

Comment combiner les objectifs et méthodes d'évaluation pour la conception itérative des EIAH

Leçons tirées de la conception de AMBRE-add

Sandra Nogry (LIRIS, Lyon), Stéphanie Jean-Daubias (LIRIS, Lyon),
Nathalie Guin-Duclosson (LIRIS, Lyon)

■ **RÉSUMÉ** • Cet article discute des évaluations à réaliser dans le cadre de la conception itérative d'un EIAH. Au cours de la conception, différents aspects de l'interface doivent être évalués et des méthodes différentes ont été proposées pour chacun de ces aspects. Lors de la conception de l'EIAH AMBRE-add, nous avons combiné différentes méthodes, successivement ou simultanément, afin de faire progresser la conception du logiciel. La démarche d'évaluation adoptée dans ce projet est analysée ici en précisant les apports et les limites des différentes évaluations réalisées, ceci afin de proposer une démarche d'évaluation plus générale pouvant être appliquée dans le cadre de la conception itérative d'EIAH.

■ **MOTS CLÉS** • conception itérative, utilisabilité, utilité, usages, démarche d'évaluation.

■ **ABSTRACT** • This paper deals with the evaluation to conduct during the iterative design of an ILE. During the design of an ILE a set of aspects of the system must be evaluated and a lot of evaluation methods exist. For the design of the ITS AMBRE-add, we combined several methods in order to evaluate and refine the system. In this paper, the approach we adopted to evaluate AMBRE-add is described and analysed in order to specify its contributions and its limits and we propose a general approach of evaluation to implement during iterative design of an ILE.

■ **Keywords** • iterative design, usability, utility, usage analysis, evaluation processes.

1. Introduction

Depuis longtemps déjà, de nombreux auteurs militent pour l'intégration de l'évaluation dès la conception des EIAH (voir par exemple Nanard et Nanard, 1998) afin de corriger rapidement des problèmes qu'il serait coûteux voir impossible de corriger en fin de conception.

Mais quelle démarche d'évaluation suivre au cours de la conception d'un EIAH ? Différents objets d'évaluations ont été mis en évidence, différentes méthodes ont été décrites et proposées. Dans cet article, à la lumière de l'évaluation que nous avons conduite dans le cadre du projet AMBRE, nous proposons une démarche d'évaluation dans un contexte de conception itérative d'un EIAH. Cette démarche intègre l'évaluation de l'utilisabilité, de l'utilité du logiciel et la prise en compte de ses usages en proposant de conjuguer différentes méthodes adaptées à chacun de ces aspects.

Après avoir présenté les différents objets d'évaluation habituellement pris en compte lors de la conception de systèmes informatiques, ainsi que quelques-unes des méthodes associées, nous présenterons les spécificités de l'évaluation des EIAH en discutant des méthodes adaptées à cette évaluation. Ensuite, nous présenterons les évaluations réalisées dans le cadre de la conception de AMBRE-add, un EIAH destiné au domaine des problèmes additifs ; dans cette partie, nous insisterons sur les avantages et les limites des différentes méthodes utilisées et en tirant les conséquences des évaluations que nous avons conduites. Enfin, nous décrivons la démarche d'évaluation que nous proposons dans le cadre de la conception itérative d'un EIAH.

2. Démarche de conception itérative

En génie logiciel, différentes méthodes de conception ont été décrites. Une méthode souvent utilisée est la conception itérative, qui consiste en une succession de phases permettant d'affiner progressivement les spécifications, d'évaluer les solutions retenues, puis d'intégrer les modifications choisies jusqu'à obtention d'un produit satisfaisant (Van Eylen et Hiraclidès, 1996 cité par Jean, 2000). Cette démarche de conception s'appuie sur la réalisation de maquettes et de prototypes qui, présentés aux clients et/ou utilisateurs, sont évalués puis éventuellement modifiés. Cette méthode est spécialement adaptée au développement de logiciels pour lesquels il est difficile de définir entièrement les spécifications au début de projet. Elle permet en effet la prise en compte en cours de conception d'objectifs nouveaux, notamment par la large place qu'elle laisse à l'évaluation, à la prise en compte de l'avis des clients mais aussi des utilisateurs. Cette méthode permet également l'évaluation de différentes alternatives de conception, possibilité particulièrement intéressante dans un contexte de recherche. Pour ces raisons, la démarche itérative, appropriée au développement d'EIAH, a été adoptée dans nombre de projets de recherche de ce type (voir par exemple Jean, 2000 ; Nogry, Jean-Daubias et Duclosson, 2004).

Évaluation des systèmes informatiques en conception

De nombreuses méthodes destinées à évaluer les systèmes informatiques existent. Ces méthodes permettent d'évaluer différents aspects d'un système. Senach (1993) distingue deux facettes particulièrement importantes : l'utilisabilité d'un système informatique (sa facilité d'utilisation) et son utilité (l'adéquation du logiciel aux objectifs de haut niveau). Pour évaluer ces deux aspects, il est possible de procéder à une évaluation analytique ou à une évaluation empirique. En situation de conception ces évaluations doivent non seulement informer sur la qualité du logiciel mais aussi permettre d'identifier les erreurs de conception et les déficits éventuels du logiciel afin de pouvoir proposer une nouvelle version plus aboutie du logiciel.

En premier lieu, il est important d'évaluer l'utilisabilité d'un système. L'utilisabilité concerne l'adéquation entre la manière dont une tâche est réalisée par un utilisateur et les capacités cognitives de cet utilisateur (Farenc, 1997). D'après la définition ISO 9241-11, un logiciel est utilisable lorsque l'utilisateur peut réaliser sa tâche (efficacité), qu'il consomme un minimum de ressources pour le faire (efficience) et que le système est agréable à utiliser (satisfaction de l'utilisateur).

L'évaluation analytique de l'utilisabilité consiste à étudier les interfaces selon un ensemble de référents afin de contrôler qu'elles possèdent bien certaines qualités et de détecter les problèmes qu'elles peuvent poser. Il existe plusieurs manières de conduire une évaluation analytique. Cette évaluation peut être une évaluation par inspection, qui consiste à examiner chaque aspect du logiciel afin d'identifier précisément les problèmes d'utilisabilité. Cette inspection est conduite par des experts qui peuvent s'appuyer sur des listes de critères conçues par des ergonomes (voir par exemple Bastien et Scapin, 1993 ; Nielsen, 1993 ; Lewis et al., 1990). Ainsi Nielsen (1993) et Schneiderman (1992) distinguent cinq attributs de l'utilisabilité :

- l'apprentissage (de l'utilisation du système),
- l'efficacité (la facilité à réaliser la tâche souhaitée),
- la mémorisation (la capacité à reprendre en main rapidement le système lors d'utilisations espacées),
- la gestion des erreurs (liées à l'utilisation du logiciel),
- la satisfaction subjective de l'utilisateur.

Bastien et Scapin (1993) proposent, quant à eux, huit critères ergonomiques pour l'évaluation des interfaces (qui recoupent parfois les attributs présentés précédemment). Citons pour exemple le critère de gestion des erreurs : « Le critère gestion des erreurs concerne tous les moyens permettant d'une part d'éviter ou de réduire les erreurs, et d'autre part de les corriger lorsqu'elles surviennent. Les erreurs sont ici considérées comme des saisies de données incorrectes, des saisies dans des formats inadéquats, des saisies de commandes avec syntaxe incorrecte, etc. Trois sous-critères participent à la gestion des erreurs : protection contre les erreurs, qualité des messages d'erreurs et correction des erreurs. » (Bastien et Scapin, 1993).

Pour faciliter l'évaluation analytique, il existe également des check-lists qui permettent de vérifier point par point que le logiciel est conforme à un certain nombre de critères.

Une autre méthode d'évaluation analytique consiste à spécifier des tâches et des séquences d'actions dans le logiciel et à imaginer ce que ferait l'utilisateur dans ces situations. Ces inspections cognitives (cognitive walkthroughs) (Lewis et al., 1990) permettent ainsi de s'assurer que le système peut réaliser les actions souhaitées et d'identifier quels problèmes peuvent se poser.

L'évaluation empirique consiste à recueillir des données relatives au comportement de l'utilisateur lors de l'utilisation du système. Ce type d'évaluation nécessite l'existence d'un système réel (maquette, prototype ou système final) et la présence d'utilisateurs. L'utilisation qui est faite du système par ces utilisateurs est observée et analysée. Une observation individuelle détaillée de l'interaction entre l'utilisateur et le système permet d'identifier les capacités de l'utilisateur, de détecter les difficultés éventuelles, ou encore de noter les caractéristiques inattendues de la situation (Gagné, Briggs et Wager, 1988). Ainsi, pour évaluer empiriquement l'utilisabilité d'un système, Nielsen (1993) propose d'observer des utilisateurs durant l'utilisation du logiciel suivant des critères déterminés et d'interroger ensuite les utilisateurs. Ses études montrent qu'un panel de cinq utilisateurs « représentatifs » permet d'identifier 80% des problèmes (Nielsen et Landauer, 1993). Ces techniques d'observation peuvent être complétées par un questionnaire afin d'avoir des indications sur la compréhension ou la satisfaction de l'utilisateur.

Il est également nécessaire d'évaluer l'utilité du système. Rappelons que l'utilité est l'adéquation entre les fonctions fournies par le système et celles nécessaires à l'utilisateur pour atteindre les objectifs de haut niveau du client. Si des critères et des méthodes d'évaluation de l'utilisabilité ont été bien formalisés, l'évaluation de l'utilité l'est beaucoup moins du fait que les objectifs à évaluer sont spécifiques à chaque logiciel. Classiquement, l'utilité est évaluée empiriquement en observant si des utilisateurs représentatifs sont capables d'accomplir leur tâche à partir des fonctionnalités du système. Pour cela, il est nécessaire de bien formaliser la tâche que l'utilisateur doit accomplir en proposant un scénario représentatif d'une situation d'utilisation.

3. L'évaluation des EIAH

Spécificité des EIAH

Les EIAH ont certaines spécificités qu'il est nécessaire de prendre en compte pour évaluer de manière adéquate leur utilisabilité et leur utilité.

La première spécificité concerne leur objectif de haut niveau : faciliter l'apprentissage. Cet objectif peut différer de la tâche que doit accomplir l'apprenant (l'objectif à court terme). Par exemple, pour favoriser l'apprentissage d'une notion

particulière, un premier objectif à court terme peut être de rechercher des documents traitant de cette notion dans un hypermédia.

La seconde spécificité est liée aux utilisateurs des EIAH. Il faut en effet distinguer deux types d'utilisateurs : les apprenants et les enseignants. Si les apprenants sont clairement les utilisateurs finaux principaux de ces systèmes, les enseignants en sont prescripteurs, au sens où ce sont généralement eux qui provoquent l'utilisation par les apprenants d'un EIAH donné, mais également utilisateurs secondaires (Dubourg et Teutsch, 1997 ; Jean, 2000) du fait qu'ils peuvent parfois préparer le travail de leurs apprenants en paramétrant le système.

