

Modélisation de tâches contextualisées pour la construction d'applications collaboratives

Olivier Delotte, Bertrand David

Laboratoire ICTT
Ecole Centrale de Lyon
36, av. Guy de Collongue
69134 Ecully Cedex
olivier.delotte@ec-lyon.fr

RESUME

Cet article décrit une méthode pour la construction d'un modèle comportemental pour l'élaboration de systèmes collaboratifs capillaires. Le point de départ de cette démarche est la constitution d'un ensemble de scénarios d'usage. L'activité et les préférences des acteurs sont au cœur du modèle, et sont prises en compte en amont de l'intégration ainsi qu'au cours de l'utilisation. De plus, nous détaillons notre outil CBME (Contextualized Behavioral Model Environment) qui permet la création et la vérification d'un modèle comportemental comprenant un éditeur de tâches contextualisées. Pour les concepteurs de systèmes capillaires sensibles au contexte, il permet de visualiser et de manipuler les différents éléments du modèle comportemental pour contrôler la cohérence du modèle.

MOTS CLES : Scénarios, Systèmes Capillaires, CSCW, Modélisation des tâches, Approche Model-Based.

INTRODUCTION

Le monde du TCAO est encore aujourd'hui essentiellement tourné vers des dispositifs fixes disposant de connexions haut-débit. Cependant, le nomadisme des travailleurs, et donc des utilisateurs de systèmes collaboratifs change la donne. Dans nos précédents articles présentés à IHM [6] [7], nous avons positionné nos travaux au niveau bibliographique et par rapport aux projets existants. Le travail coopératif capillaire [4] qui marie le synchrone et l'asynchrone dans la mobilité des personnes et des informations, tente de prendre en compte le contexte des sujets et des objets de l'action. La diversité des dispositifs fixes et mobiles et leurs caractéristiques, notamment d'affichage et d'interaction, doivent être prises en compte par une démarche de développement multi plates-formes. Dans [6], nous décrivons une démarche théorique pour la construction et l'évolution de systèmes collaboratifs sensibles au contexte (figure 1). Nous nous situons dans la mouvance Model-based Approach, MDA ou MDE [10]. Il s'agit de collecter des scénarios exprimant de façon parcellaire, souvent tâche par tâche et acteur par ac-

teur, des situations d'utilisation souhaitées. La première étape vise à construire, à partir d'un ensemble de scénarios d'utilisation collectés, un modèle comportemental qui synthétise ces informations afin de caractériser le comportement du système coopératif visé. La seconde étape est une phase d'adaptation du modèle comportemental à l'architecture collaborative capillaire en tenant compte des caractéristiques des matériels utilisés et des outils logiciels déjà disponibles. La dernière étape concerne l'évolution déclenchée lors de la modification du système conformément aux souhaits des utilisateurs exprimés à travers de nouveaux scénarios. Le système collaboratif peut co-évoluer avec le modèle comportemental afin de proposer de nouveaux services et de nouvelles manières d'interagir aux utilisateurs.

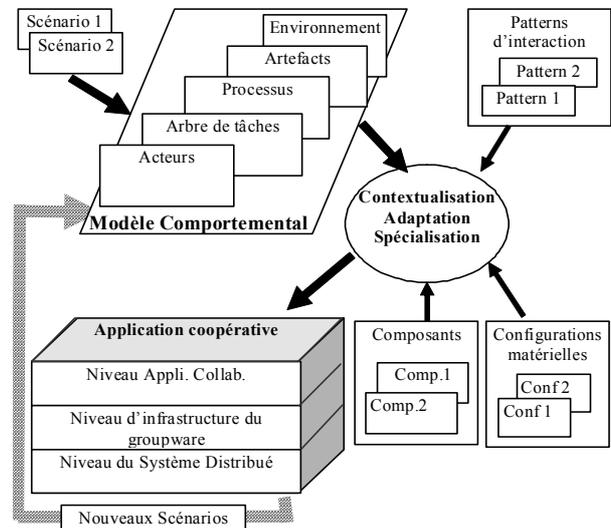


Fig. 1. Processus de conception d'applications collaboratives sensibles au contexte d'usage.

