

Utilisation d'un client WWW comme interface de l'étudiant dans un système tutoriel intelligent

Roger Nkambou*, Gilles Gauthier+, Claude Frasson*

*Université de Montréal, département d'Informatique et de Recherche Opérationnelle
Succursale Centre-ville, Montréal (Québec) H3C 3J7 Canada
Téléphone: (1) (514) 744 9601
Email: nkambou@iro.umontreal.ca

+Université du Québec à Montréal, département d'Informatique
Succursale A, Montréal (Québec) H3C 3P8 Canada
Téléphone: (1) (514) 987 8212
Email: gauthier.gilles@uqam.ca

Résumé

L'intégration d'un client "World-Wide Web" (WWW) dans un processus d'enseignement-apprentissage nécessite d'étudier la gestion des ressources pédagogiques par le client et la communication avec les autres modules du système d'enseignement. Nous présentons dans ce papier, une architecture de système tutoriel intelligent (STI) utilisant un client WWW comme interface de l'apprenant. Cette interface permet de supporter le déroulement de toute ressource pédagogique faisant partie d'une matière à enseigner par un STI.

Mots clés: STI, WWW, Interface de l'étudiant, Représentation de connaissances.

Abstract

The main object of this paper is to show how a World-Wide Web (WWW) client can be used as an Intelligent Tutoring System (ITS) student interface. An ITS architecture that integrates such an interface is presented. This ITS includes a subject-matter knowledge model component containing multimedia pedagogical resources. We also show how the proposed interface supports the delivery of a complete course.

Keywords: ITS, WWW, Student interface, Knowledge representation

1. Introduction

L'intégration du multimédia dans le processus de l'enseignement devient de plus en plus incontournable. Cependant, très peu de moyens sont disponibles dans les STI actuels pour supporter cette intégration. Par exemple, dans Sherlock (Lajoie et al., 1989), les vidéos interactifs sont utilisés.

L'avènement du WWW a résolu de nombreux problèmes relatifs à la création des documents multimédias. Plus encore, on tend vers des ressources multimédia interactives et distribuées avec l'avènement du langage Java (Friendly, 1995) et de son support WWW HotJava (SUN, 1995). Exploiter ces technologies dans un système tutoriel intelligent (STI) peut contribuer à résoudre certains problèmes reliés à l'intégration du multimédia et de l'hypermédia et à faciliter la distribution des STI. Cependant, alors que la navigation dans un client WWW reste une navigation libre (sans contrôle des actions de l'utilisateur), un STI a besoin de contrôler la navigation d'un étudiant dans une session d'enseignement-apprentissage. Il peut même contraindre l'étudiant dans certaines

situations. Ce contrôle lui permet d'être capable d'évaluer les apprentissages d'un étudiant afin de s'assurer de l'évolution cognitive de ce dernier. Le contrôle des actions de l'étudiant permet aussi au STI de suivre son raisonnement, pour être capable de l'analyser afin d'intervenir de manière pertinente au cours de l'apprentissage.

Notre problème consiste à intégrer cette couche de contrôle dans un client WWW afin de pouvoir l'exploiter dans un STI, tout en assurant la communication avec ce dernier. Dans ce qui suit, nous présentons l'approche que nous avons utilisé pour supporter cette intégration, et l'architecture du STI auquel nous avons abouti. Le domaine d'application retenu est l'enseignement du code de la route au Québec. Un autre domaine d'application est en cours de conception: l'enseignement de la géographie du Cameroun.

2. Représentation et organisation des connaissances

Nous avons spécifié et développé un modèle de représentation de la matière qui permettrait de contrôler l'acquisition, par un étudiant, des connaissances relatives à la matière enseignée. Nous présentons dans ce paragraphe un bref aperçu de ce modèle dont les détails sont dans (Nkambou et Gauthier, 1996a). Ce modèle permet de spécifier une matière au niveau domaine, pédagogique et didactique.