Par ailleurs, les situations d'utilisation des EIAH peuvent être très variées. Ces systèmes peuvent être utilisés seuls ou en binôme, à la maison, en salle informatique avec l'enseignant ou en salle de classe en autonomie pendant que les autres élèves effectuent d'autres activités. L'utilisation du système peut être extrêmement ponctuelle (utilisation unique) ou régulière pendant une période de l'année (par exemple une utilisation par semaine pendant trois mois). Or la situation d'utilisation d'un EIAH a un impact sur l'apprentissage. Il est donc nécessaire de prendre en compte cette situation lors de l'évaluation de l'apprentissage.

Suivant ces spécificités, les méthodes d'évaluation de l'utilisabilité doivent être adaptées aux EIAH (Hû et Trigano, 1998 ; Jean, 2000) et les méthodes d'évaluation de l'utilité doivent être spécifiques à l'objectif de haut niveau de ces systèmes : l'apprentissage. Par ailleurs, Tricot et al. (2003) montre qu'une troisième facette de l'évaluation d'un système, l'acceptabilité, est également importante à prendre en compte au cours de la conception d'un EIAH.

Utilisabilité et EIAH

« L'utilisabilité des EIAH se joue au niveau de l'interface (sa cohérence, sa lisibilité, la façon dont elle représente les actions possibles, etc.), de sa navigation (la cohérence, la simplicité, l'exhaustivité des déplacements possibles), et de sa cohérence avec l'objectif » (Tricot et al., 2003)

Les différentes méthodes classiquement utilisées pour évaluer l'utilisabilité d'un logiciel doivent être adaptées aux spécificités des EIAH. En ce qui concerne l'évaluation analytique, il est possible d'utiliser les attributs et critères ergonomiques pour évaluer l'utilisabilité des EIAH (Squires et Preece, 1999), mais ceux-ci doivent parfois être adaptés en fonction de l'objectif pédagogique du logiciel, des utilisateurs des EIAH et de la tâche. Ainsi, lorsque l'objectif de haut niveau (l'apprentissage) diffère de l'objectif à court terme de l'utilisateur, la définition de certains critères tels que l'efficacité doivent être reconsidérés : dans un EIAH ayant pour objectif à court terme la résolution de problèmes, on peut privilégier un parcours tortueux du point de vue de la résolution de problèmes, plus bénéfique pour l'apprentissage, même s'il est perçu comme moins efficace qu'un parcours linéaire. D'autres critères doivent être

précisés : par exemple l'application du critère de gestion des erreurs (Bastien et Scapin, 1993) aux EIAH nécessite de faire la distinction entre deux types d'erreurs : les erreurs dans l'utilisation du logiciel et les erreurs au sens de réponses erronées (erreurs conceptuelles) (Jean, 2000). Si les erreurs d'utilisation du logiciel doivent être empêchées ou corrigées, suivant ainsi la recommandation de Bastien et Scapin, la gestion des erreurs conceptuelles doit être traitée indépendamment en fonction de la théorie cognitive sous-jacente à l'EIAH. Ainsi, dans une perspective behavioriste, on choisira de corriger immédiatement chaque erreur, tandis que dans d'autres perspectives on pourra laisser l'apprenant se rendre compte lui-même de ses erreurs.

Les autres méthodes d'évaluation analytique présentées sont plus difficiles à adapter à des systèmes laissant de nombreux degrés de liberté tels que les EIAH. L'utilisation de check-lists semble être peu adaptée (Squires et Preece, 1999) du fait qu'elles ne prennent pas en compte le contexte d'utilisation réel du logiciel, contexte particulièrement important en EIAH. L'utilisation des « inspections cognitives » peut être envisagée, mais seulement pour certains types d'EIAH. En effet, cette méthode demande souvent une modélisation très détaillée de la tâche, ce qui est réalisable pour des environnements d'apprentissage qui favorisent le développement de compétences très précises, mais pas pour des environnements qui permettent une certaine créativité ou qui peuvent être employés de manières différentes suivant les apprenants, comme les micromondes.

Les méthodes d'évaluation empiriques telle que celle proposée par Nielsen sont plus flexibles et semblent plus appropriées aux EIAH dans la mesure où les utilisateurs observés sont représentatifs des apprenants qui utiliseront ensuite le système. Les méthodes classiques d'observations et d'entretiens peuvent donc être utilisées. Par ailleurs, il peut être intéressant de conduire ces observations dans le contexte d'utilisation envisagé pour le logiciel afin de déterminer si les apprenants ont le comportement et les résultats escomptés dans des conditions proches de la réalité (Hoecker et Elias, 1986).

Méthodes d'évaluation de l'utilité des EIAH

Après l'évaluation de l'utilisabilité, il est également nécessaire d'évaluer l'utilité du système. Dans le cadre des EIAH, l'objectif à atteindre comporte deux niveaux : l'apprentissage (de la discipline enseignée et non de la manipulation du système) et la réalisation des tâches proposées par le système dans le but de permettre cet apprentissage (résolution de problèmes, recherche d'informations, simulations...) (Jean, 2000). Même si ces niveaux sont connectés, il n'y a pas toujours de lien direct entre la réalisation de la tâche et l'apprentissage effectif, un échec dans la réalisation de la tâche pouvant, dans certaines conditions, être bénéfique pour l'apprentissage. Évaluer l'utilité d'un EIAH ne consiste donc pas seulement à vérifier que l'utilisateur peut réaliser la tâche qu'il souhaite accomplir (ce qui peut se faire selon les méthodes

classiques de mesure de l'utilité), mais aussi à évaluer si l'objectif d'apprentissage tel qu'il est défini par le concepteur est atteint par l'apprenant.

Cette évaluation doit permettre de définir si les déficits éventuels rencontrés relèvent d'un problème de contenu, d'une inadéquation entre le dispositif et le format de la connaissance à acquérir (pour une synthèse, voir De Vries, 2001) ou si le problème relève d'un manque de correspondance entre l'objectif d'apprentissage et les besoins de l'apprenant (Tricot et al., 2003).

L'évaluation de l'utilité devrait également prendre en compte la mise en œuvre des processus cognitifs impliqués dans l'activité d'apprentissage (Tricot et al., 2003 ; Nogry et Didierjean, à paraître ; Mayer, 2001 ; Sander et Richard, 1997 ; Lim, 1998).

Il existe de nombreuses méthodes pour évaluer l'apprentissage, pour la plupart empiriques. Ces méthodes, issues de techniques non spécifiques aux EIAH, peuvent être adaptées pour évaluer de tels systèmes. Elles peuvent permettre de quantifier l'impact de l'EIAH sur l'apprentissage, d'informer sur l'activité de l'apprenant durant l'utilisation de l'EIAH ou encore de prendre en compte la situation dans laquelle se déroule l'apprentissage. Nous proposons ici une rapide description de ces différentes méthodes.

La **méthode comparative**, développée par la psychologie cognitive, est souvent utilisée pour évaluer les EIAH. Elle consiste à comparer l'effet de plusieurs situations sur l'apprentissage qui en résulte. Shute et Regian (1993) présentent en détail l'application de cette méthode aux EIAH. La difficulté réside dans le choix de la condition contrôle : faut-il comparer l'EIAH à un enseignement oral ? À un autre système ? À une version tronquée de l'EIAH testé ? Malgré cette difficulté, cette méthode permet d'observer le résultat d'un changement dû au système et d'inférer les connaissances acquises par l'apprenant avec un certain degré de généralité.

Toutefois, cette méthode ne permet pas de comprendre ce qui se passe au cours de l'apprentissage ou au cours de l'utilisation du dispositif de formation. Or pour comprendre et interpréter les résultats obtenus avec la méthode comparative, il peut être important de savoir ce qu'a fait l'apprenant durant l'utilisation du logiciel. Des **méthodes « on-line »** (Rouet et Passerault, 1999) permettent d'identifier l'activité de l'apprenant au cours de l'utilisation du logiciel. Certaines méthodes (telles que le paradigme de double tâche) issues de recherches en compréhension de texte, permettent d'identifier sur quels éléments l'apprenant a focalisé son attention. D'autres méthodes permettent de mieux comprendre les processus d'apprentissage mis en œuvre. Ainsi, le **recueil des verbalisations** (Caverni, 1988), qui consiste à demander à l'apprenant de penser à haute voix durant l'activité proposée, permet d'identifier les raisonnements qu'il peut mettre en œuvre pour la réaliser.

La collecte de documents et de traces d'interactions peut compléter les informations verbales. En complément de ces méthodes, les **méthodes ethnographiques** permettent de prendre en compte la situation dans laquelle l'apprentissage se déroule.

Ces méthodes consistent à observer une situation « de l'intérieur ». L'observateur fait parti de la situation qu'il observe, il est pleinement conscient de la subjectivité de sa situation et adopte donc une position réflexive sur sa pratique d'observation. Pour l'évaluation des EIAH, l'observation peut porter sur un apprenant, un groupe d'apprenants ou une classe entière (Barfurth et al., 1994).

Acceptabilité des EIAH

Enfin, une troisième dimension à évaluer concerne l'acceptabilité de l'EIAH. Celle-ci peut se définir comme la valeur de la représentation mentale (attitudes, opinions, etc.) de l'utilisateur ou du prescripteur à propos de l'EIAH, de son utilisabilité et de son utilité (Tricot et al., 2003). Cette représentation mentale, individuelle ou collective conditionnerait la décision d'utilisation de l'EIAH. Un EIAH étant conçu pour être utilisé, cette dimension doit également être prise en compte. Comme le précise Tricot et al. (2003), l'acceptabilité peut être sensible à des facteurs tels que la culture et les valeurs des utilisateurs, leurs affects, leur motivation, l'organisation sociale et les pratiques dans lesquelles s'insère plus ou moins bien l'EIAH. Ce critère d'évaluation peut faire l'objet d'une évaluation par inspection en étudiant l'acceptabilité en termes d'adéquation aux besoins de l'institution, aux attentes des apprenants, aux caractéristiques des apprenants. Elle peut également être envisagée en termes de compatibilité avec l'organisation du temps, l'organisation des lieux, la présence du matériel nécessaire, la visibilité des résultats. Enfin, il peut faire l'objet d'une évaluation empirique par observations, entretiens ou questionnaires en prenant en compte la motivation, les affects, les cultures et les valeurs des utilisateurs et prescripteurs de l'EIAH.

Selon le modèle proposé par Nielsen (1993), l'acceptabilité d'un système comporte une dimension pratique et une dimension sociale. L'utilisabilité et l'utilité d'un système font partie des critères qui déterminent son acceptabilité pratique. Selon ce modèle, pour être acceptable, un EIAH doit donc nécessairement être utilisable et utile : l'utilisabilité et l'utilité apparaissent comme des conditions nécessaires à une bonne acceptabilité, mais non suffisantes.

Des études plus récentes ont proposé d'autres modèles et montrent que les relations entre ces trois dimensions ne sont pas faciles à évaluer et sont plus complexes que ce que laisse supposer le modèle de Nielsen, chacune de ces trois dimensions ayant une influence sur les autres (voir Tricot et al., 1993 pour une revue).

4. Comment conjuguer les différents objets d'évaluation et quelles méthodes utiliser lors de la conception itérative d'un EIAH ?

