

RAPPORT DE STAGE

MASTER 2 RECHERCHE INFORMATIQUE  
CONNAISSANCES ET RAISONNEMENT

---

**ADAPTE**

**GENERATEURS D'EXERCICES PERSONNALISES  
DANS LE CADRE DU PROJET PERLEA**

---

Encadrée par

**STEPHANIE JEAN-DAUBIAS**

MAITRE DE CONFERENCE AU LIRIS  
UNIVERSITE CLAUDE BERNARD LYON1

## RESUME

---

L'étude réalisée dans le cadre de ce stage de Master 2 Recherche se situe dans le domaine de recherche des EIAH (Environnements Informatiques pour l'Apprentissage Humain) et concerne plus particulièrement la personnalisation de l'apprentissage par l'exploitation de profils d'apprenants à travers le projet PERLEA.

Un *profil d'apprenant* rassemble un ensemble d'informations sur l'apprenant caractérisant ses connaissances, ses compétences, ses conceptions et son comportement, identifiées à l'issue d'une activité pédagogique, qu'elle soit informatisée ou non.

Le projet PERLEA vise à concevoir un système permettant aux enseignants de gérer des profils existants en proposant une visualisation riche pour faciliter le suivi de l'apprenant. Les résultats du projet PERLEA sont mis en œuvre dans l'environnement EPROFILEA. Un des modules de cet environnement, le module ADAPTE, permet de proposer aux apprenants des activités adaptées aux compétences mises en évidence par leurs profils. Il donne ainsi les moyens à l'enseignant de réintégrer, dans sa pratique de classe, son travail avec l'environnement EPROFILEA, sur les profils d'apprenants.

L'objectif de ce stage était de concevoir un prototype d'ADAPTE gérant la proposition d'activités papier-crayon en répondant à deux problématiques de recherche. La première est de trouver un équilibre entre généralité de l'environnement EPROFILEA et spécificités disciplinaires ou liées à l'âge, au niveau scolaire ou universitaire. La seconde est d'identifier jusqu'où il est possible d'une part, et souhaitable d'autre part d'automatiser la création de telles activités.

Pour cela, nous avons proposé une typologie d'exercices pouvant être donnés à un apprenant, ainsi que l'architecture de neuf générateurs permettant de créer la totalité de ces exercices. Pour pouvoir affecter ces exercices à un profil d'apprenant, nous avons également défini un ensemble de règles d'affectation. Suite à cette étude, nous avons proposé une architecture complète pour le module ADAPTE et développé un prototype, qui a ensuite été expérimenté avec les enseignants partenaires du projet.

**Mots clés :** EIAH, personnalisation de l'apprentissage, profil d'apprenant, exercice, générateur, architecture.

## ABSTRACT

---

The study realized in this training course of Master 2 Research is in the field of ILE (Interactive Learning Environments) and concerns more particularly the personalization of learning by the exploitation of learners' profiles within the PERLEA project.

A *learner's profile* gathers information on the learner characterizing his knowledge his abilities, his conceptions or his behaviour, identified at the end of a pedagogical activity, computerized or not.

The PERLEA project aims to conceive a system allowing teachers to manage existing profiles by proposing a rich visualisation of profiles to facilitate learners tracking. Results of the PERLEA project are implemented in EPROFILEA environment. One of the modules of this environment, ADAPTE module, allows proposing to learners, activities suited to the abilities highlighted by their profiles. It gives the teacher the means to reinstating, in this class practice, his work with EPROFILEA environment, on the learner's profiles.

The aim of this training course was to conceive a prototype of ADAPTE, managing the proposal of pencil and paper activities, by treating two main issues. The first is to find a balance between genericity of EPROFILEA environment and disciplinary specificities or depending on the age, the scholar or academic level. The second is to identify how far it is from one hand possible and from the other hand desirable to automate the activities creation.

For that, we proposed an exercises typology being able to be given to a learner, and the architecture of nine generators allowing to create all of this exercises. To be able to affect this exercises to a learner's profile, we also defined a set of affectation rules. Following this study, we proposed a complete architecture for ADAPTE module and implemented a prototype, witch was tested with the teachers partners of the project.

**Keywords :** ILE, personalization of learning, learner's profile, exercise, generator, architecture.

## TABLE DES MATIERES

---

<b>1. INTRODUCTION</b> .....	<b>- 4 -</b>
<b>1.1. Le projet PERLEA</b> .....	<b>- 4 -</b>
1.1.1. <i>Les profils d'apprenants</i> .....	- 4 -
1.1.2. <i>Le projet PERLEA</i> .....	- 4 -
1.1.3. <i>L'environnement EPROFILEA</i> .....	- 4 -
<b>1.2. Présentation du stage</b> .....	<b>- 5 -</b>
1.2.1. <i>Le module ADAPTE</i> .....	- 5 -
1.2.2. <i>Problématique</i> .....	- 6 -
1.2.3. <i>Démarche adoptée</i> .....	- 6 -
<b>2. ÉTAT DE L'ART</b> .....	<b>- 7 -</b>
2.1. Les générateurs d'exercices .....	- 7 -
2.2. Les normes et standards.....	- 8 -
2.3. Les scénarios pédagogiques .....	- 9 -
2.4. L'affectation d'un exercice à un apprenant .....	- 10 -
<b>3. CONCEPTION</b> .....	<b>- 10 -</b>
<b>3.1. Qu'est-ce qu'une activité papier-crayon ?</b> .....	<b>- 10 -</b>
3.1.1. <i>Définition des termes utilisés</i> .....	- 10 -
3.1.2. <i>Proposition d'une typologie d'exercices</i> .....	- 11 -
<b>3.2. Les générateurs d'exercices</b> .....	<b>- 14 -</b>
3.2.1. <i>Quels types de générateurs pour ADAPTE ?</i> .....	- 14 -
3.2.2. <i>Utilité des générateurs existants pour ADAPTE</i> .....	- 15 -
3.2.3. <i>Les générateurs d'ADAPTE</i> .....	- 15 -
<b>3.3. L'affectation d'une feuille d'exercices à un profil d'apprenant</b> .....	<b>- 22 -</b>
3.3.1. <i>Les types de profils</i> .....	- 22 -
3.3.2. <i>Définition des critères d'affectation</i> .....	- 22 -
<b>4. REALISATION D'ADAPTE</b> .....	<b>- 25 -</b>
4.1. L'architecture proposée.....	- 25 -
4.2. Le prototype réalisé .....	- 26 -
4.2.1. <i>La définition des stratégies pédagogiques</i> .....	- 26 -
4.2.2. <i>Les générateurs</i> .....	- 27 -
4.2.3. <i>La génération des feuilles d'exercices</i> .....	- 27 -
4.3. Évaluation du prototype.....	- 28 -
<b>5. CONCLUSION ET PERSPECTIVES</b> .....	<b>- 28 -</b>
<b>BIBLIOGRAPHIE</b> .....	<b>- 30 -</b>

## 1. INTRODUCTION

---

Le travail proposé dans le cadre de ce stage de Master 2 Recherche Informatique s'inscrit dans le domaine de recherche des Environnements Informatiques pour l'Apprentissage Humain (EIAH) et concerne plus particulièrement la personnalisation de l'apprentissage par l'exploitation de profils d'apprenants.

Arrivée au terme de ce stage, effectué au LIRIS, Laboratoire d'InfoRmatique en Image et Systèmes d'information, au sein de l'équipe Cognition, Expérience et Agents Situés, nous allons présenter dans ce rapport le travail réalisé. Nous présenterons tout d'abord le projet PERLEA sur lequel se fonde notre étude ainsi que les problématiques de ce stage et l'état de l'art s'y rapportant. Nous en viendrons ensuite à la conception du module ADAPTE, puis à sa réalisation. Nous terminerons en analysant les résultats proposés et en indiquant les perspectives de recherches pour ADAPTE.

### 1.1. Le projet PERLEA

Le projet PERLEA (Profils d'Elèves Réutilisés pour L'Enseignant et l'Apprenant) s'intéresse au problème de la réutilisation par les enseignants de profils d'apprenants existants et à leur exploitation par l'ensemble des acteurs de la situation d'apprentissage (enseignants, apprenants, familles, institutions). Cette étude théorique est appliquée grâce à l'environnement EPROFILEA (Exploitation de PROFILs par les Enseignants et les Apprenants), qui propose un ensemble d'outils permettant la réutilisation et l'exploitation de profils d'apprenants pour toutes disciplines et tous niveaux scolaires ou universitaires.

Dans cette section, nous présenterons, après avoir rappelé la définition des profils d'apprenants, le projet PERLEA, puis l'environnement EPROFILEA.

#### 1.1.1. Les profils d'apprenants

Un profil est un ensemble d'informations concernant un apprenant, collectées ou déduites à l'issue d'activités pédagogiques, qu'elles soient informatisées ou non [JEAN-DAUBIAS et al. 05].

Ce *profil de l'apprenant* est à différencier du *modèle de l'apprenant* comme il est défini en EIAH. Le modèle de l'apprenant étant, à l'image des modèles manipulés en Intelligence Artificielle, le reflet exact des toutes les connaissances d'un apprenant, le profil comme défini à l'instant n'est qu'une instance de ce modèle.

Les profils peuvent être de sources différentes : soit ils sont issus d'un EIAH, soit ils sont sous forme papier-crayon. De plus, les profils peuvent porter sur différents objets : les connaissances de l'apprenant, ses compétences, ses conceptions ou son comportement. Enfin, ils sont sous diverses formes : textuelles, numériques, graphiques...

#### 1.1.2. Le projet PERLEA

Les EIAH conçus dans les laboratoires de recherche sont peu utilisés dans l'enseignement et lorsqu'ils le sont, les profils d'apprenant qu'ils produisent ne sont pas utilisés dans le processus de gestion des compétences au sein de la classe, et ce malgré l'intérêt qu'ils peuvent représenter. Cela peut s'expliquer par la difficulté de récupérer et d'exploiter les informations issues de ces profils.

Le projet PERLEA propose donc d'améliorer l'intégration des EIAH dans le système éducatif grâce à des outils liant l'utilisation des EIAH et les pratiques quotidiennes des enseignants [JEAN-DAUBIAS et al. 05]. Ces outils d'aide à l'enseignant permettent de fusionner des profils issus des pratiques des enseignants et des profils issus des EIAH, afin de les exploiter ensemble au sein de la classe. L'exploitation de ces profils devrait permettre, en facilitant le travail des enseignants, de favoriser la personnalisation de l'apprentissage grâce notamment à la proposition d'activités adaptées aux profils.

Dans un premier temps, il a été nécessaire aux chercheurs du projet d'étudier les usages, les pratiques et les attentes des enseignants en ce qui concerne les profils et leur réutilisation. Ensuite, il leur a fallu déterminer différentes exploitations possibles des profils d'apprenant permettant de réintégrer au mieux les informations contenues dans les profils dans le processus d'apprentissage. L'exploitation principale consiste en une visualisation de ces informations, en s'adaptant au public auquel elle s'adresse, accompagnée d'activités favorisant leur appropriation. Une autre exploitation possible est la création d'un profil de classe à partir des profils individuels afin d'offrir une vue des connaissances d'un groupe d'apprenants.

Cette étude a été faite en lien avec des enseignants selon la méthode de conception différenciée, attribuant un rôle à chacun des types d'intervenants dans le projet [JEAN-DAUBIAS 04].

#### 1.1.3. L'environnement EPROFILEA

Le projet PERLEA a donné lieu au développement d'un EIAH, l'environnement EPROFILEA qui comporte deux phases : la normalisation des profils existants que l'on souhaite intégrer à EPROFILEA et l'exploitation des profils

ainsi restructurés. Afin de mieux comprendre son fonctionnement, présentons l'architecture de cet environnement (cf. Figure 1).

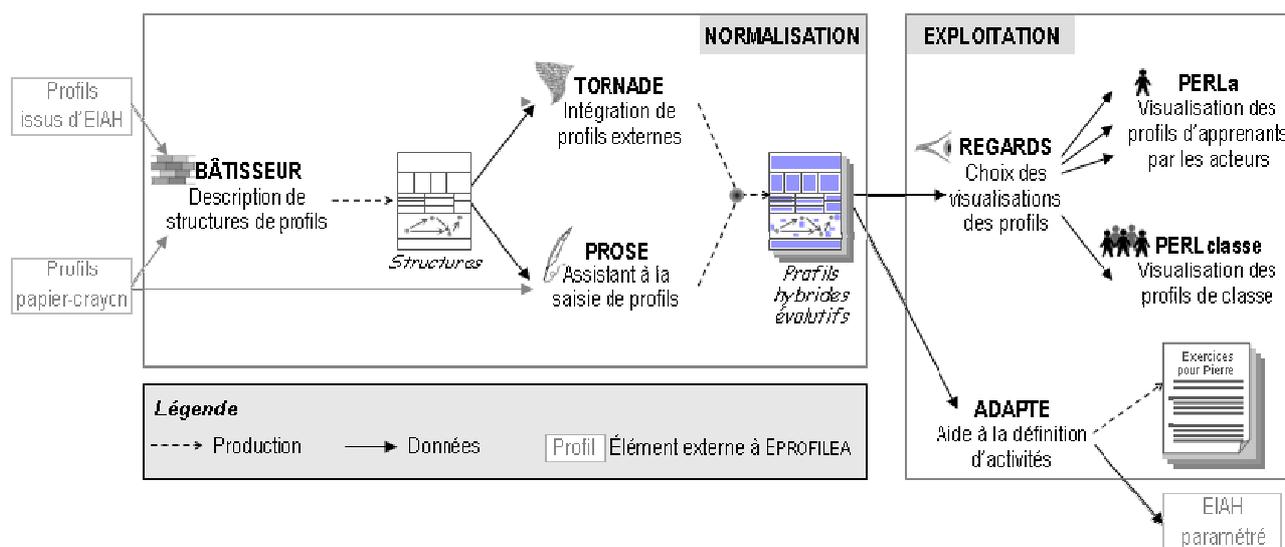


Figure 1 : Architecture de l'environnement EPROFILEA

La réutilisation des profils d'apprenant nécessite de connaître leur structure. Cette description est faite par l'enseignant grâce au module BÂTISSEUR. Ce module opérationnalise le langage de description de profils défini dans PERLEA. Ce langage doit permettre de décrire des profils qu'ils soient issus d'EIAH ou papier-crayon, et ce quels que soient les types d'informations contenus dans ces profils.

Il faut ensuite remplir la structure de profils issue de BÂTISSEUR. Cela se fait de deux manières différentes selon que les données sont issues d'un profil papier-crayon ou d'un EIAH (appelé profil externe).

Dans le cas des profils papier-crayon, le module PROSE (PROfils Saisis par l'Enseignant) aide l'enseignant à saisir les données de chacun de ses élèves selon la structure de profils définie dans BÂTISSEUR.

Dans le cas des profils externes, EPROFILEA propose des systèmes de conversion de profils, les Tourbillons. Le problème étant qu'il faut développer un Tourbillon pour chaque nouveau logiciel externe. Pour faciliter cette étape, le module TORNADO assiste un enseignant expert dans la constitution des Tourbillons adaptés aux EIAH dont on souhaite réutiliser les profils.

Pour qu'une structure soit complètement remplie, il se pourra, dans le cas où les données sont issues d'EIAH externes et de profils papier-crayon, qu'il faille la remplir à la fois à l'aide de TORNADO et à l'aide de PROSE.

Des prototypes de chacun des modules constituant la phase de normalisation des profils externes ont été développés et de premiers tests ont été menés auprès d'enseignants.

À partir des profils résultants, l'enseignant pourra établir dans le module REGARDS la visualisation des profils qui sera proposée à chaque acteur de la situation d'apprentissage : les enseignants eux-mêmes, les apprenants, leur famille... Pour construire ces différentes vues, l'enseignant choisira les parties des profils qui seront consultables par les différents acteurs, ainsi que le vocabulaire utilisé ou le mode de représentation (graphique, textuel ou numérique).

Les modules PERL (Profils d'Elèves Réutilisés) permettent la visualisation interactive des différentes vues déterminées précédemment par l'enseignant. Différentes visualisations ont été définies et développées sous forme d'un premier prototype.

Enfin, le module ADAPTE propose aux apprenants des activités adaptées aux compétences et connaissances mises en évidence par leurs profils. Ce module sera détaillé par la suite puisqu'il est l'objet de notre stage.

## 1.2. Présentation du stage

Dans cette section, nous allons expliquer le principe du module ADAPTE. Nous présenterons ensuite les problématiques liées à sa mise en place, ainsi que la démarche que nous avons adoptée pour ce travail.

### 1.2.1. Le module ADAPTE

ADAPTE est le module d'EPROFILEA qui vise à proposer aux apprenants des activités adaptées aux compétences et connaissances mises en évidence par leurs profils. Ces activités peuvent être des activités papier-crayon proposées par le système ou des activités informatisées gérées par un autre EIAH ; dans ce cas ADAPTE paramètrera un EIAH externe pour proposer une session adaptée à l'apprenant.

Ce module donne les moyens à l'enseignant de réintégrer son travail sur les profils d'apprenants avec l'environnement EPROFILEA, dans des activités adaptées aux profils, dans sa pratique de classe. Pour cela, l'enseignant devra fournir à ADAPTE les profils des apprenants qu'il souhaite faire travailler ainsi que la définition théorique de ces profils selon le langage d'EPROFILEA. L'enseignant va ensuite, avec l'aide d'ADAPTE, créer des feuilles d'exercices personnalisées ou paramétrer un EIAH externe à l'environnement EPROFILEA.

### **1.2.2. Problématique**

L'objectif de ce stage était de concevoir un prototype d'ADAPTE gérant la proposition d'activités papier-crayon variées, adaptées aux difficultés des élèves. Pour cela, il est nécessaire de générer les exercices qui seront proposés, ce afin d'avoir une variété suffisante de problèmes et de disposer de questions spécifiquement conçues pour chaque apprenant.