2.1. Connaissances du domaine

L'aspect domaine est mis en oeuvre à travers une représentation des connaissances impliquées dans la matière (concepts, procédures, faits, stratégies...) et une définition des liens existants entre ces connaissances (analogie, abstraction, généralisation...). Le modèle obtenu possède un intérêt certain dans un processus d'enseignement-apprentissage: l'explicitation des connaissances que l'on veut faire acquérir à l'étudiant. Il sert aussi à des inférences qui peuvent être fait dans un modèle de l'apprenant (Nkambou, Lefebvre et Gauthier, 1996). La figure 1 montre un exemple de multi-graphe représentant une portion du modèle de connaissances pour le code de la route. Dans ce graphe, chaque forme de noeud correspond à un type de connaissance, et chaque forme de lien représente un type de relation entre deux connaissance du domaine. Par exemple, les concepts sont représentés par des ronds, alors que les règles (connaissances procédurales) sont représentées par des carrés arrondies.

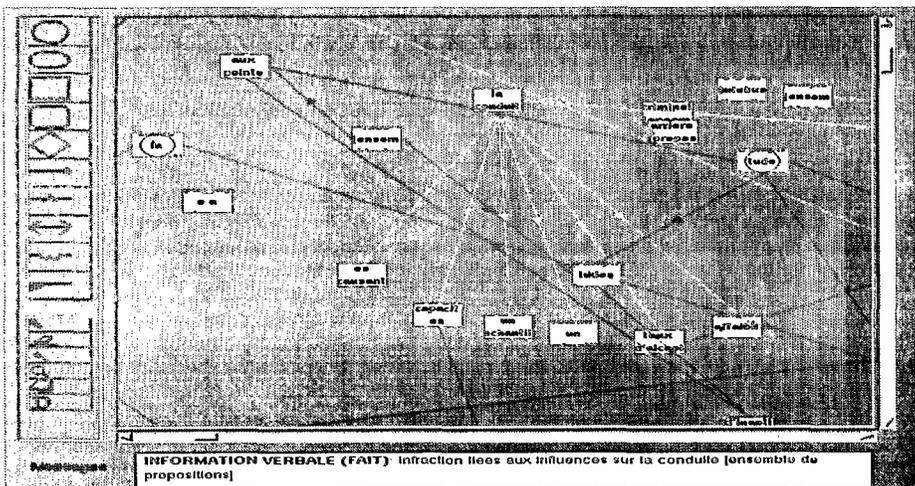


Figure 1: Une portion du modèle des connaissances sur le code de la route au Québec.

2.2. Aspects pédagogiques

Sur le plan pédagogique, notre approche offre la possibilité de spécifier, d'un point de vue du système et de l'étudiant, les objectifs d'enseignement poursuivis. Cette spécification permet notamment de prévoir les connaissances qui seront considérées comme préalables à l'atteinte d'un objectif, et d'étudier l'impact que peut avoir une telle atteinte sur l'acquisition des connaissances du domaine (par l'apprenant). Ceci abouti à un modèle pédagogique qui sera exploité à tous les niveaux du STI, notamment pour la génération automatique d'un cours, pour la planification de l'enseignement et pour supporter le déroulement d'un cours (Nkambou et al., 1995a).

2.3. Aspects didactiques

Une fois que le domaine est défini et que les objectifs d'enseignement sont fixés, il est utile d'étudier les moyens à mettre en oeuvre pour pouvoir permettre la réalisation de ces objectifs. Ceci consiste à définir les tactiques (ressources pédagogiques) qui sont susceptibles de favoriser l'atteinte des objectifs, et par là, l'acquisition des connaissances désirées. C'est à ce niveau que l'apprentissage a effectivement lieu à travers l'interaction entre l'étudiant et la ressource pédagogique (problème à résoudre, démonstration, enseignement de concepts, tests...).

La présentation d'une ressource pédagogique à un étudiant nécessite de disposer d'un support de présentation approprié. Étant donné que notre approche de modélisation prévoit des ressources multimédia et hypermédia dont l'intérêt pédagogique n'est plus à démontrer (Marchionini, 1988; Jonassen et Grabinger, 1990), nous avons opté pour l'utilisation d'un client WWW. Cette préférence est principalement due à la capacité d'un client WWW à supporter l'hypermédia et le multimédia qui, comme on le verra dans le prochain paragraphe, se retrouve à tous les niveaux des ressources pédagogiques nécessaires pour l'enseignement de certaines matières (la géographie par exemple). Une autre raison est notre désir d'arriver plus tard à une architecture distribuée des STI qui pourrait représenter un pas important dans l'apprentissage à distance.