Comme nous l'avons vu différents objets d'évaluation doivent être pris en compte au cours de la conception et de multiples méthodes existent pour les prendre en considération. Quelles méthodes utiliser pour évaluer chacune de ces facettes ? Voilà la question à laquelle nous avons été confrontées lors de la conception de l'EIAH

AMBRE-add. S'il existe différents articles portant sur la description des méthodes, leur intérêt et leurs limites, il n'existe pas à notre connaissance de guide qui permette d'orienter la démarche d'évaluation en conception. Aussi, nous avons mis en place une démarche d'évaluation à partir des références existantes afin d'évaluer l'utilisabilité puis l'utilité de AMBRE-add. Dans cette section, nous présenterons les différentes évaluations réalisées dans le cadre de la conception de l'EIAH AMBRE-add en en présentant les intérêts et les limites et nous procéderons à une réflexion critique sur la démarche d'évaluation ainsi mise en place.

L'EIAH AMBRE-add

L'EIAH AMBRE-add, conçu dans le cadre du projet AMBRE (Guin-Duclosson, Jean-Daubias et Nogry, 2002), propose de guider la résolution de problèmes additifs afin de conduire les apprenants à acquérir des classes de problèmes (Vergnaud, 1982 ; Riley, Greeno et Heller, 1983 ; Guin 1991) et des techniques associées à ces classes de problèmes. Suivant le principe du projet AMBRE, l'EIAH présente à l'apprenant des problèmes résolus puis assiste l'apprenant dans la résolution d'un nouveau problème en le guidant à travers les différentes étapes du cycle du Raisonnement à Partir de Cas (Figure 1) :

- Après avoir lu l'énoncé du problème, l'apprenant doit reformuler le problème afin d'identifier les éléments pertinents pour la résolution.
- Ensuite, il choisit, parmi les problèmes-types, un problème proche du problème à résoudre.
- Puis il adapte la solution du problème type au problème à résoudre.
- Enfin il range le nouveau problème dans la base de problèmes déjà rencontrés.

Ces étapes sont donc guidées par le système mais effectuées par l'apprenant. Dans chacune de ces étapes, un diagnostic est réalisé par le système. Celui-ci évalue la production de l'apprenant et lui propose si nécessaire des explications pour l'amener à comprendre ses erreurs et à corriger ses réponses. Nous considérons que ce diagnostic remplace l'étape de révision présente dans le cycle du RâPC ; ainsi, la révision est distribuée entre les différentes étapes.

L'objectif du projet AMBRE est de montrer que des EIAH conçus suivant ce principe peuvent faciliter l'acquisition de classes de problèmes et des techniques de résolution associées, dans le domaine des problèmes additifs comme dans d'autres domaines d'application.

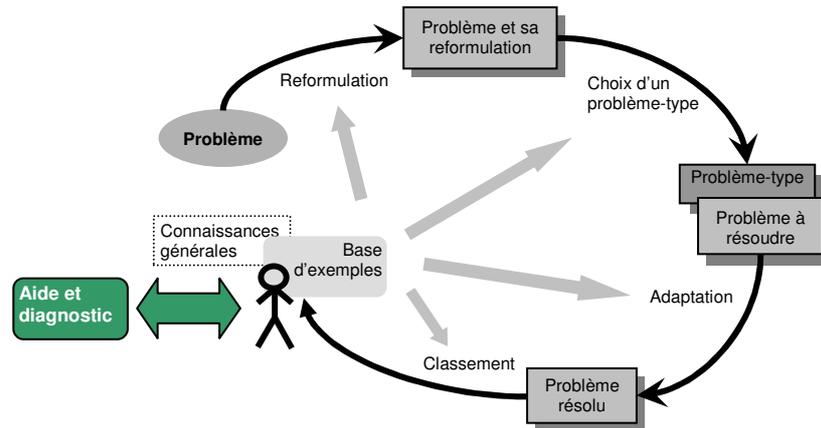


Figure 1 : Le cycle du Raisonnement à Partir de Cas adapté au projet AMBRE (« cycle AMBRE ») ; ce cycle décrit les activités de l'apprenant lors de l'utilisation du logiciel

Les problèmes présentés par AMBRE-add sont des problèmes additifs à une étape étudiés en cycle 2 de l'école primaire française.

Les problèmes additifs décrivent une situation concrète, par exemple un jeu de billes : « Karim avait 32 billes. À la fin de la récréation, il en a 45. Combien a-t-il gagné de billes pendant la récréation ? ». Pour un problème de ce type, la résolution proposée dans AMBRE-add est décomposée en quatre étapes :

- Décrire le problème à l'aide d'une "opération à trou" : $32 + ? = 45$ (équation représentant le problème)
- Écrire comment on effectue le calcul : $45 - 32 = ?$
- Effectuer le calcul : 13
- Écrire la réponse à la question : Karim a gagné 13 billes.

Seule l'équation représentant la résolution est considérée comme étant la technique de résolution associée à la classe, le reste de la résolution mobilise d'autres compétences relevant du calcul numérique.

Certains problèmes additifs sont résolus dès la maternelle, alors que d'autres posent encore des difficultés en fin de troisième (Marthe, 1982). Greeno et Riley (1987) montrent que les difficultés que rencontrent les jeunes enfants en résolvant des problèmes additifs viennent essentiellement du fait qu'ils n'arrivent pas à se représenter correctement la situation décrite dans l'énoncé. De nombreuses études montrent que la catégorie du problème ainsi que la nature de l'inconnue interviennent dans la difficulté du problème. Le projet AMBRE postule que le fait de demander à l'apprenant de reformuler le nouveau problème puis de rechercher un problème proche afin d'adapter sa solution pour résoudre le nouveau problème peut lui apprendre à modéliser le problème et à acquérir les classes de problèmes et les techniques de résolution associées.

Cet EIAH a été développé au sein d'une équipe pluridisciplinaire en suivant un cycle de conception itérative alternant des phases de conception et de développement et des phases d'évaluation.

Cycle de conception et d'évaluation de l'EIAH AMBRE-add

La conception itérative de AMBRE-add est fondée sur la spécification et l'implémentation de prototypes qui sont évalués puis conduisent à la production de nouvelles spécifications qui permettent d'implémenter une nouvelle version du logiciel modifiée et complétée par de nouvelles fonctionnalités. Cette démarche a été choisie afin de valider les choix de conception pluridisciplinaires et de détecter précocement les problèmes. Le déroulement du projet AMBRE peut être décrit par un cycle inspiré du schéma proposé par Jean (2000) qui représente l'élargissement progressif des utilisateurs, des tests et des validations des logiciels.

De part la nature pluridisciplinaire du projet AMBRE, plusieurs partenaires sont intervenus dans chacune de ces étapes : des chercheurs en informatique, mais aussi en psychologie cognitive et en didactique des mathématiques, une conseillère pédagogique, des enseignants et des apprenants.

Des premières spécifications générales définissant le principe du projet AMBRE ont d'abord été proposées (Figure 2, 2a) par l'équipe de conception. Ensuite, ces spécifications ont été appliquées au domaine des problèmes additifs afin de concevoir et d'implémenter l'EIAH AMBRE-add (Figure 2, 3b). Dans le cadre d'une conception itérative, ce système a fait l'objet d'une évaluation technique par les développeurs, d'une évaluation analytique par les concepteurs, une conseillère pédagogique et des enseignants et de plusieurs évaluations empiriques destinées à évaluer l'utilisabilité puis l'utilité du système (Figure 2). Chaque évaluation a été suivie de nouvelles spécifications et de modifications du système.

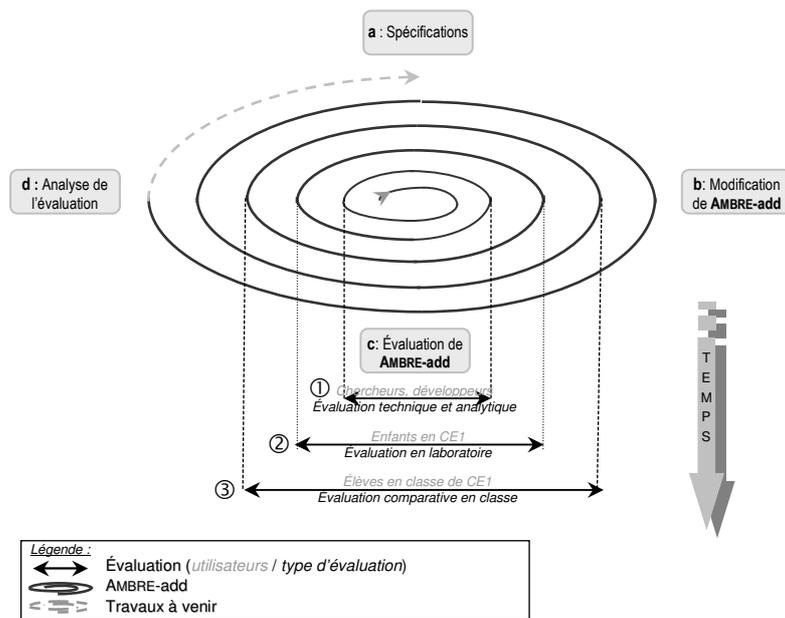


Figure 2 : cycle de conception itérative de l'EIAH AMBRE-add.

Dans les sections suivantes, nous décrirons d'abord les méthodes mises en place dans chacune des évaluations réalisées, en précisant à quels types de spécifications elles ont permis d'aboutir et en insistant sur les intérêts et limites des méthodes utilisées. Puis nous clôturerons cette section en faisant une analyse critique des évaluations réalisées.

Les évaluations menées durant la conception de AMBRE-add

Pour évaluer l'utilisabilité de AMBRE-add et son impact sur l'apprentissage, nous avons conduit plusieurs expériences combinant différentes méthodes présentées dans la partie précédente. Nous avons d'abord réalisé une première expérience consistant à observer cinq enfants individuellement afin d'identifier les principaux problèmes d'utilisabilité. Ensuite, nous avons réalisé une seconde expérience afin d'évaluer l'utilité du logiciel, c'est-à-dire son impact sur l'apprentissage. Nous avons adopté une méthode comparative en comparant AMBRE-add avec deux maquettes contrôle. Nous avons réalisé cette expérience dans une situation « réelle » d'utilisation, dans une école avec trois classes d'élèves en CE1. Pour prendre en compte les caractéristiques de la situation d'utilisation, nous avons complété cette expérience en mettant en œuvre différentes méthodes qualitatives (observation, entretiens, questionnaire).

Nous avons également réalisé deux évaluations auprès d'élèves en classe de CE2. La première évaluation, qui consistait à observer l'utilisation du logiciel par 21 élèves,

avait pour but d'analyser l'usage du logiciel afin de vérifier si AMBRE-add était adapté à des élèves de CE2. La seconde consistait à comparer les performances d'élèves ayant utilisé AMBRE-add à celles des élèves n'ayant pas effectué d'activité en lien avec la résolution de problèmes additifs.