DEFINITION DU CONTEXTE D'UTILISATION

Nous nous appuyons sur la définition de A.K. Dey [8], pour qui le contexte regroupe toutes les informations qui peuvent être utilisées pour caractériser la situation d'une entité. Une entité est une personne, un endroit ou un objet qui relève de l'interaction entre un utilisateur et une application, eux-mêmes inclus. Nous choisissons cette définition avec comme objectif d'associer

au contexte d'usage d'un utilisateur les activités des autres acteurs. Ainsi la première étape dans le processus d'intégration du contexte est la recherche des informations et, ensuite, leur interprétation dans le système collaboratif. Beaucoup de recherches [5] se font actuellement autour du captage de l'environnement de l'utilisateur avec l'aide de puces électroniques (bruit, position,...). Pour utiliser ces informations dans un processus de conception d'applications sensibles au contexte, nous pensons qu'elles doivent être couplées avec l'arbre des tâches afin de faciliter l'adaptation (des IHM et des Services). Dans le cadre de systèmes collaboratifs, la conscience des autres (awareness) est un élément supplémentaire nécessaire à l'activité de groupe et le système doit proposer à chaque utilisateur une interface adaptée en fonction des tâches qu'il est en train d'effectuer et des possibilités des autres utilisateurs. C'est pourquoi, nous avons opté pour l'utilisation de scénarios d'utilisation comme formalisme de description des activités dans leur contexte d'utilisation.

Les scénarios d'utilisation comme point de départ

Un scénario peut prendre plusieurs formes : soit un texte libre décrivant une activité, soit un texte structuré avec un certain nombre de champs, soit une forme graphique ou textuelle telle que proposée par les cas d'utilisation d'UML [9]. Carroll [2] [3] définit les scénarios comme des récits d'utilisation et comme une description narrative d'une activité, située dans un contexte précis. Le système capillaire, que l'on souhaite construire, a pour but de prendre en considération les différents scénarios ponctuels collectés. Ceux-ci sont donc progressivement repris et « recoupés » pour servir lors de la construction du modèle comportemental. Nous allons donc détailler ses principaux constituants de ce dernier.

Le modèle comportemental

La section Acteurs contient la liste de tous les acteurs évoluant dans le système avec leurs caractéristiques qui, outre leurs noms (génériques), comportent leurs rôles potentiels. Ces derniers expriment la façon dont ils interviennent dans les activités et agissent avec et sur les artefacts.

La section Processus regroupe à la fois les arbres des tâches avec leurs pré-conditions et les processus donnant l'organisation (succession, parallélisme, dépendances) logique, fonctionnelle ou temporelle de ces tâches.

Quant à la section Artefacts, elle identifie les outils logiciels (avec quoi on travaille) et les classes d'objets (sur quoi on travaille).

La section Environnement réunit les informations relatives à la localisation potentielle des acteurs (position géographique, situation d'utilisation), la situation de l'activité dans le temps ou l'action déclenchante. En effet, la situation « chez le client » (localisation) est cataloguée comme contexte d'utilisation, tout comme « après la réparation » (position dans la séquence de tâches) ou « en se connectant au serveur de l'entreprise » (événementiel).

DES SCENARIOS VERS LE MODELE COMPORTEMENTAL

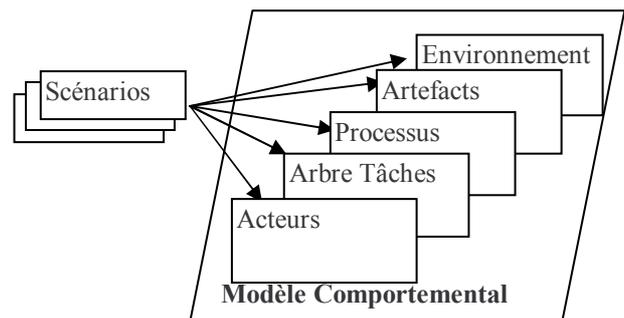


Figure 2 : Transformation des scénarios en modèle

La transformation des scénarios en modèle consiste donc à extraire les informations contenues dans les scénarios en les positionnant dans les différentes sections du modèle comportemental. Nous prenons chaque récit et nous appliquons un découpage.

Exemple de décomposition d'un scénario :

« Le matin, le technicien consulte sur son PDA la liste des opérations de la journée. »

Le matin devient une étiquette de l'environnement. Le technicien est l'acteur principal. De cette manière, on décompose intégralement le scénario. Comme dit précédemment, un scénario ne peut couvrir intégralement l'ensemble du modèle comportemental. C'est pourquoi, un corpus de scénario pour chaque activité doit être créé. Aujourd'hui, la décomposition d'un ensemble de scénarios a permis de fabriquer un corpus de tâches récurrentes pour la journée d'un technicien d'une société de dépannage. La mise à disposition de ces informations et leur hiérarchisation forment le modèle comportemental. Reste le problème de la mise en forme de plusieurs scénarios disjoints ou complémentaires. Comment reconnaître et éliminer les redondances au sein du modèle ? Quelle méthode de construction à employer ? C'est à ces réponses que je souhaite répondre dans le paragraphe suivant.

Construction du MC avec plusieurs scénarios

La construction du modèle comportemental se fait par recoupement de plusieurs scénarios et par réification.