Nous avons développé un environnement auteur permettant de construire des curriculums et des cours destinés à un STI suivant l'approche précédente (Nkambou et al., 1995b). Cet environnement contient un ensemble d'outils et de méthodologies permettant de produire des objets curriculum et cours directement utilisables par les autres modules d'un système tutoriel intelligent.

3. Ressources pédagogiques utilisables sur le WWW

Les ressources pédagogiques jouent un rôle important dans le processus d'apprentissage. En effet, elles représentent l'interface entre l'étudiant et les autres modèles de connaissances. C'est à travers l'interaction entre l'étudiant et une ressource pédagogique que ce dernier acquiert des connaissances et est évalué. Il est important d'étudier les différentes catégories de ressources selon le rôle qu'elles peuvent jouer dans le processus d'apprentissage.

3.1. Typologie

L'analyse de différentes ressources pédagogiques et de leur impact dans le processus d'apprentissage nous a permis de produire une typologie (figure 2). Cette typologie fait ressortir plusieurs catégories de ressources pédagogiques:

- Les *ressources supports* qui sont utilisées dans le processus d'enseignement-apprentissage pour favoriser l'acquisition de connaissances (problèmes, démonstrations, document d'enseignement de concepts...);
- Les *ressources de type expertise* qui oeuvrent comme des experts virtuels pouvant

intervenir dans le processus d'enseignement-apprentissage pour aider, conseiller ou critiquer l'apprenant dans une activité précise;

- Les *ressources physiques et médiatiques* représentant du matériel d'enseignement pouvant être utilisées dans certaines situations d'apprentissage.

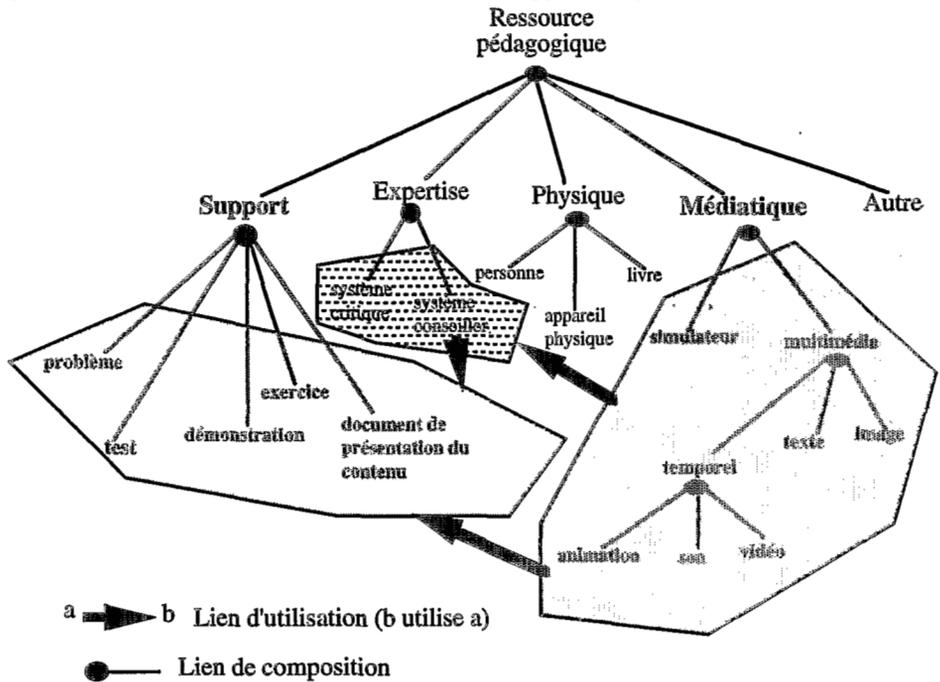


Figure 2: une typologie des ressources pédagogiques

Les ressources sont reliées dans cette typologie par des liens de composition (ce qui abouti à une hiérarchie de ressources), et des liens d'utilisation (une ressource peut utiliser une autre). Ainsi, les ressources supports peuvent utiliser les ressources médiatiques et expertises, alors que les ressources expertises peuvent utiliser les ressources médiatiques. Par exemple, les problèmes et les démonstrations portant sur un simulateur (*simulation-based problem*, *simulation-based demonstrations* (Nkambou et al., 1995c)) utilisent un simulateur. Un document de présentation de contenu peut utiliser des image, des vidéos...