Nous n'avons pas mis en place d'évaluation dédiée aux questions d'acceptabilité, mais au cours des différentes évaluations, nous avons recueilli différentes informations relatives à cette dimension à l'aide d'observations ou des questionnaires.

Nous présentons ici chacune de ces évaluations avant de faire une analyse critique des résultats obtenus.

Évaluations par inspection

Nous avons d'abord procédé à différentes évaluations analytiques concernant plusieurs aspects du logiciel. Outre l'évaluation technique, nous avons réalisé une évaluation de l'utilisabilité en inspectant les différents aspects de l'interface suivant les critères proposés par Bastien et Scapin (1993) en accordant une attention particulière à la cohérence de l'interface, au guidage et à la gestion des erreurs.

Concernant la cohérence, nous nous sommes assurées que les codes, les dénominations et les interactions proposées soient bien homogènes. Nous avons évalué le guidage en nous assurant que l'utilisateur puisse savoir à tout moment où il se situe dans la séquence d'action et quelles actions lui sont permises. De plus, nous avons vérifié que les groupements proposés facilitent bien la compréhension et la mise en relation des différents éléments d'interface. En outre, nous avons essayé de réduire le nombre d'erreurs d'utilisations possibles, et nous avons vérifié qu'un message d'aide était systématiquement prévu en cas d'erreur d'utilisation.

AMBRE-add a également fait l'objet d'une évaluation par inspection par des enseignants et une conseillère pédagogique afin de valider l'interface sur le plan pédagogique. Cette inspection a permis d'évaluer différents aspects du logiciel. Tout d'abord, une inspection globale de l'EIAH a été réalisée afin que les enseignants vérifient que les différentes activités proposées par le logiciel dans ses différentes étapes étaient adaptées à des élèves de cycle 2. Cette inspection avait également pour objectif de vérifier la qualité des consignes et des messages d'erreur et de diagnostic, afin d'ajuster le vocabulaire utilisé aux élèves de CE1. Cette inspection a par ailleurs porté sur le contenu proposé par le logiciel : une analyse systématique des problèmes présentés a permis de vérifier que les thèmes abordés étaient adaptés et que les valeurs utilisées dans les problèmes correspondaient bien au niveau des apprenants.

Ainsi, cette inspection pédagogique nous a permis de compléter l'inspection ergonomique de l'utilisabilité en apportant un avis d'experts sur l'adéquation des consignes et des messages au public visé et sur l'adéquation du contenu proposé au niveau des apprenants.

Évaluation de l'utilisabilité du logiciel

Nous avons d'abord évalué l'EIAH AMBRE-add afin de vérifier que le système était utilisable par des élèves de CE1. Comme le préconisent Nielsen et Landauer (Nielsen et Landauer, 1993), nous avons choisi d'observer individuellement cinq élèves de CE1 durant l'utilisation de l'EIAH en laboratoire durant 45 minutes avant de les interroger sur différents aspects de l'utilisation du logiciel. Pour observer l'utilisation, l'expérimentateur était placé derrière l'enfant. Cette observation était guidée par des critères sélectionnés parmi les critères ergonomiques proposés par Bastien et Scapin (1993), Nielsen (1993) et Schneiderman (1992). Ainsi, nous avons plus particulièrement observé la prise en main du logiciel, la compréhension générale, l'efficacité, la gestion des erreurs, la surcharge cognitive et la satisfaction. Pour chaque critère, un ensemble d'observables ont été définis.

Apports de cette évaluation

Cette observation et l'analyse des réponses des enfants nous ont permis de décrire l'utilisation du logiciel suivant les différents critères choisis et de mettre en évidence différentes difficultés d'utilisabilité rencontrées par les enfants lors de l'utilisation de l'interface. Par exemple, dans la phase d'analyse des exemples, les utilisateurs cherchaient comment interagir avec l'interface du fait que certains éléments d'interface évoquaient des cases à remplir mais n'étaient pas associées à une action. Nous avons également observé certaines difficultés pour comprendre le principe général du logiciel ; une séance d'utilisation ne semblait pas suffisante pour prendre en main le logiciel. Enfin, nous avons mis en évidence une difficulté d'ordre mathématique récurrente chez tous les utilisateurs. Dans l'étape d'adaptation, après avoir construit l'addition à trou (ex : $32 + ? = 43$), les apprenants éprouvent de très grosses difficultés à trouver l'opération correspondante (ex : $43 - 32 = ?$) (Figure 3). Les enfants de CE1 ne disposent pas des outils mathématiques pour effectuer cette transformation. Cette difficulté n'avait pourtant pas été évoquée avant l'évaluation empirique ni par la conseillère pédagogique ni par les enseignants ayant collaboré à la conception.

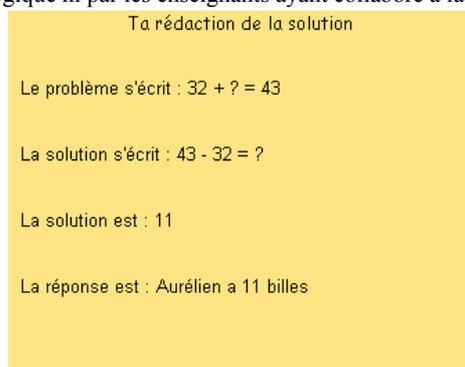


Figure 3 : Plan de rédaction de la solution

Ces différentes observations nous ont amenée à proposer des recommandations qui ont conduit à modifier certains éléments d'interface, à revoir la présentation du logiciel afin de faciliter sa prise en main et sa compréhension, et à modifier l'étape d'adaptation afin d'éliminer la difficulté liée au calcul.

Cette évaluation a donc rempli les objectifs fixés, évaluer l'utilisabilité du logiciel pour faire évoluer la conception, et a également permis de mettre en évidence d'autres difficultés d'ordre mathématiques. Cependant, cette évaluation comporte certaines limites.

Limites

Durant l'observation, nous avons remarqué qu'une séance d'utilisation n'était pas suffisante pour prendre en main le logiciel, mais nous n'avons pas eu la possibilité de déterminer le nombre de séances nécessaires pour prendre en main le logiciel, les enfants n'étant venu au laboratoire que pour une seule séance.

Par ailleurs, les conditions d'observation ont pu influencer le comportement des enfants ; la présence de l'observateur a pu les pousser à être plus concentrés et plus attentifs lors de l'utilisation de logiciel. Par ailleurs, ils ont pu avoir la tentation de demander de l'aide à l'observateur plutôt que d'utiliser l'aide du logiciel (même si l'observateur minimisait ses interventions), ce qui a pu limiter la qualité des observations portant sur l'utilisation du système d'aide et de diagnostic.

Évaluation de l'impact du logiciel sur l'apprentissage

Après avoir évalué l'utilisabilité du logiciel en laboratoire et identifié ainsi les principales difficultés rencontrées lors de l'utilisation du logiciel, nous avons apporté des modifications à AMBRE-add., Nous avons ensuite conduit une seconde évaluation afin d'évaluer l'impact du logiciel sur l'apprentissage. Nous avons choisi de conduire cette évaluation dans une situation proche d'une situation réelle d'utilisation à l'école. Trois classes d'élèves de CE1 ont utilisé l'un des logiciels proposés, en salle informatique à l'école, pendant six séances de 45 minutes, à raison d'une séance par semaine. Ce cadre ne permet pas de contrôler rigoureusement l'environnement, mais offre la possibilité de pouvoir tester un nombre important d'élèves dans une situation familière pour eux, pendant plusieurs séances, en limitant la mortalité expérimentale (liée par exemple aux abandons). À la vue des conditions d'utilisation, il nous a semblé important d'évaluer l'impact du logiciel sur l'apprentissage, mais aussi de récolter des données concernant l'utilisation du logiciel dans une situation « écologique » d'utilisation et de prendre en compte les interactions entre apprenants et encadrants. Pour cela, nous avons combiné différentes méthodes qualitatives et quantitatives. Dans cette partie nous allons d'abord présenter les différentes techniques mises en place pour évaluer chacun de ces aspects, puis nous présenterons sommairement les résultats obtenus avant de discuter l'intérêt et les limites de ces techniques au vu des objectifs visés.

Impact de AMBRE-add sur l'apprentissage

Pour tester l'impact de AMBRE-add sur l'apprentissage de classes de problèmes et des techniques associées, nous avons comparé l'utilisation de AMBRE-add à celle de deux logiciels contrôles. À travers cette comparaison, nous voulions savoir si le principe de AMBRE-add, à savoir guider la résolution d'un problème à travers les étapes du cycle AMBRE, a un impact plus important sur l'apprentissage que d'autres formes de résolutions de problèmes arithmétiques sur ordinateur.

Dans chacune des trois classes de CE1 participant à l'évaluation, la moitié des élèves utilisait le logiciel AMBRE-add tandis que l'autre moitié utilisait l'un des deux logiciels contrôle implémentés pour cette expérience (Figure 4).

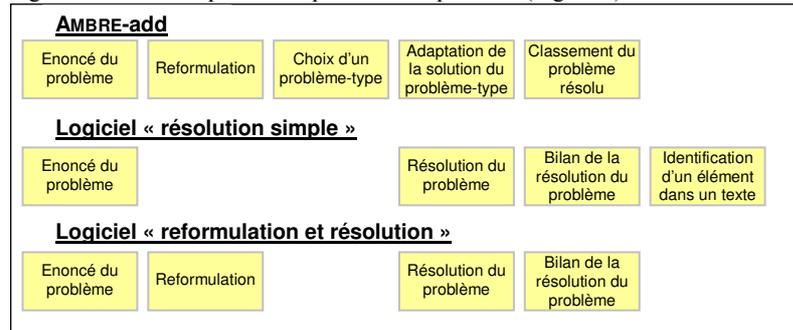


Figure 4 : Logiciels utilisés pour évaluer l'impact de AMBRE-add dans une approche comparative

Le premier logiciel contrôle (« résolution simple ») présentait d'abord des problèmes résolus puis, contrairement à AMBRE-add, proposait de résoudre directement le problème sans le reformuler ou identifier un problème proche. Après la résolution, un écran présentait à l'apprenant un bilan du problème. La comparaison entre AMBRE-add et ce logiciel avait pour but de vérifier si AMBRE-add facilite davantage l'apprentissage de la méthode qu'un logiciel proposant une résolution simple. Comme ce dernier logiciel comportait moins d'étapes que les autres logiciels, une étape complémentaire a été ajoutée pour limiter le nombre de problèmes résolus par l'élève pendant une séance avec ce logiciel. Le second logiciel contrôle (« reformulation et résolution ») présentait des problèmes résolus puis demandait à l'apprenant de reformuler le problème avant de le résoudre et de proposer un bilan de la résolution. Contrairement à AMBRE-add, ce système ne proposait pas de choisir et d'adapter un problème-type. La comparaison de l'impact sur l'apprentissage de ce logiciel avec l'impact de AMBRE-add devait permettre d'évaluer si le choix d'un problème type et son adaptation avaient un impact réel sur l'apprentissage, ou si l'étape de reformulation peut expliquer à elle-seule l'impact du système.

