

Retour d'Expérience en Enseignement de la Modélisation de Tâches

Sybille Caffiau^{1,2}, Dominique L. Scapin², Loé Sanou¹

¹LISI

ENSMA - 1, Avenue Clément Ader
BP 40109, F-86961 Futuroscope Chasseneuil Cedex
{sybille.caffiau, loe.sanou}@ensma.fr

²INRIA

Domaine de Voluceau, Rocquencourt,
BP 105, F-78153 Le Chesnay
Dominique.Scapin@inria.fr

RESUME

L'enseignement de la conception de logiciel centrée utilisateur traite de nombreux aspects. Un des aspects les plus importants est celui de la modélisation des tâches. En plus de constituer une base indispensable à la conception, cette phase permet de se familiariser avec le domaine d'application, avec les utilisateurs, et de faciliter la communication entre les intervenants dans la conception (utilisateurs, experts métiers, concepteurs, etc.). La modélisation des tâches contribuant fortement à la qualité ergonomique et à l'acceptation du logiciel résultant, il est indispensable qu'elle soit la mieux comprise possible par les futurs concepteurs de logiciels. Dans ce but, cette étude cherche à évaluer à la fois la méthode d'enseignement utilisée, et son contenu, i.e., le modèle et son outil associé. Cet article décrit la problématique de l'étude, le modèle et le logiciel enseigné, les différentes phases de l'enseignement, et le retour d'expérience d'utilisation concrète du logiciel. Ce retour d'expérience identifie les apports et les inconvénients liés à cette utilisation sur le plan pédagogique, mais aussi du point de vue pratique, notamment en matière d'évolution des modèles obtenus et d'édition de ces modèles.

MOTS CLES : modèles de tâches, enseignement, évaluation.

ABSTRACT

Teaching user-centered design of software covers many aspects. One of the most important ones is the modelling of tasks. In addition to being a required basis for design, this phase allows becoming familiar with the application domain, the users, and facilitating communication with stakeholders (users, domain experts, designers, etc.). Task modelling contributing largely to the ergonomic quality and acceptance of the resulting software, it is essential that its concepts are well understood by future software designers. To this end, this study aims at evaluating both the teaching method used, and its contents, i.e. the model and its associated tool. This article describes the rationale of the study, the model and the software used, the teaching steps, and the experience feedback on the practical use of the software. The latter identifies the benefits and disadvantages of this practice use from a pedagogical point of view, but also from a practical

standpoint, including in terms of model evolution and model editing.

KEYWORDS: task model, teaching, evaluation

INTRODUCTION

Prendre en compte l'utilisateur dans la conception des systèmes interactifs est aujourd'hui une évidence [5]. De ce fait, il est nécessaire d'enseigner les méthodes pour y parvenir aux étudiants qui, une fois leurs études terminées, concevront les applications. Dans ce cadre, nous leur enseignons comment modéliser les activités humaines depuis plusieurs années. Cependant, concevoir des modèles de tâches décrivant des activités pour des applications complètes est complexe. De plus, la durée impartie à notre enseignement de la modélisation est limitée. Ces deux impératifs rendent indispensable la recherche de moyens rentables en temps et pédagogiquement profitables aux étudiants.

Dans ce but, depuis quelques années, nous utilisons un logiciel comme support lors de la mise en pratique de la modélisation des tâches. Parmi les formalismes de tâches orientés utilisateur et qui disposent d'outils logiciel dédiés et gratuits, nous en avons enseigné deux : CTT [11] (CTTE) et depuis 2007, K-MAD [10] (K-MADe [2]). Le but de cet article n'est ni d'expliquer ces choix, ni de comparer l'utilisation de ces deux formalismes mais d'exposer une étude que nous avons menée sur la manière dont un groupe d'étudiants modélise un cas et utilise un logiciel de modélisation comme support à l'enseignement. Nous avons réalisé cette étude en utilisant comme logiciel K-MADe. La principale raison de ce choix est le fait que K-MAD est issu de recherches visant à produire un noyau pour la modélisation des tâches et qu'il contient, par conséquent, l'ensemble des concepts minimaux nécessaire à une modélisation complète des tâches ; de plus le logiciel K-MADe est actuellement en phase d'évaluation et de refonte. Il s'agit donc, suite à l'observation des différentes phases d'apprentissage de la modélisation des tâches, de disposer de données sur l'enseignement de la modélisation, mais aussi de permettre l'amélioration de l'utilisation du logiciel support.

Dans cet article seront tout d'abord rapidement présentés le formalisme enseigné et le logiciel associé. Puis, nous détaillerons les différents aspects de l'évaluation que nous avons menée, les participants, les différentes phases de l'expérience, la méthode d'évaluation utilisée, les données recueillies et leur analyse. Enfin, nous ferons un bilan de cette évaluation et des perspectives de travaux futurs pour l'évolution de K-MADe.

FORMALISME ET LOGICIEL

Nous allons dans un premier temps, présenter le formalisme K-MAD puis l'outil K-MADe qui a été utilisé lors de l'enseignement de la modélisation des tâches.

Le formalisme K-MAD

Le formalisme K-MAD permet d'exprimer hiérarchiquement l'activité de l'utilisateur en utilisant une sémantique formelle via des arbres de tâches.

