

DEVELOPPEMENT D'UNE BOITE A OUTILS INFORMATISEE ET DE SCENARIOS DE PROCESSUS COGNITIFS PERMETTANT DE REPONDRE AUX BESOINS D'UTILISATEURS DEFICIENTS MENTAUX

Thierry Lepoutre, Michel Mercier et Martine Galand

ORIGINE DU PROJET

Le but de notre communication est de présenter une démarche qui consiste à élaborer des logiciels d'aide à la structuration cognitive: aide à la décision et aide à la construction mentale de l'objet. Il s'agit, pour les psychologues et les psychopédagogues, d'élaborer des cahiers des charges pour la réalisation de prothèses cognitives destinées à des personnes handicapées mentales. Nous présentons la manière dont notre démarche s'est construite, à partir de la réalisation de différents projets. Il s'agit d'une note technique qui pourra paraître ardue au lecteur mais qui reprend de manière précise les différentes étapes du projet.

Dans le cadre du département, il existe un centre, le centre PSINHA (PSYchologie, INformatique, HAndicap) qui vise à mettre en oeuvre des moyens pour que l'informatique et la psychologie soient utilisées dans leur développement respectif, au service des personnes ayant une déficience. Différents types d'objectifs sont poursuivis: la recherche, la formation, la documentation, la consultation clinique, le développement de logiciels. Dans les recherches entreprises au cours des dernières années, nous nous sommes rendus compte qu'il fallait repenser le paradigme de développement des logiciels. Même si nous avons déjà le souci de ne pas répondre à des demandes particulières mais de globaliser les demandes, nous nous sommes aperçus que le coût du développement nous obligeait à aller plus loin dans la généralisation. Il fallait concevoir des générateurs de logiciels dans le domaine du développement cognitif qui devraient, eux, répondre aux demandes. C'est pourquoi fut

introduit le concept de développement en plusieurs "couches": un langage de départ souple et sophistiqué, une boîte à outils, des scénarios de processus cognitifs et la mise en oeuvre d'exercices particuliers. Ce concept devrait permettre une conception itérative regroupant les psychologues, pédagogues, informaticiens et les utilisateurs ainsi que l'augmentation de l'efficacité du développement.

UNE DEMARCHE DE DEVELOPPEMENT: UNE APPROCHE PAR PROTOTYPAGE

Une caractéristique essentielle du milieu dans lequel nous travaillons est la pluridisciplinarité des différents acteurs de la conception. En effet, la réalisation d'un outil pédagogique requiert une collaboration étroite entre psychologues, pédagogues, informaticiens et utilisateurs. Un processus de développement traditionnellement linéaire où les utilisateurs ne sont finalement que peu impliqués n'est pas requis pour des applications à haut degré d'interactivité. Il est plus adéquat de suivre une démarche itérative basée sur le principe du prototypage. Le processus global adopté correspond donc à un ensemble d'itérations sur le couple (spécifications implémentation) avec la naissance d'un nouveau prototype à chacune de ces itérations.

Le caractère essentiellement interactif des applications nous amène à scinder les spécifications en deux parties où différents acteurs vont intervenir. Tout d'abord les spécifications des fonctionnalités contenues dans l'énoncé de base du travail et découpant le logiciel à réaliser en grandes fonctions. Ces fonctionnalités sont les

seules fonctions du logiciel; tout le reste n'étant que du dialogue. Ce type de spécifications est réalisé par l'équipe de psychologues.

Les spécifications du dialogue, quant à elles, sont scindées en deux parties. Tout d'abord les spécifications des scénarios, c'est-à-dire le déroulement de la tâche que l'utilisateur va devoir réaliser, l'enchaînement des actions à effectuer pour réaliser la tâche. Ce type de spécifications ne présume en rien de la représentation concrète des différents scénarios. Dans cette étape de spécification, seuls les représentants des utilisateurs (psychologues et pédagogues) interviennent. Leur aide se porte essentiellement sur la création de scénarios efficaces d'un point de vue pédagogique et adaptables aux différents niveaux cognitifs de la population cible. La seconde étape du processus de spécifications consiste en l'élaboration d'exercices concrets. Ce travail se réalise en collaboration avec des représentants des utilisateurs et les utilisateurs eux-mêmes. Le problème épineux est celui de la communication avec les personnes handicapées mentales. Ce problème peut être résolu par deux types de participation:

- participation directe concrétisée par des tests sur des représentations graphiques, sur la compréhension de mots de vocabulaires spécifiques au domaine étudié;
- participation indirecte faisant référence à une phase d'évaluation.

