



HAL
open science

Modèle pour la conception et le réalisation de formations au travers de réseaux : l'environnement ECSAIWEB

Charun Sanrach

► **To cite this version:**

Charun Sanrach. Modèle pour la conception et le réalisation de formations au travers de réseaux : l'environnement ECSAIWEB. Informatique [cs]. Institut National Polytechnique de Lorraine, 2001. Français. NNT : 2001INPL050N . tel-01750339

HAL Id: tel-01750339

<https://hal.univ-lorraine.fr/tel-01750339>

Submitted on 29 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

01JNPL050N



Département de formation doctorale en informatique

Institut National
Polytechnique de Lorraine

École doctorale IAE + M

Modèles pour la conception et la réalisation de formations au travers de réseaux L'environnement ECSAIWeb

THÈSE

présentée et soutenue publiquement le 14 Septembre 2001

pour l'obtention du

Doctorat de l'Institut National Polytechnique de Lorraine
(spécialité informatique)

par

Charun SANRACH

Service Commun de la Documentation
INPL
Nancy - Brabois

Composition du jury

- | | | |
|--|--|--|
| <i>Président et rapporteur interne :</i> | M. Jean-Claude DERNIAME | Professeur à l'INPL |
| <i>Rapporteurs :</i> | M. Jean-Pierre PECUCHET
M. Pierre TCHOUNIKINE | Professeur à l'INSA, Rouen
Professeur à l'Université
du Maine |
| <i>Examineurs :</i> | Mme Monique GRANDBASTIEN
Mme Elisabeth GAVIGNET | Professeur à l'UHP Nancy I
(Directrice de Thèse)
Maître de Conférence
à l'IUFM de Bourgogne |



Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503



136027639 9

(M) 2001 SANRACH, C

Mis en page avec la classe thloria.

Remerciements

Je remercie vivement toutes les personnes qui m'ont aidé et encouragé tout au long de ce travail et plus particulièrement :

Mme Monique GRANDBASTIEN, professeur à l'Université Henri Poincaré, mon directeur de recherche. Je la remercie de m'avoir accueilli au sein de l'équipe «*Informatique et Formation*» et de m'avoir apporté un soutien régulier et efficace. Ses compétences concernant l'informatique d'une part et l'enseignement d'autre part, sa disponibilité malgré des charges importantes et son amabilité constante, m'ont été très précieuses, je lui en suis très profondément reconnaissant.

M. Jean-Claude DERNIAME, professeur à l'Institut National Polytechnique de Lorraine, pour avoir accepté d'être rapporteur interne de cette thèse. Son soutien au début de ce travail a été très important pour me permettre de continuer.

Je remercie également **M. Pierre TCHOUNIKINE**, professeur à l'Université du Maine et **M. Jean-Pierre PECUCHET**, professeur à l'INSA Rouen pour avoir accepté d'être rapporteurs de cette thèse.

Je remercie **Mme Elisabeth GAVIGNET**, maître de conférence à l'IUFM de Bourgogne, pour avoir accepté d'être examinatrice de cette thèse et pour le travail qu'elle a proposé dans ECSAI, sur lequel j'ai pu m'appuyer au début de ma thèse.

Je remercie tous les membres de l'équipe *Information et Formation* et plus particulièrement **Mme Marilynne ROSSELLE** qui m'a encouragé tout au long de ce travail.

Je tiens aussi à remercier tous ceux qui ont participé à l'expérimentation d'ECSAIWEB, spécialement **M. Mönchai TIANTONG**, professeur associé, directeur de département Information et formation et les étudiants maîtres en «*master degree*» du même département, King Mongkut's Institute of Technology North Bangkok, Thaïlande.

Je remercie très profondément le Président de l'Institut National Polytechnique de Lorraine et l'équipe des Relations Internationales qui m'ont accueilli très chaleureusement à l'INPL, et en particulier **Mme Catherine SOMMIER**.