Dans K-MAD, une tâche est définie par son nom, son but et son numéro. De plus, elle est caractérisée par plusieurs éléments : sa durée, son feedback, son exécutant (utilisateur, système, interactif ou abstrait), sa modalité (lorsqu'un utilisateur est impliqué), sa fréquence, son importance, les événements qu'elle engendre et les observations la concernant.

L'ordonnement de la tâche est défini par ses caractéristiques sur son interruptibilité, sa nécessité (facultative ou obligatoire), son itération (nombre de fois ou expression devant être vérifiée pour exécuter la tâche), sur les pré-conditions à son exécution, sur son événement déclencheur (événement qui contraint l'exécution de la tâche) et sur les modifications qu'elle effectue sur les objets (permet de donner une dynamique aux objets). Une fois définis, les objets sont utilisés pour définir les pré-conditions d'exécution des tâches, les expressions des itérations de tâches, les expressions de manipulation des objets.

Enfin, une tâche décompose ses sous-tâches en déterminant un type d'ordonnement entre elles (séquentiel, alternatif, parallèle, pas d'ordre, élémentaire).

L'outil K-MADe

Un outil a été développé pour supporter le formalisme K-MAD : K-MADe [2]. Celui-ci permet l'édition, l'interrogation des modèles de tâches et leur simulation. Cet environnement est particulièrement destiné aux ergonomes et aux spécialistes d'IHM. C'est cet outil qui a été utilisé par les étudiants, lors des séances pratiques de modélisation.

K-MADe permet de définir l'ensemble des caractéristiques du formalisme K-MAD en l'enrichissant parfois par des attributs utilisés pendant la conception des modèles ; par exemple, l'ajout d'un exécutant « inconnu » dédié à la phase de conception.

Afin de permettre la production d'un modèle de tâches répondant au formalisme K-MAD, le logiciel dispose de plusieurs composants :

- une zone d'édition graphique de l'arbre des tâches (repère 1 sur la Figure 1)
- des zones d'édition des caractéristiques de la tâche sélectionnée. Le repère 2 de la Figure 1 présente une des zones d'édition de caractéristiques.
- une zone d'édition pour les objets (abstrait et concrets), pour les utilisateurs et pour les événements. Les onglets (à gauche dans la Figure 1) permettent d'accéder aux espaces de ces éléments.

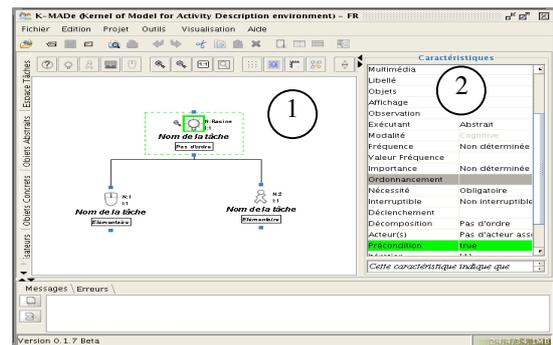


Figure 1: La fenêtre principale de K-MADe

- une calculatrice pour éditer les pré-conditions, post-conditions et itérations. L'expression évaluable de ces conditions est construite à partir d'une grammaire, laquelle peut être évaluée.
- des outils d'aide à l'analyse des modèles de tâches (statistiques, vérification de la grammaire, etc.)
- un outil de simulation permettant l'animation des modèles de tâches. Lorsque celui-ci est lancé, une vérification de la cohérence syntaxique ("vérification grammaire") du modèle est systématiquement réalisée. Dès qu'une erreur est identifiée, une fenêtre s'ouvre, énumérant les problèmes rencontrés.
- un outil de découpage et d'impression de l'arbre de tâches et des caractéristiques des tâches.

Plus d'informations sur le logiciel K-MADe peuvent être trouvées dans son manuel d'utilisation [3].

LES OBJECTIFS

Toutes les données recueillies lors de cette évaluation avaient pour but de connaître le rôle joué par le logiciel K-MADe lors du processus d'apprentissage du formalisme K-MAD. Pour cela, nous avons voulu évaluer son rôle lors de trois phases de la mise en pratique du formalisme étudié : la découverte de la modélisation, l'appropriation du formalisme et l'utilisation de ce formalisme aux fins d'apprentissage.

Lors de la phase de découverte du formalisme, nous avons souhaité connaître le sentiment des étudiants face à l'utilisation d'un logiciel de modélisation des tâches. Ensuite, nous avons cherché à connaître l'impact de K-

MADe sur l'appropriation des concepts du formalisme. Nous avons ainsi cherché à déterminer si le logiciel a ajouté de la complexité à l'analyse, ou au contraire a apporté une aide aux apprenants.

Enfin, une fois les étudiants familiarisés avec le formalisme, nous avons cherché à évaluer comment ils utilisaient K-MADe pour concevoir leurs modèles de tâches : quels sont les outils utilisés, comment les différents concepts sont-ils définis (quand, comment et pourquoi). Cette dernière étude vise à identifier les concepts du formalisme qui ont été assimilés et à établir si K-MADe a été utilisé pour les définir.

LES PARTICIPANTS

Les étudiants ayant participé à cette évaluation sont des étudiants de 4^{ième} année après le bac, inscrits en informatique appliquée en biologie (Master Génie Physiologique et Informatique : <http://www-gphy.univ-poitiers.fr>). Ils sont au nombre de 22, âgés de 21 à 25 ans. Tous ces étudiants ont comme langue maternelle le français.