Dans ce papier, nous nous intéressons uniquement aux ressources exploitables par un client WWW (ressources apparaissant en gras dans la figure 2).

Quatre types de ressources sont réalisables sous la forme d'un document HTML:

- des documents de présentation du contenu d'un cours
- des documents d'évaluation présentant des tests à l'apprenant
- des exercices et des problèmes
- des documents de démonstration et de simulation.

3.2. Approches de construction des ressources pédagogiques utilisables sur WWW

3.2.1. Construction systématique. Parmi les différents types de documents cités plus haut,

nous nous sommes intéressé aux documents de présentation du contenu d'un cours et aux documents d'évaluation. En ce qui concerne le contenu du cours, nous nous sommes inspiré essentiellement de l'analyse du contenu effectué dans le cadre du projet SAFARI. Cette analyse fait ressortir un certain nombre de connaissances (concepts, règles et information verbales) reliées à la matière, pour lesquelles des ressources de type document d'enseignement doivent être prévues. Par exemple, dans le cas de la conduite automobile au Québec, il est important de prévoir du matériel pour l'enseignement de la signalisation routière (panneaux de signalisation, marques sur la chaussée, feux de signalisation, et leur rôle) de même que des règles de la circulation routière pour ne citer que celles-ci. Le type de ressource à considérer dépend de la connaissance impliquée. Par exemple, l'enseignement de la signalisation routière nécessite la présentation d'images correspondant aux signaux, alors que celui des règles de circulation routière nécessite des vidéos de simulation. Ces ressources ont été recueillies des document traitant du code de la route.

L'analyse du contenu a conduit à un curriculum impliquant la construction de plusieurs ressources de présentation. Ces ressources ont été explicitement implémentées (avec le langage HTML (W3 Consortium, 1995)) comme des documents HTML. La granularité des documents dépendait de la structuration des connaissances impliquées par la matière concernée. Par exemple, nous avons créé un document présentant l'information verbale sur l'acuité visuelle considérée comme concept défini dans un curriculum du code routier québécois. Les documents sont modulaires et correspondent aux différents concepts associés à cette matière. Les liens entre les différents concepts définis dans le curriculum serviront pour guider l'exploitation de ces ressources lors du processus d'apprentissage et/ou d'enseignement.

3.2.2. Génération automatique: cas des ressources de type test. Étant donné un objectif d'un cours, il existe une multitude de questions d'évaluation pouvant mesurer l'atteinte de cet objectif. La génération dynamique de test présente des avantages certains qui sont:

- le gain en place mémoire: il n'est plus nécessaire de produire explicitement tous les tests possibles et de les stocker pour les présenter ensuite à l'apprenant;
- la variété des tests: la génération dynamique offre au système d'enseignement la possibilité de décider du contenu exact du test ainsi que de l'objet du test en tenant compte de l'objectif à évaluer et de paramètres concernant l'apprenant.

Les tests que nous avons réalisés portent essentiellement sur la signalisation routière. Ces tests sont générés à partir de ressources qui sont en fait les panneaux de signalisation numérisés et stockés dans une base d'images, et intégrés dans le STI comme la partie physique d'un concept concret (Nkambou, Lefebvre et Gauthier, 1996) de type signalisation. Les tests générés ont essentiellement pour but de mesurer la capacité de l'apprenant à identifier, reconnaître, classifier ou généraliser les concepts relatifs aux panneaux de signalisation. Les quatre niveaux d'acquisition retenu proviennent du vocabulaire d'évaluation des concepts de Klausmeier (1990). Les types de test à considérer varient suivant le niveau d'acquisition à vérifier chez l'apprenant. Nous avons aussi implémenté un module de génération de tests de type QVF (question vrai ou faux), QCM (Question à choix multiples) et des tests de type *phrase trouvée*, moyennant une spécification de paramètres à entrer par l'auteur. Tous les tests générés sont directement exploitables par un clients WWW (Nkambou, Gauthier, 1996b).