Ces deux types de participation permettent de mettre en évidence certains aspects peu ou mal compris des composantes de l'interface concrète. Les corrections effectuées dans les spécifications permettent de réitérer le processus.

Les exercices constituent l'objet des demandes des utilisateurs potentiels (éducateurs, institutions, personnes handicapées elles-mêmes); ils doivent pouvoir être réalisés par des programmeurs moins spécialisés que ceux qui constituent la boîte à outils et les scénarios. Le coût de réalisation des

exercices particuliers doit pouvoir être pris en charge par les utilisateurs; tandis que le coût de la boîte à outils et des scénarios doit être pris en charge par des subventions de recherche et de développement. Un travail de formation doit être réalisé pour faire prendre conscience aux utilisateurs potentiels des possibilités des scénarios existants qui doivent eux-mêmes être suffisamment généraux pour répondre à des demandes étendues. Parallèlement, les chercheurs et analystes informaticiens doivent eux-mêmes concevoir de nouveaux scénarios pour pouvoir répondre à de nouvelles demandes potentielles (processus itératif).

UNE BOITE A OUTILS ET DES SCENARIOS POUR UNE APPLICATION INTERACTIVE

Etant donnée la multiplicité des intervenants dans la création d'outils adaptés, il est indispensable d'offrir des outils souples et modelables (c'est-à-dire non figés, adaptables à un grand nombre de personnes). Un type d'architecture mis en oeuvre par modules permet de rencontrer cet objectif fondamental. Nous avons donc imaginé un système à quatre couches: la couche reprénant la machine et son environnement de programmation ou (un PC et Windows), la couche représentant une boîte à outils (niveaux de l'informaticien), le niveau des scénarios (imaginés par les chercheurs) et le niveau de l'application interactive (utilisée par la personne handicapée). Cette architecture permet de générer relativement facilement des applications dans différents domaines d'activité.

La boîte à outils

L'objectif de cette boîte à outils est d'assurer une indépendance graphique des applications vis-à-vis de la machine. On pourrait alors penser que cette couche n'est pas nécessaire si l'on décide de ne travailler que sur un seul système! Cependant elle permet une transparence vis-à-vis de la complexité des environnements graphiques que l'on pourrait utiliser. Cette couche doit limiter la puissance des fonctions graphiques de l'environnement et proposer

une interface graphique qui soit plus conviviale (au niveau de l'utilisation de l'application). Limiter la puissance consiste à effectuer certains choix pour produire un interface adapté à un domaine applicatif.

Les scénarios

Un scénario est un modèle générique définissant des messages interactifs, des opérations associées à ces messages interactifs et des règles d'enchaînement de ces messages.

Un message interactif est une unité abstraite de saisie ou de communication d'informations destinées à l'utilisateur. Les opérations que l'utilisateur peut effectuer sur un message interactif sont, par exemple, la saisie et la correction d'une chaîne de caractères, ou encore la clôture du message qui va déclencher les traitements associés à sa terminaison. Les opérations associées aux messages interactifs ne peuvent pas être effectuées à n'importe quel moment. La troisième partie de la tâche consiste en la définition des contraintes de déclenchement des opérations sur les messages interactifs. Ces règles d'enchaînement du dialogue constituent la dynamique de l'exécution de la tâche. Elles spécifient les contraintes liées à l'ordre dans lequel les messages interactifs seront présentés à l'utilisateur.