LES PHASES DE L'ETUDE

L'étude a été réalisée sur une période de un mois (Janvier 2008) et s'est déroulée en deux étapes. Tout d'abord un cours magistral est dispensé afin d'exposer les principes fondamentaux de la conception centrée utilisateur et de la modélisation des tâches. Lors de ce cours, le formalisme K-MAD est présenté, suivi d'une démonstration de l'utilisation de K-MADe. La seconde étape consiste à mettre en pratique ces connaissances théoriques en modélisant sous forme d'arbre K-MAD diverses activités (décrites en texte libre). C'est pendant cette étape qu'ils se familiarisent avec l'outil K-MADe en l'utilisant pour concevoir leurs modèles de tâches.

Déroulement de l'enseignement théorique

Pendant le cours proposé aux participants, leurs sont expliqués les concepts de modélisation de tâches et d'évaluation. L'enseignement de l'IHM proposé est destiné à la conception d'applications interactives centrées utilisateur.

Les modèles de tâches [1, 9] permettent d'analyser un système en mettant son utilisation au centre du processus, et d'exprimer les différentes activités que l'utilisateur souhaite pouvoir réaliser à l'aide de l'application [6]. Ils ont été conçus pour satisfaire des buts différents [1], comme l'assistance à la conception, à la validation, ou pour la génération d'une partie de l'application. Notre but n'étant pas d'enseigner exhaustivement les forts nombreux modèles de tâches publiés dans la littérature, nous concentrons notre enseignement sur le formalisme K-MAD [10], lequel repose sur la définition d'un noyau générique.

La deuxième partie du cours portant sur l'évaluation (enseignement des concepts de base de l'évaluation et des

principales méthodes utilisées en évaluation [7]), le protocole que nous avons utilisé pour pratiquer notre évaluation a été donné comme exemple aux étudiants, afin qu'ils puissent pratiquer cette évaluation avant d'en concevoir une nouvelle par eux-même (ce qu'ils ont fait ensuite sur un logiciel différent).

Déroulement des séances pratiques

La mise en pratique des concepts de modélisation des tâches s'est déroulée sur trois séances. Nous allons décrire les exercices que les étudiants ont eu à faire pendant ces séances et les consignes correspondantes.

Pendant la première séance pratique, nous avons demandé la production de deux modèles de tâches distincts (MdT1 et MdT2). Le premier pouvait être réalisé soit sur papier, soit en utilisant K-MADe. Il s'agissait, pour chaque étudiant de construire un modèle de tâches correspondant à son activité pour venir en cours chaque jour de la semaine. Cet exemple fait intervenir très peu de concepts, puisque seul un exécutant humain est concerné. L'objectif pédagogique était surtout de démontrer la capacité du formalisme à modéliser les différents scénarii proposés par chaque étudiant. Pour ce premier exercice d'initiation à la modélisation, nous avons seulement demandé la décomposition des tâches (pas d'objets, ni de conditions). Les étudiants qui ont utilisé K-MADe, l'ont de ce fait uniquement utilisé comme un éditeur. Après ce premier exercice, l'utilisation de K-MADe fut imposée. Tous les autres exercices nécessitaient la définition d'objets, de conditions et parfois d'acteurs et d'événements. La seconde activité à modéliser était celle d'un agent d'accueil d'une société de location de véhicule (MdT2). Ce second modèle de tâches était à faire par binôme et sans limite de temps. Lors de cette modélisation, un enseignant était présent et assistait les étudiants pendant la première heure et demie de travail. Ensuite, les étudiants ne disposaient plus d'aide extérieure, à part celle qu'ils s'apportaient au sein de leur binôme. C'est lors de cette modélisation qu'ils ont pu explorer le logiciel.

Le lendemain, lors de la seconde séance, le modèle de tâche à produire concernait l'activité de tenue d'une feuille de match de volley ball (MdT3). La description de l'activité a été fournie sur papier et était constituée des consignes officielles de la Fédération Française de Volley Ball et d'exemples de ces feuilles de match (une feuille vierge et une feuille remplie). Les étudiants disposaient d'1h45 pour réaliser cette modélisation et devaient l'accomplir seuls sans aide ni de l'enseignant (bien sur présent pour observer les erreurs à rectifier ultérieurement), ni d'un camarade.

Enfin, une dernière séance de 4 heures concernait K-MADe. Par groupe de 3 ou 4, les étudiants ont eu à réaliser une modélisation K-MAD (MdT4) qui sert de base à

la conception d'une application interactive (pour un laboratoire d'analyses).

METHODE D'EVALUATION UTILISEE

Nous avons voulu évaluer l'utilisation de K-MADe tout au long du processus d'apprentissage de la modélisation. Pour cela, nous avons eu recours à l'observation. Nous avons observé à deux niveaux, une observation de l'ensemble du groupe (observation globale) et une observation du comportement de chaque étudiant (observation individuelle). Cette section décrit les moyens mis en œuvre pour chacune de ces observations.

