

Exploitation d'assistances épiphytes en contexte éducatif

Le Vinh THAI^{1,2}

Blandine GINON^{1,3}

1^{ère} année de doctorat

3^{ème} année de doctorat

¹ Université de Lyon, CNRS,

² Université Lyon 1, LIRIS, UMR5205, F-69622, France

³ INSA-Lyon, LIRIS, UMR5205, F-69621, France

le-vinh.thai@liris.cnrs.fr

blandise.ginon@liris.cnrs.fr

Résumé :

L'adjonction d'un système d'assistance épiphyte dans une application utilisée en contexte éducatif est une solution pour faciliter l'acquisition de connaissances et pour pallier les difficultés de prise en main et d'utilisation qui peuvent compromettre l'atteinte de l'objectif pédagogique de l'enseignant. Un tel système d'assistance doit être capable de fournir une assistance technique, mais aussi une assistance pédagogique. Nous présentons dans cet article de quelle manière le système SEPIA peut être utilisé afin de permettre la mise en place de systèmes d'assistance épiphytes dans une application utilisée en contexte éducatif. Nous discuterons les forces et les limites de SEPIA avant d'identifier les questions de recherches identifiées pour enrichir et adapter pleinement ce système au contexte éducatif.

1. Introduction

Dans le contexte éducatif, de plus en plus d'applications informatiques sont utilisées, qu'il s'agisse de logiciels dédiés spécifiquement à l'apprentissage, les EIAH (Environnements Informatiques pour l'Apprentissage Humain) ou de logiciels « classiques » utilisés comme support d'une activité pédagogique. L'acquisition des connaissances pédagogiques en jeu peut néanmoins être compromise par des difficultés de prise en main et d'utilisation de l'application. L'ajout d'un système d'assistance adapté à la fois à l'application et aux objectifs pédagogiques de l'activité est une solution pour pallier ces difficultés et éviter l'abandon de l'activité et la démobilité de l'utilisateur (Gapenne 2006).

Le travail présenté dans cet article se situe dans le contexte du projet AGATE que nous présentons dans la section suivante et qui a donné lieu au développement du système SEPIA (cf. section 2.3) qui permet l'ajout de systèmes d'assistance dans des applications existantes. Cet article a pour but de présenter un travail de thèse qui débute et qui concerne la proposition de modèles et l'enrichissement de SEPIA afin de permettre la mise en place de systèmes d'assistance spécifiquement adaptés au contexte éducatif.

Dans cet article, nous présentons tout d'abord le projet AGATE, ainsi que les propositions théoriques qui lui sont associées et leur mise en œuvre dans le système SEPIA. Nous présentons ensuite les acteurs et les applications concernés par l'assistance dans le contexte éducatif, avant d'identifier les besoins auxquels une telle assistance doit pouvoir répondre. Enfin, nous montrons de quelle manière SEPIA peut être utilisé en EIAH, en présentant une expérimentation suivie d'une discussion sur les forces et les limites de SEPIA dans ce contexte. Nous concluons en exposant les questions de recherches qui seront abordées dans la thèse qui commence.

2. Contexte : le projet AGATE

Le projet AGATE (Approche Générique d'Assistance aux Tâches complexEs) vise à proposer des modèles génériques et des outils unifiés pour permettre la mise en place de systèmes d'assistance dans des applications existantes, que nous appelons *applications-cibles*. Ce projet adopte une démarche entièrement générique et épiphyte. Une application épiphyte est une application externe capable de réaliser des actions dans une application-cible sans perturber son fonctionnement (Paquette et al. 1996). Le fonctionnement d'un système d'assistance épiphyte ne perturbe donc pas celui de l'application-cible. Les modèles et outils proposés dans la cadre du projet AGATE ne sont spécifiques ni à une application ni à un domaine, mais peuvent au contraire être exploités pour ajouter une assistance à des applications les plus variées, sans que celles-ci aient été spécifiquement conçues pour permettre l'intégration d'une assistance.

