



**HAL**  
open science

## Systemes à base de connaissances pour des EIAH fondés sur le RàPC

Nathalie Guin, Marie Lefevre

► **To cite this version:**

Nathalie Guin, Marie Lefevre. Systemes à base de connaissances pour des EIAH fondés sur le RàPC. Journée EIAH&IA 2013, May 2013, Toulouse, France. pp.1. hal-00824949v2

**HAL Id: hal-00824949**

**<https://hal.science/hal-00824949v2>**

Submitted on 23 May 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Systemes à base de connaissances pour des EIAH fondés sur le RàPC

Nathalie Guin, Marie Lefevre

Université de Lyon, CNRS  
Université Lyon 1, LIRIS, UMR5205, F-69622, France  
{Nathalie.Guin, Marie.Lefevre}@liris.univ-lyon1.fr

**Résumé.** Cet article présente tout d'abord comment nous nous sommes inspirés d'un paradigme issu de l'Intelligence Artificielle, le Raisonnement à Partir de Cas, pour proposer un processus permettant l'apprentissage de méthodes de résolution de problèmes fondées sur une classification des problèmes du domaine. Ce processus d'apprentissage ayant été mis en œuvre dans un EIAH destiné à enseigner la résolution de problèmes arithmétiques à l'école primaire, nous décrivons également les bases de connaissances permettant au système de diagnostiquer les réponses de l'élève et de lui fournir des explications.

**Mots-clés :** EIAH, système à base de connaissances, Raisonnement à Partir de Cas, diagnostic, explications.

## 1 Introduction

Le projet AMBRE (Apprentissage de Méthodes Basé sur le Raisonnement à partir de l'Expérience) est une étude pluridisciplinaire dont l'objectif est de concevoir des Environnements Informatiques pour l'Apprentissage Humain (EIAH) destinés à favoriser l'acquisition de méthodes. Issues de recherches en didactique des disciplines, de telles méthodes permettent, dans un domaine donné, de reconnaître la classe d'un problème et d'être capable de savoir quelle technique de résolution appliquer pour résoudre ce problème.

Nous montrons dans cet article comment nous nous sommes inspirés du Raisonnement à Partir de Cas (RàPC) pour concevoir un processus favorisant l'apprentissage d'une méthode par un apprenant. Ce processus a été mis en œuvre dans un EIAH destiné à enseigner une méthode de résolution de problèmes arithmétiques aux élèves de l'école primaire. Dans un deuxième temps, nous nous intéressons au système à base de connaissances permettant à cet EIAH d'assister l'apprenant dans sa résolution et dans son apprentissage de la méthode, en lui fournissant de l'aide, en diagnostiquant ses réponses et en lui proposant des explications adaptées.

## **2 Le projet AMBRE : s'inspirer du RàPC pour enseigner des méthodes**

Nous présentons dans cette partie quelles sont les méthodes que nous souhaitons enseigner dans le projet AMBRE, et pourquoi nous nous sommes inspirés du RàPC pour concevoir le principe des EIAH AMBRE.

### **2.1 Enseigner des méthodes**

Le projet AMBRE a pour objectif de concevoir des environnements informatiques pour l'apprentissage de méthodes. Il se fonde sur des recherches en didactique des disciplines qui proposent d'enseigner explicitement des méthodes [1]. Pour le projet AMBRE, une méthode permet de reconnaître la classe d'un problème et d'être capable de savoir quelle technique de résolution appliquer pour résoudre ce problème.

Pour concevoir un EIAH, Rogalski propose de partir des connaissances telles qu'elles devraient fonctionner chez l'apprenant après l'apprentissage [1]. Toutefois, même s'il existe une méthode, il n'est pas forcément souhaitable de la présenter explicitement à l'apprenant. En effet, les termes définissant les classes de problèmes et les techniques de résolution ne sont pas toujours connus des apprenants. Il semble de plus préférable que l'apprenant soit actif et se construise sa propre méthode.

Pour aider l'apprenant à construire sa propre méthode, nous avons choisi de l'inciter à mettre en œuvre un raisonnement par analogie, c'est-à-dire à se remémorer une expérience précédemment vécue (ici, un problème proche déjà rencontré) et à l'adapter pour résoudre un nouveau problème. En effet, la littérature en psychologie cognitive montre que cette forme de raisonnement permet d'acquérir des connaissances abstraites (par exemple [2, 3]). Ces travaux ont montré que certaines activités telles que la comparaison de problèmes [3], ou l'adaptation d'un problème déjà vu pour résoudre un nouveau problème [4] favorisent la généralisation de connaissances. Néanmoins, le raisonnement par analogie n'est pas toujours spontané [2] et l'acquisition de connaissances lors de la résolution par analogie n'est pas toujours automatique. Aussi, nous avons choisi d'inciter l'apprenant à utiliser de manière systématique un paradigme de raisonnement issu de l'Intelligence Artificielle : le RàPC.