Je veux enfin remercier toute ma famille pour l'aide logistique qu'elle m'a apportée.

*À ma grand-mère,
mon père et ma mère.*

Table des matières

Remerciements	i
Introduction	1
Partie I Etat de l'art et cahier des charges : d'ECSAI à ECSAIWEB	7
Chapitre 1 Motivations et Systèmes existants	9
1.1 Introduction	9
1.2 Motivations	10
1.2.1 Plateforme indépendante	10
1.2.2 À prix réduit	10
1.2.3 Facilité d'entretien	11
1.2.4 Portabilité	11
1.2.5 Distribué	11
1.2.6 Intégration Internet	11
1.2.7 Médias riches	11
1.2.8 Serveur extensible.	11
1.3 Caractéristiques de SEW	12
1.4 Environnements intégrés	12
1.5 Systèmes WEDS existants	12
1.6 Études comparatifs des systèmes WEDS	13
1.6.1 Entités gérées par la plate-forme	13
1.6.2 Fonctionnalités offertes aux différents intervenants	17
1.6.3 Conclusion	20
1.7 Conclusion	20
Chapitre 2 Évolution d'ECSAI vers le Web	23
2.1 Introduction	23
2.2 Système ECSAI	23

2.2.1	Prototype du système ECSAI	24
2.2.2	Modèles d'ECSAI	24
2.2.3	Générateur du parcours d'apprentissage	31
2.2.4	Modes d'apprentissage	33
2.2.5	Points forts et faibles du système ECSAI	33
2.3	Passage d'un système autonome à un système sur le Web	34
2.3.1	Architecture des systèmes basés sur le Web	34
2.3.2	Avantages et désavantages des architectures	36
2.4	Evolution des «formations pilotées par des enseignants» à des «formations centrées sur des apprenants»	37
2.5	Evolution des formations traditionnelles aux formations à distance	38
2.6	Application à l'évolution d'ECSAI	39
2.6.1	Liste des conditions pour un système sur le Web	39
2.6.2	Contraintes d'implantation	40
2.7	Conclusion	40
Chapitre 3 Adaptation d'Hypermédia		43
3.1	Introduction	43
3.2	Adaptation dans les systèmes d'apprentissage basés sur le Web	43
3.2.1	Présentation adaptative	44
3.2.2	Support adaptatif de Navigation	45
3.3	Application des adaptations d'hypermédia	47
3.3.1	Techniques Adaptatives de Présentation	47
3.3.2	Techniques Adaptatives de Support de Navigation	50
3.4	Conclusion	54
Conclusion vers un «cahier des charges» pour ECSAIWEB		55
Partie II Conception et Réalisation d'ECSAIWEB		59
Chapitre 4 Modélisation des ressources pédagogiques		61
4.1	Introduction	61
4.2	Items d'évaluation	61
4.2.1	Rôle des items d'évaluation	61
4.2.2	Expression des objectifs pédagogiques	62
4.2.3	Opérations sur les items	64
4.2.4	Définition d'opérations sur un ensemble d'items	66