Dans cette section, nous présentons le processus d'adjonction de systèmes d'assistance épiphytes, ainsi que le langage de définition de systèmes d'assistance épiphytes qui ont été proposés dans le cadre du projet AGATE. Nous présentons ensuite le système SEPIA qui met en œuvre ces propositions théoriques.

2.1. Processus d'adjonction d'un système d'assistance épiphyte

La Figure 1 présente le processus d'adjonction d'un système d'assistance épiphyte proposé dans le cadre du

projet AGATE. Ce processus comporte deux phases : la spécification de l'assistance, puis l'exécution de cette assistance de manière épiphyte. La première phase est effectuée par un expert de l'application-cible, appelé par la suite *concepteur de l'assistance*. Cette phase préparatoire permet au concepteur de spécifier l'assistance qu'il souhaite pour une application-cible. La seconde phase concerne les utilisateurs finaux de l'application-cible. Elle consiste en l'exécution de l'assistance souhaitée par le concepteur. Cette phase a lieu à chaque utilisation de l'application-cible par un utilisateur, et se compose de trois processus. La surveillance de l'application-cible exploite un ensemble de détecteurs épiphytes (Ginon et al. 2013a) et permet d'observer en continu et de tracer toutes les interactions entre l'utilisateur et l'interface de l'application-cible. Ces observations sont exploitées par le processus d'identification d'un besoin d'assistance qui s'exécute en parallèle et déclenche le processus d'élaboration d'une réponse au besoin identifié en fonction de l'assistance spécifiée par le concepteur. La réponse se fait sous la forme d'une ou plusieurs actions d'assistance, réalisées dans l'application-cible par un assistant épiphyte.

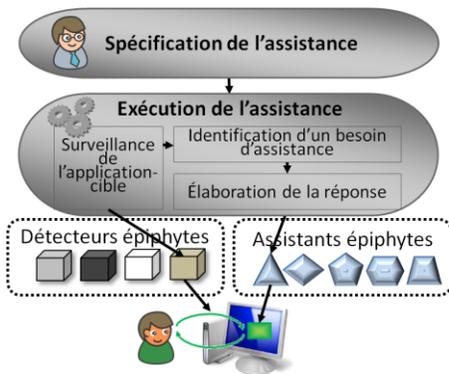


Figure 1 : Processus d'adjonction d'un système d'assistance épiphyte

2.2. Le langage aLDEAS

aLDEAS (a Language to Define Epi-Assistance Systems) est un langage qui permet la définition de systèmes d'assistance pour des applications existantes sous la forme d'un ensemble de règles (Ginon et al. 2014). Ce langage est au cœur du processus d'adjonction de systèmes d'assistance épiphytes : il permet au concepteur de modéliser l'assistance souhaitée lors de la phase de spécification de l'assistance, puis est exploité lors de la phase d'exécution de l'assistance au travers de sa mise en œuvre. aLDEAS est constitué de différents types de composants qui peuvent être combinés pour créer des règles et actions d'assistance, afin de répondre à des besoins d'assistance variés. Les principaux éléments d'aLDEAS sont les attentes d'événements sur l'application-cible, les consultations (de l'utilisateur, de l'historique de l'assistance, du profil de l'utilisateur ou des traces de l'utilisateur) et les actions d'assistance. Une attente d'événement permet d'attendre qu'un événement

donné ait lieu pour déclencher une action du système d'assistance. Les événements attendus peuvent notamment concerner des actions de l'utilisateur, comme le clic sur un bouton donné, l'absence d'action pendant une durée donnée ou la réalisation d'une tâche de « haut-niveau » telle que la correction des yeux rouges sur une photo, combinant plusieurs actions de « bas-niveau ». Les consultations permettent de personnaliser et contextualiser l'assistance en conditionnant les actions du système d'assistance à des informations relatives aux actions passées de l'utilisateur, à son profil, à ses choix et à l'état de l'application-cible. Enfin, aLDEAS propose un large choix d'actions d'assistance : messages, mises en valeur ou actions automatisées sur un composant de l'interface de l'application-cible, propositions de ressources externes (comme des vidéos de démonstration, forum, supports de cours), etc.