### **2.2 Le Raisonnement à partir de cas**

Le RàPC est un paradigme fondé sur la réutilisation d'exemples. Il est issu des recherches en intelligence artificielle sur la résolution de problèmes. Ce principe de résolution peut être décrit par un ensemble d'étapes séquentielles (élaboration, remémoration, adaptation, test/révision, mémorisation) que l'on représente souvent par un cycle [5]. À partir d'un problème cible, on élabore le cas en utilisant des connaissances générales et en ne gardant que les informations pertinentes [6]. Puis on se remémore un cas proche (le cas source) en cherchant dans la base de cas ; on adapte ensuite la solution du cas source afin d'obtenir une solution au problème cible. On

teste cette solution qu'on révisé éventuellement, puis on mémorise le cas cible dans la base de cas pour une réutilisation future.

### 2.3 Le cycle AMBRE

Dans le projet AMBRE, le RàPC n'est pas utilisé par le système mais proposé à l'apprenant comme une stratégie d'apprentissage. Pour chaque problème, l'apprenant doit passer par toutes les étapes du RàPC, de l'élaboration du cas à sa mémorisation, pour retrouver lui-même un cas approprié au problème et l'adapter au problème à résoudre. Par ailleurs, comme l'apprenant n'a *a priori* pas d'expérience dans le domaine étudié, des problèmes résolus lui sont proposés.

En effet, lorsque les apprenants sont novices dans un domaine, ils ont peu ou pas d'expérience dans le domaine étudié. Aussi, dans cette situation, des cas représentatifs de la situation (par exemple des problèmes résolus) peuvent leur être présentés pour leur donner une première expérience dans le domaine. De nombreuses études conduites en psychologie [7, 8] ont montré l'intérêt pour l'apprentissage de présenter des problèmes résolus aux apprenants. Elles ont également mis en évidence le rôle sur l'apprentissage de la similarité entre les problèmes présentés successivement [9]. Ainsi, il semble que l'élaboration de connaissances générales soit favorisée lorsqu'on adapte un problème ayant des traits de surface proches du problème à résoudre [4]. Nous avons pris en compte l'ensemble de ces résultats de psychologie cognitive pour concevoir le principe des EIAH AMBRE.

Ainsi, une séance de travail avec un EIAH AMBRE se divise en deux phases. La première phase est destinée à présenter à l'apprenant des problèmes résolus (problèmes-types) qui serviront de référence par la suite. Après plusieurs séances, l'apprenant aura vu un problème-type correspondant à chaque classe de problèmes définie au sein de la méthode.