4.3	Unités logiques d'apprentissage	68
4.3.1	Structure des unités logiques d'apprentissage	68
4.3.2	Définition d'opérations sur les unités logiques	68
4.4	Unités physiques d'apprentissage	71
4.4.1	Unités physiques de type «PRESENTATION»	71
4.4.2	Unités physiques de type «EXERCICE»	74
4.5	Base de tests	85
4.6	Conclusion	85
Chapitre 5 Modèle des élèves		89
5.1	Introduction	89
5.2	Rôle du modèle des élèves	89
5.3	Représentation et Contenu du modèle d'élève	91
5.3.1	Modèle de recouvrement	91
5.3.2	Réseaux bayésiens	91
5.3.3	Ce qui est inclus dans le modèle d'élève	92
5.4	ECSAIWEB: Modèles de recouvrement multi-points de vue	93
5.4.1	Généralités	93
5.4.2	Modèle pour le mode guidé	94
5.4.3	Modèle pour le mode libre	94
5.4.4	Visualisation et Adaptation	96
5.5	Conclusion et Evolution	97
Chapitre 6 Fonctionnalités proposées pour ECSAIWEB		99
6.1	Introduction	99
6.2	Interfaces graphiques adaptatives d'ECSAIWEB	99
6.2.1	Type d'interfaces	101
6.2.2	Formulaires	104
6.2.3	Outils de gestion d'interfaces adaptatives	105
6.3	Description fonctionnelle de l'environnement	108
6.4	Fonctionnalités de l'environnement enseignant-auteur	108
6.4.1	Définition des items d'évaluation	112
6.4.2	Détermination des objectifs pédagogiques	112
6.4.3	Description des unités logiques d'apprentissage	112
6.4.4	Edition des unités physiques d'apprentissage	113
6.4.5	Vérification de cohérences	119
6.4.6	Simulation du parcours d'apprentissage	121

6.4.7	Création des examens	122
6.4.8	Importation et Gestion des fichiers	124
6.4.9	Gestion des profils d'élèves	124
6.5	Fonctionnalités de l'environnement élève	125
6.5.1	Mode libre	127
6.5.2	Mode guidé	130
6.5.3	Navigation adaptative	135
6.5.4	Barres de navigation	135
6.6	Fonctionnalités de l'environnement partagé	136
6.6.1	Courrier électronique	136
6.6.2	Forum de discussion	139
6.6.3	ChatRoom	140
6.7	Fonctionnalités de l'outil administratif	141
6.8	Conclusion	143
Chapitre 7 Architecture et Implantation réalisée		145
7.1	Introduction	145
7.2	Contraintes de mise en œuvre	145
7.3	Architecture du système ECSAIWEB	146
7.3.1	Environnement auteur	146
7.3.2	Environnement élève	146
7.3.3	Environnement partagé	147
7.4	Organisation des programmes	148
7.5	Programmes principaux	148
7.6	Modules intermédiaires	149
7.6.1	Module LireParam	149
7.6.2	Module DOMAIN	153
7.6.3	Module PRESENTATION	156
7.7	Bases de données	159
7.8	Générateur des menus	160
7.9	Générateur d'un parcours d'apprentissage	160
7.10	Conclusion	164
Partie III Bilan et perspectives		165
Chapitre 8 Conclusion		167

Glossaire	173
Bibliographie	177
Liste de Publications	185

Table des figures

2.1	Diagramme du système ECSAI	24
2.2	Représentation des concepts dans les différents contextes d'utilisation	25
2.3	Composants d'une unité pédagogique	28
2.4	Exemple de description d'une unité d'apprentissage	28
2.5	Liens pour l'apprenant X	31
2.6	Processus de raffinement du choix de la prochaine unité d'apprentissage	32
2.7	Architecture centralisée: PATOnline	35
2.8	Architecture décentralisée: DCG	35
2.9	Architecture distribuée: AlgeBrain.	36
2.10	Système pédagogique	37
2.11	Pédagogie centrée sur l'apprenant	38
2.12	Evolution de système autonome via le Web	40
3.1	Stretchtext adaptatif: fragment caché et affiché	49
3.2	Description de l'hérissone est produite par PEBA-II	49
3.3	Copie d'écran du système PUSH.	51
3.4	Exemple d'annotation adaptative des liens exploitant l'interface WWW dans ELM-ART.	53
4.1	Exemple d'objectifs pédagogiques	62
4.2	Transformation d'un objectif pédagogique pour ECSAIWEB	63
4.3	Modèles formels du système ECSAIWEB	63
4.4	Modèle d'une unité logique d'apprentissage	68
4.5	Structure générale de la présentation d'une unité physique	72
4.6	Modèle de la présentation du type NORMAL	72
4.7	Exemple de contenu que nous voulons présenter dans la page entière	73
4.8	Exemple d'une présentation de l'exemple	75
4.9	Modèle de présentation de la catégorie EXEMPLE	76
4.10	Exemple d'une présentation du type EXEMPLE	76
4.11	Modèle de présentation de la catégorie RAPPEL	77
4.12	Exemple d'une présentation de la catégorie RAPPEL que nous avons souhaitée	77
4.13	Contenu (à gauche) de chaque question et son modèle (à droite).	79
4.14	Exemple de la présentation d'un exercice	81
4.15	Modèle de question à choix multiples	82
4.16	Exemple d'exercice du type QCM	83
4.17	Modèle de question du type NUMERIQUE SIMPLE	83
4.18	Exemple de la question du type numérique simple	84