L'impact des différents logiciels sur l'apprentissage de méthode était mesuré à travers différents tests. La capacité à reconnaître que des problèmes appartiennent à la

même classe était testée à la fin de chaque séance à l'aide d'une tâche de « détection des similarités de structure » (Thibodeau, Dufresne & Mestre, 1989). Celle-ci consistait à lire un premier problème puis à sélectionner parmi deux autres problèmes celui qui se résolvait de la même manière que le premier. La capacité à associer une équation à une classe de problèmes était testée lors de la dernière séance en représentant un exemplaire d'une classe de problème par une reformulation, puis en demandant à l'apprenant d'écrire l'équation correspondante. Un test papier-crayon consistant à résoudre des problèmes faciles et difficiles avait pour but de mesurer si le logiciel avait un impact dans une situation papier-crayon et si l'utilisation de AMBRE-add facilitait la résolution de problèmes difficiles. Ce test était présenté avant l'utilisation des logiciels, après la quatrième séance, immédiatement après la dernière séance et un mois après.

Méthodes qualitatives

Nous avons complété cette approche expérimentale par des méthodes qualitatives telles que des observations, un questionnaire et des entretiens afin d'observer l'utilisation effective du logiciel et de prendre en compte les interactions entre apprenants et encadrants. Afin d'évaluer l'utilisation de AMBRE-add, nous avons prévu d'observer individuellement les élèves au cours de la séance d'utilisation du logiciel à l'aide d'une fiche d'observation incitant à décrire le comportement des apprenants durant chaque étape du logiciel, à noter leurs réactions face aux messages d'aide et de diagnostic et les difficultés rencontrées. Par ailleurs, les traces d'interaction étaient enregistrées et comportaient des informations telles que le nombre de problèmes résolus, le temps passé sur chaque étape ou encore le type d'erreurs rencontrées. Après la dernière séance un questionnaire et un entretien collectif étaient proposés afin de connaître l'avis subjectif des apprenants sur les logiciels, tel que les difficultés ressenties ou leur satisfaction, et de recueillir des données personnelles, telles que la fréquence d'utilisation d'un ordinateur.

En outre, afin de prendre en compte les interactions entre encadrants et apprenants, nous notions les différentes questions posées par les apprenants, les difficultés rencontrées lorsqu'ils faisaient appel aux encadrants et le type d'aide proposé par les encadrants.

Principaux résultats

L'analyse des tests de résolution de problèmes montre une amélioration des performances après l'utilisation des logiciels. Toutefois, on n'observe pas de différence de performances suivant le logiciel utilisé ; AMBRE-add semble avoir un impact équivalent aux logiciels contrôles.

Cette absence de différence peut s'expliquer de différentes manières. D'abord, les traces d'interaction des différents logiciels montrent que les enfants résolvent moins de problèmes avec AMBRE-add qu'avec le logiciel résolution simple (9 problèmes

versus 14 problèmes résolus en moyenne durant 6 séances). De plus, les questions posées durant l'utilisation de ces deux logiciels et les réponses données par les encadrants n'étaient pas de même nature, ce qui a pu biaiser la comparaison. En outre, le nombre de problèmes résolus avec AMBRE-add durant chaque séance est relativement faible (moins de 2 problèmes résolus par séance). L'observation de l'utilisation montre que les enfants rencontrent beaucoup de difficultés lors de l'utilisation du logiciel. Les élèves ont en particulier éprouvé des difficultés à lire les instructions et les différents messages ainsi qu'à identifier les raisons de leurs erreurs, ce qui a souvent limité leur progression. Les différentes difficultés observées ainsi que le faible nombre de problèmes résolus par séance nous ont conduites à conclure que AMBRE-add était trop complexe pour des enfants de CE1.

Intérêts et limites des méthodes utilisées durant cette évaluation

Les techniques d'évaluation utilisées ont-elles permis de nous informer sur l'impact du logiciel sur l'apprentissage et l'utilisation du logiciel ? Les données ainsi récoltées ont-elles permis de faire évoluer le logiciel dans le cadre d'une conception itérative ? Cette section propose une revue critique des apports et limites des différentes méthodes utilisées, par rapport à l'objectif visé dans le cadre d'une conception itérative.

Évaluation de l'impact de AMBRE-add sur l'apprentissage. Comme nous l'avons présenté précédemment, les différents tests proposés montrent que AMBRE-add semble avoir un impact équivalent aux autres logiciels contrôle. Cependant, le choix de la condition contrôle semble ici poser problème. En effet, comme le montrent les traces d'interaction, le nombre de problèmes résolus n'est pas le même suivant le logiciel utilisé, et ceci malgré la tâche complémentaire visant à compenser le temps passé dans l'étape de reformulation. Par ailleurs, les questions posées, et les aides fournies par les encadrants n'étaient pas de même nature. Dans le logiciel AMBRE-add, l'aide consistait le plus souvent à expliciter ou à reformuler les messages de diagnostic. En revanche, lors de l'utilisation du logiciel résolution simple, l'aide consistait souvent à aider l'apprenant à comprendre l'énoncé, et se rapprochait du guidage proposé par l'étape de reformulation. Afin de préserver l'équivalence entre systèmes nous avons proposé des messages d'aide et d'explication proches de ceux proposés dans AMBRE-add, mais cette expérience montre que ces messages n'étaient pas suffisants dans le cadre d'une résolution simple.

Les données quantitatives récoltées à l'issue de l'utilisation des différents logiciels ne nous permettent donc pas de montrer que AMBRE-add améliore davantage l'apprentissage de la méthode qu'un autre système. En outre, ces données, prise isolément, ne sont pas suffisamment informatives pour proposer de nouvelles recommandations dans le cadre d'une conception itérative. Qu'en est-il des autres données récoltées ?

Utilisation de AMBRE-add. Comme nous l'avons décrit précédemment, nous avons prévu d'observer individuellement les élèves au cours de la séance d'utilisation du logiciel à l'aide de fiche d'observation définissant différents critères à examiner. Cela n'a pas été réellement possible, seuls quelques enfants ont été observés de façon approfondie durant les deux dernières séances. En effet, durant les premières séances, les encadrants ont été très sollicités par les apprenants. Lors de l'analyse nous avons néanmoins pu récolter des informations sur l'utilisation du logiciel à travers la description des difficultés rencontrées faites par les encadrants lorsqu'ils intervenaient. Cette source d'information a permis d'identifier certaines erreurs et certaines difficultés récurrentes qui faisaient obstacle à l'utilisation du logiciel par les enfants. Par ailleurs, l'observation de ces enfants en difficulté a conduit à identifier certaines utilisations du logiciel assez différentes de l'utilisation prescrite, tel que des comportements d'essais-erreurs.

Les traces d'interaction ont pu compléter ces observations en apportant certaines données quantitatives concernant par exemple le nombre de problèmes résolus ou le temps passé par étape.

Ainsi, même si les informations concernant l'utilisation du logiciel ne reflètent pas l'utilisation faite par chacun des apprenants en raison de la manière dont elles ont été obtenues, elles mettent tout de même en lumière un nombre important de difficultés rencontrées par les CE1. Ces observations mettent en évidence un défaut d'acceptabilité du logiciel du fait qu'il n'est pas exactement adapté aux caractéristiques des utilisateurs.

Dans le cadre de la conception itérative, ces informations nous ont conduit à reconsidérer le public auquel AMBRE-add est destiné et nous a amenées à proposer différentes recommandations pour faire évoluer le logiciel. Cette évaluation nous a conduit à modifier les messages d'aide et de diagnostic, ou encore à ajouter un compagnon et un synthétiseur vocal destiné à assister la lecture. Par ailleurs, des activités plus simples seront proposées dans l'EIAH afin de préparer les enfants à l'utilisation du logiciel ou de leur proposer des activités de remédiation.

Enfin, les informations concernant les interactions entre encadrants et apprenants se sont révélées très utiles tant pour comprendre les différentes questions posées lors de l'utilisation des différents logiciels que pour identifier les difficultés rencontrées, comme nous l'avons présenté précédemment. Notons cependant que les informations concernant les interventions des encadrants n'ont pas été notées systématiquement, et dépendaient de la subjectivité de chacun. Aussi les phénomènes observés ne sont pas quantifiables, ce qui limite la portée de ces données.

En résumé, cette seconde évaluation combinant des méthodes quantitatives destinées à évaluer l'impact de AMBRE-add sur l'apprentissage à des méthodes qualitatives destinées à prendre en compte l'utilisation des logiciels, et le contexte dans lequel cette utilisation se déroulait n'a pas permis de montrer qu'AMBRE-add facilitait

l'apprentissage de méthodes davantage qu'un autre système. Cependant, les méthodes d'observation, et l'analyse de certaines traces d'interaction ont permis d'identifier les difficultés rencontrées par des élèves de CE1, représentatifs d'une classe normale en situation réelle d'utilisation. Certaines difficultés, erreurs ou modes d'utilisation avaient été rencontrées durant l'observation en laboratoire, d'autres non. L'observation de l'utilisation a donc permis d'observer l'évolution de l'utilisation du logiciel, de préciser les difficultés persistantes dans le temps et de faire évoluer le logiciel dans le cadre d'une conception itérative.

Cependant, du fait de la conception et de l'utilisation de logiciels contrôlés, cette évaluation a nécessité de tester le logiciel auprès d'un nombre d'enfants important et a été très coûteuse en temps de préparation (implémentation des logiciels contrôlés), de passation, et d'analyses alors que c'est l'observation de l'utilisation et des difficultés rencontrées qui a apporté le plus d'information pour la conception.

Aussi, à l'issue de cette expérience, il nous semble qu'après l'évaluation en laboratoire, il est préférable de réaliser une observation longitudinale de l'utilisation du logiciel en situation réelle avant d'entreprendre une évaluation de l'impact du logiciel sur l'apprentissage.

Observation en classe de CE2

Les problèmes additifs à une étape tels que ceux présentés par AMBRE-add sont introduits en classe de CE1 et continuent à être traités en classe de CE2, les enfants de ce niveau ayant encore des difficultés à modéliser ces problèmes. Comme AMBRE-add semble difficile à utiliser par des CE1, nous avons souhaité tester si AMBRE-add était plus adapté aux élèves de CE2, ceux-ci ayant moins de difficultés de lecture et étant plus autonomes. Pour le vérifier, nous avons réalisé une expérience avec des élèves de ce niveau afin d'observer la manière dont ils utilisent AMBRE-add. À travers cette expérience, nous voulions observer si l'utilisation effective du logiciel correspondait à l'utilisation prescrite et identifier les difficultés rencontrées par ce public.

Méthode

Dans cette expérience, 21 élèves en fin de CE2 ont utilisé AMBRE-add individuellement, dans la salle informatique de l'école, durant quatre séances de 45 minutes. Durant ces séances, nous avons observé l'utilisation effective du logiciel. Au cours de chaque séance, l'un des encadrants observait plusieurs enfants tour à tour en notant sur une fiche d'observation des informations concernant la réalisation de chaque étape, la réaction par rapport à l'aide et au diagnostic, ainsi que les différentes difficultés rencontrées. Durant les quatre séances, tous les enfants étaient observés au moins une fois. À l'issue de la quatrième séance, chaque élève était interrogé sur la stratégie employée dans l'étape d'adaptation (étape centrale dans le cycle AMBRE). De plus, nous avons recueilli certaines traces d'interactions. Enfin, à l'issue de la quatrième séance, nous avons fait remplir aux enfants un questionnaire afin d'identifier

les difficultés perçues par les apprenants et afin d'évaluer leur satisfaction ; nous avons ajouté aux questionnaires des questions relatives au vocabulaire utilisé dans l'interface afin de vérifier qu'il était bien compris.