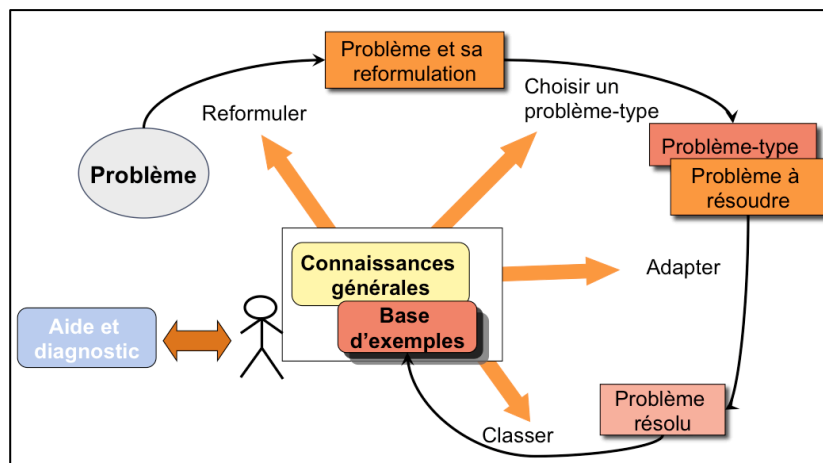


Fig. 1. Le cycle AMBRE

La deuxième phase de la séance est consacrée à la résolution par l'apprenant de plusieurs problèmes. L'EIAH guide l'apprenant dans sa résolution à travers cinq étapes inspirées du cycle du RàPC (cf. figure 1). Après une première étape qui consiste à lire l'énoncé du problème, l'apprenant doit reformuler le problème en identifiant les éléments de l'énoncé pertinents pour la résolution (étape analogue à l'élaboration du RàPC). Dans une troisième étape, l'apprenant choisit parmi les problèmes-types qui lui ont été présentés celui qui lui semble le plus proche du problème à résoudre (étape analogue à la remémoration). Il identifie ainsi de manière implicite la classe du problème à résoudre. Ensuite, l'apprenant rédige la solution du problème en s'inspirant de la solution du problème-type (étape analogue à l'adaptation). Cette adaptation devrait l'amener à identifier une technique de résolution adaptée aux problèmes de cette classe, via une généralisation des connaissances d'adaptation qu'il utilise. La littérature en psychologie cognitive [4] indique en effet que l'acquisition de connaissances générales est favorisée si l'on demande d'abord à l'apprenant d'adapter des problèmes proches en termes de traits de surface (comme sur la figure 2), avant de lui proposer d'adapter des problèmes comportant des traits de surface plus éloignés (avec par exemple d'autres objets que des billes). La dernière étape consiste pour l'apprenant à associer le problème qu'il vient de résoudre à un problème-type, de manière à constituer des groupes de problèmes correspondant aux classes, un groupe étant représenté par un problème-type (étape analogue à la mémorisation).

Ces étapes sont donc guidées par le système mais effectuées par l'apprenant. L'étape de révision du RàPC est remplacée dans le cycle AMBRE par un diagnostic du système. Celui-ci évalue la production de l'apprenant à chaque étape et lui propose des explications pour l'amener à comprendre ces erreurs et à corriger ses réponses.

Fig. 2. Étape d'adaptation dans l'EIAH AMBRE-add

## 2.4 L'EIAH AMBRE-add

Nous avons appliqué le cycle AMBRE au domaine des problèmes additifs (et soustractifs) traités à l'école primaire [10]. Il s'agit de problèmes décrivant des situations concrètes, par exemple un jeu de billes. Certaines étapes du cycle ont été découpées en sous-étapes spécifiques au domaine lors de la conception de l'EIAH AMBRE-add. Par exemple la rédaction de la solution du problème lors de l'étape d'adaptation est découpée en quatre sous-étapes (cf. figure 2).

Dans l'étape de reformulation du problème, l'élève doit utiliser un schéma pour décrire les éléments de l'énoncé pertinents pour la résolution (cf. figure 2). Cette reformulation sous forme de schéma est fondamentale pour l'apprentissage de la méthode, car si on fait abstraction des valeurs numériques, le schéma représente la classe du problème. Cette schématisation aide également l'élève pour l'étape de choix du problème-type le plus proche (appelé *modèle* sur l'interface destinée aux enfants).

## 3 Quel système à base de connaissances pour soutenir l'apprentissage ?

Comme nous l'avons vu dans la partie précédente, AMBRE propose à l'apprenant de résoudre les problèmes en utilisant le paradigme du RàPC. Cependant, Rogalski recommande que pour concevoir un tuteur intelligent, on parte des connaissances telles qu'on veut qu'elles fonctionnent chez l'apprenant *après* l'apprentissage [1]. Il précise que le résolveur d'un « tuteur donneur de leçons de méthodes » doit fonctionner selon les méthodes qu'il veut enseigner. Le résolveur que nous utilisons dans le projet AMBRE fonctionne justement selon les méthodes que nous souhaitons enseigner. Il est donc fondé sur la classification des problèmes du domaine, et n'utilise pas le paradigme du RàPC pour résoudre les problèmes. Nous sommes donc confrontés à la difficulté d'assister l'apprenant dans sa résolution de problèmes, en nous fondant sur des connaissances de type classification, alors que lui utilise l'analogie entre problèmes. C'est la notion de problème prototypique représentant une classe qui permet de faire le lien entre les connaissances du système et le raisonnement de l'apprenant. Comme nous le verrons, il est en effet nécessaire que le système dispose de connaissances pour soutenir le RàPC, comme par exemple des connaissances d'analogie.

Nous commençons par décrire les fonctionnalités d'un EIAH AMBRE pour lesquelles il est nécessaire de faire appel aux systèmes à base de connaissances. Nous précisons ensuite les différentes bases de connaissances qu'il faut définir dans un domaine donné pour mettre en œuvre ces systèmes à base de connaissances.

En effet, notre démarche dans le cadre du projet AMBRE consiste à définir une architecture fondée sur des modèles de connaissances et des processus de raisonnement sur ces connaissances indépendants de la méthode à enseigner. Pour un domaine et une méthode donnée (par exemple les problèmes arithmétiques), il faut instancier les modèles de connaissances afin de définir des bases de connaissances spécifiques du domaine que les processus génériques peuvent exploiter. Nous utiliserons le domaine des problèmes arithmétiques pour illustrer nos propos.