3.3. Encapsulation des ressources pour fin d'utilisation par le STI

Pour intégrer les ressources de type HTML créées ou générées (documents de présentation de concept, tests, ...) dans un STI, nous avons proposé une encapsulation ce celles-ci comme des objets dont la structure varie selon le type de ressource. Un objet HTML de type présentation (document d'enseignement de concepts et d'informations verbales) comprend les messages *présenter*, *arrêter-la-présentation*, *retourner-la-trace...* Un objet HTML de type test comprend les messages *générerPrésenter;*, *présenter;*,

évaluer... La méthode *générerPrésenter*: est une méthode de classe qui permet de générer et de présenter un nouveau test suivant les paramètres spécifiés. Ces paramètres varient suivant le type de test. Cette méthode se spécialise en fonction de la sous-classe de test concernée. Le STI peut décider (pour des raisons de révision par exemple), de retenir un test généré. Ce test sera alors stocké dans la base des tests comme un objet pouvant répondre à un message de présentation. Toutes les classes de ressources pédagogiques utilisables sur WWW ont été implémentées avec le langage Smalltalk.

4. Le STI

4.1. Architecture fonctionnelle

La figure 3 montre l'architecture fonctionnelle d'un STI intégrant un client WWW comme interface de l'apprenant.

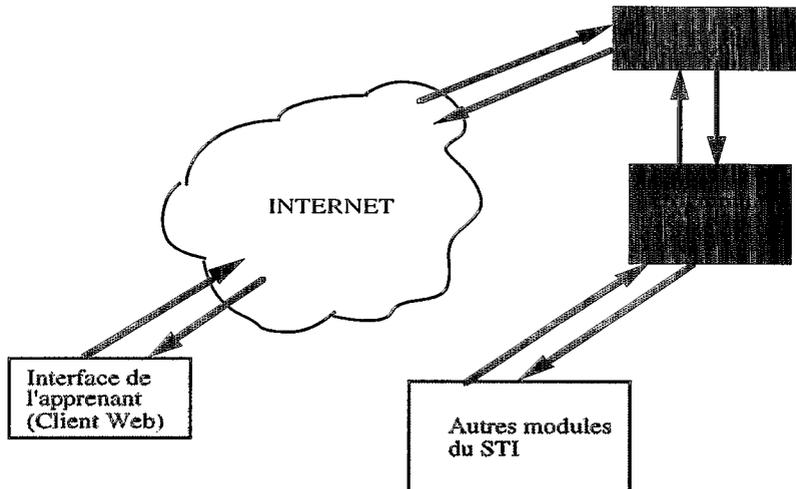


Figure 3: architecture fonctionnelle du STI proposé

- L'interface de l'apprenant est un client Web de type Netscape 2.0. Elle est chargée de supporter la présentation d'une ressource pédagogique quelconque à l'apprenant.
- *HTTPD Server* est un noeud quelconque sur Internet.
- Module CGI (*Common Gateway Interface*) (NCSA, 1995; Liu *et al.*, 1994) est le module qui contient les programmes utilitaires destinés au traitement des actions de l'étudiant.
- *Autres modules du STI* comprend les modules planificateur, tuteur, modèle de connaissances de la matière (curriculum), cours, modèle de l'apprenant et ressources pédagogiques. L'architecture interne de ce système a été présentée dans Nkambou *et al.* (1995a).

4.2. Dynamique de l'architecture

Les activités pédagogiques (enseignement de concept et d'informations verbales, résolution de problèmes et de tests, démonstrations, ...) sont planifiées par le STI (fonctionnement interne prenant en compte le modèle de l'apprenant). Un module du STI (Tuteur) commande la présentation de la ressource retenue comme activité courante, en générant une page HTML à partir de l'objet représentant cette ressource. Cette page est alors récupérée par le CGI qui va l'envoyer au serveur pour être chargée sur le client Web sur lequel la session d'apprentissage a été ouverte. Le client reçoit le document et le charge dans son interface (voir plus loin). Les actions de l'étudiant sont transmises aux programmes du CGI via le serveur qui va transmettre au STI les informations reçues. Ces

informations sont traitées par le STI et une décision est prise pour la suite de l'apprentissage, et le cycle recommence.