4.19	Modèle de question du type vrai/faux	85
4.20	Exemple de la question du type Vrai/Faux	86
4.21	Modèle de question du type TEXTE	86
4.22	Exemple de la question du type Texte	87
4.23	Exemple d'une base de test	87
4.24	Présentation d'un contenu du test	88
5.1	Modèles de recouvrement d'élèves	91
5.2	Modèle d'élève de ECSAIWEB	93
5.3	Modèle, Visualisation et Annotation du mode guidé	95
5.4	Exemple d'un parcours d'apprentissage en mode guidé de l'élève X	95
5.5	Modèle, Visualisation et Annotation des concepts en mode libre	96
6.1	Schéma d'interaction entre les clients et le didacticiel par l'intermédiaire du WWW	100
6.2	Page d'accueil du système ECSAIWEB	100
6.3	Exemples d'interfaces utilisées dans ECSAIWEB	102
6.4	Liens d'hypertexte	103
6.5	Menu adaptatif	104
6.6	Processus de la barre des menus	105
6.7	Exemple de feuille de styles utilisé dans ECSAIWEB	106
6.8	Menu POP-UP utilisant le DHTML	107
6.9	L'ensemble des menus pop-up	107
6.10	Organigramme du système ECSAIWEB	109
6.11	Espace de travail d'enseignants-auteurs	110
6.12	Six catégories d'outils dans l'environnement auteur	110
6.13	Organigramme des fonctionnalités dans l'environnement auteur	111
6.14	Saisie des items d'évaluation (à gauche) et des objectifs pédagogiques (à droite)	112
6.15	Description de l'unité logique d'apprentissage U3	113
6.16	Editeur de l'unité de présentation normale (a) et sa page obtenue (b)	114
6.17	Editeur de l'unité de présentation exemple (a) et sa page obtenue (b)	115
6.18	Editeur de l'unité de présentation rappel (a) et sa page obtenue (b)	115
6.19	Editeur HTML	116
6.20	Editeur d'une unité exercice	117
6.21	Partie différente de chaque type de question	118
6.22	Affichage d'un exercice	118
6.23	Vérifier la pertinence des items d'évaluation	119
6.24	Vérifier la validité des objectifs pédagogiques	120
6.25	Vérifier l'accessibilité des unités d'apprentissage	120
6.26	Simulation du parcours d'apprentissage	121
6.27	Interface pour créer un examen	123
6.28	Processus de transformation des fichiers du système ECSAIWEB	124
6.29	Commands de gestion des profils d'élèves	125
6.30	Informations personnelles	125
6.31	Connaissances en mode libre	125
6.32	Informations en mode guidé	126
6.33	Espace de travail pour les élèves	127
6.34	Organigramme de l'environnement élève	128
6.35	Exemple du générateur d'un parcours d'apprentissage en mode libre	128