Le premier point délicat était ainsi de bien cibler la difficulté qu'il y a à générer des exercices. En effet, cette génération n'est pas un problème trivial, elle nécessite des connaissances dédiées aux stratégies de construction des exercices, et de leur solution, sous forme de règles d'expertise et de méta-connaissances, ainsi que des connaissances propres au domaine d'application de l'exercice, comme les formules et théorèmes du domaine, les supports pédagogiques (textes, illustrations...) ou encore des connaissances permettant d'écrire un énoncé en langue naturelle. Rassembler ces connaissances, sous forme d'expertises, ou permettre à une personne ou un système de fournir ces connaissances est long et compliqué. Et faut ensuite exploiter correctement ces connaissances pour générer des exercices mais également leurs solutions. Dans le contexte du projet PERLEA, il faut de plus que ces connaissances couvrent tous les niveaux scolaires et universitaires, et ce dans toutes les disciplines.

La seconde problématique de recherche était de trouver un équilibre entre généralité de l'environnement EPROFILEA et spécificités disciplinaires ou liées à l'âge, au niveau scolaire ou universitaire. En effet, l'environnement EPROFILEA est destiné à être utilisé par des enseignants de toutes disciplines pour suivre des élèves pouvant se trouver de l'école primaire à l'université. Or, proposer des activités se générant de la même manière pour un élève qui apprend à lire au CP et un élève qui travaille sur des expressions algébriques en fin de collège paraissait complexe. Jusqu'à maintenant, la plupart des logiciels proposant des exercices le font dans une matière donnée pour un niveau précis ou le font de façon générique mais en demandant à l'enseignant de fournir toutes les connaissances sur l'énoncé et sa correction. Le but du stage était de voir comment générer des exercices pour une matière quelconque et pour un niveau quelconque sans qu'ADAPTE soit la somme de logiciels traitant chaque matière et chaque niveau.

Enfin, la troisième problématique en rapport avec ce sujet de stage était d'identifier jusqu'où il est possible d'une part, et souhaitable d'autre part d'automatiser la création d'activités. Nous nous sommes donc demandée s'il était possible de créer des exercices automatiquement, sans intervention de l'enseignant utilisateur du logiciel, et cela pour toutes les matières et pour tous les niveaux de formation. De plus, si cela est possible, est-ce souhaitable ? Est-ce vraiment le souhait, le besoin des enseignants qui se serviront d'ADAPTE ? Ou au contraire ne serait-il pas plus intéressant pour eux de leur fournir un logiciel proposant des outils qu'ils pourraient paramétrer à leur guise ? Pour répondre à ces questions, nous avons choisi de travailler avec des enseignants pour pouvoir discuter avec eux de leurs besoins, de leurs pratiques et de leurs habitudes de travail.

### **1.2.3. Démarche adoptée**

Avant de pouvoir proposer une architecture pour le module ADAPTE, nous avons défini les activités pouvant être proposées. Pour la partie d'ADAPTE qui nous intéresse, les activités correspondent à des feuilles d'exercices papier-crayon. Nous avons donc fait l'inventaire des types d'exercices proposés aux apprenants par les enseignants. Nous avons ensuite regardé comment les générer, en étudiant des générateurs d'exercices existants ainsi que leur possible réutilisation. Nous nous sommes ensuite intéressée aux normes et standards en vigueur en EIAH pour stocker les objets pédagogiques et donc les exercices proposés aux apprenants de manière à favoriser l'interopérabilité entre les systèmes. Pour finir, nous avons étudié les différentes manières dont un exercice pouvait être affecté à un apprenant selon ses compétences et/ou ses difficultés.

Une fois notre étude de l'existant terminée, nous avons pu proposer une architecture pour ADAPTE comprenant deux parties : premièrement, la définition de stratégies pédagogiques permettant à un enseignant de décrire les critères conduisant à l'affectation d'un exercice à un apprenant et ainsi à la constitution d'une feuille d'exercices personnalisée, deuxièmement, la génération même de cette feuille d'exercices. Pour cette deuxième partie du logiciel, nous avons également proposé l'architecture des différents générateurs utiles à ADAPTE.

Ce travail de conception du module ADAPTE s'est fait en travaillant avec des enseignants selon la méthode de conception différenciée [JEAN-DAUBIAS 04].

Pour valider cette architecture, nous avons développé un prototype permettant aux enseignants de générer des feuilles d'exercices personnalisées à ses apprenants selon ses propres critères. Ce prototype a ensuite été présenté aux enseignants partenaires du projet.

## 2. ÉTAT DE L'ART

---

Dans cette partie, nous allons exposer ce qui existe en matière de générateurs d'exercices. Nous regarderons ensuite les normes et standards en vigueur dans l'enseignement et nous expliquerons ce qu'est un scénario pédagogique. Enfin, nous nous intéresserons à ce qui a déjà été proposé en matière d'affectation d'exercices à un apprenant.

### 2.1. Les générateurs d'exercices

Dans la plupart des EIAH fondés sur la résolution de problèmes, les exercices proposés aux apprenants sont issus d'une bibliothèque prédéfinie. Ce mode de fonctionnement a deux inconvénients majeurs. Le premier est que le nombre d'exercices et leur diversité s'en trouvent limités. Le second, et sûrement le plus important, est que les exercices ne sont pas toujours adaptés aux besoins des enseignants et à leur contexte de travail. Il y a donc un problème d'adaptation et donc d'intégration des EIAH à leur démarche pédagogique. Une des solutions proposées a été de créer des générateurs d'exercices.

Les premiers systèmes génératifs ont vu le jour dans les années 1970 avec l'intégration des techniques informatiques dans les environnements d'apprentissage. Ces systèmes génèrent des problèmes et leurs réponses. Ils vont permettre de rendre quasiment infini le nombre d'exercices proposés et dans certains cas de paramétrer ces exercices [BRUILLARD 97].

Après plus de 30 ans de recherches et de développements, nous pouvons dire qu'il existe trois grands types de générateurs. Tout d'abord, les générateurs automatiques, qui vont générer des exercices sans aucune intervention de l'utilisateur. A l'opposé, les générateurs manuels, appelés outils auteurs, qui vont aider et guider l'utilisateur lors de la conception des exercices. Et enfin, entre les deux, les générateurs semi-automatiques, qui vont générer des exercices automatiquement mais selon des contraintes définies par l'utilisateur.

Nous allons à présent définir plus précisément ces types de générateurs et donner des exemples pour chacun d'eux.

**Les générateurs automatiques** créent des exercices automatiquement, sans que l'utilisateur ne puisse influencer les choix du système. Cet utilisateur peut être soit le module pédagogique d'un EIAH, dans ce cas le système produit des exercices qui seront proposés à l'apprenant par l'EIAH lui-même, soit un enseignant, dans ce cas le système crée des exercices à la demande de l'enseignant qui les proposera ensuite à ses élèves.

Si l'utilisateur du système est le module pédagogique d'un EIAH, deux cas de figure vont se présenter : soit le système va créer une batterie de tests afin d'évaluer l'apprenant et de tester les connaissances acquises ; soit le système utilise le modèle de l'apprenant afin de générer des exercices adaptés à l'apprenant. Il va être capable de créer automatiquement toute une batterie de problèmes axés sur une notion précise du cours que l'apprenant doit acquérir et cela grâce à l'état de ses connaissances et à l'historique de son parcours. Ainsi, le système IDEBUGGY [BURTON 82], pour le diagnostic des causes d'erreur dans la soustraction écrite, crée un problème après chaque erreur de l'apprenant.

Si l'utilisateur du système est un enseignant, celui-ci va demander la génération d'un ou plusieurs exercices sur un domaine, ou sur une connaissance d'un domaine, pour que l'apprenant puisse travailler ou être évalué. Un exemple est l'environnement APLUSIX [NICAUD 93], micromonde pour l'apprentissage de l'algèbre élémentaire dans les lycées et collèges. Cet environnement comporte un générateur automatique d'exercices [BOUHINEAU et al. 05]. Le but de ce générateur est de construire automatiquement des listes d'exercices pour l'entraînement ou l'apprentissage de différents types de problèmes algébriques (calculer, développer, factoriser, résoudre). Il est composé d'une hiérarchie de patrons d'exercices qui définit l'ensemble des exercices possibles, et d'une carte de tests qui regroupe l'ensemble des exercices définis.

L'avantage majeur de ces générateurs automatiques est qu'ils peuvent créer rapidement une grande quantité d'exercices. Mais leur principale limite est que les enseignants ne peuvent pas intervenir dans la génération des exercices et ne peuvent donc pas adapter ceux-ci à leurs habitudes de travail.

**Les générateurs manuels** ont pour principale caractéristique le fait qu'ils ne construisent pas eux-mêmes les énoncés des problèmes, ni leurs solutions. Cette tâche revient à l'enseignant, d'où le nom d'outils auteurs. L'enseignant construit donc chaque énoncé et fournit la ou les réponses attendues. Les générateurs manuels peuvent ainsi créer de nombreux exercices, comme des exercices à trous, des reconstitutions de textes, des appariements, des exercices sous forme de jeux (mots croisés, puzzle, quiz...), des exercices multimédias (lecture, écoute)... et ce dans divers domaines d'enseignement. Les exercices créés vont être de deux types : ouverts ou

fermés. Ainsi les exercices fermés (exercice à trous, QCM...) permettent une correction automatique des réponses de l'apprenant, dans la mesure où l'utilisateur du système auteur a décrit la manière de corriger l'exercice, alors que les exercices de type ouvert, qui demandent une rédaction de l'apprenant sur ordinateur ou sur papier, nécessitent une correction de l'enseignant, le système n'ayant pas de représentation du problème et des réponses de l'apprenant, il ne pourra pas les corriger.

La seconde caractéristique des outils auteurs est qu'ils visent des enseignants ou formateurs sans connaissances informatiques particulières. Ils se doivent donc d'être des logiciels simple d'utilisation et possèdent, dans la plupart des cas, des outils de conception graphique ou visuelle.

Enfin, leur dernière caractéristique est qu'ils ne possèdent aucune connaissance sur l'exercice qu'ils proposent et sur les réponses préenregistrées. Ils ne possèdent donc pas de fonctionnalité d'aide (sauf si celle-ci est préalablement définie par l'enseignant) ni de diagnostic autre que vrai/faux pour les réponses de l'apprenant.

Un exemple est l'outil auteur GENEVAL (GÉNÉrateur d'exercices pour auto-EVALuation) [COGNE et al. 98]. Il permet de créer des exercices hypermédiés ayant tous la même structure : un énoncé et la réponse associée qui peut être donnée selon trois niveaux progressifs, qui correspondent à deux niveaux d'aide puis à une réponse détaillée. À l'inverse d'autres systèmes, les apprenants devront résoudre sur papier les exercices proposés avant de comparer leur réponse à celle(s) indiquée(s) par le système.

L'avantage majeur des générateurs manuels est qu'ils laissent une totale liberté à l'enseignant tant dans le domaine d'application que dans le contenu pédagogique de l'exercice. En contrepartie, ils ont une limite extrêmement forte du fait que l'enseignant doit définir entièrement l'exercice et ses solutions.

**Les générateurs semi-automatiques** sont intermédiaires aux deux autres types de générateurs. Ils construisent eux-mêmes les énoncés des problèmes, mais en laissant intervenir l'utilisateur dans le processus de création. Ainsi l'utilisateur influe sur les problèmes générés en spécifiant un ensemble de contraintes. Cet utilisateur peut être soit un enseignant, soit le module pédagogique d'un EIAH.

Un premier exemple est le système CEP [GIROIRE 89]. Il a pour but de générer des énoncés d'exercices dans des domaines variés, liés au monde réel, comme la physique ou la chimie. La génération se fait selon les directives introduites par l'utilisateur, et donne lieu à un énoncé en langue naturelle.

Dans une optique de continuation du travail amorcé à travers CEP, SYGEP (Système de Génération d'Énoncé de Problèmes) a été réalisé [PECEGO 98]. C'est un Système à Base de Connaissances qui crée des énoncés de problèmes dans des domaines divers utilisant des dispositifs ou des expérimentations pour donner lieu à des questions (circuits électriques, expériences de mécanique...). Dans ce système, basé essentiellement sur la manipulation des formules du domaine, l'utilisateur fixe ses objectifs pédagogiques en spécifiant un ensemble de contraintes.

Un autre exemple nous est proposé avec le projet AMBRE (Apprentissage de Méthodes Basé sur le Raisonnement à partir de l'Expérience) [GUIN-DUCLOSSON et al. 02], qui a pour but de concevoir des EIAH utilisant un processus d'apprentissage inspiré du cycle du Raisonnement à Partir de Cas pour l'enseignement de méthodes. AMBRE est fondé sur un Système à Base de Connaissances qui s'appuie sur un résolveur de problèmes SYRCLAD [GUIN 97] qui permet de fournir aide et diagnostic à l'apprenant ainsi que des explications sur ses erreurs. Un EIAH AMBRE comporte plusieurs modules, dont le module AMBRE-enseignant [DUCLOSSON et al. 05]. Ce module est un outil d'assistance à l'enseignant contenant, entre autres, un générateur de problèmes pour créer les exercices de leur choix et des séquences personnalisées d'apprentissage.

Les générateurs semi-automatiques possèdent donc les avantages des générateurs automatiques (génération rapide d'une quantité importante d'exercices), mais en donnant une solution à leur principale limite : les enseignants peuvent paramétrer les exercices générés. C'est pourquoi nous choisiront cette dernière approche pour notre travail.

## 2.2. Les normes et standards

La multiplicité des applications pédagogiques, notamment diffusées sur le web, entraînent une quantité d'objets pédagogiques importante et une hétérogénéité des termes et des concepts utilisés. Afin de permettre une réutilisation des contenus de formation, il est nécessaire au préalable de se mettre d'accord sur leurs formats, pour pouvoir les manipuler. Ces formats partagés sont les normes et standards. Ces normes et standards vont permettre d'indexer les ressources d'apprentissage, de faciliter les recherches et leur réutilisation. Ils vont donc permettre une interopérabilité entre différentes applications et une pérennité des ressources. Nous allons dans cette section rappeler la différence entre norme et standard puis présenter les principaux dans le domaine de l'enseignement et plus précisément ceux proposés pour la formation à distance. Nous verrons ensuite comment nous comptons prendre en compte ces normes et standards dans ADAPTE.

Pour pouvoir distinguer une norme d'un standard, nous allons nous inspirer des définitions données par [GUIDON 01] : une **norme** est un ensemble de règles de conformité édictées par un organisme de

normalisation, comme l'ISO au niveau international ou l'AFNOR au niveau national, alors qu'un **standard** est un ensemble de recommandations développées et préconisées par un groupe représentatif d'utilisateurs, comme les recommandations du W3C (World Wide Web Consortium) de l'IEEE (Institute of Electrical and Electronics Engineers).

Dans le domaine de l'enseignement, plusieurs normes et standards sont utilisés, et ce dans des buts différents. Nous présenterons ici une norme permettant de décrire une grande variété de ressources, le Dublin Core, puis nous nous intéresserons aux normes et standards liées aux objets pédagogiques, les exercices que nous souhaitons proposer dans ADAPTE pouvant être considérés ainsi. Pour cela, nous nous appuyerons sur [PERNIN 03] qui compare LOM, SCORM et EML.

Le **Dublin Core**, créé en 1995, est un schéma de métadonnées génériques qui permet de décrire une ressource numérique ou physique et d'établir des relations avec d'autres ressources. Il comprend 15 éléments de description (titre, auteur, éditeur, sujet, description, langue...) ou relatifs à la propriété intellectuelle.

La **LOM (Learning Object Metadata)**, adoptée comme standard par l'IEEE en 2002, est un ensemble de métadonnées permettant de décrire des ressources de formation. La LOM comprend 45 éléments descriptifs regroupés en 9 catégories. Ces éléments sont détaillés notamment dans [DE LA PASSARDIERE et al. 03]. D'après Pernin, la « LOM constitue la première tentative cohérente de classification à l'aide de métadonnées des entités mises en jeu dans un processus d'apprentissage. » [PERNIN 03].

**SCORM (Sharable Content Object Reference Model)**, proposé par le consortium ADL (Advanced Distributed Learning), est issu d'une initiative du Département américain de Défense. Son but est de fournir un modèle de référence permettant de garantir la qualité des contenus d'ensemble en termes de réutilisabilité, accessibilité et pérennité, grâce à deux types de recommandations. Premièrement, le modèle d'agrégation de contenu (Content Aggregation Model) qui fournit un guide pour l'identification des ressources de base et leur agrégation dans un contenu structuré de formation. Ce modèle prend en compte les informations de la LOM. Deuxièmement, l'environnement d'exécution (Run-Time Environment) qui fournit un guide pour l'exécution des contenus, la communication et le suivi des activités dans un environnement web. SCORM propose une répartition des objets pédagogiques en trois niveaux : les ressources numériques élémentaires, les objets de contenu partageables et les agrégats de contenu.

Enfin, le modèle **IMS LEARNING DESIGN** a été proposé par R. KOPER de l'Open University of the Netherlands pour décrire les situations effectives d'apprentissage [KOPER 01]. Cette proposition se fonde sur le langage de modélisation pédagogique **EML** qui repose sur la structuration des situations d'apprentissage en « Unités d'Étude » qui peuvent être des cours, des leçons, des études de cas, des travaux pratiques, etc.

Pour bien comprendre la distinction entre LOM, SCORM et EML, nous pouvons citer la classification des modèles définie par [PERNIN 06]. On peut donc distinguer :

- les langages de modélisation pédagogique, tels qu'EML ;
- les langages d'indexation, tels que la LOM ;
- les langages de mise en œuvre informatique, tels que SCORM.