Ces messages interactifs, opérations et règles d'enchaînement constituent le scénario de l'exécution de la tâche: c'est le niveau abstrait du dialogue avec l'utilisateur. Concrètement, un message interactif sera matérialisé par un objet interactif (une fenêtre, une icône...) et à chaque opération correspondra une action physique (cliquer sur une icône au moyen de la souris, valider la saisie d'une chaîne de caractères en appuyant sur "Return", etc.).

Dans le cadre actuel, cinq scénarios ont été définis: le déplacement d'un ou deux objets sous contrainte d'un trajet prédéfini, une forme d'animation par déplacement d'un curseur, la

désignation d'objets, l'association spatiale et ordonnée d'objets, la mesure et la classification d'objets.

Déplacement d'un ou deux objets sous contrainte d'un trajet prédéfini

L'objectif de ce scénario est de gérer le déplacement d'un ou deux objets graphiques à l'écran. Il vérifie, pour chaque objet, la conformité du déplacement par rapport à un trajet prédéfini (un trajet par objet peut être prédéfini). Lorsque l'utilisateur déplace un des objets en dehors des limites permises (c'est-à-dire lorsqu'il rencontre un obstacle), le scénario réagit en exécutant un traitement particulier.

Sur chaque trajet, il est possible de définir des points chauds. Lorsque l'objet est déplacé sur un point chaud, le scénario réagit en exécutant un traitement particulier. Le scénario peut également réagir à une action de l'utilisateur autre que le déplacement d'un objet. Il est possible d'ouvrir ou de fermer des portes positionnées sur le trajet; une porte fermée se comportant comme un obstacle. Enfin, une fonction particulière doit pouvoir être appelée en fonction d'un "timer" prédéfini.

Animation par déplacement d'un curseur

Considérons un curseur multiple composé de sept images différentes: trois images pour les déplacements vers la gauche, trois images pour les déplacements vers la droite et une image intermédiaire assurant une transition douce lors d'un changement de direction gauche-droite (ou droite-gauche). Lorsque le curseur est déplacé vers la gauche ou vers la droite, les trois images dédiées au déplacement sont affichées de façon cyclique afin de donner une impression d'animation. Notons qu'un déplacement selon l'axe des Y ne donne pas lieu à un changement d'image.

Désignation d'objets

L'objectif de ce scénario est de désigner

successivement différentes parties d'une image affichée à l'écran selon les consignes reçues (la nature de ces consignes est du ressort du programmeur). Le scénario doit vérifier si l'utilisateur a correctement désigné la partie attendue et réagir en exécutant un traitement particulier.

Ce traitement particulier doit être spécifié par le programmeur et peut servir à produire des feedback positifs et négatifs.

Le scénario doit pouvoir proposer les désignations dans un ordre séquentiel (c'est-à-dire tel que défini lors de la préparation des données) ou aléatoire. Lorsque l'utilisateur commet une erreur, le scénario doit pouvoir réagir en répétant, après un certain nombre d'autres énoncés, l'énoncé mal désigné. Normalement à chaque énoncé correspond une seule partie à désigner mais le scénario doit pouvoir gérer une technique de désignation par groupes d'énoncés. Dans ce cas, pour le premier énoncé du groupe, un ensemble de parties de l'image est considéré comme zone de désignation correcte. Cet ensemble de parties est alors diminué de la partie qui a été effectivement désignée. Les parties restantes sont les zones de désignation considérées comme correctes pour le second énoncé du groupe; et ainsi de suite jusqu'à épuisement des énoncés du groupe. Il est à noter que l'ordre de prise en compte des énoncés au sein d'un même groupe est toujours séquentiel.

Association spatiale et ordonnée d'objets

L'objectif de ce scénario est de déplacer successive-ment chaque objet graphique d'une zone de travail pour le positionner sur un autre objet graphique situé dans une zone d'association. Chaque objet déplacé dans la zone d'association peut, à son tour, servir de graphique support pour un objet de la zone de travail; il y a donc un ordre de "précédence" au sein des associations à réaliser.