6.36	Procédure du mode libre	129
6.37	Plan de progression et le résultat en mode libre	129
6.38	Plan de progression du mode guidé	130
6.39	Procédure du mode guidé	131
6.40	Parcours d'apprentissage de l'apprenant X en mode guidé	131
6.41	Méthode raffinée: Présentation d'abord	132
6.42	Méthode raffinée: Item d'abord	133
6.43	Méthode raffinée: Item atteint	133
6.44	Méthode raffinée: Apport global	134
6.45	Algorithme de raffinement du parcours d'apprentissage	134
6.46	Menus adaptatifs a) en mode libre, et b) en mode guidé	135
6.47	Informations dans la partie haute de la page de présentation	136
6.48	Barres de navigation	136
6.49	Organigramme des outils de l'environnement partagé	137
6.50	Outils de communication: courrier électronique	138
6.51	Outils de communication: forum de discussion	139
6.52	Visualisation des messages	140
6.53	Outils de communication: salle de discussion en temps réel (Chatroom)	141
6.54	Outil administratif: diagramme des fonctionnalités	142
6.55	Exemple de la fonction « <i>modifier une matière</i> »	142
7.1	Architecture d'ECSAIWEB	147
7.2	Organisation des programmes du système ECSAIWEB	148
7.3	Sous-programme pour filtrer des requêtes de commandes: partie de définition des variables	149
7.4	Partie de traitement des formulaires du type « <i>GET</i> »	150
7.5	Partie de traitement des formulaires du type « <i>multipart/form-data</i> » ou « <i>POST</i> »	150
7.6	Script d'une partie de module ECSAIWeb::Domain, opérations des items	153
7.7	Opérations de l'unité logique d'apprentissage	154
7.8	Script de module ECSAIWeb::Domain, calculer les liens	155
7.9	Script de module ECSAIWeb::Presentation	156
7.10	Contenu d'une unité physique U1: Présentation normale	158
7.11	Liste de sous-programme pour gérer la base de donnée d'un profil d'élève en mode libre	159
7.12	Liste de sous-programmes pour gérer la base de donnée d'un profil d'élève en mode guidé	161
7.13	Script du Java pour manipuler la barre de menus	162
7.14	Relation entre le menu adaptatif et le contenu dans le profil de l'apprenant	163
7.15	Sous-programme de raffinement du parcours d'apprentissage	163

Liste des tableaux

1.1	Intervenants	14
1.2	Objets gérés par la plate-forme	15
1.3	Présentation du contenu	16
1.4	Possibilités de navigation	17
1.5	Ressources de communication	18
1.6	Étapes de création d'un cours	19
1.7	Intégration du contenu : outils	19
1.8	Développement du contenu	20
3.1	Présentation adaptative utilisée dans MetaDoc	48
6.1	Liste des objectifs pédagogiques.	122

Introduction

Le domaine de recherche

Le travail de recherche présenté dans cette thèse porte sur la conception et la réalisation de formations au travers de réseaux. L'une des difficultés rencontrées est que les systèmes proposés sont rarement acceptés par les enseignants car ils leur sont fermés : très peu de personnalisation des contenus et de prise en compte de leur pédagogie. Certains d'entre eux offrent des outils pour modifier ou créer les matériaux pédagogiques, mais ils sont difficiles pour les enseignants parce que les ouvertures à l'auteur sont limitées ou bien les modèles sont trop compliqués pour enrichir les contenus. Même si certains systèmes permettent aux enseignants-auteurs de modifier les contenus, il est presque impossible de mettre à jour immédiatement les contenus si les produits didacticiels sont installés sur plusieurs postes de travail.

De l'EAO aux tuteurs intelligents sur le Web

Dès le début des années 70, on a pu remarquer une première vague d'intérêt pour l'utilisation des technologies de l'information dans la formation. On parlait alors d'enseignement assisté par ordinateur ou EAO¹. Depuis lors, de nouvelles technologies ont révolutionné le domaine de l'EAO : l'avènement du PC² et du multimédia, la mise au point du CDROM³, le déploiement de l'Internet et du Web en particulier. La reconnaissance d'approches pédagogiques nouvelles et l'intérêt pour une pédagogie active sont également à inscrire au tableau des perspectives nouvellement ouvertes. Enfin, on constate un développement de la demande de formation continue, de formation en horaires décalés, de formation à distance et de formation adaptée. Ces différents facteurs expliquent la présence de plus en plus étendue sur le marché de systèmes d'auto-formation à distance.