Le langage aLDEAS est complété par plusieurs patrons qui facilitent la définition d'actions d'assistance (Ginon et al. 2014).

2.3. Le système SEPIA

Le système SEPIA met en œuvre ces propositions théoriques à travers deux principaux outils opérationnels : l'éditeur d'assistance et le moteur générique d'assistance.

L'éditeur d'assistance opérationnalise la phase de spécification de l'assistance du processus d'adjonction d'un système d'assistance épiphyte. Il fournit une interface à destination des concepteurs d'assistance, afin de leur permettre de définir l'assistance qu'ils souhaitent pour une application-cible, sous la forme d'un ensemble de règles d'assistance respectant le langage aLDEAS.

Le moteur générique d'assistance opérationnalise la phase d'exécution du processus d'adjonction d'un système d'assistance épiphyte et permet de fournir de l'assistance à chaque utilisateur d'une application-cible en fonction de l'assistance spécifiée en aLDEAS par le concepteur.

Le moteur est complété par un ensemble de détecteurs qui permettent la surveillance épiphyte d'une application-cible donnée. Actuellement, trois détecteurs ont été développés : un détecteur pour les applications Windows natives, un détecteur pour les applications Java et un détecteur pour les applications Web. Enfin, le moteur exploite un ensemble d'assistants épiphytes qui peuvent agir sur une application-cible donnée à la demande du moteur. Les assistants épiphytes développés permettent de réaliser les actions d'assistance proposées par aLDEAS, en particulier afficher ou lire des messages, mettre en valeur un composant, faire appel à des agents animés capables de s'exprimer oralement, textuellement, par gestes et animations (par exemple le compagnon Merlin peut applaudir et dire « Bravo, tu as réussi »). Il existe également des assistants épiphytes capables de réaliser des actions sur l'interface de l'application-cible à la place de l'utilisateur, par exemple pour cliquer

automatiquement sur un bouton, saisir un texte ou sélectionner un item dans une liste déroulante.

3. Assistance dans le contexte éducatif

Dans la cadre du projet AGATE, nous souhaitons utiliser le système SEPIA pour mettre en place des systèmes d'assistance épiphytes dans le contexte éducatif, tout en conservant une approche générique. Plus particulièrement, nous souhaitons permettre à un concepteur pédagogique de définir un système d'assistance épiphyte pour une application-cible utilisée par un ensemble d'apprenants.

3.1. Quels sont les acteurs concernés?

Dans le contexte éducatif, le concepteur de l'assistance peut être un *concepteur pédagogique* qui souhaite ajouter un système d'assistance à une application qui en est dépourvue ou qui possède une assistance incomplète ou inadaptée aux souhaits du concepteur. Le terme de *concepteur pédagogique* peut désigner un enseignant ou une équipe pédagogique, éventuellement assistés par un informaticien ou un expert de l'application-cible.

Les utilisateurs finaux concernés par l'assistance dans le contexte éducatif sont les apprenants, c'est-à-dire les personnes en situation d'apprentissage (élève, étudiant, adulte en formation...).

Un autre usage de SEPIA dans un contexte éducatif concerne l'adjonction d'un système d'assistance à un outil à destination de l'enseignant, mais il sort du cadre de cet article.

3.2. Quelles sont les applications concernées?

Deux catégories principales d'applications sont utilisées dans le contexte éducatif : les EIAH et les logiciels « classiques » exploités avec un objectif pédagogique. Ces deux catégories peuvent être concernées par l'adjonction d'un système d'assistance épiphyte dans le contexte éducatif.