Résultats

Cette observation systématique montre d'abord que, contrairement aux élèves de CE1, les élèves de CE2 n'ont pas eu de difficulté à lire et à comprendre les consignes et les messages d'aide et de diagnostic ou à mettre en œuvre des techniques opératoires. De plus, ils sont très autonomes, prennent en main le logiciel rapidement et mettent moins de temps que les élèves de CE1 pour résoudre un problème.

L'observation de l'utilisation du système et l'entretien individuel ont également permis de mettre en évidence des différences d'utilisation importantes entre élèves de CE1 et élèves de CE2 concernant la réalisation des différentes étapes, le comportement face au logiciel, ou la manière d'utiliser les messages d'aide et d'explication. Cependant, cette observation a montré que certaines difficultés étaient toujours présentes lors de l'utilisation du logiciel par des élèves de CE2. L'analyse de ces difficultés a conduit à concevoir un tutoriel mieux adapté aux apprenants, qui présente davantage comment naviguer dans le logiciel et facilite la compréhension du principe général du logiciel.

Intérêts et limites de cette évaluation

Dans le cadre de cette évaluation, nous avons limité le nombre de méthodes employées à l'observation individuelle, l'analyse de traces d'interaction et de questionnaires. Sur le plan logistique, ce dispositif a l'avantage de pouvoir être mis en place facilement. De plus, cette méthode ne nécessite pas d'avoir un nombre de participants important. En outre, il permet de décrire relativement finement des différentes manières d'utiliser le logiciel suivant des dimensions choisies préalablement et les difficultés rencontrées par les apprenants. Il permet aussi de prendre en compte des variations individuelles. Néanmoins, ce dispositif ne permet ni de quantifier les différents comportements observés, ni d'observer l'évolution de l'utilisation du logiciel par un apprenant au cours du temps. Cette difficulté pourrait être palliée en étant capable de mettre en relation les comportements observés par les observateurs avec les traces d'interactions enregistrées. Ainsi, l'analyse des traces pourrait ensuite permettre de détecter automatiquement différents comportements et de pouvoir ainsi les quantifier. Plus simplement Baker, Corbett, Koedinger et Wagner (2004) proposent d'adapter des méthodes d'observation quantitative (Lloyd & Loper 1986) afin de quantifier différents comportements prédéfinis en observant de manière aléatoire chaque élève sur plusieurs par séance.

Évaluation comparative en classe de CE2

Les observations encourageantes réalisées avec des élèves de CE2 et l'intérêt que présentent pour les enseignants les problèmes traités et la démarche proposée par le

logiciel nous ont amenées à tester l'impact de AMBRE-add sur l'apprentissage auprès de ce public. Pour cela nous avons réalisé une étude comparative conjuguant à des méthodes d'observation et à une analyse des traces d'interaction. Dans cette évaluation, la condition contrôle consistait à ne pas travailler sur les problèmes additifs, mais à réaliser une autre activité en classe avec l'enseignant.

Méthode

Cinquante élèves de CE2 appartenant à deux écoles différentes ont utilisé AMBRE-add en salle informatique durant quatre séances de 45 minutes. Durant cette évaluation, les enfants passaient d'abord un premier test (T1) consistant à résoudre six problèmes additifs. Ensuite, dans chaque classe, la moitié des enfants utilisait AMBRE-add en salle informatique durant quatre séances tandis que l'autre moitié restait en salle de classe avec l'enseignant pour réaliser une autre activité sans rapport avec la résolution de problèmes. À l'issue de ces quatre séances, les élèves passaient un second test (T2), constitué de six problèmes de même nature que dans le premier test. Ensuite, les deux groupes étaient inversés : les enfants n'ayant pas encore utilisé le logiciel utilisaient le logiciel pendant quatre séances tandis que les autres restaient en classe (Figure 5). A l'issue des ces 4 séances, un post-test consistant à résoudre de nouveau résoudre 6 problèmes additifs était proposé (T3) Par ailleurs, durant chaque session, plusieurs apprenants étaient observés ; de plus différents comportements prédéfinis étaient quantifiés en observant chaque élève dans un ordre aléatoire trois fois par séance. Enfin, un questionnaire était proposé aux élèves après la dernière séance d'utilisation.

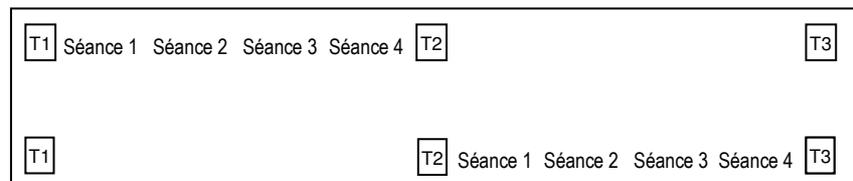


Figure 5 : déroulement de l'expérience évaluant l'utilisabilité du logiciel en CE2

Résultats

Les données issues de l'observation ont confirmé les données observées dans l'expérience précédente. La quantification des comportements a montré que le nombre d'élèves adoptant un comportement d'essai-erreur était limité. Les résultats quantitatifs ont d'abord montré que les performances lors du pré-test étaient très variables. Nous avons comparé les performances du groupe ayant utilisé AMBRE-add au groupe contrôle et les performances de chaque élève en situation d'utilisation du logiciel et en situation contrôle. Les résultats montrent que le premier groupe qui utilise AMBRE-add a de meilleures performances que le second groupe après utilisation du logiciel. Cependant cette différence n'est pas reproduite lorsque le second groupe utilise le logi-

ciel ; on observe que le second groupe a de meilleures performances après utilisation du logiciel, mais que cette différence n'est pas significative. Une analyse plus fine montre que la plupart des élèves qui ne progressent pas résolvent déjà cinq problèmes sur six lors du pré-test. Par ailleurs, une confrontation des performances en résolution de problèmes avec les traces d'interaction et les données relatives au comportement montre que le nombre de problèmes résolus durant l'utilisation du logiciel n'est ni corrélé aux performances lors du pré-test, ni corrélé au comportement : certains élèves adoptant un comportement d'essai-erreur résolvent beaucoup de problèmes tandis que d'autres en résolvent très peu.

Cette analyse montre donc qu'AMBRE-add a un impact sur l'apprentissage, mais que cet impact dépend des connaissances préalables des apprenants.

Intérêts et limites

Cette expérience met en évidence un effet de AMBRE-add sur l'apprentissage qui dépend des connaissances préalables de l'apprenant. À terme, les informations relevées dans cette expérience devraient pouvoir être utilisées pour construire un modèle de l'apprenant qui permette d'adapter le système aux connaissances de l'apprenant.

Sur le plan méthodologique, montrer que AMBRE-add a un impact sur l'apprentissage en le comparant à une situation neutre était un préalable à la comparaison à un logiciel contrôle qui permettra de valider le principe du cycle AMBRE. Ce point de comparaison permet de s'assurer que l'amélioration observée provient bien de l'utilisation du logiciel et non de l'habituation aux tests proposés. Par ailleurs, l'utilisation d'observations quantitatives permet de compléter les observations qualitatives et les traces d'interaction et permet de quantifier les différentes stratégies observées.

Ainsi, nous avons décrit les différentes évaluations réalisées pour évaluer l'utilisabilité du logiciel puis son utilité par des élèves de CE1 puis l'usage qui en était fait par des élèves de CE2 en présentant les intérêts et les limites de chaque méthode suivant le stade de conception du logiciel. Au cours de ces différentes évaluations, nous avons également récolté différentes informations concernant l'acceptabilité du logiciel à partir de l'analyse des usages, des questionnaires et des interviews. La section suivante synthétise les leçons tirées de cette expérience de conception et propose une démarche d'évaluation.

5. Quelle démarche d'évaluation mettre en œuvre dans le cadre d'une conception itérative ?

Leçons tirées la conception itérative de AMBRE-add

Dans le cadre de la conception itérative de AMBRE-add, nous avons conduit différentes évaluations afin d'évaluer l'utilisabilité du logiciel puis son utilité, c'est-à-dire

son impact sur l'apprentissage. Nous avons d'abord procédé à une évaluation de l'utilisabilité par inspection, puis nous avons observé cinq enfants en laboratoire suivant la procédure proposée par Nielsen (1993). Celle-ci a permis de mettre en évidence les plus grosses difficultés d'utilisabilité et a permis d'identifier une difficulté d'ordre mathématique non détectée par les conseillers pédagogiques avec qui nous avons travaillé. En revanche, le fait que nous n'observions qu'une séance d'utilisation n'a pas permis de différencier les difficultés de prise en main du logiciel de celles qui persistent. En outre, les enfants volontaires pour participer (plutôt bons élèves aimant les mathématiques) n'étaient pas représentatifs de l'ensemble des enfants de leur niveau.

Nous avons ensuite réalisé une seconde évaluation afin de tester l'impact du cycle AMBRE proposé par le logiciel sur l'apprentissage en comparant AMBRE-add avec deux logiciels contrôle. L'utilisation des logiciels avait lieu dans la salle informatique de l'école durant six séances d'une demi-heure chacune. Il est apparu que cette évaluation était trop ambitieuse à ce niveau de conception. En effet, l'observation de l'utilisation du logiciel a mis en évidence des difficultés d'utilisation, ainsi que des formes d'utilisations non prévues qui pouvaient laisser présager de l'impact limité de Ambre-add sur l'apprentissage. Aussi, avant d'évaluer l'impact du cycle AMBRE sur l'apprentissage, il aurait été préférable d'observer l'utilisation du logiciel en situation réelle et de s'assurer de l'influence du logiciel lui-même sur l'apprentissage en le comparant à une situation neutre. Néanmoins, à travers les données qualitatives et les traces d'interaction, cette évaluation nous a apporté de nombreuses informations sur l'utilisation du logiciel et les difficultés rencontrées par les élèves lors de cette utilisation qui ont conduit à modifier différents aspects du logiciel.

Nous avons ensuite conduit deux évaluations auprès d'élèves en classe de CE2. Ces évaluations se sont également déroulées à l'école, en salle informatique, durant les heures de cours. Ces évaluations ont été plus légères à mettre en place mais ont néanmoins permis de recueillir des données intéressantes concernant les différents usages du logiciel. De plus, la seconde évaluation, qui consistait à comparer l'utilisation de AMBRE-add à une situation neutre a permis de mesurer l'impact du logiciel lui-même sur l'apprentissage et a donné des éléments permettant de caractériser dans quelles situations l'utilisation du logiciel était associé à une amélioration des performances.