Pour ce qui est de l'utilisation des normes et standards dans ADAPTE, nous avons décidé de ne pas les intégrer directement mais d'ajouter un module de conversion de nos exercices dans la norme ou le standard voulu. Cette décision vient du fait que « l'état actuel de la standardisation est celui d'un processus en cours d'élaboration et non celui d'une base sur laquelle construire des solutions pérennes » [CROZAT 04]. Les termes choisis pour servir de métadonnées et leur signification évoluent encore et des ambiguïtés subsistent sur le sens de certaines métadonnées. De plus, dans le même article, Crozat insiste sur le fait que « les standards proposés doivent être considérés en tant que standards d'échange entre systèmes et non comme des standards de création. » [CROZAT 04]. C'est pourquoi nous allons créer des exercices que nous stockerons au format XML, format lisible par une majorité de personnes et de systèmes, et nous proposerons un module de conversion de nos exercices vers les normes et standards pour permettre leur interopérabilité et préserver leur pérennité.

### 2.3. Les scénarios pédagogiques

D'après [LEJEUNE et al. 04], un scénario d'apprentissage ou scénario pédagogique représente « la description, effectuée a priori ou a posteriori, du déroulement d'une situation d'apprentissage ou unité d'apprentissage visant l'appropriation d'un ensemble précis de connaissances, en précisant les rôles, les activités ainsi que les ressources de manipulation de connaissances, outils, services et résultats associés à la mise en œuvre des activités ». Cette définition montre bien qu'un scénario n'est pas seulement un agencement de ressources ou d'exercices. Il met en place des buts et des situations d'apprentissage. Il définit comment les ressources pédagogiques seront mises en œuvre dans la situation d'apprentissage. En fait, un scénario dépend principalement du contexte d'apprentissage.

Or notre but dans ADAPTE est de fournir des feuilles d'exercices adaptées à un moment donné aux compétences d'un apprenant. Ensuite l'enseignant peut les utiliser de la manière dont il le souhaite : devoir à la maison, soutien dans une matière, évaluation de certains points du cours, etc. Une même feuille d'exercices pourra donc être utilisée dans différents contextes. C'est pour cela que nous allons permettre à l'enseignant de donner une description des structures d'exercices qu'il crée en précisant un but pédagogique, mais nous ne mettrons pas en œuvre de description de scénario pédagogique. Ainsi les enseignants pourront se servir de ce que le système leur propose, les feuilles d'exercices, pour les inclure dans des scénarios qu'ils définiront en dehors d'ADAPTE.

## 2.4. L'affectation d'un exercice à un apprenant

Une des problématiques du stage était de trouver comment faire correspondre une activité à un profil d'apprenant. Si l'on regarde les travaux déjà proposés en EIAH, on peut remarquer les travaux mis en place dans PEPISTEREO [VINCENT et al 05]. PEPISTEREO est une partie du projet PEPITE [JEAN-DAUBIAS 02], projet qui s'attache au diagnostic de la compétence algébrique des élèves de fin de collège - début de lycée, en proposant un profil cognitif de l'apprenant. PEPISTEREO s'appuie sur ces profils cognitifs ainsi que sur des stéréotypes pour proposer à l'enseignant des stratégies d'enseignement. Pour [VINCENT et al 05], un stéréotype est un ensemble de profils « équivalents » au sens où on leur associe les mêmes objectifs prioritaires d'apprentissage. Les élèves sont donc classés par stéréotype à partir de leurs taux de réussite et de leurs modalités de fonctionnement, et ceci en prenant en compte des seuils pour s'assurer que les élèves ont répondu à un nombre significatif d'exercices mobilisant une compétence donnée. Cela permet aux enseignants d'obtenir une « géographie cognitive » de leur classe. Dans PEPISTEREO, trente-six stéréotypes ont été définis pour faciliter l'établissement de parcours d'apprentissage et la gestion des groupes d'apprenants.

Ce travail est celui qui se rapproche le plus de ce que nous voulons mettre en place dans ADAPTE. Toutefois, les contextes sont différents. Dans ADAPTE, nous laissons les enseignants, en les aidant, construire leurs propres groupes de travail selon leurs propres critères. Nous ne souhaitons pas proposer aux enseignants des stratégies d'enseignement appropriées à des stéréotypes auxquels appartiendraient les élèves, mais nous voulons leur permettre de créer leurs propres stratégies utilisant des exercices qu'ils choisissent comme support de leur enseignement. D'une part, nous souhaitons ainsi laisser une totale liberté aux enseignants, d'autre part, cette attitude nous permet de conserver le caractère générique de notre approche.

Cette différence de contexte implique que les propositions d'affectation de parcours d'apprentissage à un profil d'apprenant proposées dans PEPISTEREO ne sont pas applicables pour ADAPTE. Nous avons donc défini nos propres règles permettant à un enseignant d'affecter un exercice à un profil, règles détaillées dans la section 3.3.2 de ce rapport.

## 3. CONCEPTION

---

Dans cette partie, nous allons reprendre point par point les étapes de la conception qui nous ont mené à la définition de l'architecture d'ADAPTE présentée dans le chapitre suivant. Rappelons pour cela qu'ADAPTE est le module d'EPROFILEA qui va permettre aux enseignants de créer des activités adaptées à leurs élèves. Dans le cadre de ce stage, nous nous sommes intéressée à la génération d'activités papier-crayon. Pour générer ces activités, l'enseignant va décrire ses stratégies pédagogiques qui vont permettre d'associer un exercice, dont la création est paramétrée par l'enseignant, à une compétence précise du profil de l'apprenant. Pour permettre cela, nous avons défini en quoi consistait une activité papier-crayon, ainsi que les exercices qui la constituent. Nous avons ensuite proposé des générateurs permettant de créer ces activités. Enfin, nous avons défini, à travers un ensemble de règles, les connaissances nécessaires pour affecter un exercice à un profil.

### 3.1. Qu'est-ce qu'une activité papier-crayon ?

Dans notre contexte, une activité papier-crayon peut être vue comme une feuille d'exercices à imprimer. Cette feuille d'exercices va contenir un nombre variable d'exercices de différents types, exercices différents selon le profil de l'apprenant pour lequel la feuille est créée. Nous allons à présent définir les termes que nous utilisons pour caractériser les exercices puis nous présenterons la typologie des exercices que nous proposons.

#### 3.1.1. Définition des termes utilisés

Pour pouvoir définir très précisément les exercices manipulés dans ADAPTE, nous avons posé un ensemble de termes. Nous allons voir la définition de chacun de ces termes et revenir ensuite sur leur utilisation dans le cadre de ce stage. Des exemples illustrant chacun de ces termes seront donnés dans la section suivante.

Un **exercice** est constitué d'éléments d'énoncé et de réponse pour l'apprenant ainsi que de la solution pour l'enseignant. Un exercice sera en fait une instanciation d'une structure d'exercices.

Une **structure d'exercices** contient l'ensemble des contraintes permettant de créer un exercice d'un type particulier.

Un **patron d'exercices** contient la définition théorique d'une catégorie de structures d'exercices. Il existe deux types de patrons : les patrons de base qui seront utilisés dans les générateurs d'exercices et les patrons opérationnels qui vont déterminer l'interface précise à travers laquelle un enseignant définira les contraintes nécessaires aux générateurs pour créer des exercices.

Une **structure de patrons** est constituée de la caractérisation d'un patron et des points communs à tous les patrons. La caractérisation d'un patron correspond à la définition générale d'un patron : un patron est l'ensemble des éléments discriminants que doit contenir une structure d'exercices pour que l'exercice qu'elle permet de créer appartienne à une classe précise d'exercices. Une structure d'exercices va contenir les différents champs que le générateur d'exercices va devoir remplir pour créer un exercice.

Les points communs des différents patrons sont les suivants : un identifiant et un nom de structure d'exercices, le niveau scolaire auquel est destiné l'exercice, ainsi que la discipline et la matière concernées par l'exercice, la description de la structure d'exercices et ses buts pédagogiques, le statut de cette structure (privé ou public) et enfin l'identifiant de l'enseignant qui a créé la structure d'exercices. Une structure d'exercices aura un statut public ou privé donné par l'enseignant pour indiquer s'il souhaite ou non partager avec d'autres enseignants les exercices qu'il a créés.

### 3.1.2. Proposition d'une typologie d'exercices

Nous avons défini une activité papier-crayon d'ADAPTE comme une feuille d'exercices. Nous voulons donc proposer aux enseignants des outils permettant de générer ces feuilles d'exercices. Pour cela, nous avons identifié les différents exercices pouvant être proposés à l'enseignant, en étudiant le programme scolaire 2005-2006 tiré des textes officiels du Ministère de l'Education Nationale, de la Recherche et de la Technologie ainsi que les différents exercices proposés par les générateurs d'exercices existants, puis nous en avons discuté avec les enseignants partenaires du projet PERLEA.

Rappelons que nous nous intéressons ici à des exercices de tous domaines et de tous niveaux. Un exercice va donc dépendre du niveau scolaire de l'apprenant et de la matière sur laquelle l'enseignant souhaite faire travailler son élève. Un exercice peut être soit de type ouvert, soit de type fermé. Une activité de type ouvert va nécessiter une rédaction de la part de l'apprenant (dissertation, études de document...), alors qu'une activité de type fermé aura des réponses fixes (QCM, texte à trous...). Cette diversité des activités va entraîner une diversité dans la façon de les générer.

Nous avons identifié quinze sortes d'exercices et mis en place un classement par type d'activités. Nous avons ainsi pu construire une hiérarchie de patrons d'exercices (cf. Figure 2). Cette typologie d'exercices a été définie de façon générique et nous l'avons implémentée de façon spécifique dans ADAPTE. Elle a été obtenue en prenant le point de vue de l'enseignant lors de la création d'un exercice. Elle aurait été tout autre si l'on avait pris le point de vue de l'apprenant lors de la résolution d'un exercice. Cette typologie comprend neuf catégories d'exercices représentées par neuf patrons d'exercices (cf. A à I dans la Figure 2) auxquels correspondront neuf générateurs. Ces générateurs seront détaillés par la suite (cf. section 3.2.3. de ce rapport). Nous allons à présent voir les différents niveaux de la typologie d'exercices en expliquant à quoi ils font référence techniquement, puis nous verrons chacune des catégories d'exercices pour la préciser et montrer à chaque fois un exemple d'exercice correspondant.

La typologie d'exercices proposée est constituée de quatre niveaux : les structures de patrons, les patrons, les structures d'exercices et enfin les exercices. La définition de chacun de ces termes a été donnée dans la section 3.1.1 de ce rapport. D'un point de vue technique, les points communs des patrons sont décrits dans une DTD (*Document Type Definition*). Cette DTD est purement théorique, elle ne sert pas concrètement dans l'utilisation du logiciel, mais elle a servi à implémenter le logiciel. Ensuite, pour chaque patron, une DTD théorique contiendra les différents éléments d'une structure d'exercices associée à ce patron. Pour créer une structure d'exercices, l'enseignant choisira un patron. Il existe deux sortes de patrons : les patrons et les patrons opérationnels. Un patron va définir une catégorie d'exercices se générant tous de la même façon, grâce au même générateur d'exercices. Un patron opérationnel va définir une sous-catégorie d'exercices se générant grâce au même générateur mais avec des contraintes de génération spécifiques. On peut donc remarquer que les patrons opérationnels sont plus précis que les patrons et que deux patrons opérationnels peuvent appeler le même générateur d'exercices. Dans ADAPTE, si le patron choisi par l'enseignant n'est pas un patron opérationnel, il lui faudra préciser son choix. Une fois ce choix effectué, le système présentera à l'enseignant l'interface correspondante au patron opérationnel retenu. Cette interface donne la possibilité à l'enseignant de préciser les

contraintes permettant au générateur de créer un exercice. Ces contraintes seront mémorisées dans la structure d'exercices. Et c'est lors de l'instanciation de cette structure que les exercices seront réellement créés.

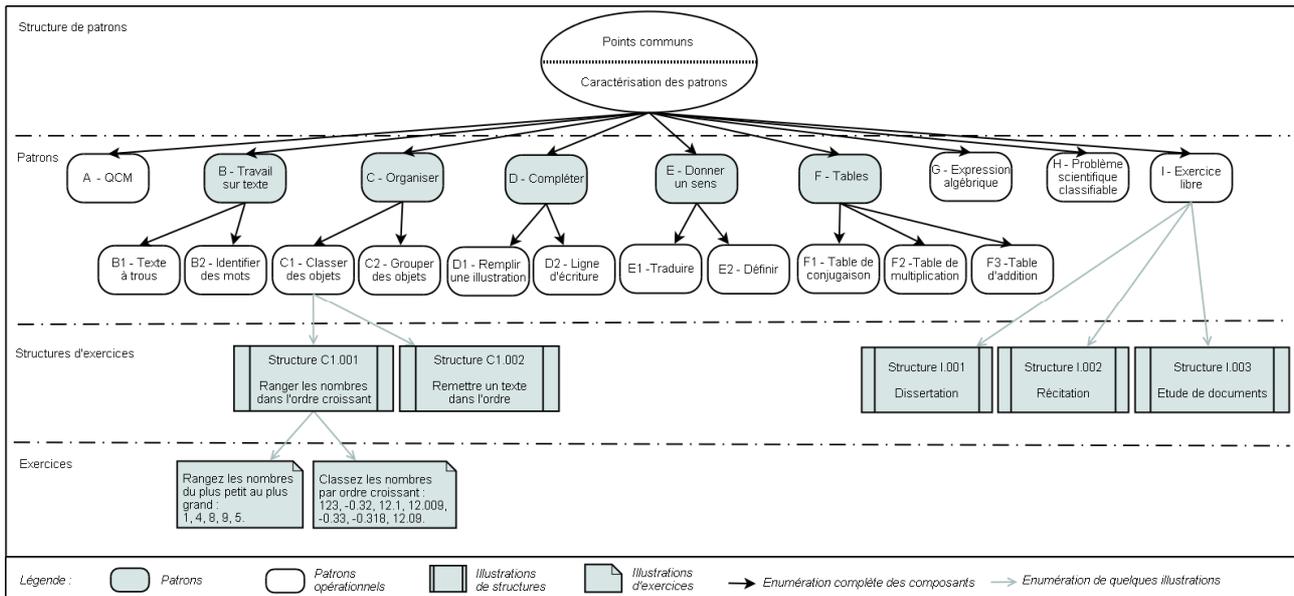


Figure 2 : Typologie des exercices dans ADAPTE

Donnons un exemple concret d'utilisation d'ADAPTE pour mieux comprendre cette typologie d'exercices : celui d'un enseignant voulant créer des exercices pour que ses élèves travaillent en mathématiques sur l'ordonnancement de nombres relatifs. Pour cela, il va sélectionner le patron « Organiser » (cf. C sur la Figure 2), puis affiner son choix en choisissant le patron opérationnel « Classer des objets » (cf. C1 sur la Figure 2). À partir de ce patron, le système présentera à l'enseignant une interface lui permettant de spécifier les contraintes de génération d'exercices. C'est à ce moment là que l'enseignant va formuler son souhait de travailler sur des nombres relatifs et spécifier que l'organisation de ces nombres doit se faire par ordre croissant. Il va également préciser des contraintes sur le choix des nombres contenus dans l'exercice (pas de fraction, pas de multiple de 10...). Toutes ces contraintes vont être mémorisées dans une structure d'exercices (cf. Structure C1.001 sur la Figure 2). C'est à partir de cette structure d'exercices que le système générera les exercices contenus dans les feuilles d'exercices personnalisées. Il pourra ainsi générer des exercices différents à partir de la même structure d'exercices.

**Les exercices de type « QCM »** (cf. A sur la Figure 2) sont des questionnaires à choix multiples. L'apprenant doit donc répondre à une série de questions où pour chacune plusieurs réponses lui sont proposées. Selon les QCM, zéro, une ou plusieurs réponses peuvent être correctes. Les exercices de ce type peuvent servir à faire travailler un apprenant en langue sur les accords, la conjugaison ; en mathématiques sur le calcul, les conversions ; en histoire sur les dates ; pour les apprenants plus jeunes, les QCM peuvent servir à faire travailler la lecture, la reconnaissance de sons... et plus généralement, dans n'importe quelle matière pour contrôler la connaissance d'un cours ou d'un point précis du cours. Ce type d'exercices est destiné à toutes les disciplines et tous les niveaux. Générer automatiquement des QCM demande d'avoir des bases de connaissances sur le domaine d'application du QCM. Dans le cas d'ADAPTE, les enseignants vont générer des QCM pour de multiples domaines. Il n'est donc pas envisageable qu'ADAPTE génère automatiquement les QCM. Le système pourra donc proposer une aide à la création en demandant à l'enseignant de fournir les connaissances et ADAPTE fera le choix des questions et réponses proposées pour diversifier les exercices.

*Exemple : Que veut dire le mot anglais CAR en français : o Chat o Bus o Voiture*

**Les exercices de type « Travail sur texte »** se décomposent en deux sous-catégories : les textes à trous et les exercices demandant d'identifier des mots (cf. B1 et B2 sur la Figure 2).

Les textes à trous peuvent être de deux types : avec une liste d'éléments permettant de remplir les trous ou sans liste d'éléments. Dans les deux cas, ces exercices permettent de faire travailler les apprenants en langue sur la lecture, la compréhension de texte, la conjugaison, l'orthographe, le vocabulaire et tous types de travaux lexicaux. Ils vont permettre dans de nombreuses matières de faire travailler les notions de cours, apprises par cœur ou devant être comprises, et en mathématiques de faire travailler les apprenants sur les conversions, les expressions algébriques, les équivalences... Pour la création des textes à trous, le logiciel peut proposer à l'enseignant de choisir un texte ou de le saisir. Mais le logiciel peut également proposer des textes selon un jeu de contraintes

précisé par l'enseignant (matière, langue, longueur du texte). Une fois le texte choisi, l'enseignant pourra déterminer les mots à effacer ou laisser faire le logiciel. L'enseignant décidera enfin si l'exercice contiendra la liste des mots à placer ou non pour faire varier le niveau de difficulté de l'activité.