Tout au long de cette dynamique, l'interface doit dynamiquement changer en fonction du déroulement des activités. Par exemple, lors d'une activité de résolution de problème ou de traitement de test, le conseiller peut réagir par rapport aux actions de l'étudiant (en changeant de face ou de couleur, pour exprimer une situation. Par exemple couleur rouge pour indiquer qu'il y a un problème). L'étudiant peut aussi accéder au conseiller pour savoir plus sur sa réaction. Pour permettre cet aspect, nous avons pensé à représenter le conseiller dans l'interface par une image dynamique. Ceci a été possible grâce à l'utilisation d'un *server push* et d'un *client pull* (Davison, 1995). Ces deux concepts nous permettent d'assurer une communication continue entre le client et le serveur sans que cela ne paraisse à l'étudiant.

L'interface proposée donne la possibilité à un apprenant de faire appel aux services d'un conseiller *on-line* concernant l'activité en cours.

5. Implémentation

Le système construit avec cette approche offre deux modes d'utilisation: le mode exploration libre (WebExplorer) et le mode tuteur (WebITS). Au début d'une session d'apprentissage, une interface permet à l'apprenant de s'identifier (pour permettre de récupérer son modèle) et de préciser le mode qu'il désire.

Dans le premier mode, l'initiative est laissée à l'apprenant. Ainsi, le rôle du STI est limité au contrôle des activités réalisées par l'étudiant et à la mise-à-jour de son modèle. Le contrôle consiste à intervenir pour conseiller l'étudiant ou pour rendre compte de ses performances. Après le choix d'un cours à suivre, WebExplorer récupère les activités disponibles pour ce cours (à partir du ou des curriculums concernés) et les met à la disposition de l'étudiant (figure 4). La figure 5 montre un exemple de réaction de WebExplorer à la performance de l'étudiant.