Ainsi, il apparaît qu'après avoir évalué l'utilisabilité suivant les méthodes classiques développées en IHM (Interaction Homme-Machine) et adaptées aux EIAH, l'évaluation de l'impact du logiciel conduite en CE1 était peut-être trop précoce. En effet, l'observation de l'utilisation du logiciel a montré qu'il était trop complexe pour des élèves de CE1. Une analyse des usages sur plusieurs séances, moins coûteuse à

mettre en œuvre qu'une évaluation comparative, aurait également permis de mettre en évidence ce problème.

En outre, il apparaît que parmi les données recueillies durant cette évaluation comparative, les plus informatives pour la conception concernaient l'analyse des usages et de l'évolution de ces usages au cours du temps. Cette analyse a apporté des informations sur le temps nécessaire pour que l'EIAH soit pris en main, sur les difficultés persistantes ou encore sur l'utilisation effective du logiciel. La prise en compte des usages est donc très informative au cours de la conception itérative des EIAH. L'intérêt de l'analyse des usages avait déjà été souligné en ergonomie cognitive (voir par exemple Béguin et Rabardel, 2000, Decortis et al., 2001). Néanmoins, dans le domaine des EIAH, l'analyse des usages est souvent réservée à des logiciels déjà commercialisés ou à des sites web déjà accessibles aux apprenants (voir par exemple Trouche, 2002 ; De Vries et Tricot, 1998). L'analyse des usages en conception n'est prise en compte que depuis peu de temps (voir Hautecouverture et al., 2004 ; Cottier et Choquet, 2005) et est pour l'instant peu répandue.

Concernant l'évaluation de l'utilité, nous avons également été confrontées à des difficultés pour le choix de la condition contrôle. Comme nous l'avons décrit précédemment, le logiciel contrôle utilisé comme point de comparaison a été utilisé assez différemment de AMBRE-add, ce qui limite la portée des résultats obtenus. De plus, il semble qu'une situation neutre soit une meilleure condition contrôle pour avoir des indications sur l'utilisabilité du logiciel. En effet, la comparaison à une situation neutre permet de mesurer l'impact du logiciel lui-même sur l'apprentissage tout en s'assurant que la progression observée n'est pas due à une habitude aux tests ou à un facteur extérieur. À partir de ces différentes remarques, nous proposons une démarche d'utilisation en conception.

Proposition d'une démarche d'évaluation en situation de conception itérative

À partir de l'expérience que nous avons acquise lors de l'évaluation de AMBRE-add, nous proposons une démarche préconisant des évaluations successives dans le cadre de la conception itérative d'un EIAH. Cette démarche propose de combiner différentes méthodes et d'alterner différents types d'évaluations au cours du cycle de conception itérative afin de prendre en compte à la fois l'utilisabilité, l'utilité du système et son usage en situation au cours de la conception¹.

Après avoir précisé les spécifications d'un premier prototype et l'avoir implémenté, une première étape consiste à faire une évaluation analytique du logiciel afin d'évaluer son utilisabilité. Dans le cadre de la conception des EIAH, cette analyse

¹ Cette démarche d'évaluation devrait être plus particulièrement adaptée aux EIAH destinés à être utilisés individuellement dans une situation de formation en présence.

peut être faite par des experts en IHM, en didactique ainsi que par des experts en pédagogie.

Ensuite une seconde étape consiste à évaluer empiriquement l'utilisabilité du prototype implémenté à l'aide des méthodes présentées précédemment (cf. section 2). Comme le décrit Nielsen (1993), cette évaluation peut se dérouler en laboratoire en observant et en interrogeant cinq utilisateurs représentatifs. La mise en évidence des principaux défauts d'utilisabilité peut conduire à de nouvelles spécifications et à une modification du prototype. Cette évaluation peut être répétée plusieurs fois si la première évaluation conduit à des modifications importantes du logiciel.

L'étape suivante consiste à observer et à analyser les usages du logiciel en situation réelle d'utilisation. En effet, avant d'évaluer l'impact du logiciel sur l'apprentissage, il nous semble nécessaire d'observer la manière dont les apprenants se l'approprient (observation de la genèse instrumentale du logiciel, Rabardel, 1995) et l'utilisent effectivement. L'analyse des usages peut apporter des informations sur l'utilisabilité (temps nécessaire pour que l'EIAH soit pris en main, difficultés persistantes), l'acceptabilité (par exemple l'évolution de la motivation ou la compatibilité avec l'organisation du temps et des lieux) et sur les usages du logiciel, c'est-à-dire l'utilisation effective du logiciel, l'évolution des activités réalisées durant l'utilisation au fur et à mesure des séances, ou encore sur des usages non prévus par le concepteur. Elle peut également permettre de prendre en compte les variations interindividuelles dans l'utilisation du logiciel. L'analyse des usages peut être réalisée à partir d'observations et éventuellement de l'analyse des traces d'interaction. Cette analyse des usages peut conduire à des modifications du système et éventuellement à la constitution d'un scénario pédagogique pour faciliter la prise en main du logiciel et orienter son utilisation (Hautecouverture et al., 2004 ; Trouche, 2002). Elle pourra être répétée au cours du cycle de conception, afin de « raffiner » la conception du produit à travers des observations d'usages régulières (Hautecouverture et al., 2004).

L'analyse des usages peut éventuellement être combinée à une analyse de l'impact du logiciel sur l'apprentissage à l'aide d'une méthode comparative. Pour mettre en œuvre cette méthode, une question importante concerne la condition à laquelle comparer le logiciel utilisé. Il semble que pour évaluer l'utilité du logiciel stricto sensu (le système conçu a-t-il une influence sur l'apprentissage ?) la comparaison de l'EIAH à une situation sans intervention (cf. Ainsworth, Wood et O'Malley, 1998) apporte des informations intéressantes. En effet, sous réserve que les tests proposés soient adaptés à l'apprentissage visé, si un effet du logiciel est observé, cette situation pourra permettre de vérifier que l'effet obtenu n'est pas du à la répétition des tests. En outre, une analyse individuelle de l'évolution des performances en fonction des comportements observés ou enregistrés par les traces d'interaction peut permettre de mieux comprendre quel type d'utilisation du logiciel est associé à un apprentissage.

Une analyse comparative peut également être mise en œuvre pour atteindre d'autres objectifs : elle peut permettre de tester l'effet d'une fonction particulière du logiciel en comparant l'EIAH avec une autre version du logiciel (Aleven et al., 1999 ; 2002 ; Lucken et al., 2001).

Enfin, lorsque les concepteurs estiment que l'utilisabilité, l'utilité et l'acceptabilité sont satisfaisantes, le logiciel pourra être validé en comparant son efficacité à l'efficacité d'un enseignement classique dispensé par un enseignant (cf. Shute et Glaser, 1990 ; Koedinger et al., 1997 ; Meyer et al., 1999), ou à l'efficacité d'autres systèmes.

Conclusion

Devant la multiplicité des objectifs à évaluer et des méthodes proposées, quelle démarche d'évaluation mettre en œuvre en conception ?

Nous avons été confrontées à cette question lors de la conception de l'EIAH AMBRE-add, et, pour y répondre, nous nous sommes appuyées sur les études existantes afin de mettre en place plusieurs évaluations combinant différentes méthodes existantes afin d'évaluer l'utilisabilité et l'utilité du logiciel. Dans cet article, nous avons décrit les méthodes existantes puis nous avons procédé à une revue critique des différentes évaluations conduites lors de la conception de AMBRE-add. Nous avons ainsi mis en évidence les apports à la conception et les limites des différentes méthodes utilisées. Cette analyse critique nous a conduites à mettre en évidence l'intérêt d'analyser les usages très tôt au cours de la conception et à proposer une démarche d'évaluation en situation de conception. Ainsi, nous proposons d'abord d'évaluer l'utilisabilité de l'EIAH en utilisant les méthodes décrites en IHM puis d'analyser les usages en situation réelle d'utilisation pendant plusieurs séances afin de compléter l'évaluation de l'utilisabilité, d'identifier les différentes formes d'usage du logiciel et de récolter des données sur l'acceptabilité. Cette analyse des usages peut être combinée à une évaluation de l'impact de l'EIAH sur l'apprentissage afin d'identifier les formes d'usages associées à un apprentissage. Chacune de ces évaluations pourra conduire à produire des spécifications pour implémenter une nouvelle version du logiciel dans le cadre d'une conception itérative.

BIBLIOGRAPHIE

Ainsworth, Wood et O'Malley, 1998

AINSWORTH S. E., WOOD D., O'MALLEY C. (1998). There is more than one way to solve a problem: Evaluating a learning environment that supports the development of children's multiplication skills, *Learning and Instruction*, Vol 8 n°2, pp. 141-157.

Aleven et al., 2002

ALEVEN V., KOEDINGER K. R. (2002). An Effective Meta-cognitive Strategy: Learning by Doing and Explaining with a Computer-Based Cognitive Tutor. *Cognitive Science*, Vol 26 n° 2, pp. 147-179.

Aleven et al., 1999

ALEVEN V., KOEDINGER K. R., CROSS K. (1999). Tutoring answer explanation fosters learning with understanding. In *Proceedings of AIED-99*, Amsterdam, Netherland, IOS Press, pp. 199-206.

Baker, Corbett, et Koedinger, 2004

BAKER R.S., CORBETT A.T., KOEDINGER K.R. (2004). Detecting Student Misuse of Intelligent Tutoring Systems. In *Proceedings of International Conference on Intelligent Tutoring Systems (ITS'2004)*, Maceio, Brasil, Springer, pp. 531-540.

Bastien et Scapin, 1993

BASTIEN C., SCAPIN D. (1993). Critères ergonomiques pour l'évaluation des interfaces utilisateurs, *RT n°156, INRIA*.

Barfurth et al. 1994

BARFURTH M.A., BASQUE J., CHOMIENNE M., WINER, L.R. (1994). Les instruments de collecte de données de recherche qualitative dans des environnements pédagogiques informatisés. In *Apprendre dans des environnements pédagogiques informatisés*, Bordeleau, P. eds, Editions Logiques, pp. 485-548.

Béguin et Rabardel, 2000

BEGUIN P., RABARDEL P. (2000). Concevoir des activités instrumentées. *Revue d'Intelligence Artificielle*, Vol 14, pp. 35-54.

Caverni, 1988

CAVERNI J.P. (1988). La verbalisation comme source d'observables pour l'étude du fonctionnement cognitif. In *Psychologie cognitive, modèles et méthodes*. Caverni, J.P., Bastien, C., Mendelsohn, P., Tiberghien, G. eds., Presse Universitaire de Grenoble, pp. 253- 273.

Cottier et Choquet, 2005

COTTIER P., CHOQUET C. (2005). De l'usager construit à l'usager participant. In *actes de EIAH'2005*, Montpellier, France, pp. 449-454.

Decortis et al., 2001

DECORTIS F., DAELE L., POLAZZI L., RIZZO A., SAUDELLI B. (2001). Nouveaux instruments actifs et activités narratives. *Revue d'interactions homme-machine*, Vol 2 n°2, pp. 1-30.

De Vries, 2001

DE VRIES E. (2001). Les logiciels d'apprentissage, panoplie ou éventail ? *Revue Française de Pédagogie*, Vol 137, pp. 105-116.