Observation Globale

Après chacune des séances, l'enseignant prenait des notes sur ce qu'il avait observé. Il notait particulièrement d'un point de vue global les difficultés liées à la compréhension de l'énoncé, les concepts abordés lors de la séance et éventuellement les points qui n'avaient pas pu l'être. Toutes ces notes sont constituées de deux informations : une indication sur l'exercice (ou la question) pendant lequel l'observation a été faite et l'observation en elle-même. Pendant l'ensemble de la formation, l'enseignant a également pris en note les questions particulières posées par les étudiants sur le formalisme ou l'outil. Ces questions concernaient leurs productions particulières telles qu'une demande de vérification de leurs modèles ou une difficulté à utiliser le logiciel. Elles étaient posées selon deux modalités ; de vive voix lors d'une séance pratique ou par email. Ces données sont constituées des mêmes types d'informations que les notes prises lors des séances pratiques mais comportent en plus une indication sur l'étudiant ou le groupe d'étudiants initiateur. De ce fait, nous les avons jointes d'une part aux notes prises pendant la séance correspondante, et d'autre part à l'observation des étudiants.

Si, dans la plupart des cas, ces notes ont pour but d'améliorer notre compréhension des autres données obtenues, lors de certaines séances, nous avons récolté certaines données particulières. Ainsi, l'enseignant a recueilli les données nécessaires au premier point de notre évaluation (lors de la phase de découverte du formalisme) et des compléments d'information lorsque nous n'arrivions pas à interpréter les données dont nous disposions.

Observation Individuelle

L'observation individuelle du comportement des étudiants lors de l'utilisation de K-MADe pour concevoir leurs modèles de tâches a nécessité l'utilisation d'une méthode particulière. Nous avons réalisé cette observation uniquement lors de la séance 2, une fois la phase d'exploration de l'outil terminée. Cette séance se déroule en deux temps d'1h45 chacune, le même après-midi (1/4 d'heure sépare les deux). Les participants sont répartis en 11 groupes de deux personnes. Lors de la première moitié du temps, nous désignons quel est celui des deux

qui joue le rôle de l'utilisateur (et par conséquent, quel est celui qui sera l'observateur). Nous prenons la précaution de déterminer nous-mêmes le rôle de chacun, afin de garantir que tous les niveaux d'utilisateurs sont représentés. Ils reçoivent alors les consignes. Pendant 1H45, les utilisateurs vont utiliser K-MADe pour modéliser sous forme d'arbre de tâches K-MAD, l'activité de tenue de la feuille de match de volley ball. Pendant la durée de la modélisation, l'un des rôles de l'observateur est de s'assurer que l'utilisateur décrive oralement ce qu'il veut faire, afin d'obtenir le plus clairement possible le processus mental de modélisation. Il a donc pour consigne de poser des questions pour inciter l'utilisateur à raisonner à haute voix.

La seconde de ces tâches est de transcrire les difficultés que l'utilisateur rencontre dans l'utilisation du logiciel. Il doit détecter ces difficultés d'utilisation en observant l'utilisateur (hésitations, aller-retour, exploration de diverses parties du logiciel sans actions dans chacune de ces parties...). Afin de permettre de prendre en note les observations, des fiches sont fournies aux observateurs. Une illustration d'observations prises en note est présentée Table 1. Ces fiches sont principalement composées d'un tableau de trois colonnes correspondant aux trois informations à fournir pour chaque observation :

- le type d'observation parmi les catégories définies (buts de l'utilisateur (B), fonctionnalités de l'outil proposées (F), l'utilisation de celle-ci (UF) et information (I)). Cette information a pour but de les guider lors de la prise de notes.
- l'observation sous forme textuelle.
- l'heure à laquelle elle est faite.

Type	Observation	Temps
UF	la fenêtre principale n'est plus accessible (« simulation » écrit dessus mais la fenêtre de simulation est inaccessible) => redémarrage de K-MADe	14h32
B	recherche où définir un objet	14h34 14h37
F	l'utilisateur ne comprend pas à quoi peut servir le bouton avec un entonnoir dans la fenêtre de simulation	14h40

Table 1 : Extrait d'une feuille de prise de notes.

Afin de compléter les notes des observateurs, nous fournissons aux utilisateurs une version de K-MADe conforme à celle qu'ils ont utilisée jusque-là, mais contenant des mouchards. Ces derniers permettent d'enregistrer dans un fichier-texte certaines actions réalisées par les utilisateurs et le moment auquel elles ont été exécutées. Il s'agit d'enregistrer précisément l'heure d'entrée et de sortie dans chacun des composants de K-MADe (espace de définition des tâches, d'objets abstraits et concrets, labels, événements, utilisateurs, fenêtre de simulation, fenêtres d'édition des conditions (pré, post et itération)...). Un exemple d'informations contenues dans ce fichier est présenté sur la Figure 2.

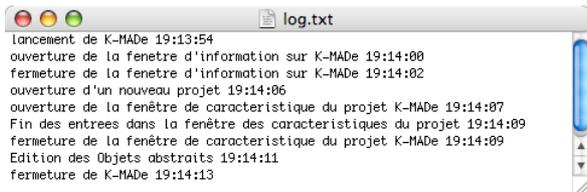


Figure 2 : Illustration des données dans le fichier-texte.

Après cette séance, chaque étudiant doit, en utilisant les données récoltées, réaliser un rapport de quelques pages détaillant trois aspects : le processus de modélisation de l'utilisateur, l'utilisation de l'outil et l'évaluation du modèle de tâches obtenu.