### 3.1 Quelles fonctionnalités d'assistance requièrent des connaissances ?

Afin d'assister l'apprenant dans sa résolution de problème, et dans le but de favoriser l'apprentissage, le système peut d'une part répondre à une demande d'aide de l'apprenant, et d'autre part diagnostiquer ses réponses pour chaque étape du cycle et l'aider à réviser une réponse erronée en lui proposant des explications pertinentes. L'EIAH propose des aides et des explications de différentes natures selon l'étape en cours de réalisation et des erreurs commises. Pour AMBRE-add, la conception de ces messages a été réalisée à partir d'une analyse *a priori* des difficultés rencontrées et des erreurs possibles dans chacune des étapes et sous-étapes proposées par le logiciel.

Les messages d'aide consistent souvent à rappeler à l'apprenant ce qu'il doit faire. Cette aide peut également renvoyer l'élève vers des pages de rappel (par exemple les différents schémas utilisés pour reformuler un problème arithmétique) ou vers des outils favorisant la comparaison entre problème-type et problème à résoudre à travers la coloration des éléments de l'énoncé et de la reformulation ayant la même fonction. (cf. figure 3).

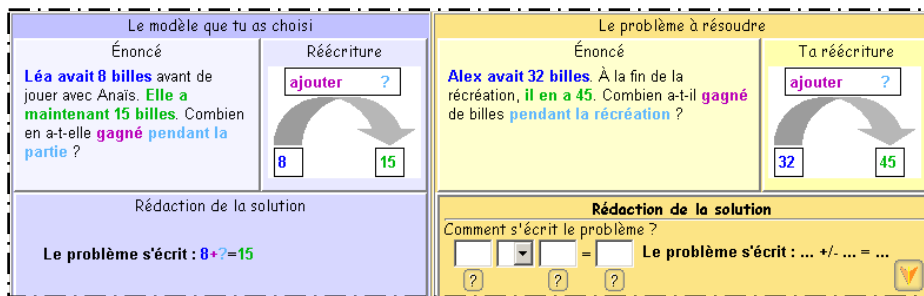


Fig. 3. Mise en évidence de l'analogie entre problème-type et problème à résoudre

Le système doit pouvoir effectuer un diagnostic de la réponse fournie par l'apprenant à chaque étape et sous-étape du processus de résolution. Afin de favoriser son apprentissage, il lui explique comment réviser une réponse erronée. Ce diagnostic nécessite des connaissances, et se fonde sur un résolveur des problèmes du domaine.

Les messages d'explication faisant suite au diagnostic peuvent prendre plusieurs formes. Ils peuvent présenter une phrase explicative de l'erreur. Par exemple, dans AMBRE-add, lors de la rédaction de la phrase de réponse, il est fréquent que les élèves écrivent une phrase qui est dans l'énoncé (par exemple « Alex avait 32 billes »). Dans ce cas, on explique « Ce que tu as écrit est vrai, mais ce n'est pas la réponse à la question ».

Ces messages d'explication peuvent aussi contenir un énoncé généré par le système correspondant à la réponse de l'élève. Cette génération d'énoncé peut souvent lui permettre de comprendre en quoi sa réponse est erronée. Cette fonctionnalité de génération d'énoncé est par exemple utilisée dans AMBRE-add pour expliquer à l'apprenant pourquoi sa reformulation à l'aide d'un schéma n'est pas correcte : le système génère un énoncé de problème correspondant au schéma donné par l'élève.

Des éléments graphiques peuvent aussi être utilisés lorsque le système ne sait pas interpréter la réponse de l'apprenant, ou parce que ce format est plus facile à comprendre que des explications textuelles complexes. On peut ainsi mettre en vert les éléments de la réponse qui sont corrects et en rouge ceux qui sont erronés.

Nous allons à présent décrire le système à base de connaissances qui permet de réaliser ces fonctionnalités d'aide, de diagnostic et d'explications.

### 3.2 CHAMADE : un système à base de connaissances pour les EIAH AMBRE

Afin de proposer des messages d'aide et d'explications adaptés, l'interface d'un EIAH AMBRE interagit avec un système à base de connaissances, CHAMADE. Ce système, réalisé en Prolog, diagnostique les différentes réponses des apprenants en les comparant aux réponses proposées par un résolveur, SYRCLAD, puis compose des messages d'aide et d'explication adaptés à la situation à partir des connaissances du domaine et des connaissances pédagogiques qu'il intègre.