Reprenons l'exemple 1 et un autre scénario pioché dans le corpus. La première étape reprend le découpage précédent avec le second scénario. Ainsi, on a obtenu 2 arbres de tâches. La mise en relation de ces 2 éléments est donc une partie importante dans la réification. Décomposons étape par étape :

- Découpage de tous les éléments qui ne sont pas des actions. Intégration de l'activité dans le module Arbre des tâches.
- Association de l'activité avec les éléments du modèle déjà insérés.

A travers la construction, on discerne 4 modes d'actions dans la journée d'un réparateur : transmission d'informations, d'interaction, d'organisation-supervision, approche métier.

Construction du modèle de tâches

De nombreux travaux se sont penchés sur les formalismes de modélisation des tâches. Peu ont aboutis à des outils utilisables [11]. Cependant, les travaux sur CTT [12] et sur la modélisation de tâches sensibles au contexte [13] sont des recherches sur lesquelles nous pouvons nous appuyer pour construire notre modèle de tâches. Nous utilisons actuellement le formalisme CTT pour sa complétude. L'outil d'édition et de visualisation graphique CTTE n'étant pas intégrable dans notre projet, nous avons décidé de mettre en place notre propre outil (édition et visualisation). Il permet d'exprimer visuellement les liens entre plusieurs tâches collaboratives, comme CTTE, mais en intégrant les modèles précédemment définis: les acteurs, les artefacts et leurs environnements. De plus, l'outil doit gérer les liens entre les autres éléments de chaque modèle du MC.

Validation du modèle comportemental

La validation du modèle comportemental se fait selon trois axes : complétude, exactitude et cohérence. Il s'agit donc de mener des validations formelles et informelles, avec recours aux utilisateurs. Les validations formelles portent sur la complétude de l'arbre des tâches et sur la complétude de l'utilisation d'opérations proposées par des classes. Les utilisateurs doivent intervenir lors de la validation sémantique qui peut s'appuyer sur différentes représentations. Ces représentations peuvent montrer pour chaque acteur (ou rôle) son implication dans le système, c'est-à-dire dans les opérations issues de l'arbre de tâches, les classes manipulées et les opérations autorisées et/ou utilisées.

Pour pouvoir manipuler correctement le modèle, il est nécessaire de conserver, tout au long de la construction, les liens entre les scénarios sources et les éléments

du modèle. De cette manière, il est possible de valider les différentes parties du modèle ou d'apporter des modifications pour le rendre cohérent. Par exemple, la liste des acteurs issus des scénarios peut être redondante avec des synonymes, ... On peut alors « nettoyer » cette liste et, par conséquent, modifier le modèle et les scénarios sources. Il en est de même pour les autres aspects du modèle.

Plus globalement, les représentations basées respectivement sur les acteurs, tâches, processus, artefacts et les environnements permettent d'étudier formellement et sémantiquement le modèle pour l'observer, le compléter, le modifier et finalement le valider. La transposition des scénarios vers le modèle comportemental est réalisée manuellement grâce à l'outil CBME (fig. 3), explicité ci-après.

L'OUTIL CBME

Contextualized Behavioral Model Environment (CBME) est un outil de visualisation graphique du modèle comportemental. Il permet le passage entre les scénarios (sous forme de texte) et le modèle comportemental (fig. 3). En effet, cet outil est capable de prendre un scénario et d'aider le concepteur à l'insérer dans le modèle comportemental.

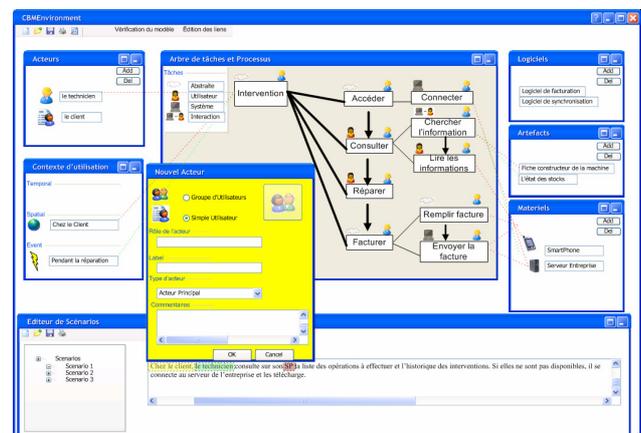


Fig.3 Prototype CBME

FONCTIONNALITES DU PROTOTYPE CBME :

Construction graphique : l'utilisateur sélectionne les mots du scénario et les dispose dans la section correspondante. Une zone de description permet de caractériser le nouvel élément du modèle.

Edition des liens entre les tâches : Il s'agit de construire l'arbre des tâches pour chaque activité et d'établir des liens appropriés entre les tâches, les processus et les artefacts concernés.

Vérification du modèle : L'outil permet de vérifier la complétude du modèle en s'assurant la cohérence des relations entre les éléments du modèle. La complétude

sémantique est contrôlée par observation des liens entre les éléments. Des modifications peuvent être effectuées afin de corriger le modèle.