Un **EIAH** est une application informatique qui vise à favoriser l'apprentissage. Il peut être de différents types : micromondes, tuteurs intelligents, environnements de simulation, hypermédias... (Grandbastien et Labat 2006) Par ailleurs, un **logiciel « non pédagogique »**, c'est-à-dire qui n'est pas spécifiquement conçu pour l'apprentissage, peut également être utilisé dans le contexte éducatif. Tout d'abord, une activité pédagogique peut avoir pour objectif la découverte ou la maîtrise d'un logiciel classique. Par exemple, une entreprise peut organiser un stage pour former ses employés à l'utilisation des outils utilisés pour leur travail, comme un logiciel de gestion de commandes ou des outils de conception assistée par ordinateur. Certaines formations à destination du grand public ont également comme but d'apprendre à utiliser un outil, par exemple un logiciel de retouche d'images.

Ensuite, certaines activités pédagogiques, dont l'objectif n'est pas la maîtrise d'un outil, nécessitent toutefois

l'utilisation d'un logiciel « non pédagogique ». Ainsi, dans le cadre d'un cours de bureautique, un enseignant peut faire utiliser Word à ses élèves pour apprendre à présenter des lettres officielles, ou utiliser Excel pour la résolution de problèmes d'algèbre (Ritter et Koedinger 1995). Un formateur en langues étrangères peut également faire utiliser à ses stagiaires un logiciel d'enregistrement de son pour leur faire travailler la prononciation. Dans ces contextes, la non-maîtrise du logiciel « non pédagogique » utilisé comme support de l'activité, peut freiner l'acquisition des connaissances pédagogiques en jeu.

Enfin, dans certains cas, les enjeux d'une activité pédagogique peuvent porter à la fois sur des connaissances relatives à un logiciel « non pédagogique » et sur d'autres connaissances du domaine. Par exemple, dans le cadre d'un cours d'informatique, un enseignant peut imposer à ses étudiants d'utiliser un environnement de développement particulier, par exemple (NetBeans 2013) ou (Visual Studio 2014), pour réaliser des travaux pratiques dans le but d'apprendre à utiliser cet environnement tout en acquérant des connaissances en programmation.

3.3. Quels sont les besoins d'assistance?

L'identification des besoins d'assistance est une tâche clé pour la définition d'un système d'assistance. En nous appuyant sur une large étude bibliographique (Ginon et al. 2013b) et sur une étude de systèmes d'assistance existants, nous avons identifié les principaux besoins d'assistance des apprenants concernant l'utilisation de logiciels, pédagogiques ou non, utilisés dans un contexte éducatif (cf. première colonne du Tableau 1). Ces besoins d'assistance sont de deux types : techniques (T1 et T2) ou pédagogiques (P1 à 10).

Besoins d'assistance dans le contexte éducatif	SEPIA
T1. Prise en main	✓
T2. Utilisation	✓
P1. Choix de l'activité	A
P2. Apprentissage des prérequis	A
P3. Explications sur les étapes	✓
P4. Indices	✓
P5. Exemples	✓
P6. Diagnostic intermédiaire	~
P7. Explication sur les erreurs	~
P8. Automatisation de sous-tâches	~
P9. Bilan, suivi	✓
P10. Guidage pédagogique	✓

Tableau 1 : Besoins d'assistance en EIAH

Besoins d'assistance technique

Les apprenants qui utilisent une application pour la première fois peuvent être confrontés à des difficultés de prise en main de cette application (cf. T1-Tableau 1). Ils peuvent également rencontrer des difficultés pour l'utilisation courante d'une application (T2),

particulièrement lorsqu'il s'agit d'une utilisation occasionnelle ou lors de la découverte d'une fonctionnalité inconnue. Ces difficultés techniques ne concernent généralement pas directement la discipline enseignée et se retrouvent également dans un contexte autre qu'éducatif. Néanmoins, elles peuvent empêcher ou ralentir l'acquisition des connaissances en jeu, ou entraîner l'abandon de l'activité pédagogique.

La définition d'un système d'assistance pour répondre à ces besoins d'assistance technique permet d'éviter que les utilisateurs renoncent à utiliser une application. Plus particulièrement, dans le contexte éducatif, un tel système d'assistance permet d'éviter que les apprenants perdent du temps ou se découragent en raison de difficultés techniques. De plus, lors de l'utilisation d'une application en présentiel, un système d'assistance technique peut rendre les apprenants plus autonomes et ainsi permettre à l'enseignant d'alléger ses interventions d'ordre technique et de se recentrer sur les aspects pédagogiques.