Même s'il intègre des connaissances de classification, de comparaison et d'adaptation, le système à base de connaissances de AMBRE n'est pas un système de RàPC. En effet, ce système est destiné à assister l'apprenant dans la résolution de problèmes suivant le cycle AMBRE, et donc pour des tâches telles que l'adaptation par exemple, mais ne résout pas les problèmes en utilisant le cycle complet du RàPC.

#### L'architecture SYRCLAD

Nous avons conçu l'architecture SYRCLAD (SYstème de Résolution de problèmes fondé sur la CLAssification du Domaine) [11] qui nous permet de réaliser un résolveur de problèmes appliquant une méthode. Cette architecture permet d'explicitier de manière déclarative une classification de problèmes et les connaissances de reformulation et de résolution qui y sont liées (cf. figure 4).

Pour un domaine donné, un expert (didacticien, enseignant...) définit une hiérarchie de classes de problèmes. Pour pouvoir utiliser cette hiérarchie afin d'identifier la classe d'un problème, il faut définir des connaissances de reformulation qui permettent de déterminer les valeurs des attributs discriminants de la hiérarchie de classification. Certaines classes de la hiérarchie, dites opérationnelles, sont suffisamment spécifiques pour qu'on puisse associer à chacune une technique de résolution adaptée aux problèmes qui relèvent de cette classe.

Résoudre un problème consiste alors tout d'abord en une phase d'opérationnalisation, où le système utilise les connaissances de reformulation et le graphe de classification pour déterminer de quelle classe relève le problème et pour construire un nouveau modèle de ce problème (appelé modèle opérationnel). Ce modèle opérationnel est essentiellement composé des attributs discriminants de la hiérarchie, mais aussi d'attributs qui instancient le problème par rapport à la classe dont il relève. Le modèle opérationnel est débarrassé des traits de surface du modèle descriptif (c'est-à-dire des éléments du problème non pertinents pour la résolution). La résolution proprement dite consiste alors à appliquer au modèle opérationnel la technique de résolution associée à la classe du problème, afin d'obtenir une solution au problème posé.



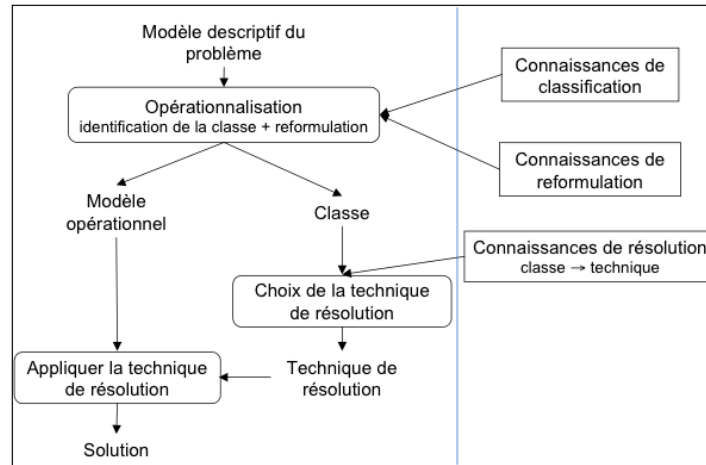


Fig. 4. L'architecture SYRCLAD

Dans l'architecture SYRCLAD, le formalisme de représentation des trois types de connaissances (classification, reformulation et résolution), ainsi que les processus (opérationnalisation, choix et application de la technique de résolution) sont indépendants du domaine. Lorsque l'on définit pour un domaine les trois bases de connaissances, on obtient un résolveur de problèmes du domaine (partie gauche de la figure 4). Cette architecture a été appliquée à quatre domaines [11].

Un résolveur SYRCLAD ne permet pas à lui seul de fournir de l'aide à l'apprenant, ni de diagnostiquer ses réponses ou de lui expliquer ses erreurs. Pour assurer ces fonctionnalités, nous avons ajouté autour de SYRCLAD des bases de connaissances qui constituent l'architecture CHAMADE (arCHitecture pour l'Apprentissage de Méthodes permettant Aide, Diagnostic et Explications).

#### L'architecture CHAMADE

L'architecture CHAMADE contient tout d'abord un résolveur SYRCLAD ainsi que les connaissances sur la méthode associée. Elle contient de plus deux grands groupes de connaissances : celles destinées à la communication avec l'apprenant, et celles destinées à diagnostiquer les réponses de l'apprenant et à construire les messages d'aide et d'explication des erreurs. De plus, une base de problèmes contient les problèmes-types déjà présentés à l'apprenant ainsi que les problèmes qu'il a déjà résolus.