Pendant la seconde partie de la séance, les rôles sont inversés, l'observateur devient l'utilisateur et l'utilisateur devient l'observateur. L'expérimentation est réalisée dans les mêmes conditions et avec les mêmes consignes. La production de cette seconde phase d'évaluation se compose des mêmes éléments que la première, un fichier log, le modèle réalisé, les notes d'observation et un rapport écrit.

LES DONNEES RECUEILLIES

Comme il ne s'agissait pas d'évaluer les mêmes aspects de l'utilisation de K-MADE lors de chacune des séances, les données recueillies pour chacune de ces séances pratiques étaient différentes. La Table 3 résume le contenu et la production de chaque séance. De plus, afin de conserver des données exploitables, nous avons éliminé certaines productions. Dans un second temps, nous détaillerons les raisons de cette sélection.

Les types de données

Lors de la première séance pratique, l'objectif était de connaître le choix des étudiants pour apprendre un formalisme. Pour cela, nous avons laissé la liberté aux étudiants d'utiliser ou non le logiciel K-MADE pour réaliser leur premier modèle (l'exercice ne requérant pas particulièrement l'utilisation de K-MADE). L'enseignant a alors comptabilisé le nombre d'étudiants pour chacun des choix. C'est la seule donnée que nous avons retenue de cette modélisation.

Dans un second temps, l'objectif était d'observer comment les étudiants concevaient leurs modèles de tâches et quelles fonctionnalités de K-MADE ils utilisaient pour cela. Il a été demandé de produire deux modèles (MdT2 et MdT3). La production de MdT2 avait pour but de les familiariser avec tous les concepts de K-MAD et avec l'outil K-MADE afin que la conception de MdT3 soit exemptée de la phase de découverte de l'outil. L'intérêt de ces données réside dans le fait qu'elles n'ont pas été obtenues sous contrainte temporelle.

Lors de la réalisation de MdT3 (la tenue de feuille de match), plusieurs dispositifs ont été mis en place, pour connaître le processus de conception (mouchards, obser-

vation). Nous avons recueilli l'ensemble des productions de ces dispositifs (fichier log et notes). À partir de ces deux sources de données et du modèle de tâche produit (MdT3) par leur camarade, les observateurs devaient produire un rapport d'évaluation sur l'usage de K-MADE. Nous avons également recueilli ces documents (MdT3 et rapport).

De plus, lors de chacune des séances (y compris la dernière), lorsque l'enseignant était présent, des notes ont été prises.

séance 1	MdT1 MdT2	- rapport de l'enseignant (nombres) - modèles de tâches - notes de l'enseignant
séance 2	MdT3	- fichiers avec les étapes (log) - notes d'observation des étudiants - rapports des observateurs - modèles de tâches - notes de l'enseignant
séance 3	MdT4	- notes de l'enseignant

Table 3 : Les données récoltées.

Les données retenues

Lors de la première séance, le modèle de tâches de MdT2 étant à faire par groupe de deux, nous aurions dû obtenir 11 modèles. Seuls 9 nous ont été rendus. En raison d'un virus informatique, un des binômes nous a fourni une précédente version. Ces modèles étant utilisés pour étudier ce que les étudiants définissent pour concevoir un modèle de tâches qu'ils considèrent terminé, il ne correspondait pas à notre protocole. Comme les étudiants ont prévenu rapidement l'enseignant, nous avons pu isoler leur modèle afin de ne pas en tenir compte lors de l'analyse des résultats.

De même, nous n'avons pas utilisé les 22 documents réalisés lors de la séance 2. Tous les modèles MdT3 étaient réalisés cependant, certains observateurs n'ont pris que très peu de notes sur les actions de l'utilisateur. Ces notes nous aidant à interpréter les données du fichier avec les étapes lorsqu'elles étaient ambiguës, nous avons isolé ces documents (au nombre de 4). L'élimination des documents étant principalement due à un manque d'implication de la part des étudiants, il s'est avéré que les 4 documents inexploitable du MdT3 correspondaient aux modèles MdT2 non rendus. De plus, nous voulions prendre les productions de MdT2 et MdT3 des mêmes étudiants afin de pouvoir les comparer notamment lors de l'évaluation sémantique de leurs modèles, nous avons donc également isolé les documents MdT3 correspondant aux étudiants dont le MdT2 avait lui-même été mis de côté. Nous avons finalement retenu 18 documents (fichier avec étapes, rapport et notes de l'observateur et modèles de tâches de l'utilisateur) pour la modélisation MdT3.

RESULTATS

Nous avons analysé ces diverses sources d'information sur l'utilisation du logiciel K-MADE en suivant 3 axes :

le point de vue pédagogique ; l'autonomie des étudiants pour la correction de leurs modèles ; et l'évaluation des modèles terminés.

Enseignement des modèles avec le logiciel

Sur l'ensemble des 22 étudiants présents lors de la première séance, seuls 4 ont choisi de ne pas utiliser K-MADe pour réaliser leurs premiers modèles de tâches. Ce choix peut être expliqué par deux raisons : la majorité a choisi l'utilisation du logiciel après avoir demandé s'ils auraient à s'en servir par la suite, souhaitant de ce fait, factoriser l'apprentissage de la modélisation et de l'utilisation du logiciel ; les autres ont exprimé le fait que l'utilisation du logiciel les rassurait lors de l'apprentissage d'un formalisme. Notons que ces étudiants en bioinformatique n'ont aucune réticence à utiliser un logiciel. De plus, il nous est impossible de déduire de nos données le choix qui aurait majoritairement été fait si nous n'avions pas imposé l'utilisation de K-MADe lors des autres séances. Nous ne pouvons donc conclure sur l'apport d'un logiciel sur l'appréhension de l'apprentissage d'un nouveau formalisme par des étudiants en informatique, et a fortiori sur les autres publics.