Besoins d'assistance pédagogique

En plus des classiques besoins d'assistance technique, les apprenants qui utilisent une application dans un contexte éducatif peuvent être confrontés à des difficultés auxquelles une assistance pédagogique permettrait de répondre.

Le but d'une assistance pédagogique n'est pas nécessairement de permettre à un apprenant de terminer sans erreur et au plus vite une activité pédagogique. Si fournir à un apprenant la solution d'une activité pédagogique paraît très peu pertinent, certaines formes d'assistance pédagogique peuvent cependant être jugées pertinentes par l'enseignant pour faciliter l'acquisition des connaissances en jeu.

C'est le cas de l'aide au choix de l'activité pédagogique (P1), en particulier lorsque ce choix est déterminé par le concepteur pédagogique de manière personnalisée pour chaque apprenant. L'aide à l'acquisition des prérequis (P2) peut être utile pour conseiller à un apprenant de réaliser d'autres activités (exercices, consultation de cours, rappels) afin de l'aider pour une activité dont il ne maîtrise pas encore tous les prérequis. Des explications sur les étapes à suivre (P3) pour réaliser une activité peuvent être intéressantes pour guider un apprenant et lui faire acquérir une méthodologie. Par exemple, pour résoudre un calcul de type « $4 + 5 * 8 - (9 / (2 + 1))$ », il peut être pertinent de conseiller à l'apprenant de commencer par traiter les calculs entre parenthèses, puis les opérations prioritaires. Fournir à un apprenant des indices (P4) ou des exemples (P5) peut également faciliter la résolution de l'activité et par conséquent faciliter l'acquisition des connaissances en jeu tout en motivant l'apprenant. Le diagnostic intermédiaire (P6) permet de confirmer à l'apprenant que ce qu'il a fait est correct ou au contraire qu'il a commis des erreurs. Par exemple, un système d'aide peut dans un premier temps indiquer à un apprenant qu'il a fait une erreur, puis si besoin lui montrer précisément où se situe cette erreur, et

enfin lui fournir des explications sur l'erreur (P7). Dans certains cas, il peut être pertinent d'automatiser une partie de la tâche (P8) sans que cela ne compromette l'acquisition des connaissances en jeu. Par exemple, dans le cas d'un exercice dont le but est d'exploiter le théorème de Pythagore, le système d'aide pourrait automatiser les calculs de racines carrées que l'apprenant ne sait pas résoudre et qui l'empêchent de réaliser l'exercice. Un bilan ou un suivi du travail (P9) permet de montrer à l'apprenant ce qu'il a fait et ce qui lui reste à faire, afin de l'encourager. Enfin, un système d'assistance peut suivre un scénario pédagogique (P10) défini par le concepteur pédagogique afin de répondre aux différents besoins d'assistance identifiés.

En conclusion, un système d'assistance peut avoir pour but de répondre à des besoins d'assistance technique comme à des besoins d'assistance pédagogique. L'assistance fournie à un apprenant doit faciliter l'acquisition des connaissances en jeu, dans le respect de la stratégie pédagogique du concepteur. Dans la partie suivante, nous présentons de quelle manière SEPIA peut être utilisé pour mettre en place de tels systèmes d'assistance.

4. Exploitation de SEPIA dans le contexte éducatif

Le système SEPIA a été conçu afin de permettre la mise en place de systèmes d'assistance épiphytes capables de répondre aux besoins d'assistance technique des utilisateurs d'application-cibles variées. Cependant, nous montrons dans cette section que SEPIA permet également la mise en place de systèmes d'assistance respectant les souhaits d'un concepteur pédagogique et capables de répondre à des besoins d'assistance pédagogiques.

4.1. Expérimentation avec NetBeans

Afin d'évaluer les forces et les faiblesses de SEPIA pour la mise en place de systèmes d'assistance dans le contexte éducatif, nous avons réalisé une expérimentation dans le cadre d'un cours d'IHM (Interface Homme-Machine) de troisième année de licence à l'Université Lyon 1.