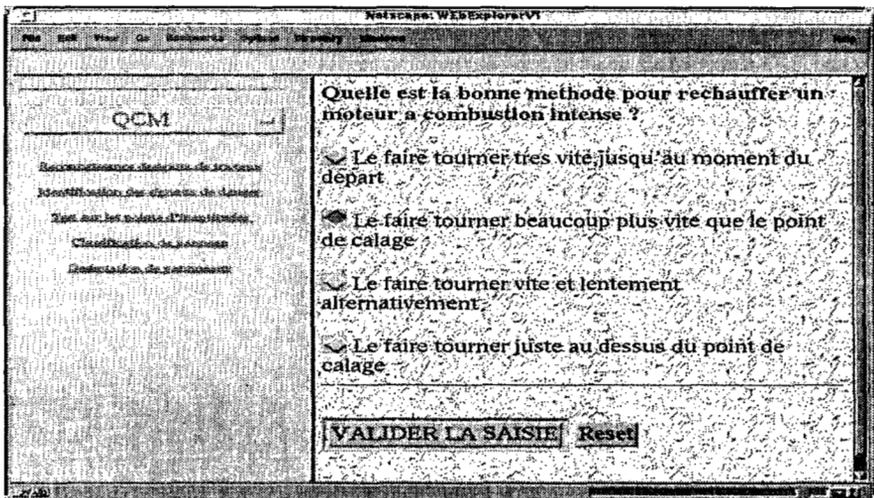


Figure 4: L'activité choisie est une question à choix multiples

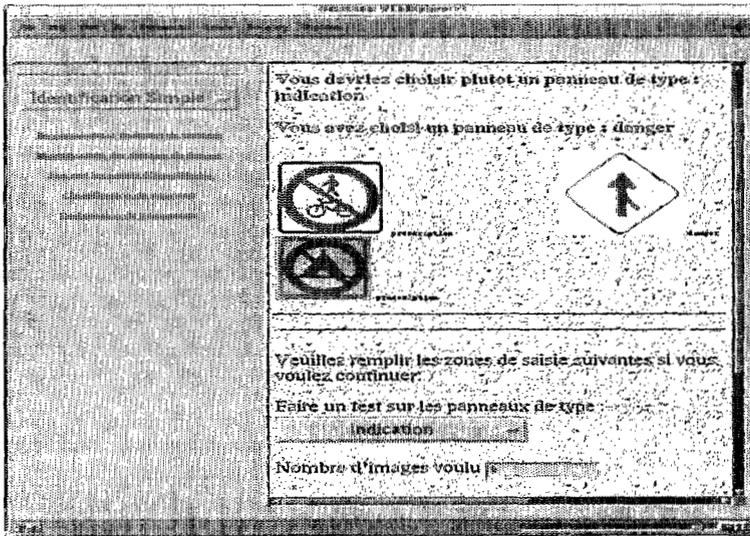


Figure 5: Réaction du système à la performance de l'étudiant.

Ces activités sont regroupées par catégories suivant leur type. L'étudiant peut alors choisir le type d'activité qu'il veut faire et les ressources disponibles pour la catégorie choisie apparaîtront dans une liste dans laquelle il peut choisir une ressource à exécuter.

Dans le second mode, c'est le tuteur qui prend l'initiative. WebITS décide de l'activité à faire en fonction des connaissances de l'étudiant et des objectifs poursuivis par l'enseignement. Il a l'initiative d'abandonner la présentation d'une ressource et de replanifier une autres ressources suivant le déroulement de l'enseignement (figure 6). En plus, il contrôle le déroulement des activités et décide de la mise à jour des connaissances de l'étudiant. Un conseiller on-line est disponible pour pouvoir répondre à des questions du type 'pourquoi', 'quoi' et 'comment' concernant l'activité courante.

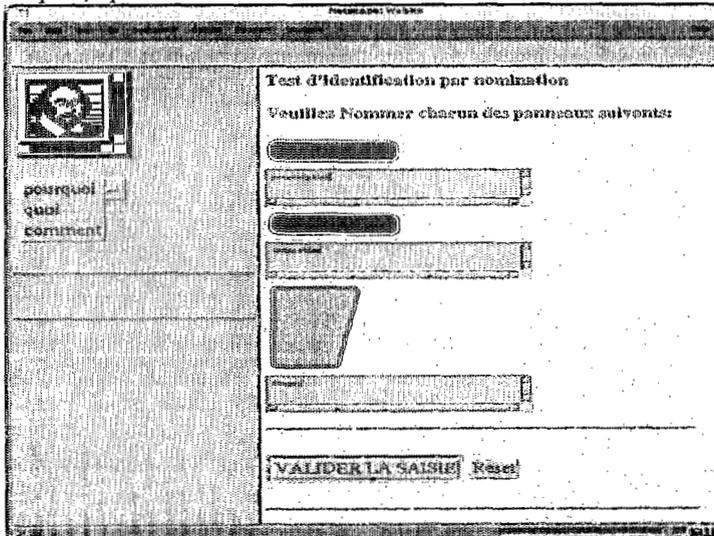


Figure 6: Test de désignation généré.

6. Conclusion

Nous avons présenté une approche d'intégration d'un client Web dans un STI comme interface de l'apprenant. Cette intégration représente une solution à l'exploitation des ressources multimédia et hypermédia dans un système tutoriel intelligent. Cette approche a été exploitée pour la construction des cours sur l'enseignement du code la route au Québec (Nkambou, Gauthier, 1996b). Un cours complet utilisant l'approche proposée pour l'enseignement de la géographie du Cameroun est en cours de construction. Une démonstration de ce système pourrait être faite à la conférence. Un URL sera disponible pour permettre de l'utiliser.

Recherches futures: L'idée d'une architecture de STI distribuée devient un axe important de recherche dans le domaine des STI. Une telle distribution favoriserait l'enseignement à distance (*tele-learning*). Le système proposé dans ce papier constitue un pas vers cette architecture: la séparation du poste d'apprentissage (interface de l'étudiant) des autres composantes du STI. Cette séparation a été rendu possible grâce à l'utilisation d'un client Web comme interface de l'étudiant. Nous nous préoccupons maintenant à passer à une distribution complète des autres composantes. On imaginerait d'ici là, un STI dont le module tuteur serait sur une machine à Libreville, le planificateur à Paris, et le module curriculum à Montréal. Comment peut-on mettre en oeuvre une telle architecture? Nous pensons à exploiter les outils tels que SGML et JAVA pour répondre à cette question.