**Les connaissances destinées à la communication** avec l'apprenant sont de trois types :

- les connaissances liées à la langue naturelle, qui permettent de construire des messages grammaticalement corrects,
- les connaissances de génération, qui permettent de générer un énoncé de problème, ou une formulation en langue naturelle d'une connaissance du domaine,
- les connaissances de configuration de l'interface, qui sont destinées à fournir à l'interface des éléments qu'elle interprète afin de par exemple mettre en couleur des éléments du problème ou de la réponse de l'apprenant, ou de construire des

listes déroulantes (par exemple, pour rédiger la phrase de réponse, l'apprenant utilise des listes déroulantes contenant des éléments choisis par CHAMADE-add, grâce à sa connaissance du problème).

**Les connaissances destinées à l'aide et au diagnostic** sont de quatre types :

- les connaissances permettant de connaître le rôle des différentes parties d'un problème (énoncé, reformulation et solution), ainsi que les liens entre ces parties. Elles permettent de mettre en évidence (grâce à des couleurs) les éléments analogues de l'énoncé, de la reformulation et de la solution, et de les mettre en relation avec les éléments analogues d'un autre problème (cf. figure 3). Ces connaissances sont similaires aux connaissances d'adaptation du RàPC.
- les connaissances sur le cycle AMBRE tel qu'il est mis en œuvre dans l'EIAH (étapes et sous-étapes), afin de pouvoir rappeler à l'apprenant ce qu'il doit faire en cas de demande d'aide.
- les connaissances définissant comment comparer les réponses de l'apprenant à l'ensemble des réponses correctes fournies par le résolveur SYRCLAD, afin de pouvoir déterminer si une réponse est correcte ou non.
- les connaissances permettant de donner à l'élève des explications pertinentes sur ses erreurs : il s'agit de définir un ensemble de réponses erronées attendues, et d'y associer des messages explicatifs. Il faut également savoir comment aider l'élève à remédier à une erreur que le système n'a pas su interpréter. Enfin, il peut être nécessaire de recontextualiser un message d'erreur.

L'ensemble des connaissances que nous venons de décrire peut être utilisé par CHAMADE pour répondre à trois types de demande :

- L'interface fait appel à CHAMADE, et plus précisément au résolveur SYRCLAD, pour présenter à l'apprenant en début de séance des problèmes-types résolus. L'interface a également besoin de connaître les problèmes-types qu'a déjà vus l'apprenant pour lui demander de choisir un problème-type ; elle doit aussi savoir quels problèmes a déjà résolus l'apprenant et à quel problème-type il les a associés afin de lui permettre de consulter les problèmes rangés avec un problème-type. L'interface peut également faire appel à CHAMADE pour constituer des éléments d'interface tels que les listes déroulantes décrites ci-dessus.
- Lorsque l'apprenant demande de l'aide, l'interface indique à CHAMADE quel est exactement l'avancement de l'élève dans le cycle AMBRE. CHAMADE construit alors un message d'aide en utilisant ses connaissances sur le cycle AMBRE et ses connaissances liées à la langue naturelle. Afin de pouvoir mettre en évidence les éléments analogues du problème-type et du problème à résoudre, CHAMADE utilise également les connaissances sur les liens entre ces éléments ainsi que les connaissances du résolveur SYRCLAD sur les deux problèmes.
- Lorsque l'apprenant demande un diagnostic de ses réponses, CHAMADE procède en deux temps. Dans un premier temps, le système utilise ses connaissances de diagnostic pour comparer les réponses de l'élève à celles du résolveur. Dans un deuxième temps, il construit en cas d'erreur un message d'explication ou des informations que l'interface mettra en œuvre visuellement. Pour cela, il utilise les connaissances d'explication des erreurs, en s'appuyant sur le résolveur

SYRCLAD, ainsi que sur l'ensemble des connaissances destinées à la communication avec l'apprenant.

Nous avons dans cette partie présenté les connaissances de l'architecture CHAMADE et comment elles sont utilisées pour répondre aux demandes de l'interface ou de l'apprenant. En appliquant SYRCLAD à quatre domaines [11], nous avons démontré la généralité de cette architecture pour la réalisation de solveurs de problèmes utilisant des méthodes. En ce qui concerne l'architecture CHAMADE, même si nous avons procédé selon une démarche analogue, il faudrait l'utiliser dans un EIAH AMBRE appliqué à un autre domaine afin de montrer sa généralité. Une application au domaine de la conjugaison française est en cours.