*Exemple 1 : Joan (to play) \_\_\_ football every day.*

*Exemple 2 : 12 cm = \_\_\_ dm.*

L'activité d'identification des mots consiste à demander à l'apprenant de trouver un ou plusieurs mots dans un texte. Cela permet de le faire travailler, par exemple en langue sur le vocabulaire, les synonymes. La création de l'activité ressemblera à la précédente pour le choix du texte et le choix des mots à trouver.

*Exemple : Entourer tous les adverbes présents dans le texte. (...texte...)*

**Les exercices de type « Organiser »** se décomposent en deux sous-catégories : classer et grouper des objets (cf. C1 et C2 sur la Figure 2).

L'activité de classement d'objets va demander à l'apprenant de classer des objets selon des consignes. Cela peut être utilisé en mathématiques pour classer les nombres dans un ordre précis (cf. Structure C1.001 sur la Figure 2), mais également pour de jeunes apprenants à qui on va par exemple demander de classer des mots dans l'ordre alphabétique ou des objets selon leur taille. Ces activités peuvent être créées semi-automatiquement facilement.

*Exemple : Ranger les nombres suivants par ordre croissant : 7/5, 6, -0.7, 0.2, 100, 18, 0, 3/4.*

Un cas particulier du classement d'objets est le fait de remettre un texte dans l'ordre (cf. Structure C1.002 sur la Figure 2). Cela permet de faire travailler un élève en langue sur la compréhension d'un texte, de reconstruire correctement des démonstrations géométriques ou de remettre dans l'ordre des événements, en histoire, en biologie... Cette activité peut être construite semi-automatiquement selon les contraintes définies par l'enseignant : sélection du texte, nombre de découpage à faire, où découper le texte (à la fin d'une phrase, d'un paragraphe, au milieu de phrases...).

*Exemple : Remettre la phrase suivante dans l'ordre :*

*sick He made by eating much chocolate himself too*

Le groupement des objets va consister à demander à l'apprenant de grouper des objets selon certaines contraintes. Il peut s'agir en mathématiques de regrouper les fractions qui sont égales, ou dans d'autres cas de regrouper des mots s'ils ont des points communs, au niveau syntaxique ou linguistique. Un dernier exemple est de rassembler des figures ayant des caractéristiques communes : la couleur pour les petits, les thèmes pour les plus âgés.

*Exemple : Entourer les triangles orientés vers la droite : ● ▲ ■ ● ◀ ▲ ▶ ▲ ■ ● ▶*

**Les exercices de type « Compléter »** se décomposent en deux sous-catégories : remplir une illustration et faire des lignes d'écriture (cf. D1 et D2 sur la Figure 2).

Remplir une illustration est surtout destiné à la géographie mais également à la biologie. L'enseignant choisit une illustration et saisit directement les questions associées.

*Exemple : Colorier sur la carte les pays de l'Union Européenne et écrire leur nom. (...carte...)*

Les lignes d'écriture peuvent se faire sur une lettre de l'alphabet ou sur des mots, mais également sur des figures, par exemple pour apprendre à écrire ou à dessiner des notes de musique.

*Exemple 1 : Faire trois lignes de « m ».*

*Exemple 2 : Recopier 10 fois la clef de sol.*

**Les exercices de type « Donner un sens »** se décomposent en deux sous-catégories : traduire et définir (cf. E1 et E2 sur la Figure 2).

La traduction est destinée aux langues étrangères. Elle consiste à fournir une phrase ou un mot précis à l'apprenant et à lui demander la traduction dans une langue donnée. Elle peut être créée semi-automatiquement en laissant l'enseignant fixer le choix du texte à traduire ou le type de vocabulaire (soutenu, familier).

*Exemple : Traduire en espagnol : « Manuela joue au volley ».*

Les définitions consistent à demander à l'apprenant de fournir des notions vues ou non en cours. Elles peuvent être utilisées en langues mais aussi en mathématiques, en biologie...

*Exemple : Donner la définition de LITOTE.*

**Les exercices de type « Tables »** se décomposent en trois sous-catégories : table de conjugaison, table de multiplication et table d'addition (cf. F1, F2 et F3 sur la Figure 2). Dans ces trois types d'activités, l'apprenant doit écrire le verbe précisé au bon temps ou la table de multiplication/addition pour un nombre donné. Ces activités font travailler les connaissances théoriques des apprenants. Elles peuvent être créées semi-automatiquement.

*Exemple 1 : Conjuguer le verbe MANGER au présent de l'indicatif.*

*Exemple 2 : Écrire la table de multiplication du nombre 9.*

**Les exercices de type « Expression algébrique »** consistent à fournir à l'apprenant une expression algébrique en lui demandant de faire certaines opérations dessus. Ces exercices permettent de faire travailler les apprenants en algèbre et plus précisément sur les opérations élémentaires, les priorités de calculs, la factorisation et le développement, les identités remarquables, la résolution d'équations et d'inéquations. Ces exercices d'algèbre peuvent être utilisés tout au long de la scolarité et être créés semi-automatiquement. Pour cela, on peut s'inspirer du générateur automatique d'exercices d'algèbre de l'EIAH APLUSIX [BOUHINEAU et al. 05].

*Exemple 1 : Calculer  $2 + 3$ .*

*Exemple 2 : Développer  $(2x + 3)(5x + 4) + 6(2x + 3)$ .*

*Exemple 3 : Résoudre  $1,5x + 8 < -4,5x$ .*

*Exemple 4 : Résoudre* 
$$\begin{cases} x - 2y = 4 \\ x + 4y = -3 \end{cases}$$

**Les exercices de type « Problème scientifique classifiable »** sont communs à toutes les matières dites scientifiques (mathématiques, physique, chimie...). Ils consistent à fournir un énoncé de problème pour que l'apprenant le résolve. Toute la difficulté de ce genre d'exercices est de fournir une solution. Pour cela, un résolveur de problème est nécessaire, et SYRCLAD [GUIN 97] nous a semblé une bonne solution, dans la mesure où il permet de fournir une solution aux « énoncés des problèmes correspondant à des situations concrètes alors que le cours est donné sous forme de connaissances théoriques » [GUIN 97]. Nous avons spécifié qu'il s'agissait de problèmes scientifiques « classifiables » car SYRCLAD permet de résoudre les problèmes d'un domaine uniquement si une classification des problèmes dans ce domaine peut être définie par les experts du domaine. En acceptant cette limite dans la définition de ce type d'exercices, nous pouvons dire qu'un exercice sera constitué d'un énoncé et de sa correction. Ces exercices pourront donc être créés semi-automatiquement dans la mesure où les experts fourniront les connaissances nécessaires à la génération d'un problème pour ce domaine et les enseignants fourniront le jeu de contraintes permettant de personnaliser la génération du problème.

*Exemple : Tu avais 4 billes, tu en as gagné 10 pendant la récréation. Combien as-tu de billes dans la poche maintenant ?*

**Les exercices de type « Exercice libre »** vont contenir tous les exercices ayant une structure variable. Ainsi, un exercice libre pourra être constitué soit uniquement d'un énoncé, soit d'un énoncé accompagné d'un document, soit contenir un énoncé, un document à étudier ainsi que des questions sur ce document. Les exercices libres vont correspondre aux exercices qui ne sont pas compris dans les autres types d'exercices que nous venons de présenter. De cette manière, la typologie présentée est exhaustive : n'importe quel exercice imaginé par un enseignant pourra être catégorisé grâce à cette typologie. Nous allons à présent voir quelques exemples d'exercices libres.

L'étude de documents peut servir en français ou en histoire/géographie à partir du collège. Sa construction va se faire en deux étapes. L'étape de sélection du document se fera semi-automatiquement à la manière des sélections de textes vues précédemment. La seconde étape consistera aux choix des questions. Cette étape peut se faire semi-automatiquement mais paraît compliquée à mettre en œuvre étant donnée la diversité possible des documents. Les rédactions, les dissertations sont communes à de multiples matières. L'enseignant devra fournir l'énoncé ou le choisir parmi une base de sujets mémorisés. La récitation, le chant, la lecture font travailler la mémoire et dans le cas de poèmes, procurent une culture générale. Ces activités dépendront du texte choisi semi-automatiquement par l'enseignant. Les activités sportives et artistiques vont consister pour l'élève à suivre un plan de travail défini semi-automatiquement par l'enseignant. Le travail d'un instrument de musique se fait grâce à l'entraînement gestuel et ne peut donc être fait à l'écrit. Cette activité consistera donc à choisir semi-automatiquement les morceaux de musique à travailler.

Cette typologie d'exercices est générique, elle a été implémentée dans ADAPTE.

### 3.2. Les générateurs d'exercices

Nous avons détaillé précédemment les différents types d'exercices que nous avons identifiés comme pouvant constituer une feuille d'exercices et donc une activité papier-crayon pouvant être proposée à un apprenant dans ADAPTE. Nous allons maintenant montrer comment nous proposons de générer ces exercices. Pour cela, nous avons étudié l'existant en termes de générateurs, réfléchi à la possibilité d'en réutiliser certains dans notre système et conçu l'architecture des générateurs utiles à ADAPTE.

#### 3.2.1. Quels types de générateurs pour ADAPTE ?

Chacun de types de générateurs présentés dans la section 2.1. a des qualités mais également des défauts. Les générateurs automatiques n'offrent aucune interaction avec leur utilisateur et ne permettent donc pas aux enseignants d'adapter les exercices à leurs méthodes de travail. Mais leur point fort est qu'ils permettent de créer rapidement un nombre important d'exercices et sont pour cette raison ceux qui sont le plus souvent intégrés au

sein des EIAH. Les générateurs manuels quant à eux laissent une totale liberté à l'utilisateur mais, en contrepartie, l'enseignant doit créer l'exercice entièrement sans presque aucune aide. À part le côté interactif des exercices, les générateurs manuels n'apportent donc pas réellement d'avantages par rapport à des exercices créés à la main. Enfin, les générateurs semi-automatiques sont ceux qui nous semblent les plus intéressants. Ils permettent de créer un ensemble d'exercices diversifiés tout en permettant à un enseignant de les adapter à ses méthodes de travail. La limite principale que nous avons pu identifier pour les systèmes de ce type est qu'ils n'en sont qu'au stade expérimental et ne sont pas réellement utilisés par les enseignants au sein de leurs parcours pédagogiques.

Pour ADAPTE, le plus intéressant serait donc qu'il intègre des générateurs semi-automatiques. Mais le problème est de savoir si cela sera faisable pour les neuf types d'exercices que nous avons identifiés. Par exemple, fournir un générateur semi-automatique pour la création de QCM pour tous les domaines et tous les niveaux semble irréaliste. Il faudrait en effet des bases de connaissances énormes pour couvrir toutes les disciplines et tous les niveaux scolaires et universitaires. Une solution serait de proposer à l'enseignant un générateur manuel permettant de fournir une base de connaissances pour un exercice : l'enseignant saisirait une série de questions, avec pour chaque question plusieurs réponses. Il indiquerait de même si les réponses fournies sont justes ou fausses ainsi que le nombre de réponses que le QCM aura pour chaque question. À partir de là, selon ces contraintes renseignées par l'enseignant, le générateur pourra créer semi-automatiquement des exercices en choisissant un certain nombre de questions parmi celles saisies par l'enseignant et à chaque fois un ensemble de réponses. Cela pourra permettre, pour une même définition de QCM, et donc une même structure d'exercices, de générer des QCM différents pour chaque élève : ils n'auront que certaines questions en commun, et pour ces questions communes, ils n'auront pas forcément les mêmes propositions de réponse. C'est cette démarche, qui consiste à demander en quelque sorte à l'enseignant de fournir la base de connaissances pour pouvoir avoir ensuite des générateurs semi-automatiques, qui sera utilisée dans les cas où les générateurs semi-automatiques classiques semblent irréalistes dans notre contexte générique.

### 3.2.2. Utilité des générateurs existants pour ADAPTE

La question est maintenant de savoir si l'on peut dans ADAPTE se servir de générateurs existants. Nous allons pour cela reprendre chaque type d'exercices identifiés (cf. Figure 2) et voir si l'on peut se servir de générateurs existants.

Pour ce qui est des QCM (cf. A sur la Figure 2), il existe de nombreux générateurs, comme GENEVAL [COGNE et al. 98], mais tous ceux que nous avons pu rencontrer sont des outils auteurs. En les utilisant, nous n'aurions pas pu proposer aux enseignants une part d'aléatoire dans la génération de leurs QCM. Nous n'avons donc pas retenu de générateurs existants pour ce type d'exercices. Pour les mêmes raisons, nous n'en avons pas retenu non plus pour les exercices de type B à F de la Figure 2.

Pour ce qui est des exercices basés sur les expressions algébriques (cf. G sur la Figure 2), un générateur nous a semblé très intéressant : le générateur de l'environnement APLUSIX [BOUHINEAU et al. 05]. Le problème est que ce générateur est un générateur automatique. Les enseignants ne peuvent donc pas intervenir dans la création d'un exercice. Nous ne pouvons donc pas l'utiliser directement, mais nous avons décidé de nous en inspirer, en retenant notamment la carte détaillée des exercices possibles qui a été mis en place dans l'environnement APLUSIX.

Pour ce qui est des problèmes scientifiques classifiables (cf. I sur la Figure 2) nous allons intégrer GENAMBRE, le générateur d'AMBRE-enseignant [DUCLOSSON et al. 05]. Nous reviendrons sur son utilisation lors des explications portant sur le générateur correspondant.

### 3.2.3. Les générateurs d'ADAPTE

À chaque type d'exercices présenté sur la Figure 2 va correspondre un générateur d'exercices. Ce générateur va créer l'exercice pour l'apprenant, mais aussi la correction pour l'enseignant. Cette correction sera soit définie par le générateur lorsque que cela est possible, soit saisie par l'enseignant. De même, si des contraintes ne sont pas spécifiées par l'enseignant, elles sont définies par le système. De plus, lors de la génération d'un exercice, la structure d'exercices pourra contenir des contraintes de régénération. Ces contraintes serviront dans le cas où l'enseignant demande un deuxième exercice pour la même structure. Le système tiendra dans ce cas compte des contraintes de régénération pour ne pas fournir le même exercice.

Nous avons défini chacun des générateurs présentés par la suite de façon générique. Leur architecture est donnée selon quatre niveaux. Le niveau *général* contient les connaissances indépendantes du domaine pour lequel on souhaite générer un exercice. Le niveau *domaine* contient les connaissances propres au domaine d'application. Ces connaissances seront diverses, par exemple pour le générateur « Tables » (cf. Figure 8), les connaissances du domaine vont être soit en mathématiques, soit en langues. Le niveau *génération* contient les processus propres à la création d'un exercice. Cette création va se décomposer en deux grandes étapes : la construction d'une structure

d'exercices à partir d'un patron (cf. Figure 2), puis l'instanciation de cette structure pour créer un exercice. Enfin, le niveau *exercice* contient les documents propres à l'exercice créé, avec entre autres, la structure d'exercices et son instanciation, l'exercice contenant un énoncé et une correction.

Nous allons à présent regarder, pour chacun de ces générateurs, leur architecture en détaillant chacun de leur composant.

### GENERATEUR « QCM »

Le générateur de QCM va tout d'abord demander à l'enseignant de définir l'ensemble des contraintes permettant de générer des QCM. Ces contraintes seront stockées dans une structure d'exercices respectant la définition des QCM se trouvant dans le patron (cf. Figure 3). Ensuite, à partir de cette structure d'exercices, le générateur va choisir les questions du QCM ainsi que les réponses proposées, puis organiser le tout de manière à obtenir un exercice.

**Dictionnaire des synonymes :** permet de faire varier les termes d'un énoncé pour augmenter la diversité des énoncés générés.

**Définition des contraintes :** l'enseignant indique, grâce à une interface adaptée, les questions pouvant se trouver dans le QCM ainsi que les réponses à ces questions. Il peut donner plus de questions que n'en contiendra le QCM. Il devra alors indiquer le nombre de questions que devra contenir le QCM ainsi que le nombre de réponses par question. L'enseignant définira également les contraintes de mise en page (une réponse par ligne, toutes les réponses sur une ligne, image avant ou après le texte...). Toutes ces informations seront contenues dans la structure d'exercices.

**Choix des questions :** le système choisit les questions aléatoirement parmi celles contenues dans la structure d'exercices.

**Choix des réponses :** le système choisit aléatoirement les réponses aux questions en respectant la contrainte donnée par l'enseignant : X réponses justes parmi Y réponses, sachant que  $X \leq Y$  et que Y est au moins inférieur au nombre total de réponses.

**Mise en page :** selon les contraintes de mise en page choisies par défaut ou modifiées par l'enseignant, le système crée l'exercice avec les questions choisies et leurs réponses.