Lors du premier TP du cours d'IHM, les étudiants doivent réaliser une application graphique simple en langage Java à l'aide de l'environnement de développement NetBeans. Chaque année, les enseignants de cette matière constatent le niveau très hétérogène des étudiants concernant la maîtrise de la programmation Java et de l'utilisation de NetBeans. Ce premier TP est donc l'occasion de donner aux étudiants des bases solides sur lesquelles s'appuyer pour les TP suivants. En complément des explications données en cours magistral, deux vidéos de démonstrations sont mises à disposition des étudiants afin de présenter NetBeans et quelques éléments de programmation Java indispensables pour le cours d'IHM.

Dans la première partie de l'expérimentation, quatre concepteurs pédagogiques ont utilisé les outils de SEPIA pour définir un système d'assistance pour NetBeans. Pour cela, ils ont travaillé en concertation avec les enseignants et se sont inspirés des vidéos de démonstration du cours.

Le système d'assistance défini est un tutoriel qui a pour but de répondre aux principaux besoins d'assistance identifiés par les enseignants d'IHM lors du premier TP de chaque semestre. Ce tutoriel est composé de cinq parties : création d'une application graphique, propriétés d'un projet NetBeans, ajout de composants dans une fenêtre, création d'un menu et gestion des événements de l'utilisateur sur l'interface d'une application graphique. Pour chaque partie, le concepteur de l'assistance a utilisé SEPIA durant environ 3 heures afin de spécifier l'assistance en définissant en moyenne 14 règles et 20 actions d'assistance. Les actions d'assistance produites étaient de type message, mise en valeur de composants de l'interface et exemple. Ce travail est certes conséquent, mais il pourra être ré-exploité pour les prochains semestres. La Figure 2 présente un écran de NetBeans pendant l'exécution du tutoriel ainsi produit.

Dans un second temps, nous avons proposé aux étudiants volontaires de suivre ce tutoriel lors du premier TP d'IHM. Sur les 85 étudiants présents lors de ce TP, 64 ont demandé à suivre ce tutoriel. Néanmoins, par manque de machines équipée de SEPIA, seuls 52 étudiants ont pu participer à cette expérimentation.

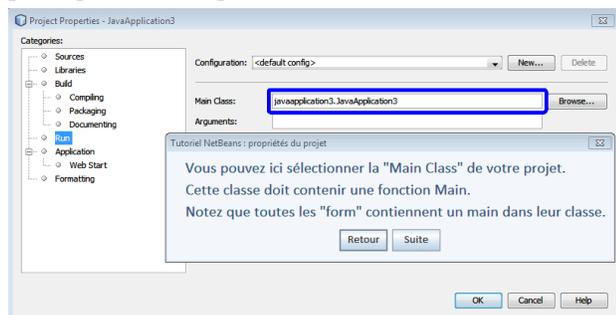


Figure 2 : Exemple d'une assistance pour NetBeans

Avant le début de l'expérimentation, nous avons distribué aux étudiants un questionnaire visant à connaître leurs expériences et connaissances préalables concernant Java et NetBeans.

Après ce questionnaire préliminaire, les étudiants ont pu suivre le tutoriel proposé ; ils étaient libres de passer autant de temps qu'ils le souhaitent sur chaque partie du tutoriel, et de ne pas faire les parties qui ne les intéressaient pas. L'utilisation du tutoriel était intégrée à la séance de TP et a duré entre 30 minutes et 1h30 selon les étudiants.