Les 4 étudiants ayant choisi d'utiliser papier et crayon au début de l'exercice l'ont terminé en utilisant K-MADe, bien que l'enseignant n'ait pas encore imposé son utilisation. Ce changement a eu lieu après discussion avec l'enseignant sur la validité de leur modèle. Ils étaient alors contraints de le modifier en partie ou entièrement. Ces modifications rendaient leurs modèles moins lisibles que ceux de leurs camarades utilisant le logiciel. C'est pourquoi ils ont préféré refaire, puis terminer leur modèle en utilisant K-MADe.

L'utilisation du logiciel pour cet exercice de modélisation d'une activité courte (produisant des modèles de tâche de petite taille), ne permet pas d'illustrer une autre raison pouvant justifier l'utilisation du logiciel ; la navigation dans le modèle de tâches. K-MADe met à disposition deux fonctionnalités pour faciliter cette navigation : le zoom de tout le modèle de tâches et la possibilité d'afficher ou non les sous-tâches d'une tâche. Les données recueillies lors de la séance 2, nous ont permis de constater que 93,75% des étudiants utilisent une de ces fonctionnalités pour faciliter la navigation dans le modèle de tâches.

Aux facilités qu'offre l'utilisation d'un logiciel tel que K-MADe pour l'édition d'un modèle de tâches, la technique papier-crayon peut répondre par une plus grande liberté en matière de personnalisation. En effet, il a été montré dans une étude menée sur 5 ergonomes experts [4] que ceux-ci utilisaient principalement (4/5) un formalisme « maison ». Il semble évident que, pour ces ergonomes, il est difficile de concevoir un logiciel capable de s'adapter. Cependant, K-MADe permet de personnaliser la présentation (sans modification du formalisme). Bien

que cette fonctionnalité n'ait été présentée par aucun enseignant, 37,6% des étudiants ont modifié les préférences (principalement la présentation des tâches) lors de la conception du MdT3, il leur est donc indispensable de pouvoir personnaliser leur outil pour la conception. De plus, dans le cadre de l'enseignement de modélisation des tâches, il est nécessaire que les étudiants se tiennent à un formalisme et à ses règles afin de permettre l'évaluation de leurs apprentissages. Pour cela, l'utilisation d'un logiciel n'offrant pas trop de libertés sur le formalisme est appréciable.

K-MADe est un tel logiciel. Il dispose notamment d'une vérification de cohérence du modèle lors de l'utilisation de l'outil de simulation. Cet outil permet la détection de certaines erreurs comme le fait qu'une tâche dite « élémentaire » ne doive pas être décomposée en tâches filles. Dès la première séance de modélisation (et donc d'utilisation de K-MADe), l'enseignant a indiqué l'existence de cette aide afin de concentrer ses interventions sur les difficultés liées au processus de modélisation, laissant ainsi le soin aux étudiants de faire la détection des erreurs et de chercher à les corriger par eux-mêmes. En plus de cette utilisation prévue, il a permis de mettre en évidence une difficulté de compréhension sur ce qu'est la décomposition des tâches. En effet, une partie des étudiants (près d'un tiers) confondait la relation de filiation de deux tâches avec la relation d'héritage et obtenaient de ce fait, des tâches décomposées en une seule sous-tâche. Ce type d'incohérence étant relevé par l'outil, l'enseignant a pu rapidement remédier à cette incompréhension.

Correction du modèle de tâches par les étudiants

Lorsque les étudiants ont eu à modéliser une activité sans aucune assistance (MdT3), 62,5% d'entre eux ont utilisé au moins un des outils de correction. De plus, 12,5% des étudiants ne l'ayant pas utilisé prévoient de le faire s'ils en avaient eu le temps (l'exercice était d'une durée limitée).

L'étude sur le processus de modélisation a montré que les étudiants lançaient l'outil de simulation plus particulièrement à deux moments, après la décomposition des tâches (37,5%) et une fois qu'ils estimaient la modélisation terminée (75%). Sur l'ensemble des étudiants qui ont lancé l'outil de simulation, 90% ont eu la fenêtre de cohérence qui s'est ouverte (le modèle était incohérent). Nous avons également constaté que lorsque les étudiants lançaient l'outil de simulation, dans 87,5% des cas, une fois que l'outil de cohérence ne détectait plus d'erreurs, l'outil était refermé. Cette observation tend à montrer que pour tous ces étudiants, le lancement de l'outil de simulation n'avait pour but que d'atteindre l'outil de cohérence. De plus, bien que cet outil soit disponible seul, aucun étudiant n'a activé la vérification de la grammaire sans passer par la simulation. Nous pouvons expliquer cela par deux raisons : soit ils n'ont pas remarqué l'outil