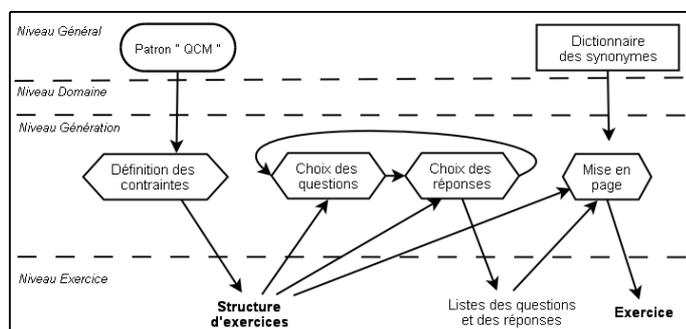


Figure 3 : Architecture du générateur « QCM »

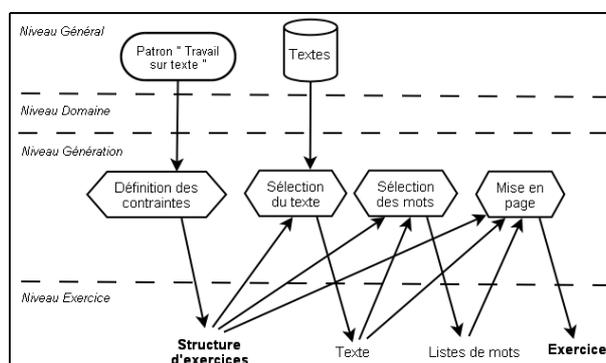


Figure 4 : Architecture du générateur « Travail sur texte »

### GENERATEUR « TRAVAIL SUR TEXTE »

Le générateur permettant de faire des exercices du type « Travail sur texte » va tout d'abord demander à l'enseignant de spécifier les contraintes permettant de créer semi-automatiquement l'exercice pour les stocker dans la structure d'exercices définie par le patron (cf. Figure 4). Le système va ensuite choisir le texte et les mots à sélectionner. Enfin, selon le type précis de l'exercice (cf. B1 et B2 sur la Figure 2), il proposera l'exercice contenant le texte avec éventuellement la liste de mots sur lesquels portera le travail.

**Textes :** base de données contenant tous les textes traités par le système : les textes d'origine, ainsi que ceux importés par les enseignants.

**Définition des contraintes :** l'enseignant indique comment choisir le texte : soit il le saisit ou l'importe, soit il définit des contraintes de sélection automatique d'un texte par le système. L'enseignant indique également comment choisir les mots à sélectionner dans le texte. Dans le cas de la saisie ou de l'importation d'un texte, il peut indiquer quels mots précisément seront sélectionnés. Mais il peut aussi demander au système de sélectionner automatiquement des mots selon certaines contraintes (taille des mots, exclure les mots commençant par une majuscule, exclure les nombres...). C'est ce que fera le système si le texte est choisi automatiquement. Dans tous

les cas, l'enseignant peut fournir plusieurs textes et des contraintes de sélection des mots différentes pour chaque texte ou type de textes.

**Sélection du texte / des mots :** le système choisit le texte et les mots parmi ceux proposés par l'enseignant ou selon les contraintes définies.

**Mise en page :** pour créer l'exercice, le système regardera si la liste des mots sélectionnés doit être fournie ou non. Il présentera ensuite l'énoncé, le texte, et le cas échéant la liste de mots, selon les contraintes de mise en page définies.

### GENERATEUR « ORGANISER »

Le générateur des exercices demandant à l'apprenant d'organiser des objets va stocker les contraintes définies par l'enseignant dans la structure d'exercices, selon la définition contenue dans le patron (cf. Figure 5). Le système va alors créer l'énoncé en sélectionnant, grâce aux contraintes, les objets sur lesquels va porter l'exercice. Le système va également définir la façon dont les objets seront organisés et créer la solution de l'exercice. Il proposera ensuite le tout à l'enseignant.

**Textes / Images :** bases de données contenant les images et les textes utilisés par le système.

**Calcul sur les nombres :** connaissances permettant de comparer les nombres.

**Dictionnaire :** connaissances permettant de classer les mots dans un ordre donné (alphabétique, par taille...).

**Définition des contraintes :** l'enseignant indique précisément quel type d'exercices il souhaite, puis complète les contraintes propres à ce type d'exercices (ordre de classement des objets, choix des objets...).

**Définition du contenu de l'organisation :** pour ce qui est du travail sur un texte, le système choisit un texte dans la base de données selon les contraintes spécifiées ou au hasard dans la liste de textes fournis. Ensuite, le système découpe le texte selon les contraintes spécifiées ou aux endroits indiqués sur les textes fournis, pour atteindre le nombre de morceaux de texte voulu ; pour ce qui est du travail sur les nombres, le système choisit les nombres aléatoirement, en respectant les contraintes fournies ; pour ce qui est du travail sur les images, le système choisit des images parmi celles données ou aléatoirement selon les contraintes fournies.

**Mise en page :** le système mélange les objets choisis avant de proposer l'exercice.

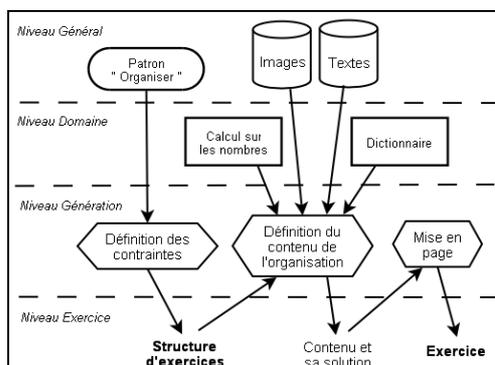


Figure 5 : Architecture du générateur « Organiser »

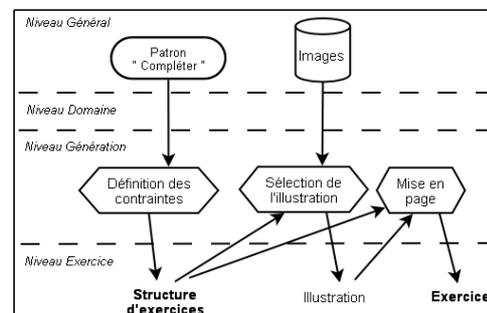


Figure 6 : Architecture du générateur « Compléter »

### GENERATEUR « COMPLÉTER »

Le générateur demandant à l'apprenant de compléter un exercice va stocker les contraintes de génération choisies par l'enseignant dans la structure d'exercices en respectant le patron (cf. Figure 6). Le système va ensuite avec ces contraintes sélectionner l'illustration soit à compléter, soit à recopier (cf. D1 et D2 sur la Figure 2), puis formuler l'exercice avec une mise en page correcte.

**Images :** base de données contenant les illustrations : les images, les cartes...

**Définition des contraintes :** l'enseignant indique si l'objectif de l'exercice sera de compléter une illustration ou de remplir des lignes d'écriture. Il fournit une liste d'illustrations, indique les mots à placer sur les illustrations ou ce que l'apprenant doit recopier dans ses lignes d'écriture.

**Sélection de l'illustration :** le système choisit l'illustration sur laquelle portera l'exercice parmi la liste que contient la structure d'exercices. La sélection automatique de l'illustration dans une base de données demande de trier les illustrations par disciplines et d'avoir des connaissances permettant de les choisir. Dans l'environnement EPROFILEA, cela paraît difficile à mettre en place étant donné le nombre de disciplines et les différents niveaux envisagés. Cela demanderait une charge de travail importante pour avoir au final peut d'intérêt puisque quand un enseignant souhaite faire remplir une illustration par ses élèves, il a souvent une idée précise de ce qu'il veut faire.

**Mise en page** : le système affiche la consigne de l'exercice, l'illustration et si besoin la liste de mots à placer ou les lignes permettant à l'apprenant de compléter des lignes d'écriture.

### GENERATEUR « DONNER UN SENS »

Le générateur permettant de créer des exercices demandant à l'apprenant de fournir le sens d'un mot ou d'un texte (cf. E sur la Figure 2) va mémoriser les contraintes de génération d'exercices définies par l'enseignant, dans la structure d'exercices selon la définition contenue dans le patron d'exercices (cf. Figure 7). Ensuite le système va choisir la liste des mots dont le sens est à trouver et chercher soit dans le traducteur, soit dans le dictionnaire, le sens de ces mots. Enfin, il met l'exercice en page.

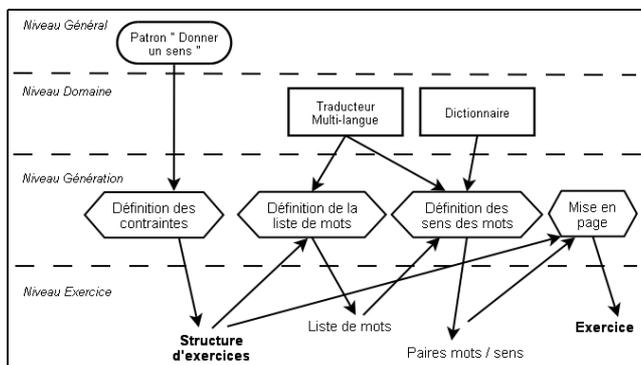


Figure 7 : Architecture du générateur « Donner un sens »

**Traducteur Multi-langue** : base de connaissances contenant les mots et leurs traductions dans plusieurs langues.

**Dictionnaire** : base de connaissances contenant les mots et leurs définitions, a priori en français.

**Définition des contraintes** : l'enseignant choisit s'il veut créer un exercice de définition ou de traduction. Il indique également comment choisir les mots et/ou donne des mots précis sur lesquels faire travailler les apprenants.

**Définition de la liste de mots** : deux cas se présentent selon que le générateur crée un exercice de définition ou de traduction. Pour la définition, le système choisit un nombre de mots parmi ceux sélectionnés par l'enseignant dans la base de connaissances (le dictionnaire). Pour la traduction, le système prend un certain nombre de mots prioritaires et choisit les autres mots dans la base de connaissances (le traducteur) selon les critères donnés par l'enseignant. Au final, on a une liste de mots, tous présents dans la base de connaissances, avec s'il y a confusion possible, le sens précis pour un mot.

**Définition des sens des mots** : le système cherche dans les bases de connaissances le(s) sens des mots choisis.

**Mise en page** : le système affiche soit le mot, soit son sens, pour que l'apprenant fournisse la partie manquante puisqu'il est intéressant dans le cas « X a pour sens Y » de demander à l'apprenant de donner parfois X, parfois Y.

### GENERATEUR « TABLES »

Le générateur de tables de conjugaison, multiplication et d'addition va permettre à l'enseignant de définir les contraintes de génération. Le système va mémoriser ces contraintes dans la structure d'exercices selon le patron (cf. Figure 8). Ensuite le système va sélectionner les objets (verbes ou nombres) pour lesquels on va demander de fournir les tables et va, grâce aux connaissances contenues dans le générateur, créer la solution de l'exercice. Enfin, l'étape de mise en page permettra à l'apprenant d'avoir l'énoncé de l'exercice, et à l'enseignant d'en avoir la correction.

**Connaissances de conjugaison** : propre à la langue choisie pour faire travailler l'apprenant, cette base de connaissances contient les verbes avec leur conjugaison à tous les temps et tous les modes.

**Connaissances de multiplication / addition** : communes à toutes les langues, ces bases de connaissances contiennent les tables de multiplication/addition.

**Définition des contraintes** : l'enseignant choisit le type d'exercices et les contraintes de génération de tables (choix du temps et des personnes pour les tables de conjugaison, de l'intervalle des nombres pour les tables de multiplication ou d'addition...). Il peut également préciser des objets (verbes ou nombres) prioritaires i.e. des objets qu'il souhaite avoir en priorité dans les exercices qui seront générés.

**Sélection des objets** : le système regarde le nombre total d'objets (verbes ou nombres) à mettre dans la liste d'objets de l'exercice ainsi que le nombre d'objets à prendre dans la liste des objets prioritaires. Trois cas de

figure se posent alors. Si ces deux nombres sont égaux, la liste d'objets de l'exercice correspond à la liste des objets prioritaires. Si l'on a plus d'objets prioritaires que d'objets à sélectionner, le système effectue un choix aléatoire parmi les objets prioritaires. Sinon, dans le cas où l'on doit sélectionner plus d'objets que le nombre d'objets prioritaires, le système prend ces objets prioritaires et les complète avec des objets choisis aléatoirement selon les contraintes contenues dans la structure d'exercices. Ces contraintes dépendent du domaine d'application.

**Création de la solution :** selon la liste d'objets fournis, le système cherche les tables correspondantes dans la base de connaissances propre aux objets.

**Mise en page :** le système présente la consigne de l'exercice suivi de la liste des objets et on fournit également la correction pour l'enseignant.

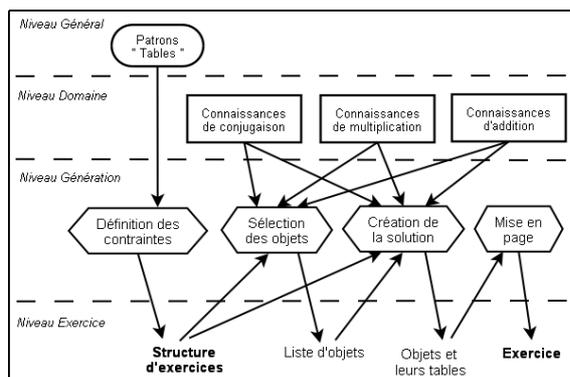


Figure 8 : Architecture du générateur « Tables »

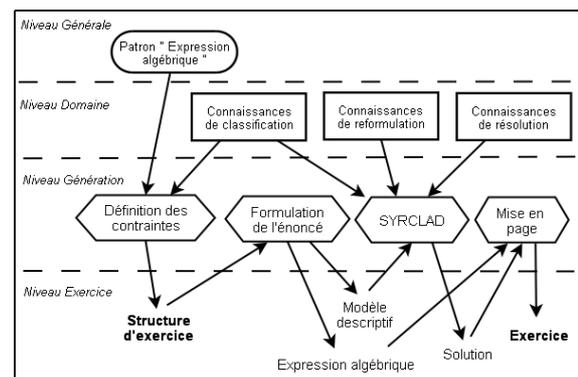


Figure 9 : Architecture du générateur « Expression algébrique »

### GENERATEUR « EXPRESSION ALGEBRIQUE »

Le générateur d'expressions algébriques s'appuie sur SYRCLAD [GUIN 97]. Ce générateur va stocker les contraintes de génération définies par l'enseignant dans une structure d'exercices définie par le patron (cf. Figure 9). Grâce à ces contraintes, le système va pouvoir créer une expression algébrique et l'énoncé qui lui est associé, mais également le modèle descriptif de l'exercice pour que le résolveur SYRCLAD puisse fournir une correction. Enfin, la mise en page permettra de fournir l'énoncé contenant l'expression algébrique ainsi que la correction de l'exercice.

**Connaissances de classification :** un expert fournit au résolveur SYRCLAD, et donc par extension au générateur, un graphe de classification des expressions. Cette classification peut être inspirée de la carte de tests du générateur d'expressions du système APLUSIX [BOUHINEAU et al. 05]. Cette carte de tests comprend six types d'expressions : calculer, développer, factoriser, résoudre une équation, résoudre une inéquation, résoudre un système. Pour chacun de ces types, il existe ensuite plusieurs classes de problèmes selon les méthodes de résolution qui leur sont associées.

**Connaissances de reformulation :** les attributs discriminants intervenant dans le graphe de classification sont des attributs relevant de la théorie du domaine. La plupart du temps, ces attributs ne sont pas présents dans les modèles descriptifs des problèmes. Il est donc nécessaire d'avoir des connaissances permettant de déterminer la valeur de ces attributs. Ces connaissances permettent alors de passer d'un modèle descriptif à un modèle opérationnel, plus adapté au choix d'une méthode de résolution dans SYRCLAD. Pour que SYRCLAD puisse utiliser ces connaissances de reformulation, l'expert doit les exprimer par des règles « si-alors ».

**Connaissances de résolution :** pour choisir la méthode de résolution à employer, il faut savoir quelle méthode est adaptée pour un problème d'une classe donnée.

**Définition des contraintes :** selon la classe de l'expression qu'il a choisie, l'enseignant va préciser les contraintes de génération. Ces contraintes peuvent porter sur les nombres utilisés (entiers, décimaux, fractions, radicaux), sur leur signe, leur intervalle de grandeur... Mais elles peuvent aussi porter sur les règles utilisées pour construire l'expression : les opérateurs utilisés, le nombre de termes, l'utilisation ou non de la retenue pour les additions/soustractions, l'utilisation ou non de parenthèses, le degré des équations, le nombre d'inconnues dans les systèmes ou les équations... Ces contraintes vont permettre de construire l'expression algébrique.

**Formulation de l'énoncé :** selon les contraintes définies par l'enseignant, les nombres contenus dans l'expression sont choisis aléatoirement. Le système produit également le modèle descriptif de l'expression pour SYRCLAD.

**SYRCLAD [GUIN 97] :** résolveur utilisé pour trouver une solution à l'expression algébrique créée.

**Mise en page** : assemblage de l'expression algébrique et de la solution fournie par SYRCLAD.

### GENERATEUR « PROBLEME SCIENTIFIQUE CLASSIFIABLE »

Le générateur de problèmes scientifiques classifiables est un générateur qui inclut le générateur GENAMBRE [DUCLOSSON et al. 05] en lui ajoutant une interface générique pour que l'enseignant puisse définir ses contraintes de génération d'exercices (cf. Figure 10). On utilise également le résolveur de problème SYRCLAD [GUIN 97] pour avoir la solution du problème proposé par GENAMBRE. Enfin, une étape de mise en page permet d'avoir un exercice ayant la même présentation que les autres exercices proposés par ADAPTE.

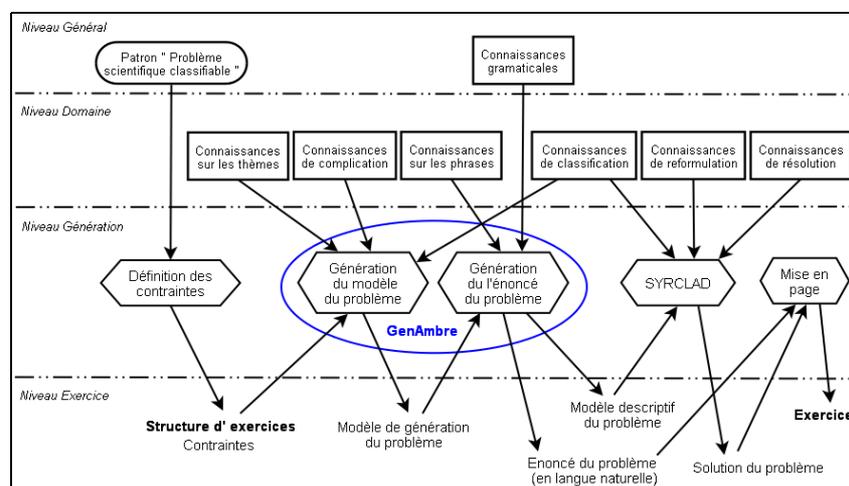


Figure 10 : Architecture du générateur « Problème scientifique classifiable »

**Connaissances grammaticales** : l'expert fournit une grammaire au système de génération, c'est-à-dire un ensemble de structures de phrases qui pourront être utilisées. Les unités syntaxiques de base sont les groupes sujets, les verbes et les compléments. Ces grammaire est indépendante du domaine.