### **3.3 Vers un outil auteur pour le projet AMBRE : formalisation des connaissances à acquérir**

Les bases de connaissances d'un EIAH AMBRE prennent des formes variées (graphes de classification, règles d'inférences, etc). Pour le premier EIAH AMBRE proposé, ces bases de connaissances ont été définies par des experts directement en langage Prolog. L'avantage du langage Prolog est de permettre une représentation explicite et déclarative des connaissances, la puissance du moteur d'inférences de Prolog permettant quant à lui de faciliter l'écriture des processus de raisonnement utilisant ces bases de connaissances. Notre objectif est maintenant de concevoir un outil permettant à un auteur de définir les connaissances relatives à une méthode de résolution de problèmes dans un domaine donné. Cet auteur sera le concepteur de l'EIAH mais ne sera pas nécessairement un informaticien. Il est donc nécessaire de proposer un formalisme et une interface qui permettrait à un utilisateur non informaticien d'explicitier les connaissances nécessaires aux systèmes à bases de connaissances des EIAH AMBRE. Les connaissances explicitées par l'utilisateur pourront ensuite être traduites via cette interface en Prolog afin d'enrichir les bases de connaissances des EIAH AMBRE.

Dans cet objectif, nous avons séparé en deux catégories les connaissances présentes dans le système à base de connaissances CHAMADE afin d'identifier celles dépendantes d'un domaine et d'une méthode à enseigner de celles propres aux processus de l'architecture CHAMADE. Nous avons ensuite regroupé au sein de la première catégorie les connaissances sur la méthode, celles pour communiquer avec l'apprenant, celles nécessaires à l'aide et celles pour le diagnostic.

À partir de cette catégorisation, nous avons identifié la structure et les invariants des différentes connaissances d'une même catégorie. Ces différentes caractéristiques nous ont permis de proposer un méta-modèle permettant d'acquérir les connaissances nécessaires à la mise en œuvre de CHAMADE pour un domaine et une méthode donnée. Ce méta-modèle est décrit en XML Schema, et les modèles de connaissances pour un EIAH AMBRE donné sont définis par un fichier XML. Sur la figure 5, nous pouvons ainsi voir une partie du méta-modèle décrivant les connaissances de diagnostic. Celles-ci sont définies pour un contexte précis et font référence à un message qui sera proposé à l'apprenant. Sur la figure 6, nous pouvons voir deux exemples de connaissances décrites selon ce méta-modèle : l'une pour l'EIAH AMBRE-add sur

les problèmes arithmétiques, et l'autre pour l'EIAH AMBRE-conjugaison sur la conjugaison en langue française.

L'étape suivante pour atteindre notre objectif est d'exploiter le méta-modèle proposé afin de spécifier l'outil auteur et les processus permettant de traduire en Prolog les connaissances exprimées selon ce formalisme par un auteur non informaticien.

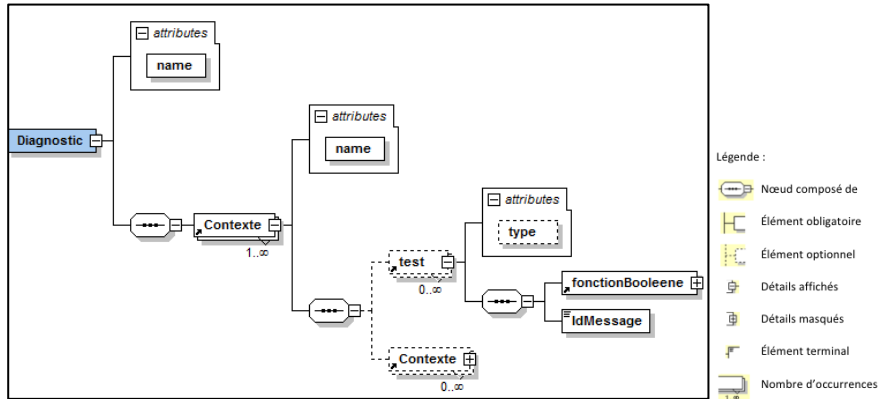


Fig. 5. XML Schema décrivant la structure des connaissances de type Diagnostic

```

<<Contexte name="Etapel">
  <<Contexte name="Etape2">
    <<Contexte name="SE1">
      <<test>
        <<fonctionBooleene>
          <<operateurNot>
            <<fonctionBooleene>
              <<comparaisonUnitaire type="Va Vs">
                <attributNonDiscriminant name="type_p"/>
              </comparaisonUnitaire>
            </fonctionBooleene>
          </operateurNot>
        </fonctionBooleene>
      </test>
    </Contexte>
  </Contexte>
</Contexte>

<<Contexte name="Radical_Je">
  <<test>
    <<fonctionBooleene>
      <<operateurNot>
        <<fonctionBooleene>
          <<comparaisonUnitaire type="Va Vs">
            <attributDiscriminant name="radical[0]"/>
          </comparaisonUnitaire>
        </fonctionBooleene>
      </operateurNot>
    </fonctionBooleene>
  </test>
</Contexte>
  