Lorsque l'utilisateur place un graphique dans la

zone d'association, le scénario doit vérifier si le positionnement est correct (validation spatiale, d'ordonnancement et de séquençement) et réagir en exécutant un traitement particulier. Ce traitement doit être spécifié par le programmeur et peut servir à produire les feedback positifs et négatifs. Le scénario n'effectue aucune vérification quant au positionnement correct des graphiques lorsque l'utilisateur les place dans la zone de travail. Au début de l'exercice, le scénario positionne aléatoirement les objets graphiques dans la zone de travail, excepté un graphique particulier considéré comme point de départ de l'ordre de précédence qui, lui, est déjà positionné dans la zone d'association (une contrainte d'ordonnancement implique la définition d'un point de départ). Le scénario doit permettre à l'utilisateur de faire tourner les graphiques avant de les positionner. L'utilisateur doit pouvoir faire et défaire les associations en respectant les contraintes définies et il doit être possible de spécifier une tolérance plus ou moins grande pour le positionnement des objets graphiques.

Classification d'objets

Le but de ce scénario est de classer des objets graphiques selon un ordre de classification déterminé. Lorsqu'un certain type de classement est demandé, l'utilisateur doit choisir les objets successifs dans la zone de travail de manière à répondre au classement. Lorsqu'un objet graphique est choisi dans la zone de travail, le scénario vérifie si le choix est correct et réagit en exécutant un traitement particulier selon la réponse (vraie ou fausse). Ce traitement particulier doit être spécifié par le programmeur et peut servir à produire les feedback positifs et négatifs.

Lorsque le choix de l'utilisateur est correct, l'objet graphique disparaît de la zone de travail et est réaffiché dans la zone de classement, à l'endroit spécifié lors de la phase de préparation de l'exercice. Le scénario doit pouvoir gérer simultanément plusieurs classifications d'objets. Appelons "pile de classement" l'une de ces

classifications, c'est-à-dire un ensemble d'éléments ayant un certain ordre de classification. La fonction doit donc permettre d'avancer parallèlement dans différentes "piles de classement". Il est ainsi possible de réaliser un exercice dont l'énoncé serait: "Classez les objets par couleurs et, pour une même couleur, par hauteurs croissantes". Dans cet exercice, il y a autant de "piles de classement" qu'il y a d'objets de couleurs différentes. Chaque "pile de classement" peut posséder son propre ordre de classement mais, ici, l'ordre ("hauteurs croissantes") est identique pour toutes les piles.

Un ordre de classement peut ne pas être strict: deux objets graphiques distincts peuvent occuper la même position de classement au sein d'une "pile de classement". Ainsi dans l'exemple précédent, plusieurs objets ayant même couleur et même hauteur peuvent être définis. Par contre, un objet graphique ne peut appartenir qu'à une seule "pile de classement".

Au début de l'exercice, la fonction affiche sur la zone de travail tous les objets graphiques spécifiés en input; certains objets pouvant ne faire partie d'aucune "pile de classement" (objets parasites).

Ce scénario doit également permettre de comparer entre elles des grandeurs abstraites pouvant prendre

des significations concrètes (grandeur, volume, distance, valeur,...). Il doit pouvoir mettre en oeuvre des comparaisons de mesures dont chacun des éléments combinés (additionnés ou soustraits) doit donner une mesure déterminée. La comparaison doit pouvoir être jugée correcte ou non correcte. Des significations concrètes doivent pouvoir être associées aux mesures.

APPEL AUX UTILISATEURS POTENTIELS

Nous sommes conscients que décrits en peu de pages, la boîte à outils et les scénarios peuvent sembler complexes. Il s'agirait pour nous de mettre au point des formations permettant de sensibiliser des utilisateurs potentiels afin de susciter des demandes d'exercices particuliers. Sur cette base les utilisateurs potentiels pourraient demander des exercices particuliers à des coûts abordables: par exemple, apprentissage du code de la route, apprentissage des parties d'une maison, représentations des tâches domestiques par une animation, gestion de l'alimentation, etc. Nous faisons appel à des demandes et nous sommes prêts à mieux informer les utilisateurs potentiels sur les possibilités de développement de logiciels particuliers ajustés à leurs besoins.