**Connaissances sur les thèmes** : pour pouvoir générer un problème, nous avons besoin de connaître le thème concerné et les traits de surface qui lui sont associés (par exemple les objets, personnages, les actions). De telles connaissances sont dépendantes du domaine pour lequel elles sont utilisées, mais leur représentation est indépendante du domaine. Les connaissances sur les thèmes sont fournies par l'expert.

**Connaissances de complication** : pour pouvoir compliquer l'énoncé de l'exercice, il est nécessaire de fournir des connaissances de complication au système de génération. Ces connaissances vont permettre de changer l'ordre des phrases ou d'ajouter des phrases inutiles par exemple.

**Connaissances sur les phrases** : les énoncés générés, notamment la structure de leurs phrases, dépendent de la classe à laquelle le problème appartient. Il est donc nécessaire pour pouvoir générer l'énoncé en langue naturelle, de savoir quelles structures de phrases (de la grammaire) pourront être utilisées pour le problème à générer. Ce sont ces informations que contiennent les connaissances sur les phrases. Elles permettent d'associer à la classe du problème les structures de phrases utilisables, et les éléments du problème qui seront associés à ces structures.

**Connaissances de classification** : dans chaque domaine d'application, un expert fournit au résolveur SYRCLAD, et donc par extension au générateur GENAMBRE, un graphe de classification de problèmes. Ce graphe est utilisé dans le résolveur pour classer le problème. C'est une hiérarchie de classes dépendante du domaine, mais dont la représentation est la même quelque soit le domaine.

**Connaissances de reformulation** : les attributs discriminants intervenant dans le graphe de classification sont des attributs relevant de la théorie du domaine. La plupart du temps, ces attributs ne sont pas présents dans les modèles descriptifs des problèmes. Il est donc nécessaire d'avoir des connaissances permettant de déterminer la valeur de ces attributs. Ces connaissances permettent alors de passer d'un modèle descriptif à un modèle opérationnel, plus adapté au choix d'une méthode de résolution dans SYRCLAD. Pour que SYRCLAD puisse utiliser ces connaissances de reformulation, l'expert les exprime par des règles « si-alors ».

**Connaissance de résolution** : pour choisir la méthode de résolution à employer, il faut savoir quelle méthode est adaptée pour un problème d'une classe donnée.

**Définition des contraintes** : l'enseignant définit les contraintes de génération du problème (intervalle des nombres utilisés, ajout de phrases inutiles...) utiles à GENAMBRE pour créer une structure d'exercices.

**GENAMBRE – Génération du modèle du problème** : à partir des trois bases de connaissances décrites précédemment et des contraintes entrées par l'enseignant, le système doit générer le modèle de génération du

problème. Ce modèle est en fait un modèle descriptif étendu, puisqu'il contient le modèle descriptif du problème, mais précise également quelques informations absentes du modèle descriptif, telles que la classe du problème ou encore son thème. Pour construire ce modèle, le processus complète les contraintes définies par l'enseignant, en choisissant notamment aléatoirement des valeurs pour celles qui n'ont pas été précisées.

**GENAMBRE – Génération de l'énoncé du problème :** pour générer un énoncé en langue naturelle, le processus utilise les connaissances sur les phrases et la grammaire du domaine, ainsi que le modèle de génération du problème créé auparavant. Les connaissances sur les phrases permettent au système de prendre les décisions d'ordre conceptuel (« décider quoi dire ») : on spécifie ici la structure des phrases qui seront générées en précisant pour chaque phrase les éléments qui la composeront (issus du modèle du problème : objets, personnages...). Une fois les décisions d'ordre conceptuel prises, le processus passe à l'étape de génération de texte, et met donc en place différents mécanismes linguistiques : traitements syntaxiques (« décider comment le dire »), lexicaux et morphologiques (« décider comment l'écrire »).

**SYRCLAD [GUIN 97] :** résolveur utilisé pour trouver une solution au problème créé.

**Mise en page :** le système assemble l'énoncé du problème et la solution fournie par SYRCLAD.

### GENERATEUR « EXERCICE LIBRE »

Le générateur d'exercices libres permet de créer les exercices qui ne sont pas proposés par les autres générateurs d'ADAPTE (cf. I sur la Figure 2). Dans ce générateur, l'enseignant définit un ensemble de contraintes stockées dans une structure d'exercices selon un patron d'exercices (cf. Figure 11). Ensuite, selon le type d'exercices défini par les contraintes, le système peut choisir un texte et des questions s'y rapportant (cf. Structure I.003 sur la Figure 2) ou juste fournir un énoncé (cf. Structure I.001 sur la Figure 2). Enfin, une mise en page définie dans les contraintes permet d'afficher l'exercice correctement pour l'apprenant.

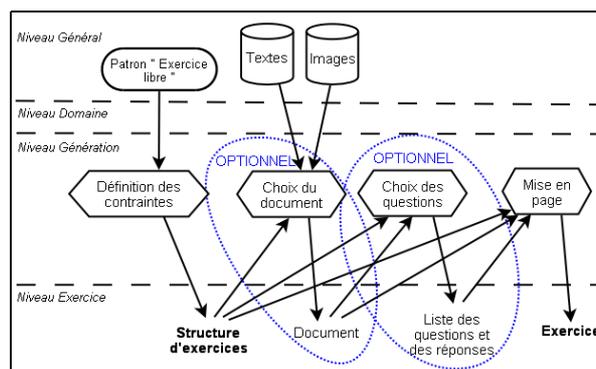


Figure 11 : Architecture du générateur « Exercice libre »

**Textes / Images :** bases de données contenant les textes et les images utilisés par le système.

**Définition des contraintes :** l'enseignant spécifie l'énoncé, le document à ajouter et le cas échéant les questions. Il définit également la mise en page de l'exercice. Dans le cas d'un exercice portant sur un texte (récitation, lecture...) ou un document n'entraînant pas de question (une partition en musique...), l'enseignant peut ne pas préciser le document et définir des contraintes permettant au système de le choisir automatiquement.

**Choix du document :** le système choisit un document parmi la liste de documents fournis par l'enseignant ou selon les contraintes spécifiées.

**Choix des questions :** le système choisit N questions parmi toutes, N étant précisé par l'enseignant lors de la création de la structure d'exercices.

**Mise en page :** le système affiche l'énoncé, et éventuellement le document et des questions, selon la mise en page définie par l'enseignant et stockée dans la structure d'exercices.

Chacune des neuf architectures que nous venons de voir a été définie de façon générique. Cette généricité porte sur deux points. Premièrement, ces architectures sont toutes données selon quatre niveaux : le niveau *général* qui contient les connaissances indépendantes du domaine, le niveau *domaine* qui contient les connaissances propres au domaine d'application, le niveau *génération* qui contient les processus de création d'un exercice et enfin le niveau *exercice* qui contient les documents propres à l'exercice créé. Deuxièmement, si l'on regarde les processus de génération de chacune de ces architectures, on peut voir apparaître deux étapes dans la génération d'un exercice : la construction d'une structure d'exercices à partir d'un patron puis l'instanciation de cette structure pour créer un exercice. Ces architectures étant génériques, nous avons pu en implémenter certaines dans le cas spécifique d'ADAPTE (cf. section 4.2.2.).

### 3.3. L'affectation d'une feuille d'exercices à un profil d'apprenant

Dans cette section, nous allons expliquer comment nous définissons les feuilles d'exercices en fonction des profils d'apprenant. Nous allons pour cela rappeler les différents types de profils pris en compte dans ADAPTE puis nous verrons comment cette affectation profil - feuille d'exercices est mise en place dans ADAPTE.

#### 3.3.1. Les types de profils

ADAPTE permet de traiter deux sortes de profils : les profils d'élève et les profils de classe. Un profil d'élève va correspondre au profil d'apprenant dont la définition a été fournie dans la section 2.1. Un profil de classe est le résultat de calculs et de fusions sur les profils d'élèves afin d'offrir aux enseignants une vue des connaissances d'un groupe d'apprenants. Quelque soit le profil pris en compte, il est défini grâce au langage de description des profils mis en place dans l'environnement EPROFILEA. Nous reviendrons par la suite sur ce langage de description.

#### 3.3.2. Définition des critères d'affectation

Pour pouvoir comprendre la manière dont ADAPTE va permettre aux enseignants de définir des critères d'affectation d'une feuille d'exercices à un profil, il faut tout d'abord savoir comment sont définis les profils. Rappelons tout d'abord que l'idée du projet PERLEA est de proposer un cadre au travail de l'enseignant sur les profils, mais en aucun cas de proposer les connaissances contenues dans les profils. Dans l'environnement EPROFILEA, l'enseignant va pouvoir décrire une structure de profils vide qui définira de quelle manière sera agencé le contenu des profils qu'il souhaite obtenir à la fin de la phase de normalisation (cf. Figure 1). C'est cette structure de profils qui, une fois remplie, formera un profil d'apprenant normalisé. Nous allons donc étudier plus en détail le langage de description des structures de profils de l'environnement EPROFILEA, puis nous verrons de quelles manières l'enseignant peut définir ses critères d'affectations.

```
- <brique id="2" type="0" nom="Calcul" logiciel_externe="Carnet de Bord" date_evaluation="20/04/2006" evaluation_source=""
commentaire="COMPTER EN CHATS" indice="1">
- <infos_echelle>
  <echelle id="1" unite=""/>
</infos_echelle>
- <arbre_des_composantes niveaux="3" ponderation="faux">
- <composante nom="Multiplication">
- <sous_composante nom="mes tables de multiplication" nomabrege="tables de multiplication">
  <valeur>7</valeur>
</sous_composante>
- <sous_composante nom="la technique de la multiplication" nomabrege="technique multiplication">
  <valeur>4</valeur>
</sous_composante>
- <sous_composante nom="la technique de la multiplication pour les nombres décimaux" nomabrege="techn multi nb décimaux">
  <valeur>6</valeur>
</sous_composante>
- <sous_composante nom="à quel moment utiliser la multiplication" nomabrege="moment utiliser multi!">
  <valeur>10</valeur>
</sous_composante>
</composante>
- <composante nom="Conversion">
- <sous_composante nom="utiliser les unités de masse" nomabrege="unités de masse">
  <valeur>3</valeur>
</sous_composante>
</composante>
- <composante nom="Calcul">
- <sous_composante nom="faire des calculs complexes" nomabrege="calculs complexes">
  <valeur/>
</sous_composante>
</composante>
</arbre_des_composantes>
</brique>
```

Figure 12 : Brique d'un profil normalisé issu d'EPROFILEA

Le langage de description des profils dans EPROFILEA [JEAN-DAUBIAS et al. 05] doit être assez riche et précis pour permettre à tout enseignant de décrire les profils de ses apprenants. D'un point de vue opérationnel, la structure de profils est divisée en entités élémentaires appelées briques. Pour faciliter la création de la structure de profils, EPROFILEA propose un certain nombre de types de brique qui permettent une organisation des données.

Les briques de type **composantes** permettent d'étudier les connaissances que l'élève a assimilé ou non. Ces connaissances sont décrites sous forme de composantes éventuellement détaillées en sous-composantes. Il faut savoir que seules les composantes ou sous-composantes de dernier niveau, c'est-à-dire celles qui n'ont pas de sous-composantes, sont évaluées ; la valeur d'une composante plus globale est calculée en effectuant la moyenne des valeurs de ses sous-composantes, éventuellement pondérée. Chaque valeur est rattachée à une échelle qui

peut être numérique (par exemple [0..20]), textuelle graduée (par exemple [ABCDEF]) ou textuelle non graduée (par exemple [Attentif, Passif, Bavard]).

Les briques de type **répartition** permettent à l'enseignant d'étudier la répartition des réponses de l'élève parmi un certain nombre de composantes, éventuellement détaillées en sous-composantes, pour identifier une ou des dominantes dans le comportement de l'élève. Ces composantes peuvent être exclusives ou non.

Les briques de type **articulation** permettent d'étudier la maîtrise par l'élève du passage d'une compétence à une autre dans un réseau conceptuel.

Les briques de type **graphes** permettent de représenter des réseaux conceptuels. Elles ont une organisation proche de celle des briques de type articulation. Néanmoins, l'évaluation porte sur les concepts et non sur l'articulation entre les concepts, le but étant ici d'évaluer des compétences tout en conservant les informations sur les liens entre ces compétences.

Enfin, les briques de type **commentaire** permettent à l'enseignant d'associer au profil des commentaires ou remarques.

Règles	Conditions	Actions
<b>CONTRAINTE SUR UNE VALEUR</b>		
R 1.1	Si la note de l'apprenant appartient à l'intervalle ou à la liste de valeurs souhaitées	Alors la contrainte est validée
R 1.2	Si la note de l'apprenant n'appartient pas à l'intervalle ou à la liste de valeurs souhaitées	Alors la contrainte est non validée
R 2	Si l'apprenant n'a pas de note	Alors la contrainte est validée
<b>CONTRAINTE SUR UNE BRIQUE OU UNE COMPOSANTE</b>		
R 3	Si une des notes a une échelle textuelle non ordonnée	Alors la contrainte est infaisable
R 4	Si la contrainte est une moyenne	Alors on convertit toutes les notes en pourcentage
R 5	Si la contrainte est une somme	Alors on convertit les notes ayant une échelle textuelle ordonnée en pourcentage
R 6.1	Si le résultat de l'opération sur les notes de l'élève appartient à l'intervalle souhaité	Alors la contrainte est validée
R 6.2	Si le résultat de l'opération sur les notes de l'élève n'appartient pas à l'intervalle souhaité	Alors la contrainte est non validée
R 7.1	Si la brique est de type répartition, que la composante choisie est exclusive et que cette composante a une note	Alors la contrainte est validée
R 7.2	Si la brique est de type répartition, que la composante choisie est exclusive et que cette composante n'a pas de note	Alors la contrainte est non validée
<b>COMBINAISON DE CONTRAINTES</b>		
R 8	Si la contrainte est de la forme (a et b)	Alors la contrainte sera validé si et seulement si a est validée et b est validée
R 9	Si la contrainte est de la forme (a ou b)	Alors la contrainte sera validé si et seulement si a est validée ou b est validée
<b>CRITERE D'AFFECTION</b>		
R 10.1	Si la contrainte ou la combinaison de contraintes formant le critère est validée	Alors un exercice sera affecté à l'apprenant
R 10.2	Si la contrainte ou la combinaison de contraintes formant le critère est non validée	Alors on n'affecte pas d'exercice à l'apprenant

**Tableau 1 : Règles de mise en correspondance des critères d'affectation avec un profil d'apprenant**

**Les critères d'affectation** vont permettre au logiciel de fournir une feuille d'exercices adaptée au profil de chaque apprenant. Pour cela on va demander à l'enseignant de préciser comment créer cette feuille. L'enseignant va affecter une structure d'exercices à un certain type de difficulté ou de compétence de l'élève. L'enseignant va de plus définir des contraintes générales qui vont permettre de contrôler la longueur ou le niveau de difficulté de la feuille d'exercices de chaque élève.

Ces principes d'affectation ont été proposés et discutés avec deux enseignants de primaires, partenaires de conception, pour rester cohérent avec leur manière d'affecter des exercices aux élèves dans leur pratique d'enseignement. Nous allons à présent revenir et préciser ces principes d'affectations en donnant des exemples se basant sur un profil normalisé d'EPROFILEA (cf. Figure 12).

Un critère va être constitué de plusieurs éléments : son **nom**, un ensemble de **contraintes** sur les différentes briques d'une structure de profils, combinées ou non, la **structure d'exercices** associée à ce critère et l'**importance** de ce critère.

Ce sont les contraintes sur les briques de la structure de profils qui vont permettre à l'enseignant de préciser sur quelle difficulté ou compétence faire travailler ses élèves. Pour cela, il va choisir une valeur dans le profil selon plusieurs niveaux de granularité. L'enseignant pourra choisir :

- soit la valeur d'une composante, par exemple la valeur de la composante « mes tables de multiplication » sur la Figure 12 ;
- soit la valeur calculée pour une sous-composante à partir de l'ensemble des ses valeurs ;
- soit la valeur calculée d'une composante à partir de celle de ses sous-composantes, par exemple la valeur calculée pour la composante « multiplication » de la Figure 12 sera obtenue grâce aux valeurs des sous-composantes « mes tables de multiplication », « la technique de la multiplication », « la technique de la multiplication pour les nombres décimaux » et « à quel moment utiliser la multiplication » ;
- soit la valeur calculée d'une brique à partir des valeurs de ses composantes, par exemple la valeur calculée pour la brique « Calcul » de la Figure 12 sera obtenue grâce aux valeurs des composantes « Multiplication », « Conversion et « Calcul ».

À la valeur choisie va correspondre une échelle. C'est à partir de cette échelle que l'enseignant va définir un intervalle ou des valeurs précises pour lesquels le système devra fournir un exercice à un élève. Dans le cas de valeurs issues d'opérations sur les valeurs d'une brique ou d'une composante, l'enseignant précisera s'il faut faire une somme ou une moyenne des valeurs comprises dans la brique ou la composante. Dans le cas où un profil ne posséderait pas de valeurs pour un critère donné, comme pour la composante « Calcul » de la Figure 12, nous avons décidé en accord avec les enseignants partenaires de conception, de lui proposer par défaut l'exercice associé pour permettre à l'enseignant d'évaluer les compétences correspondantes de l'élève. Dans la pratique, le système utilise les règles R1.1 à R6.2 du Tableau 1 pour connaître la validité d'une contrainte sur le profil d'apprenant.

