'étudier la relation modèle apprenant/ autres composantes, on pourrait <i>formellement</i> identifier, justifier et caractériser d'autres services
	FITS[34]: Outil générique pour la conception de STIs; du côté apprenant, accent sur les fonctionnalités de diagnostic et de support du diagnostic dans l'instruction	<i>Idem.</i> que par rapport à <i>TAGUS</i>

Tableau A.2: Position par rapport à l'état de l'art: synthèse (2)

DOMAINE DE RECHERCHE	RÉSULTATS	NOTRE POSITION
<b>Specification formelle du processus de diagnostic cognitif</b>		
Systèmes auteurs orientés pédagogie	IDE[42]: Utilisation des objets de connaissances pour modéliser l'instruction; Modélisation du diagnostic basé sur l'état des objets d'apprentissage	Le diagnostic doit aussi tenir compte de l'apprenant
	GTE[76]: Modélisation de l'expertise pédagogique en général	Le diagnostic n'est pas considéré à part entière mais fait partie de la tâche d'enseignement qu'est l'évaluation
	Cream-Tools[48], REDEEM[1], EON[46]: Ces systèmes modélisent des méthodes et outils pédagogiques généraux qui peuvent être d'un grand support dans le processus de diagnostic cognitif en tant qu'activité pédagogique	Mais ces systèmes restent généraux et la modélisation particulière du diagnostic cognitif n'y est pas envisagée
Systèmes auteurs orientés performance	DEMONSTR8[9]: Permet à un auteur de construire <i>une technique de diagnostic</i> basée sur le <i>model tracing</i>	La technique de diagnostic utilisée dans un STI n'est qu'un aspect du processus de diagnostic cognitif qui à notre avis implique d'autres aspects plus pédagogiques mais tout aussi importants
	DIAG[75]: Permet de supporter les apprenants lors d'une tâche de <i>diagnostic d'appareils défectueux</i>	Non Applicable. Mmais les techniques de cet article pourraient inspirer une partie de notre travail sur la modélisation du diagnostic cognitif
Systèmes auteurs hybrides	IRIS[5]: Avec DETECTIVE, permet à un auteur de concevoir des exercices selon le type de connaissances qu'il vise et d'en déduire une approche de diagnostic cognitif appropriée pour cette exercice	DETECTIVE modélise les <i>techniques/approche de diagnostic cognitif dans les systèmes d'apprentissage</i> , mais ne permet pas à l'auteur de spécifier ses préférences par rapport à d'autres aspects de ce processus