Remerciements: Nous aimerions remercier le MICST (Ministère de l'Industrie, du Commerce, de la Science et de la Technologie du Québec) pour leur financement au projet SAFARI (Programme SYNERGIE) où ce travail a été initié.

Références

- Davison, A. (1995). Animation Using the Netscape Browser. Dr. Dobb's Sourcebook, November/December 1995. PP. 13-19.
- Frasson, C. et Gauthier, G. (1994). A gradual Software Environment for Developing Tutoring Systems. Proceedings of the 7th International Conferences on Systems Research, Informatics and Cybernetics, Baden-Baden: Advances in Artificial Intelligence- Theory and Application II, pp. 73-78.
- Friendly, L. (1995). The Design of Distributed Hyperlinked Programming Documentation. Dans *Proceedings of the International Workshop on Hypermedia Design*. pp. 159-183. Montpellier, France.
- Jonassen, D.H. et Grabinger, R.S. (1990). Problem and issues in designing hypertext/hypermedia for learning. In D.H. Jonassen and H. Mandl (eds.), *Designing Hypermedia for Learning*, NY: Springer Verlag.
- Klausmeier, H.J. (1990). Conceptualizing. In Jones, B.F. et Idol, L (Eds): *Dimensions of Thinking and Cognitive Instruction*; pp 93-138; LEA, NJ.
- Lajoie, S., Lesgold, A. et al. (1989) A procedural guide to the avionics troubleshooting tutor development process. Learning research and development center. Research report; University of Pittsburgh.

- Liu, C., Peek, J., Jones, R., Buus, B., et Nye, A. (1994). *Managing INTERNET Information Services*. Sebastopol, CA: O'Reilly & Associates
- Marchionini, G. (1988). *Hypermedia and Learning: freedom and chaos*. *Educational Technology*, 28(11), pp. 8-12.
- NCSA (1995). *The NCSA HTTPd Home Page*.
<http://hoohoo.ncsa.uiuc.edu/>
- Nkambou, R., Gauthier, G. and Frasson, C., Seffah, A. (1995a). *Une architecture de STI avec une composante curriculum explicite*. In *Environnement Interactif d'Apprentissage par Ordinateur*, pp.315-327, Eyrolles, Paris.
- Nkambou, R., Gauthier, G. and Frasson, C., Antaki, M. (1995b). *Integrating Expert System in an Authoring System for Curriculum and Course Building*. Dans *Proceedings of the 7th International Conference on Artificial Intelligence and Expert Systems Application*. San Francisco, CA.
- Nkambou, R., Quirion, L., Kaltenbach, M. et Frasson, C. (1995c). *Using Multimedia In Learning about Processes; DEGREE: A simulation-based Authoring System For Multimedia Demonstration Building*. Dans: *Multimedia Modelling: Toward information superhighway*. Word Scientific publishing, Singapore. PP. 365-378.
- Nkambou, R. Lefebvre, B. et Gauthier, G. (1996). *A Curriculum-Based Student Modelling for ITS*. Dans: *Proceedings of the fifth international conference on User Modeling*, pp. 91-98, Kanula-Kona, Hawaii.
- Nkambou R. et Gauthier, G. (1996a). *Un modèle de représentation du curriculum dans un système tutoriel intelligent*. Dans: *Proceedings of the third International conference on Intelligent Tutoring Systems*, Springer-Verlag, Berlin.
- Nkambou, R., Gauthier, G. (1996b). *Use of WWW resources in an Intelligent Tutoring System*. *Proceedings of the world conference on Educational Multimedia (Ed-Media-96)*. AACE, Boston.
- SUN (1995). *The HotJAVA Browser: A White Paper*. <http://java.sun.com/1.oalpha2/doc/overview/hotjava/index.html>
- W3 Consortium (1995). *HyperText Markup Language (HTML): Working and Background Materials*. <http://www.w3.org/pub/WWW/MarkUp/MarkUp.html>