De plus, l'enseignant pourra combiner plusieurs contraintes et ainsi affecter un exercice à un élève qui possède telle et telle compétence, telle ou telle difficulté... Le système se servira alors des règles R8 et R9 du Tableau 1 pour connaître le résultat de la combinaison des contraintes sur un profil d'apprenant.

De même, dans le cas de briques de type répartition, si la composante est exclusive, on pourra également choisir d'affecter un exercice en fonction de l'appartenance à une composante ou non, grâce aux règles R7.1 et R7.2 du Tableau 1. Enfin, notons que les briques de types commentaire ne contiennent pas de valeur et ne permettent donc pas d'affecter des exercices à un élève.

Règles	Conditions	Actions
<b>NOMBRE D'EXERCICES PAR FEUILLE</b>		
R 11.1	Si le nombre d'exercices affectés est supérieur au nombre d'exercices maximal demandé	Alors on supprime les exercices en commençant par ceux ayant le niveau de priorité le plus faible pour finir par ceux ayant le plus élevé
R 11.2	Si le nombre d'exercices affectés est inférieur au nombre d'exercices minimal demandé	Alors on ajoute les exercices en commençant par ceux ayant le niveau de priorité le plus élevé pour finir par ceux ayant le plus faible
<b>DUREE DE LA FEUILLE D'EXERCICES</b>		
R 12.1	Si la durée de la feuille d'exercices est supérieure à celle demandée	Alors on supprime les exercices en commençant par ceux ayant le niveau de priorité le plus faible pour finir par ceux ayant le plus élevé
R 12.2	Si la durée de la feuille d'exercices est inférieure à celle demandée	Alors on ajoute les exercices en commençant par ceux ayant le niveau de priorité le plus élevé pour finir par ceux ayant le plus faible
<b>EXERCICES IDENTIQUES</b>		
R 13	S'il a été demandé que les exercices soit uniques à chaque apprenant	Alors à chaque génération d'exercice, on vérifie que l'exercice n'a pas déjà été donné auparavant

**Tableau 2 : Règles d'exécution des contraintes générales**

Pour ce qui est des contraintes générales, elles vont permettre de « borner » la taille ou la difficulté des feuilles d'exercices proposées. Ainsi l'enseignant pourra demander d'avoir un nombre minimum ou maximum d'exercices pour toutes les feuilles créées ou seulement pour certaines. Le système se servira alors des règles R11.1 et R11.2 du Tableau 2 pour faire en sorte que les feuilles d'exercices aient le nombre d'exercices souhaité.

De même, l'enseignant pourra demander que la feuille générée ait une durée minimum ou maximum de résolution par l'apprenant. C'est pour respecter ces contraintes générales que l'on va prendre en compte l'importance des critères. Ainsi si un élève correspond à 10 critères définis par l'enseignant, mais que celui-ci a demandé à avoir au plus 5 exercices par élève, on fournira en priorité à l'élève les exercices correspondant aux critères les plus importants grâce à la règle R12.1 du tableau 2. De la même manière, si un élève n'a pas assez d'exercices sur la feuille générée, on proposera plusieurs exercices pour les critères prioritaires grâce à la règle R12.2 du tableau 2.

Toute la difficulté de cette affectation de feuilles d'exercices à un élève consiste à assister l'enseignant dans son travail pédagogique, mais surtout à ne pas prendre sa place de pédagogue au sein de la classe. C'est pour cela que

On va proposer à l'enseignant des feuilles d'exercices qu'il pourra modifier soit en demandant de régénérer un exercice si les valeurs proposées ne lui conviennent pas, soit en supprimant ou en ajoutant des exercices.

Nous avons vu dans cette partie la typologie d'exercices proposée ainsi que les architectures des neuf générateurs d'exercices permettant de créer les différents exercices contenus dans cette typologie. Nous avons de même présenté les règles pour permettre d'affecter une feuille d'exercices personnalisées à un profil d'apprenant. Nous allons à présent passer à la partie concernant la réalisation du module ADAPTE.

## 4. REALISATION D'ADAPTE

Dans cette partie, nous allons présenter l'architecture que nous proposons pour ADAPTE. Nous verrons ensuite comment nous avons mis en place un prototype en fonction de cette architecture, puis nous parlerons de l'évaluation de ce prototype par les enseignants partenaires de conception.

### 4.1. L'architecture proposée

Une fois que nous avons défini les différents exercices pouvant être proposés par ADAPTE, la manière de les générer, ainsi que les règles permettant d'affecter ces exercices à un apprenant, nous avons pu établir une architecture pour ADAPTE (cf. Figure 13). Nous allons la détailler par la suite en expliquant comment un enseignant va pouvoir se servir du logiciel.

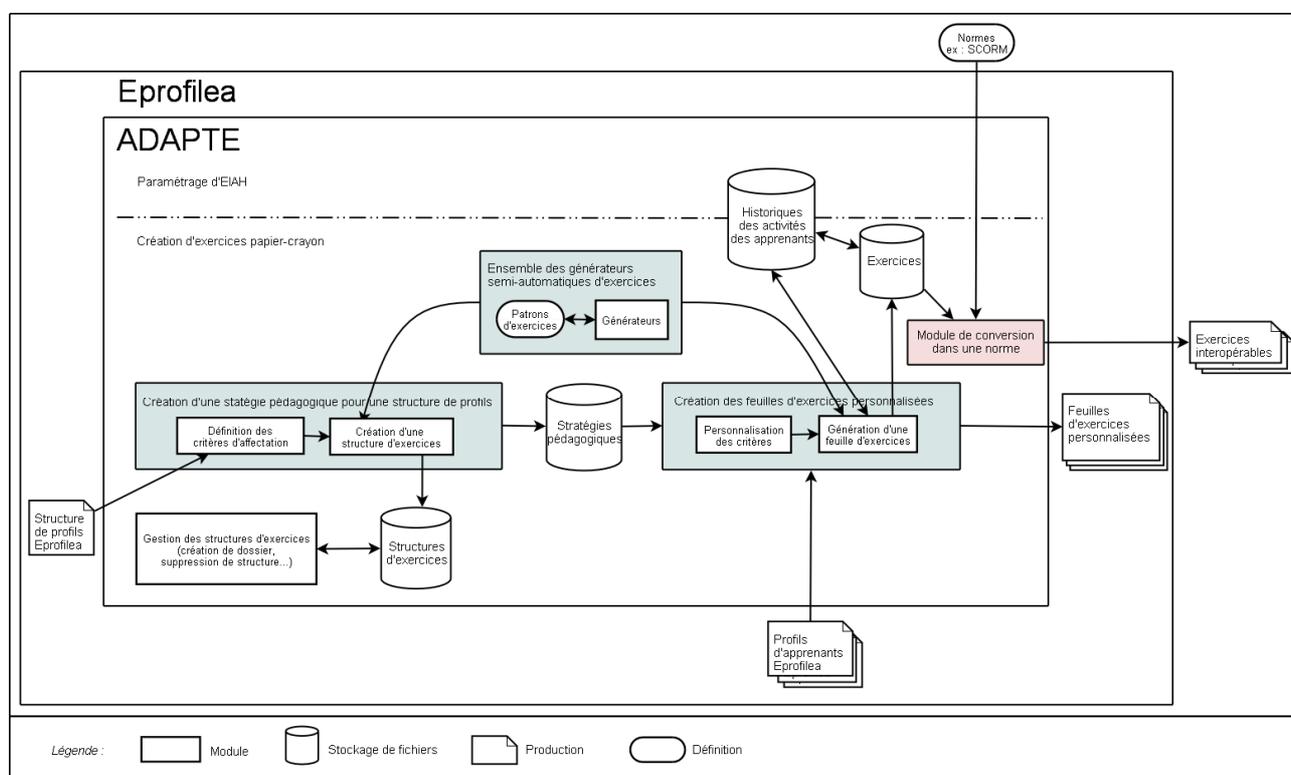


Figure 13 : Architecture d'ADAPTE

Rappelons tout d'abord le principe d'ADAPTE : proposer aux apprenants des activités adaptées aux compétences et connaissances mises en évidence par leurs profils. Ces activités peuvent être des exercices papier-crayon proposées par le système ou des activités informatisées gérées par un autre EIAH. Notre stage ne concernait que la partie proposant des activités papier-crayon.

Pour créer des activités adaptées et plus précisément des feuilles d'exercices papier-crayon, l'enseignant va devoir effectuer deux étapes.

Dans la première partie du logiciel, la **création d'une stratégie pédagogique pour une structure de profils** consiste à demander à l'enseignant de choisir la structure de profils EPROFILEA qui décrit les profils d'élèves ou de classe pour lesquels il veut définir des feuilles d'exercices. Grâce à cette structure de profils sous forme de briques, l'enseignant va définir des critères d'affectation comme indiqué dans la section 3.3.2. de ce rapport. À chaque critère, il va faire correspondre une structure d'exercices. Cette structure va être soit une structure existante que l'enseignant va sélectionner parmi celles de la base de données, soit une structure que l'enseignant va créer. Pour créer cette structure, l'enseignant va choisir un type d'exercices correspondant à l'un des neuf

patrons d'exercices définis dans la section 3.1.2. Selon le type d'exercices choisi, l'interface du générateur approprié va lui être proposée. À partir de cette interface, l'enseignant va définir les contraintes de création de l'exercice. Cette nouvelle structure d'exercices va être stockée dans la base de données correspondante. Une fois l'ensemble des critères définis, l'enseignant pourra sauvegarder sa stratégie pédagogique pour la réutiliser ou la modifier ultérieurement.

Dans la seconde partie du logiciel, la **création des feuilles d'exercices personnalisées** se fera à partir de cette stratégie pédagogique. Pour cela, on demandera à l'enseignant s'il veut définir des critères généraux ou affiner certains critères pour certains apprenants. Ensuite, l'enseignant choisira le ou les profils, d'élèves ou de classe, pour lesquels il veut créer une feuille d'exercices. C'est là que vont intervenir les générateurs proposés dans la section 3.2.3. En fonction de la mise en correspondance entre les critères de la stratégie pédagogique et les profils, ADAPTE va générer les feuilles personnalisées. Elles seront ensuite présentées à l'enseignant qui pourra les modifier, les supprimer ou les valider. Les feuilles validées seront sauvegardées sous forme de pages HTML pour une meilleure portabilité. Chaque feuille d'exercices va être constituée d'un ensemble d'exercices qui pourront, grâce à un module de conversion vers une norme ou un standard donné, être convertis pour être réutilisés par d'autres personnes ou logiciels. De plus, toute feuille créée sera mémorisée dans un historique. Cet historique comportera les données de la feuille d'exercices, mais aussi à quel élève elle est destinée, à quelle date elle a été créée et selon quelle stratégie pédagogique. Cet historique servira, en outre, à vérifier que l'on ne donne pas deux fois le même exercice à un élève, et pourra servir de feuille de route aux enseignants.

## 4.2. Le prototype réalisé

Pour pouvoir valider l'architecture proposée, nous avons mis en place une partie du logiciel décrit et l'avons présenté à deux enseignants. Ce prototype n'est pas développé entièrement mais les parties complexes ont été traitées. Il permet une utilisation complète par un enseignant, de la définition de stratégies pédagogiques jusqu'à l'enregistrement des feuilles d'exercices proposées. Par contre, tous les générateurs n'ont pas été mis en place, seuls ceux qui nous ont parus les plus intéressants ont été implémentés. Le développement du logiciel s'est fait en DELPHI pour rester cohérent avec les autres logiciels de l'environnement EPROFILEA déjà développés. Le logiciel se décompose en trois parties : la définition des stratégies pédagogiques, les différents générateurs et la génération des feuilles d'exercices personnalisées. Nous allons revenir sur chaque partie en précisant les points mis en place et ceux qui devraient l'être pour que le logiciel soit complet. Nous terminerons en parlant du point de vue des enseignants avec lesquels nous avons travaillé sur le prototype qui leur a été présenté.

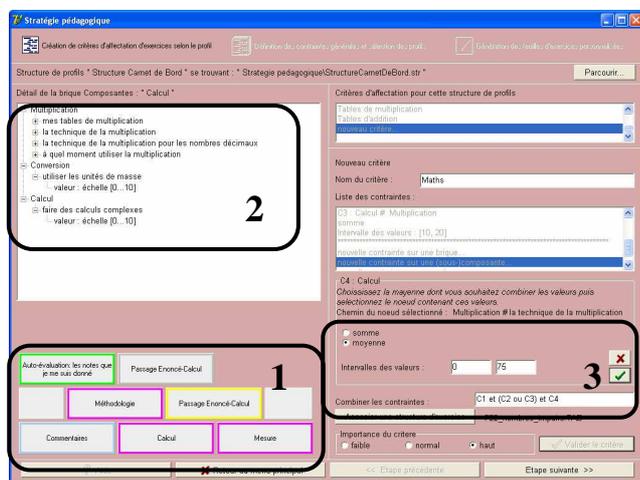


Figure 14 : Définition des critères d'affectation dans le prototype d'ADAPTE

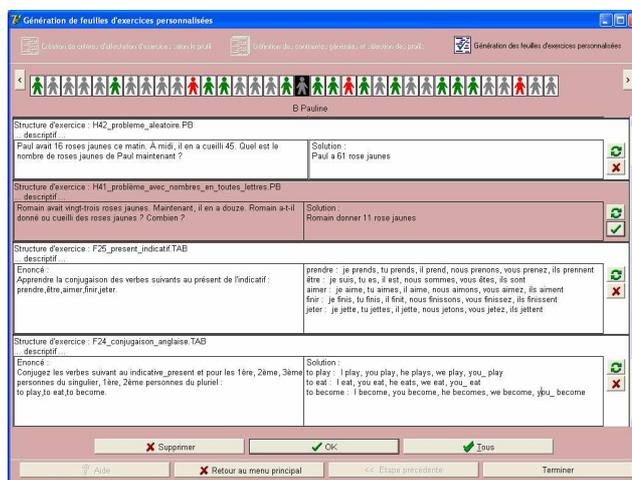


Figure 15 : Feuilles d'exercices proposées par le prototype d'ADAPTE

### 4.2.1. La définition des stratégies pédagogiques

Pour permettre à l'enseignant de définir les critères d'affectation, plusieurs choses devaient être mises en place. Tout d'abord, nous avons permis à l'enseignant de choisir une structure de profils et de l'afficher de manière compréhensible. Nous avons pour cela repris le principe du mur de briques mis en place dans BATISSEUR (cf. Figure 1). Le mur ainsi présenté est constitué des différentes briques composant les profils d'apprenants. (cf. Bulle 1 de la Figure 14). Ensuite, nous avons donné la possibilité à l'enseignant de sélectionner dans chaque brique une ou des valeurs. Les valeurs de chaque brique étant affichées sous forme d'arbre, l'enseignant n'a plus qu'à choisir un nœud dans l'arbre affiché (cf. Bulle 2 de la Figure 14). Une fois la valeur choisie, ou l'ensemble des valeurs choisies, l'enseignant définit l'intervalle pour lequel il souhaite affecter un exercice (cf. Bulle 3 de la

Figure 14). Nous avons pour cela pris en compte les échelles de valeurs définies dans BATISSEUR et mis en place la possibilité de faire des opérations (somme ou moyenne) sur des valeurs de différentes échelles. De même, pour que l'enseignant puisse définir plusieurs contraintes pour un même critère, nous lui avons laissé la possibilité de les lier grâce aux opérateurs ET et OU. Nous avons donc mis en place une analyse syntaxique et sémantique des expressions saisies par l'enseignant. Les critères ainsi définis par l'enseignant lient des valeurs de la structure de profils à des structures d'exercices. Nous parlerons de ces structures d'exercices dans le point suivant en même tant que des générateurs. Enfin, l'enseignant pourra charger des stratégies pédagogiques existantes et pourra les modifier totalement ou en partie. Les stratégies pédagogiques ainsi créées sont stockées au format XML.

#### **4.2.2. Les générateurs**

Notre étude présentée dans la section 3.2.3. de ce rapport a montré la nécessité de créer neuf générateurs d'exercices pour couvrir une grande diversité d'exercices. Chaque générateur est basé sur un patron de structure d'exercices (cf. Figure 3 à 11). Ces patrons, à l'exception du générateur des expressions algébriques, ont tous été mis en place sous forme de DTD. Nous avons ensuite défini pour chaque générateur une interface permettant à l'enseignant de spécifier simplement les contraintes de génération d'exercices. De même, ces interfaces ont toutes été mises en place, excepté pour le générateur des expressions algébriques.

Ce générateur d'expressions algébriques n'a pas été mis en œuvre puisque nous pensons qu'il serait intéressant d'utiliser la carte des expressions définies dans l'environnement APLUSIX [BOUHINEAU et al. 05] et que refaire un générateur existant n'est pas une priorité pour un stage de recherche.

Pour ce qui est des huit autres générateurs, la durée de ce stage ne nous permettait pas de tous les développer. Nous avons donc implémenté ceux qui nous paraissaient les plus intéressants. De plus, quelques générateurs suffisent à faire fonctionner le prototype d'ADAPTE et permettre ainsi d'en expliquer le principe aux enseignants.

Le premier générateur à avoir été mis en place est celui utilisant GENAMBRE. Il nous a paru intéressant de montrer que l'intégration d'un générateur déjà existant était faisable. Nous avons pour cela proposé une interface de définition de contraintes plus générale que celle proposée dans GENAMBRE, spécifique aux problèmes additifs. Cela était essentiel pour que nous puissions proposer la même interface quel que soit le domaine pour lequel il faut générer un problème. Nous nous sommes ensuite servis des exécutables PROLOG de GENAMBRE et des bases de connaissances déjà mises en place.