de vérification de la grammaire indépendante de la simulation, soit ils préfèrent lancer la simulation sachant que la vérification va être faite. 77,8% des étudiants ayant eu des erreurs de cohérence dans leurs modèles de tâches les ont modifiés afin de corriger ces erreurs. Tous ces étudiants ont réussi la correction de l'ensemble des erreurs identifiées. Les 2 étudiants restants (dont les modèles comportaient des erreurs déclenchant l'ouverture de la fenêtre de cohérence et qui n'ont pas réussi à modifier leurs modèles pour en tenir compte) sont des étudiants qui n'ont pas, ou tardivement (5 minutes avant la fin du temps imparti) compris l'utilisation de cet outil de cohérence. Cependant, les données recueillies ne permettent pas de savoir si ce sont les messages d'erreur qui n'étaient pas assez clairs ou la relation entre le message et l'erreur (lorsque l'on clique sur le message, la tâche correspondante est sélectionnée) qui n'était pas explicite. Nous pouvons tout de même souligner que seul un cinquième des étudiants ayant lancé l'outil de cohérence a rencontré des difficultés à l'utiliser et que lorsqu'une erreur est corrigée, toutes les autres le sont également (sauf limitation dans le temps).

Si cette fenêtre d'information sur les erreurs de cohérence du modèle est un outil apprécié par les étudiants pour la correction de leurs modèles, il nous est plus difficile d'avoir des informations sur la fenêtre de simulation en elle-même. Cet outil a soulevé beaucoup de questions sur la manière dont il pouvait être utilisé et à quoi il servait. Lors de la conception de leur troisième modèle de tâches, sur les 16 étudiants, 8 sont parvenus à l'ouverture de l'outil (celui-ci ne s'ouvrant qu'une fois toutes les erreurs de cohérences corrigées) et un seul a su l'utiliser et a généré des scénarios. Lors de l'utilisation de K-MADe qui a suivi, l'enseignant a présenté de nouveau comment utiliser l'outil de simulation et d'après les étudiants, le manque d'utilisation de cet outil est dû principalement à deux raisons : d'une part, lors de la première présentation (théorique), les étudiants n'avaient pas conscience de l'utilité de cet outil, ils n'ont donc pas mémorisé la manière de le faire fonctionner. D'autre part, lorsque les étudiants ont souhaité vérifier leur modèle, l'enseignant a indiqué où trouver l'outil de simulation mais pas comment réaliser les scénarios.

Évaluation des modèles obtenus

Dans cette section, nous utilisons les modèles produits après la première séance (MdT2) afin de ne pas avoir à prendre en compte les difficultés liées aux contraintes de temps. La Table 4 résume le nombre de modèles (sur les 8) ayant défini les objets abstraits, les objets concrets, au moins une postcondition, au moins une itération, au moins un événement et au moins un utilisateur. Nous n'avons pas tenu compte des définitions des préconditions car l'enseignant les a utilisées pour illustrer l'utilisation des objets pour éditer les conditions logiques. De plus, tous les binômes ont produit un modèle

ayant des tâches décomposées (au moins deux niveaux de décomposition).

objets abstraits	8
objets concrets	7
postcondition	7
itération	4
événement	1
utilisateur	2

Table 4 : Proportion des composants de K-MAD définis.

D'après ces résultats, il apparaît que le concept d'objets abstraits a été bien assimilé. La définition d'un objet abstrait demande l'attribution d'un nom pour désigner l'objet et les attributs de l'objet. Le nom de l'objet et le nom des attributs sont indiqués sous forme textuelle, l'utilisation du logiciel n'a en rien modifié ces définitions. Cependant, les attributs des objets de K-MADe possèdent un type (booléen, texte, nombre). Ceux-ci correspondant à des types informatiques, leurs définitions n'a pas posé de difficultés pour les étudiants en informatique.

Ces objets sont instanciés en objets concrets afin d'être modifiés par l'exécution des post-conditions. Sept des huit binômes ont défini des objets concrets et au moins une post condition. Cependant, bien que les conditions et les objets soient liés, il ne semble pas que la relation entre les deux concepts soit clairement comprise par les étudiants. Ce constat est appuyé par le fait qu'il ne s'agit pas du même groupe que celui qui n'a pas défini d'objets concrets et de postconditions. Afin de confirmer cette observation, nous avons étudié les moments auxquels les étudiants définissaient les objets et les conditions à partir des données obtenues lors de la séance 2. Les 7/8 des modèles rendus (MdT3) contenaient des objets et des conditions (l'exercice était à produire dans un temps limité, il est donc impossible de déduire quoi que ce soit des modèles sans objets ni conditions, l'imputation pouvant être attribué au temps). Sur ces modèles, 71,5% ont défini les objets séparément des conditions et sur ce nombre, 7/10 ne sont pas revenus sur la définition de leurs objets une fois leurs conditions définies, ce qui tend à appuyer notre observation.

Cependant, les concepts de condition (pré et post) semblent compris, les étudiants ne sachant pas définir les conditions sous forme logique les ont indiquées sous forme textuelle. K-MADe permet en effet d'éditer les conditions sous forme logique (utilisant une syntaxe proche du langage B [8]) et textuellement. La Figure 3 montre une précondition à l'exécution de la tâche « location de voiture » éditée à la fois sous forme textuelle et sous forme logique. Cependant, comme il est impossible de vérifier une condition textuelle, ce mode d'édition de conditions n'est pas utilisé lors de la simulation et donc rend impossible la vérification de ces conditions. De plus, la notion de post condition était connue des étudiants comme étant une condition logique. Ce-

pendant, le logiciel K-MADe définit les post conditions comme étant une (des) action(s). Cette définition a entraîné une confusion chez les étudiants qu'il a été nécessaire de corriger afin qu'ils puissent utiliser le logiciel de manière correcte.