VISUALISATION DU MODELE COMPORTEMENTAL DANS CBME

Le premier problème réside dans la visualisation d'informations hiérarchisées. Comment le concepteur peut visualiser l'ensemble des tâches disponibles dans telle situation et modifier le modèle comportemental ? C'est dans l'optique de répondre à cette question, que nous orientons nos recherches, en particulier, vers les travaux de visualisation d'information et d'ergonomie des logiciels. La manipulation des données est le second objectif et changer le modèle en insérant, déplaçant, supprimant des composants du modèle comportemental.

Afin d'illustrer notre démarche, nous travaillons sur un ensemble de scénarios d'utilisation tirés de l'activité d'une société de dépannage, dont voici un exemple : « Chez le client, le technicien consulte la liste des opérations à effectuer et l'historique des interventions. Si elles ne sont pas disponibles, il se connecte au serveur de l'entreprise et les télécharge. »

Nous utilisons l'éditeur de scénarios (fig. 3) de CBME, afin de le décomposer en sélectionnant chaque bloc de mots (avec conservation du marquage) vers l'une des quatre fenêtres représentant les 4 sections de la grille. Selon le type de données, une boîte de dialogue apparaît et permet une caractérisation plus fine du nouvel élément.

Le système conserve en mémoire les liens entre chaque élément du scénario et les représente graphiquement, selon une typologie de couleurs. Dans la partie Arbre de tâches et Processus, des icônes représentent les acteurs pour chaque tâche. L'édition manuelle de liens est nécessaire à la création du séquençement des actions et à la mise en relation des éléments du modèle.

CONCLUSION ET PERSPECTIVES

Dans cet article, nous avons détaillé l'étape de transformation d'un ensemble de scénarios en un modèle de comportement synthétisant toutes les activités à proposer dans un système collaboratif capillaire spécialisé en cours de construction. Nous avons également présenté l'outil chargé de cette transformation et de la validation du modèle bâti. Enfin, nos travaux sont en cours pour modifier la visualisation et l'utilisabilité de CBME. Parallèlement nos travaux futurs, comme déjà décrits dans [3], c'est-à-dire l'utilisation de ce modèle comportemental et sa projection (ou celle de ses éléments) sur une architecture logicielle dont le modèle a été prédé-

fini sont dans nos perspectives prochaines. Des recherches se font déjà dans cette direction [1].

BIBLIOGRAPHIE

1. Abi-Aad R., Sinnig D., Radhakrishnan T.; Seffah A., CoU: Context of Use Model for User Interface Designing, in Proceedings of HCI International 2003, vol. 4, (Greece), pp. 8 – 12, LEA, June 2003. HCI International 2003, ISBN: 0262032791.
2. Carroll J.M., *Making Use: Scenario-Based Design of Human-Computer Interactions*. The MIT Press, 2000, ISBN: 0262032791.
3. Carroll J., Rosson M.B., McInerney P., Scenarios in Practice, CHI Workshop 2003.
4. David B.T., Chalon R., Vaisman G., Delotte O., Capillary CSCW. In *Proc. Of IHCI*, Crète, 2003.
5. De Guzman Edward S., Yau M., Gagliano A., Park A., Dey A.K., Exploring the Design and Use of Peripheral Displays of Awareness Information, CHI 2004. ISBN:1-58113-703-6
6. Delotte, O., David, B., Chalon, R., Environnement et démarche pour la construction et l'évolution de systèmes collaboratifs. *Conférence IHM'03*, Caen, 2003, ISBN : 1-58113-803-2.
7. Delotte, O., David, B., Chalon, R., Construction d'un modèle comportemental pour le travail coopératif capillaire. *Conférence IHM'04*, Namur, 2004, ISBN : 1-58113-926-8.
8. Dey A.K., Providing Architectural Support for Building Context-Aware Applications, PhD Thesis, Georgia Institute of Technology, 2000.
9. Larman, C., *UML et les Design Patterns (2^e Ed.)*. Campus Press, 2003, ISBN: 2744016233.
10. Object Management Group : www.omg.org
11. Paris C., Lu S., Vander Linden K., *Environments for the Construction and Use of Tasks Models*, In The Handbook of Task Analysis for Human-Computer Interaction. Edited by Prof. Dan Diaper & Prof. Neville Stanton. Lawrence Erlbaum Associates., 2003.
12. Paternò, F., *Model-Based Design and Evaluation of Interactive Application*. Applied Computing Series, Springer-Verlag (eds.), 2000, ISBN: 1-85233-155-0.
13. Pribeanu C., Limbourg Q., Vanderdonck J., Task Modelling for Context-Sensitive User Interfaces; DSVIS 2001;