Ensuite, il nous a paru intéressant de mettre en place le générateur « Tables » puisque ce générateur semi-automatique permet de créer des exercices complètement aléatoires en ne fixant aucune contrainte mais également des exercices très précis en précisant au maximum toutes les contraintes. Il permet de plus de créer des exercices dans plusieurs domaines : des tables de conjugaison ou des tables de multiplication ou d'addition.

Enfin, nous avons mis en place le générateur « Travail sur texte ». Il nous paraissait intéressant dans la mesure où il demandait la mise en place de sélection automatique de textes en fonction de contraintes définies par l'enseignant.

La mise en place de ces générateurs nous a permis de valider leurs architectures (cf. Figure 10, 8 et 4) et d'avoir un prototype d'ADAPTE plus opérationnel.

#### **4.2.3. La génération des feuilles d'exercices**

Cette génération de feuilles d'exercices personnalisées se décompose en trois parties : la définition de contraintes générales et l'affinage des critères pour certains profils, la génération même et la validation ou non par l'enseignant des feuilles proposées.

Dans la première partie concernant la définition des contraintes générales, l'enseignant peut choisir de borner le nombre d'exercices ou la durée de la feuille d'exercices. Il peut de même demander à ce que tous les élèves aient des exercices différents. Pour ce qui a été mis en place, la feuille générée tient compte du nombre d'exercices maximum ou minimum et de l'affinage des contraintes pour certains apprenants. Il reste à mettre en place la prise en compte des intervalles de durée de chaque feuille. Cela se fera grâce aux estimations de temps fournies par les enseignants pour chaque structure d'exercices. Cette estimation leur est demandée lors de la création des structures d'exercices. Nous n'avons pas mis en place ce deuxième type de contraintes par manque de temps.

Dans la seconde partie concernant la génération même des feuilles d'exercices personnalisées, la mise en correspondance entre les critères d'affectation d'une stratégie pédagogique et les profils fournis par l'enseignant a été mise en place. On propose donc à l'enseignant la feuille d'exercices comprenant tous les exercices correspondant aux compétences de l'élève et l'on grise ceux que le logiciel a supprimés en fonction des contraintes générales. Tous les générateurs n'étant pas mis en place, on propose l'exercice de référence dans le cas où l'on n'a pas de générateur.

Dans la dernière partie concernant la validation de la feuille d'exercices, l'enseignant peut modifier chaque feuille une à une en demandant de régénérer un exercice de même type si celui qui lui a été proposé ne lui convient pas, en supprimant un exercice ou en ajoutant un exercice que le logiciel avait supprimé (cf. Figure 15). On laisse

donc à l'enseignant une totale liberté d'action sur les feuilles proposées. De même, il peut supprimer une feuille pour un élève donné. Enfin, s'il le souhaite, il peut demander de valider toutes les feuilles en une fois. Dans tous les cas, le logiciel fournit des feuilles d'exercices au format HTML. Il reste à mettre en place la fonction du logiciel qui permettra d'imprimer ces feuilles. Cette fonction n'a pas été implantée par manque de temps et car elle ne nous semblait pas prioritaire, puisque les feuilles sont visualisables et imprimables en dehors d'ADAPTE dans n'importe quel navigateur.

### 4.3. Évaluation du prototype

Le projet PERLEA est l'objet d'une collaboration régulière avec des enseignants. Cette collaboration est soutenue par l'IRNP dans le cadre de l'ERTÉ e-praxis.

Notre stage faisant partie du projet PERLEA, nous avons pu rencontrer deux enseignants-formateurs pour discuter avec eux de la façon dont ils affectaient des activités aux apprenants, de leur manière de créer des exercices... Ces discussions ont eu lieu lors de trois réunions :

- 1<sup>ère</sup> réunion : proposition des critères d'affectation d'un exercice à un apprenant et discussion autour de la typologie d'exercices définie ;
- 2<sup>nd</sup>e réunion : présentation d'un prototype avancé et discussion sur les interfaces de définition de contraintes des différents générateurs (avec un seul enseignant) ;
- 3<sup>ème</sup> réunion : définition du profil de classe ;
- 4<sup>ème</sup> réunion : mise à l'essai du prototype finalisé.

Lors de la seconde réunion, nous avons donc fait une démonstration du prototype-avancé. Nous avons choisi de faire une démonstration et non une expérimentation car le prototype n'étant pas terminé, il manquait des fonctionnalités et de nombreuses explications. Toutefois, l'enseignant rencontré a bien compris le principe du logiciel. Suite à cette démonstration, des remarques ont été faites, comme le fait que les noms d'enregistrement des exercices n'étaient pas assez explicites. Nous avons donc corrigé ce point en nommant les exercices grâce au nom que l'enseignant avait donné à la structure d'exercices qui permet de les générer.

Dans l'ensemble, l'enseignant rencontré a trouvé le logiciel présenté bien construit et assez simple d'utilisation. Mais il faut prendre en compte le fait qu'il connaissait déjà d'autres générateurs d'exercices et qu'il se sert souvent d'ordinateurs dans le cadre de son travail. Cela sera peut être différent avec des enseignants maîtrisant moins bien l'informatique. Pour ce qui est de la diversité des exercices pouvant être générés par ADAPTE, il l'a trouvée suffisante : tous les exercices que lui ou ses collègues ont l'habitude d'utiliser en primaire seront « générables » une fois que les neuf générateurs seront mis en place. Pour conclure, il pense que le logiciel est tout à fait utilisable par des enseignants. Ainsi, une première étape dans le développement du logiciel a été franchie puisque ce prototype avancé semblait convenir aux besoins des enseignants.

Nous avons donc poursuivi le développement du prototype d'ADAPTE et une dernière rencontre doit avoir lieu à la fin de ce stage pour permettre aux enseignants de donner leurs avis sur la version finale du prototype. Cette quatrième réunion se déroulera sous forme de mise à l'essai : nous laisseront les enseignants présents se servir du logiciel. Cette expérimentation n'aura pas d'influence directe sur notre stage, mais les remarques faites par les enseignants au sujet du prototype serviront aux personnes qui continueront à travailler sur ADAPTE. Cette rencontre permettra également de valider ou non, le prototype final.

## 5. CONCLUSION ET PERSPECTIVES

---

L'objectif de ce stage de recherche était de concevoir un prototype d'ADAPTE gérant la proposition d'activités papier-crayon adaptées à chaque profil d'apprenants. Pour cela, nous devions tout d'abord générer les exercices qui rendront possible la personnalisation de l'apprentissage, mais également trouver un équilibre entre généralité de l'environnement EPROFILEA et spécificités disciplinaires ou liées à l'âge, au niveau scolaire ou universitaire, et identifier jusqu'où il est possible d'une part, et souhaitable d'autre part d'automatiser la création d'activités. À l'issue de ce stage, nous pouvons faire le bilan du travail accompli.

Tout d'abord, pour pouvoir concevoir le prototype d'ADAPTE, nous avons proposé une typologie des exercices pouvant être donnés à un apprenant tout au long de sa scolarité. Cette typologie comprend quinze catégories d'exercices. Nous l'avons définie pour qu'elle soit générique, mais les seules évaluations que nous en avons eues ont été faites par des enseignants de primaire. Or EPROFILEA est destiné à des enseignants de tous niveaux scolaires et universitaires. Nous sommes donc bien conscient que seule une évaluation de la part d'experts en sciences de l'éducation permettrait de valider cette typologie, autant au niveau de sa généralité que de son exhaustivité.

Nous avons ensuite défini l'architecture complète d'ADAPTE permettant aux enseignants de définir leurs stratégies pédagogiques liant les compétences ou difficultés mises en évidence par le profil de l'apprenant, à des types d'exercices. Pour cela, nous avons défini de quelle manière l'enseignant allait spécifier ses stratégies pédagogiques en lui permettant de créer des critères d'affectation ainsi que des contraintes générales pour borner la taille et la difficulté des feuilles générées. Afin de faire la mise en correspondance entre les critères des enseignants et les valeurs contenues dans les profils d'apprenants, nous avons proposé un ensemble de règles. Nous avons également proposé les architectures des neuf générateurs d'exercices que nous avons jugés nécessaires pour créer les exercices classifiés dans notre typologie. Ces architectures sont génériques. Elles pourront donc être utilisées pour développer des générateurs d'exercices, et ce quel que soit leur contexte d'utilisation. Nous restons néanmoins conscient que ces architectures génériques facilitent la mise en place des générateurs dans de nouveaux domaines d'applications mais n'évitent pas le travail important d'instanciation des bases de connaissances propres au domaine. Nous avons pu tester la généricité de ces générateur grâce à l'implémentation de certains dans le cas particulier d'ADAPTE, et donc dans des domaines variés.

Suite à cette étape de conception, nous avons développé un prototype d'ADAPTE permettant à l'enseignant d'appliquer entièrement cette démarche. Dans ce prototype, l'enseignant peut définir ses stratégies pédagogiques en indiquant ses critères d'affectation d'un exercice à une ou plusieurs compétences précises d'un profil d'apprenant. L'enseignant fournit ensuite un certain nombre de profils d'apprenants et le système produit autant de feuilles d'exercices personnalisées. Le prototype réalisé ne comprend néanmoins que trois générateurs développés complètement. Par manque de temps, nous avons choisi de mettre en place les générateurs qui nous semblaient les plus pertinents : un générateur incluant un générateur existant, un générateur permettant de créer trois types d'exercices différents et un générateur proposant des exercices faisant appel à des textes sélectionnés plus ou moins automatiquement.

La conception d'ADAPTE s'étant faite en partenariat avec des enseignants et le prototype étant dès à présent utilisable, nous l'avons présenté à ces enseignants. Leurs retours semblent valider le prototype et l'architecture proposés.

Au regard du travail présenté dans ce rapport, nous pouvons donc dire que l'objectif de ce stage de Master 2 Recherche est atteint : l'étape de conception d'ADAPTE est terminée et un prototype a été mis en place. Mais ce travail nous laisse entrevoir des perspectives, autant du point de vue recherche que du point de vue opérationnalisation du prototype.

Pour ce qui est des perspectives techniques, il faudrait compléter le prototype pour permettre une utilisation plus aisée en ajoutant une aide, mais également en développant les quelques fonctionnalités non implémentées comme la proposition d'une aide à l'impression des feuilles d'exercices proposées ou la prise en compte de la durée des feuilles d'exercices. Il faudrait également développer les générateurs dont les architectures ont été proposées mais non mises en œuvre par manque de temps.

Pour ce qui est des perspectives de recherche, il serait intéressant de concevoir et de mettre en place la seconde partie d'ADAPTE qui propose des activités informatisées gérées par un autre EIAH, en paramétrant un EIAH externe pour proposer une session adaptée à l'apprenant. Ce travail ne faisait pas partie de ce stage, mais il nous semble important à mettre en place pour terminer le module ADAPTE et ainsi donner les moyens à l'enseignant de réintégrer dans sa pratique de classe son travail sur les profils d'apprenant avec l'environnement EPROFILEA à travers des activités adaptées aux profils plus variées encore car complémentaires de celles déjà proposées par le prototype développé pendant ce stage.

Enfin, une fois le prototype terminé, avec ou sans la seconde partie d'ADAPTE, il devra être soumis à une évaluation plus rigoureuse, capitale en EIAH. Cela se fera en mettant en place des expérimentations réelles avec des enseignants sans rapport avec la conception du module, expérimentations faisant appel à tous les modules d'EPROFILEA concernés et allant de la définition d'une structure de profils par l'enseignant à l'utilisation effective des activités personnalisées par les apprenants.

Nous terminerons en soulignant le fait que le travail présenté dans ce rapport s'intègre parfaitement aux thématiques actuelles de recherche dans le domaine des EIAH. Pour preuve, le cluster de recherche ISLE de la Région Rhône-Alpes dont le but est la personnalisation des EIAH. Actuellement, peu de travaux proposent cette personnalisation de l'apprentissage. Et comme nous l'avons indiqué au début de ce rapport, les générateurs semi-automatiques d'exercices sont peu nombreux malgré le fait que ce soit ceux qui s'adaptent le plus aux pratiques des enseignants. Nous avons, pour notre part, proposé neuf architectures de générateurs semi-automatiques d'exercice pouvant être mis en place dans des EIAH et permettant ainsi la personnalisation de l'enseignement recherchée.

## BIBLIOGRAPHIE

---

- [**BOUHINEAU et al. 05**] Bouhineau D., Bronner A., Chaachoua H., Nicaud J-F., « Patrons d'exercices pour APLUSIX. Une étape du développement de l'EIAH occasion d'un travail entre didacticiens et informaticiens », Conférence EIAH'2005, pp. 377-382, Montpellier, 25-27 mai 2005.
- [**BRUILLARD 97**] Bruillard E., « Les machines à enseigner », éditions Hermès, Paris, 1997.
- [**BURTON 82**] Burton R.R., « Diagnosing bugs in a simple procedural skill », Intelligent Tutoring Systems, Academic Press, pp. 157-183, London, 1982.
- [**COGNE et al. 98**] Cogne A., David J.P., Lacombe C., « Production d'exercices hypermédias et mise en œuvre pédagogique », 8èmes Journées Informatique et Pédagogie des Sciences Physiques, Montpellier, 1998.
- [**CROZAT 04**] Crozat S., « Les systèmes de production et de gestion des contenus pédagogiques numériques : vers une nouvelle approche », [http://ressources.algora.org/frontblocks/news/papers.asp?id\\_papers=1264#1](http://ressources.algora.org/frontblocks/news/papers.asp?id_papers=1264#1), mars 2004. (dernière fois visité : 13 juin 2006)
- [**DE LA PASSARDIERE et al. 03**] De La Passardière B., Grandbastien M., « Présentation de LOM v1.0, standard IEEE, Revue "Sciences et Techniques Educatives", Hors série 2003 "Ressources numériques, XML et éducation", éditions Hermès, pp. 211-218, avril 2003.
- [**DUCCLOSSON et al. 05**] Duclosson N., Jean-Daubias S., Riot S., « AMBRE-enseignant : un module partenaire de l'enseignant pour créer des problèmes », Conférence EIAH'2005, pp. 353-358, Montpellier, 25-27 mai 2005.
- [**GIROIRE 89**] Giroire H., « Un système à base de connaissances pour la génération d'exercices dans les domaines liés au monde réel », Thèse de doctorat de l'Université Paris VI, 1989.
- [**GUIDON 01**] Guidon J., « Notes sur les normes et standards pour la formation », sur EducNet, <http://www.educnet.education.fr/superieur/normes.htm>, juin 2001. (dernière fois visité : 13 juin 2006)
- [**GUIN 97**] Guin N., « Reformuler et classer un problème pour le résoudre. L'architecture SYRCLAD et son application à quatre domaines », Thèse de doctorat de l'université Paris VI, 1997.
- [**GUIN-DUCLOSSON et al. 02**] Guin-Duclosson N., Jean-Daubias S. et Nogry S., « The AMBRE ILE: How to Use Case-Based Reasoning to Teach Methods », Intelligent Tutoring Systems, ITS'2002 proceedings, Lecture Notes in Computer Science vol. 2363, pp. 782-791, 2002.
- [**JEAN-DAUBIAS 02**] Jean-Daubias S., « Un système d'assistance au diagnostic de compétences en algèbre élémentaire », Revue Sciences et techniques éducatives, Hermès, Vol.9 - n°1-2, pp. 171-200, 2002.
- [**JEAN-DAUBIAS 04**] Jean-Daubias S., « De l'intégration de chercheurs, d'experts, d'enseignants et d'apprenants à la conception d'EIAH », TICE 2004, pp. 290-297, Compiègne, novembre 2004.
- [**JEAN-DAUBIAS et al. 05**] Jean-Daubias S., Eyssautier-Bavay C., « An environment helping teachers to track students' competencies », Actes du workshop LEMORE, AIED 2005, pp. 19-23, Pays-Bas, juillet 2005.
- [**KOPER 01**] Koper R., « Modeling units of study from a pedagogical perspective. The pedagogical meta-model behind EML », Open University of the Netherlands, <http://eml.ou.nl/introduction/articles.htm>, juin 2001. (dernière fois visité : 13 juin 2006)
- [**LEJEUNE et al. 04**] Lejeune A., Pernin J.P., « A taxonomy for scenario-based engineering », Cognition and Exploratory Learning in Digital Age (CELDA 2004) Proceedings, pp.249-256, Lisboa, Portugal, decembre 2004.
- [**NICAUD 93**] Nicaud J.F., « Le projet Aplusix », Revue de l'EPI (Enseignement Public et Informatique) n°72, décembre 1993.
- [**PECEGO 98**] Pecego G., « SYGEP, un Système de Génération d'Énoncés de Problèmes dans des domaines variés », Thèse de doctorat de l'Université Paris VI, 12 juin 1998.
- [**PERNIN 03**] Pernin J.P., « Objets pédagogiques : unités d'apprentissage, activités ou ressources ? », Revue "Sciences et Techniques Educatives", Hors série 2003 "Ressources numériques, XML et éducation", éditions Hermès, pp. 179-210, avril 2003.
- [**PERNIN 06**] Pernin J.P., « Normes et standards pour la conception, la production et l'exploitation des EIAH », in Cognition et traitement de l'information, Environnements Informatiques pour l'Apprentissage Humain, éditions Hermès, pp. 201-222, mars 2006.
- [**VINCENT et al. 05**] Vincent C., Delozanne E., Grugeon B., Gélis J.M., Rogalski J., Coulange L., « Des erreurs aux stéréotypes : Des modèles cognitifs de différents niveaux dans le projet Pépite », Conférence EIAH'2005, pp. 297-308, Montpellier, 25-27 mai 2005.