À la suite du tutoriel, nous avons distribué aux 52 participants un questionnaire final, afin d'évaluer leur satisfaction, et d'essayer de mesurer la progression de leurs connaissances sur NetBeans et la programmation Java. Les étudiants ont été très satisfaits du tutoriel :

90,4% ont déclaré l'avoir apprécié et 62% souhaitent suivre d'autres tutoriels de ce type intégrés à NetBeans. De plus, le tutoriel semble avoir facilité l'acquisition de nouvelles connaissances nécessaires pour le cours d'IHM : 96,2% des étudiants ont déclaré que le tutoriel leur avait appris quelque chose. Cet excellent résultat est corroboré par l'amélioration de leurs connaissances sur NetBeans et la programmation Java mesurée par une comparaison des résultats du pré-test et du post-test. La progression moyenne des apprenants entre le pré-test et le post-test est en effet de 54%, avec une progression allant de 10% à 100% pour les apprenants dont toutes les réponses au pré-test étaient fausses et dont toutes les réponses au post-test étaient correctes. Par ailleurs, les enseignants du cours ont particulièrement apprécié ce tutoriel et pensent continuer à l'utiliser lors des prochains semestres en l'enrichissant.

5. Discussion

5.1. Limites de l'utilisation de SEPIA en EIAH

Cette première expérimentation a montré que le système SEPIA permet effectivement de mettre en place des systèmes d'assistance épiphytes dans des application-cibles utilisées par des apprenants en contexte éducatif. La seconde colonne du Tableau 1 présente les besoins d'assistance en contexte éducatif auxquels SEPIA permet de répondre : ✓ signifie que SEPIA permet de répondre de manière satisfaisante à un besoin (des améliorations visant à faciliter la tâche du concepteur pédagogiques sont néanmoins possibles), ~ signifie que SEPIA permet de répondre de manière limitée à un besoin, et A indique qu'il est possible de répondre à un besoin en exploitant SEPIA, mais que cela demande un effort conséquent de la part du concepteur pédagogique et qu'une automatisation est souhaitable pour lui faciliter la tâche.

En particulier, SEPIA ne permet pas actuellement d'exploiter les connaissances du domaine d'une application-cible. Par exemple, pour réaliser un diagnostic dans une application-cible qui vise la résolution d'équations, il est souhaitable que le système d'assistance dispose de connaissances sur les équations pour comprendre que la réponse « $5 - X + 3$ » doit être considérée comme équivalente à la réponse « $-X + 8$ » dans certaines situations (dépendant de l'objectif pédagogique de l'enseignant). Des connaissances du domaine sont également utiles pour la proposition d'exemples. En effet, pour une application-cible de résolution d'équation, le concepteur pédagogique peut actuellement définir de manière fixe un ensemble d'exemples qui pourront être utilisés par le système d'assistance. Néanmoins, des connaissances du domaine permettraient au système d'assistance de proposer des exemples plus pertinents et contextualisés en s'appuyant sur des paramètres fixés par le concepteur pédagogique. De même, des connaissances sur les activités pédagogiques proposées par un EIAH permettraient

d'améliorer la pertinence de l'assistance proposée tout en simplifiant la tâche du concepteur pédagogique. En effet, le concepteur pédagogique doit actuellement préciser lui-même lors de la définition des règles d'assistance quelles connaissances ou compétences pédagogiques doivent être maîtrisées avant de réaliser une activité, et quelles activités permettent d'acquérir une connaissance ou compétence pédagogique.

Concernant la spécification de l'assistance, la tâche du concepteur pédagogique nécessite d'être facilitée. En effet, la spécification d'un système d'assistance capable de répondre à des besoins pédagogiques implique la définition de nombreuses règles d'assistance et plus particulièrement de nombreuses consultations de l'historique de l'assistance afin d'éviter qu'une règle ne se déclenche plusieurs fois ou au mauvais moment.

5.2. Travaux connexes

Plusieurs travaux s'intéressent à l'assistance aux utilisateurs.

Une première approche consiste à proposer des applications initialement pourvues d'une assistance (tutoriel dans Connectify, bulles d'aide dans Word...).

Une deuxième approche consiste à ajouter une assistance à un environnement ou une application en développant un système d'assistance qui ne pourra pas évoluer par la suite, par exemple USCSH (Matthews et al. 2000) pour l'utilisation des commandes dans Unix.