```

Fig. 6. Exemples de connaissances de diagnostic pour AMBRE-add (gauche) et AMBRE-conjugaison (droite)

## 4 Conclusion et perspectives

Nous avons présenté dans cet article comment nous nous sommes inspirés du paradigme du RàPC pour concevoir le principe d'EIAH destinés à enseigner des méthodes de résolution de problèmes. Ces méthodes sont fondées sur des classes de problèmes que l'on souhaite faire acquérir à l'apprenant à travers la notion de problème prototypique d'une classe. Nous nous appuyons pour cela sur des travaux en psychologie cognitive montrant que le raisonnement par analogie favorise l'apprentissage de connaissances abstraites.

Nous avons ensuite décrit les bases de connaissances permettant à ces EIAH de fournir de l'aide à l'apprenant, de diagnostiquer ses réponses et de lui proposer des explications pertinentes sur ses erreurs. Afin de pouvoir spécifier un outil permettant à

un utilisateur non informaticien de définir ces bases de connaissances, nous avons proposé un méta-modèle des différents types de connaissances nécessaires à un EIAH AMBRE. Si ce méta-modèle a été validé sur le domaine des problèmes additifs (AMBRE-add), son application au domaine de la conjugaison française n'est pas encore achevée. Nous sommes en particulier confrontés au fait que lors de la conception de AMBRE-conjugaison, nous avons choisi de ne pas imposer à l'apprenant de suivre à la lettre le cycle AMBRE, mais de lui laisser la liberté d'utiliser la reformulation et l'analogie au moment où il en a besoin. Ce choix nous amène à modifier légèrement le méta-modèle défini.

La suite de ce travail consistera à spécifier un outil auteur s'appuyant sur ce méta-modèle de connaissances. Nous nous intéresserons en particulier à l'assistance que nous pouvons fournir à l'utilisateur non-informaticien dans la définition des connaissances. Nous envisageons d'une part de lui permettre de définir les connaissances sur des exemples, le système proposant ensuite une généralisation de ces connaissances. Nous projetons d'autre part d'exploiter les traces d'interaction de l'utilisateur avec l'outil auteur afin de l'assister dans sa tâche.

## Références

1. Rogalski, M. : Les concepts de l'EIAO sont-ils indépendants du domaine ? L'exemple d'enseignement de méthodes en analyse. *Recherches en Didactiques des Mathématiques* vol 14 n° 1.2, pp. 43-66. (1994)
2. Gick, M.L., Holyoak, K.J. : Schema induction and analogical transfer. In : *Cognitive Psychology* 15, pp. 1-38 (1983)
3. Cummins, D. : Role of analogical reasoning in the induction of problem categories. In : *Journal of Experimental Psychology: Learning, Memory, and Cognition* 5, pp. 1103-1124 (1992)
4. Ross, B.H., Kennedy, P.T. : Generalizing from the use of earlier examples in problem solving. In : *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16, pp. 42-55 (1990)
5. Aamodt, A., Plaza E. : Case-based reasoning: Foundational issues, methodological variations, and system approaches. In : *Artificial Intelligence Communications* 7, pp. 39-59 (1994)
6. Mille, A. : Associer expertise et expérience pour assister les tâches de l'utilisateur, Habilitation à diriger des recherches, Université Claude Bernard, Lyon1 (1998)
7. Ward, M., Sweller, J. : Structuring effective worked examples. In : *Cognition and Instruction*, 7, pp. 1-39 (1990)
8. Carrol, W. : Using worked examples as an instructional support in the algebra classroom. In : *Journal of Educational Psychology*, 86, pp. 360-367 (1994)
9. Quilici, J. H., Mayer, R. E. : Teaching students to recognize structural similarities between statistics word problems. In : *Applied Cognitive Psychology*, 16, pp. 325-342 (2002)
10. Nogry, S., Guin, N., Jean-Daubias, S. : AMBRE-add: An ITS to Teach Solving Arithmetic Word Problems. In: *Technology, Instruction, Cognition and Learning*, vol. 6, n°1, pp. 53-61, Old City Publishing (2008)
11. Guin-Duclosson, N. : SYRCLAD : une architecture de solveurs de problèmes permettant d'explicitier des connaissances de classification, reformulation et résolution. *Revue d'Intelligence Artificielle*, 13-2, pp. 67-94, Hermès (1999)