De Vries et Tricot, 1998

DE VRIES E., TRICOT A. (1998). Évaluer l'utilisation d'hypermédiat : intérêts et limites des variables de performance. *Hypertextes et Hypermédiat*, n° hors série, pp. 175-190.

Dubourg et Teutsch, 1997

DUBOURG X., TEUTSCH, P. (1997). Interface Design Issues in Interactive Learning Environments. In proceedings of IFIP WG 3.3 Working Conference, Human-Computer Interaction and Educational Tools, Sozopol.

Dubourg, Delozanne et Grugeon, 1995

DUBOURG X., DELOZANNE E., GRUGEON B. (1995). Situations d'interaction dans un environnement d'apprentissage : le système Repères. In *actes des Quatrièmes Journées ELAO*, Cachan, Eyrolles eds, pp. 223-244.

Farenc, 1997

FARENC N. (1997). ERGOVAL : Une méthode de structuration des règles ergonomiques permettant l'évaluation automatique d'interfaces graphiques. Thèse en informatique, Université Toulouse I.

Fasse et Kolodner, 2000

FASSE B.B., KOLODNER J.L. (2000). Evaluating Classroom Practices Using Qualitative Research Methods: Defining and Refining the Process. In *Proceedings of International Conference of the Learning Sciences*, pp. 93-198.

Gagné, Briggs et Wager, 1988

GAGNE R.M., BRIGGS L.J., WAGER W.W. (1988). *Principles of instructional design*. New York, Holt, Renhart and Winston Eds.

Greeno et Riley, 1987

GREENO J.G., RILEY M.S. (1987). Processes and development of understanding. In *Metacognition, motivation and understanding*, Weinert F.E. & Kluwe R.H. (Eds.), pp. 289-313.

Guin, 1991

GUIN D. (1991). La notion d'opérateur dans une modélisation cognitive de la compréhension des problèmes additifs. *Math. Inf. Sci. Hum.* Vol. 113, pp. 5-33.

Guin-Duclosson, Jean-Daubias et Nogry, 2002

GUIN-DUCLOSSON N., JEAN-DAUBIAS S., NOGRY S. (2002). The Ambre ILE: How to Use Case-Based Reasoning to Teach Methods. In *proceedings of International Conference on Intelligent Tutoring Systems (ITS'2002)*, Biarritz, France, Springer, pp. 782-791.

Hautecouverture et al., 2004

HAUTECOUCOVERTURE J.-C., GREGORI N., PAQUELIN D., CHAROY F., GODART C., PATTEN M., FAUGERAS I. (2004). Analyse d'usage d'une plate-forme de coopération : Usage et développement logiciel. *Cahiers romans de sciences cognitives*, Vol 1 n°3, pp. 45-77.

Hoecker et Elias, 1986

HOECKER D., ELIAS G. (1986). User evaluation of the LISP intelligent tutoring system. In *Proceedings of the human factors society*, Vol 32 n°3, pp. 313-324.

Holbrook et al., 2001

HOLBROOK J., GRAY J., FASSE B.B., CAMP P.J., KOLODNER J.L. (2001). Managing complexity in classroom curriculum implementation sites: Triangulating Multi-Level Assessment and Evaluation. *Paper presentation in Building Sustainable Science Curriculum: Acknowledging and Accomodating Local Adaptation. American Educational Research Association, Seattle, WA.* (<http://www.cc.gatech.edu/projects/lbd/pubconf.html>)

Hû et Trigano, 1998

HU O., TRIGANO P. (1998). Propositions de critères d'évaluation de l'interface homme-machine des logiciels multimédias pédagogiques. In *proceedings of IHM'98*, Nantes.

Hû, Trigano et Crozat, 2001

HÛ O., TRIGANO P., CROZAT S. (2001). Une aide à l'évaluation des logiciels multimédias de formation. *Sciences et Techniques Educatives*, Vol 8, n°3-4.

Jean , 2000

- JEAN S. (2000). Application de recommandations ergonomiques : spécificités des EIAO dédiés à l'évaluation. In *Actes des Rencontres Jeunes Chercheurs en IHM 2000*, pp. 39-42.
- Koedinger et al., 1997
- KOEDINGER, K. R., ANDERSON, J. R., HADLEY, W. H., & MARK, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, Vol 8, pp. 30-43.
- Lewis et al., 1990
- LEWIS C., POLSON P.G., WHARTON C., RIEMAN J. (1990). Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces. In proceedings of. CHI'90: Human Factors in Computing Systems, ACM: New York, pp. 235-242.
- Lim, 1998
- LIMC.P., (1998). The effect of a computer-based learning support package on the learning outcome of low-performance economics students. *The virtual edition*, Vol 12 n°1.
- Litmann et Soloway, 1988
- LITMANN D., SOLOWAY E., 1988. Evaluating ITSs: The cognitive science perspective. In *Foundations of Intelligent Tutoring Systems*, M. Polson and J. J. Richardson eds., Hillsdale, NJ: LEA.
- Lloyd et Loper, 1986
- LLOYD J.W., LOPER A.B. (1986). Measurement and Evaluation of Task-Related Learning Behavior: Attention to Task and Metacognition. *School Psychology Review*, Vol.15 n°3, pp. 336-345.
- Luckin et al., 2001
- LUCKIN R., PLOWMAN L., LAURILLARD D., STRATFOLD M., TAYLOR J., (2001). Narrative evolution: learning from students' talk about species variation. *International Journal of AIED*, Vol 12, pp. 100-123.
- Marthe, 1982
- MARTHE P. (1982). Problèmes de type additif et appropriation par l'élève des groupes additifs Z et D entiers relatifs et décimaux relatifs. Thèse de doctorat, EHESS, Paris.
- Mark et Greer, 1993
- MARK M.A., GREER, J.E. (1993). Evaluation methodologies for intelligent tutoring systems. *Journal of Artificial Intelligence in Education*, Vol 4 n° 2/3, pp. 129-153.
- Mayer, 2001
- MAYER, R.E. (2001). *Multimedia learning*, Cambridge University Press.
- Meyer et al., 1999
- MEYER T. N., MILLER T. M., STEUCK K., KRETSCHMER M. (1999). A multi-year large-scale field study of a learner controlled intelligent tutoring system. In *Artificial Intelligence in Education*, S. Lajoie & M. Vivet (Eds.), Vol. 50, pp. 191-198.
- Nanard et Nanard, 1998
- NANARD J., NANARD M. (1998). La conception d'hypermédiat, In *Les hypermédiat, approches cognitives et ergonomiques*, A. Tricot et J.-F. Rouet (Eds.), Paris, Hermès, pp. 15-34.
- Nielsen, 1993
- NIELSEN J. (1993). *Usability Engineering*, Academic Press.
- Nielsen et Landauer, 1993
- NIELSEN, J. & LANDAUER, T.K. (1993). A mathematical model of the finding of usability problems. In *Proceedings of ACM INTERCHI'93 Conference*, Amsterdam, The Netherlands, pp. 206-213.

- Nogry et Didierjean, à paraître
- NOGRY S., DIDIERJEAN A. (à paraître). Apprendre à partir d'exemples : interactions entre présentation du matériel, activités des apprenants et processus cognitifs. L'année Psychologique.
- Nogry, Jean-Daubias et Duclosson, 2004
- NOGRY S., JEAN-DAUBIAS S., DUCLOSSON N. (2004). ITS evaluation in classroom: the case of the Ambre-awp. In *proceedings of ITS 2004*, Maceio, Brasil, springer eds, pp. 511-520.
- Rabardel, 1995
- RABARDEL P. (1995). *Les hommes et les technologies, approche cognitive des instruments contemporains*. Paris, Colin.
- Riley, Greeno et Heller, 1983
- RILEY M.S., GREENO J.G., HELLER J.I. (1983). Development of children's problem-solving ability in arithmetic. In *The development of mathematical thinking*, Ginsburg H.P. (Ed.). New-York : Academic Press.
- Rouet et Passerault, 1999
- ROUET J.-F., PASSERAULT J.-M (1999). Analyzing learner hypermedia interaction: An overview of online methods. *Instructional Science*, Vol 27, pp. 201–219.
- Sander et Richard, 1997
- SANDER E., RICHARD J.-F. (1997). Analogical transfer as guided by an abstraction process: the case of learning by doing text editing. *Journal of Experimental Psychology: Learning, Memory and Cognition*, Vol 23, pp. 1459-1483.
- Schneiderman (1992)
- SCHNEIDERMAN B. (1992). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Reading, MA: Addison-Wesley.
- Senach, 1993
- SENACH B. (1993). L'évaluation ergonomique des interfaces homme – machine. In *L'ergonomie dans la conception des projets informatiques*. J.-C. Sperandio eds. Octares editions: 69-122.
- Shute et Regian, 1993
- SHUTE V.J., REGIAN J. W. (1993). Principles for Evaluating intelligent Tutoring Systems. *Journal of Artificial Intelligence and Education*, Vol 4 n° 2/3, 245-271.
- Shute et Glaser, 1990
- SHUTE V. J., GLASER, R. (1990). A large-scale evaluation of an intelligent discovery world: Smithtown. *Interactive Learning Environments*, Vol 1, pp. 51-77.
- Squires et Preece., 1999
- SQUIRES D., PREECE J. (1999). Predicting quality in educational software: Evaluating for learning, usability, and the synergy between them. *Interacting with Computer*, Vol 11 n°5, pp. 467-483.
- Thibodeau, Dufresne et Mestre, 1989
- THIBODEAU HARDIMAN P., DUFRESNE R. & MESTRE J.P. (1989). The relation between problem categorization and problem solving among expert and novices. *Memory and Cognition*, Vol 17 n°5, pp. 627-638.
- Tricot et al., 2003
- TRICOT A., PLÉGAT-SOUTJIS F., CAMPS J.-F., AMIEL A., LUTZ, G., MORCILLO, A. (2003). Utilité, utilisabilité, acceptabilité : interpréter les relations entre trois dimensions de l'évaluation des EIAH. In *Environnements informatiques pour l'apprentissage humain*, Strasbourg, France, C. Desmoulins, P. Marquet & D. Bouhineau (Eds), pp. 391-402.

Tricot et Lafontaine, 2002

TRICOT A., LAFONTAINE J. (2002). Une méthode pour évaluer conjointement l'utilisation un outil multimédia et l'apprentissage réalisé avec celui-ci. *Le Français dans le Monde*, pp. 41-52.

Trouche, 2002

TROUCHE L. (2002). Génèses instrumentales, aspects individuels et collectifs. In, Calculatrices symboliques. In *Transformer un outil en un instrument du travail mathématique : un problème didactique*, Guin N. & Trouche L. (eds), la pensée sauvage édition, pp. 243-275.

Vergnaud, 1982

VERGNAUD G. (1982). A classification of cognitive tasks and operations of the thought involved in addition and subtraction problems. In Carpenter P.T., Moser, J.M., & Romberg, T.A. (eds.), *Addition and subtraction: A cognitive perspective*, Hillsdale, Erlbaum, pp. 39-58.