Une troisième approche vise la construction d'un système qui permet à un expert différent du concepteur de l'application de définir des systèmes d'assistance. Par exemple, EpiTalk (Paquette et al. 1996) permet de définir et de greffer un système conseiller épiphyte sur une application-cible. Cette définition se base sur la définition de 3 graphes : celui de l'application-cible, celui des tâches réalisées par l'utilisateur et celui des conseillers. Pour définir de tels systèmes, il faut avoir des connaissances en programmation Smalltalk et en système multi-agents. Explor@ Advice Agent (Lundgren-Cayrol et al. 2001) permet quant à lui de définir et d'exécuter un système conseiller pour l'apprentissage à distance dans Explor@ Virtual Campus. Ce système fournit des conseils, des questions de diagnostics, des barres de progression. Il est spécifique au web et développé pour un environnement donné.

Cette troisième approche est proche de celle adoptée dans le projet AGATE. Mais SEPIA propose une solution à la fois : générique, épiphyte, personnalisable (selon de différents critères), disponible pour différentes plateformes (applications Windows natives, Java, Web). Cependant, en l'appliquant dans le contexte éducatif, nous avons pu identifier des limites qui nous amènent aux questions de recherche présentées dans la section suivante.

5.3. Questions de recherche

Comment rendre SEPIA plus accessible à des non-experts ? L'objectif est de permettre à des concepteurs

pédagogiques de définir plus facilement des systèmes d'assistance. Une première piste consiste à proposer des patrons d'assistance. Pour que cette solution soit pertinente, il faut au préalable définir quels patrons seront utiles aux concepteurs d'assistance en EIAH. Une deuxième piste est de permettre au concepteur de travailler à différents niveaux de granularité tant pour la création que pour la réutilisation des règles d'assistance.

Comment enrichir SEPIA pour qu'il prenne en compte les connaissances du domaine ? L'objectif est de permettre au concepteur de définir également l'assistance qui exige des connaissances du domaine (diagnostic intermédiaire, explication sur les erreurs en algèbre,...). Pour cela, l'acquisition de connaissances devient indispensable. L'extension de SEPIA envisagée devra donc permettre non seulement d'exploiter des connaissances existantes, mais également d'acquérir de nouvelles connaissances.

Références

- Gapenne, O. 2006. Relation d'aide et transformation cognitive. *Intellectica* 44(2): 7-16.
- Ginon, B.; Champin, P. A.; and Jean-Daubias, S. 2013a. An approach for collecting fine-grained use traces in any application without modifying it. In Workshop EXPORT, conference ICCBR 2013.
- Ginon, B.; Jean-Daubias, S.; and Champin, P. A. 2013b. Une typologie de l'assistance aux utilisateurs: exemple d'application aux EIAH. Rapport de recherche RR-LIRIS-2013-007.
- Ginon, B.; Jean-Daubias, S.; Champin, P. A.; and Lefevre, M. 2014. aLDEAS : un langage de définition de systèmes d'assistance épiphytes. In IC 2014.
- Grandbastien, M., and Labat, J. M. 2006. Environnements informatiques pour l'apprentissage humain. *Traité IC2, série Cognition et traitement de l'information*.
- Lundgren-Cayrol, K.; Paquette, G.; and Miara, A. 2001. Explor@ Advisory Agent: Tracing the Student's Trail. In WebNet 2001, 802-808.
- Matthews, M.; Pharr, W.; Biswas, G.; and Neelakandan, H. 2000. USCSH: an active intelligent assistance system. *Artificial Intelligence Review* 14(1-2): 121-141.
- Netbeans. 2013. <https://netbeans.org/> (consulté le 27/02/2014).
- Paquette, G.; Pachet, F.; Giroux, S.; and Girard, J. 1996. EpiTalk, a generic tool for the development of advisor systems. *International Journal Of Artificial Intelligence in Education* 7:349-370.
- Ritter, S.; and Koedinger, K. R. 1995. Towards lightweight tutoring agents. In *AIED*, 95:16-19.
- Visual Studio. 2014. <http://www.microsoft.com/france/visual-studio/> (consulté le 27/02/2014).