Figure 3 : Edition textuelle et logique d'une précondition.

Enfin, bien que l'activité à modéliser ait nécessité l'utilisation d'itérations, seule la moitié des modèles les inclut. Dans K-MADe, la définition des itérations se réalise de la même manière que les définitions des conditions (objets concrets et opérateurs logiques). Nous n'avons donc pas trouvé de raison propre à ce concept expliquant ce manque de définition. Le concept d'itération a été enseigné en même temps que ceux de pré et de post-conditions. Cependant, pendant les séances pratiques, l'enseignant n'a pas explicité ce concept alors qu'il a illustré la notion de précondition et expliqué celle de post-condition. Le manque d'itérations définies semble donc dû à un défaut dans l'enseignement, qui n'appuie pas suffisamment sur ce concept.

Cette observation est encore plus vraie pour les notions d'événement et d'utilisateur. En effet, dans le logiciel, ces concepts sont définis sous forme textuelle n'induisant de ce fait aucune difficulté particulière. Seul, le fait que l'enseignant ne les ait pas particulièrement illustrés peut expliquer leur manque d'utilisation.

BILAN-PERSPECTIVES

L'évaluation présentée dans cet article est basée sur l'exploitation des données que nous avons obtenues lors de l'utilisation du logiciel K-MADe comme support à l'apprentissage de la modélisation des tâches à des informaticiens.

Ces données nous ont révélé que ces utilisateurs trouvent avantageuses les fonctionnalités mises en place pour la navigation (93,75% des étudiants en utilisent au moins une) et qu'ils préfèrent utiliser le logiciel dans un souci de lisibilité (tous les étudiants ayant choisi l'utilisation de papier et crayon pour éditer leurs modèles se sont ravisés dès les premières modifications). D'un point de vue pédagogique, l'utilisation de K-MADe au travers ses outils, fournit une aide syntaxique nécessaire aux étudiants (90% des étudiants ayant lancé la simulation avaient des erreurs syntaxiques dans leurs modèles). Cette étude a donc montré que l'utilisation de K-MADe apportait des avantages lors de l'apprentissage de la modélisation des tâches. De plus, nous avons noté que son utilisation est positive, notamment lors de l'apprentissage des conditions. Afin d'accentuer cet apport pédagogique, des modifications sont en cours visant à incorporer d'avantage l'outil de vérification de cohérence dans la

conception et à faire en sorte que le fonctionnement de l'outil de simulation soit plus accessible. Nous avons également noté la difficulté des étudiants à définir les conditions logiques et à utiliser l'outil de simulation. Or, ces deux éléments sont nécessaires pour la vérification et la validation des modèles. Dans ce sens, des études sont menées pour identifier précisément les difficultés et améliorer leur utilisation. Enfin, il sera utile de s'intéresser à d'autres utilisateurs, non nécessairement formés en informatique.

BIBLIOGRAPHIE

1. Balbo, S., Ozkan, N. and Paris, C. *Choosing the Right Task-modeling Notation: A Taxonomy*. The Handbook of Task Analysis for Human-Computer Interaction, 2004, 445-466.
2. Baron, M., Lucquiaud, V., Autard, D. and Scapin, D. K-MADe : un environnement pour le noyau du modèle de description de l'activité. In *Proceedings of IHM'06* (18-21 avril 2006, 2006, Montréal, Canada), ACM Publishers, 287-288
3. Baron, M. and Scapin, D. Manuel d'utilisation de K-MADe. Disponible à l'adresse <http://www-rocq.inria.fr/merlin/kmade/kmadedocfr.pdf>
4. Couix, S. Usages et construction des modèles de tâches dans la pratique de l'ergonomie : une étude exploratoire. Disponible à l'adresse <http://www.biomedicale.univ-paris5.fr/taskmodelsurvey/accueil>
5. Diaper, D. *Understanding Task Analysis for Human-Computer Interaction*. The Handbook of Task Analysis for Human-Computer Interaction, 2004, 5-48.
6. Diaper, D. and Stanton, N.A. *The Handbook of Task Analysis for Human Computer Interaction*. Lawrence Erlbaum Associates, 2004.
7. Kovacs, B., Gaunet, F. and Briffault, X. *Les techniques d'analyse de l'activité pour l'IHM*. Hermes Science, 2004.
8. Lano, K. *The B Language Method: A guide to practical Formal Development*. Springer, 1996.
9. Limbourg, Q. and Vanderdonck, J. *Comparing Task Models for User Interface Design*. The Handbook of Task Analysis for Humain-Computer Interaction, 2004, 135-154.
10. Lucquiaud, V. *Sémantique et Outil pour la Modélisation des Tâches Utilisateur: N-MDA*, Poitiers, 2005, 285
11. Paternò, F., Mancini, C. and Meniconi, S. Concur-TaskTrees: A Diagrammatic Notation for Specifying Task Models. In *Proceedings of IFIP TC13 human-computer interaction conference (INTERACT'97)* 1997, Sydney, Australia), 362-369