Aujourd'hui, la grande majorité des systèmes disponibles utilisant les technologies de l'information dans la formation vise essentiellement la présentation des contenus. En toute généralité, nous pouvons dire que les matériaux pédagogiques peuvent être vus comme des livres enrichis, dans lesquels le texte est appuyé par des éléments multimédias. Ces systèmes permettent souvent une navigation non linéaire à travers la matière; ils offrent parfois à l'apprenant des possibilités sommaires de mesurer sa maîtrise de la matière. Ils sont disponibles soit sur CD-ROM (on parle alors souvent de didacticiels), soit sur le Web (c'est-à-dire sur l'Internet), soit sur un réseau d'entreprise (un Intranet).

Bien évidemment, il serait regrettable de limiter l'usage des technologies de l'information à la simple présentation de contenus. L'ordinateur offre de nombreuses autres possibilités utiles sur

1. EAO : *Enseignement Assisté par Ordinateur*

2. sigle anglais de *Personal Computer*

3. sigle anglais de *Compact Disc Read Only Memory*

le plan pédagogique et rendant l'apprentissage plus actif et vivant :

- simulations guidées par l'apprenant,
- tests de connaissances interactifs,
- apprentissage coopératif,
- accès à des sources multiples d'informations (Internet),
- interactions avec un tuteur, avec d'autres apprenants,
- individualisation et gestion de la progression dans la matière,
- etc.

Nous pouvons aller encore plus loin en essayant de créer les conditions d'un véritable dialogue entre l'apprenant et le système de formation, dont le comportement s'apparente le plus possible à celui d'un enseignant, un précepteur privé. Dans leur forme la plus avancée – dont il faut reconnaître qu'elle relève encore du domaine de la recherche scientifique – les systèmes de formation de ce type appartiennent à une classe appelée «tuteurs intelligents».

En peu de temps, les technologies éducatives pour un enseignement supérieur ont évolué de la production des multimédias et des matériaux pédagogiques basés sur l'hypertexte dans le CDROM aux scénarios gérés en réseau facilitant l'interaction, les activités communes, et le partage des ressources pédagogiques.

Les nouvelles technologies pourraient être employées pour les études préliminaires et les perspectives de l'innovation, mais l'approche pour une exécution à grande échelle pose des difficultés dès le début. En ce moment, il serait important de savoir si «*les nouvelles technologies aident réellement à améliorer ou créer de nouvelles situations d'étude pour des étudiants*» ou «*si nous nous déplaçons ou reproduisons juste les mêmes problèmes et difficultés trouvés dans l'enseignement traditionnel*». Un exemple de cette disparité : pourquoi offrir à chaque étudiant un moyen personnel de communication avec son professeur si le professeur ne peut pas consacrer assez de temps à ses étudiants. Les possibilités techniques soulèvent les espérances qui ne peuvent pas être accomplies. Un autre point est de croire que les étudiants peuvent suivre les mises à niveau technologiques rapides. Ceci mène aux situations où des étudiants n'ayant pas le dernier processeur ou le modem à grande vitesse sont mis sur la touche des expériences éducatives. Un exemple bien connu de cette situation sont les équipements visuels de communication, où une qualité minimum pour la présence virtuelle exige au moins un raccordement de RNIS⁴ et une caméra pour le Web (appelée Webcam⁵). Ces conditions, au moment où nous écrivons, sont bien au-delà des possibilités moyennes de la plupart des étudiants. Alors dans notre cas, nous voulons simplement créer des environnements simples permettant aux utilisateurs d'utiliser ou d'accéder au centre d'apprentissage situé dans le serveur de l'établissement, avec aussi peu de moyens que possible.

Actuellement, les systèmes d'éducation basés sur le Web (SEW⁶) forment un des secteurs les plus dynamiques dans la recherche et le développement de technologies éducatives. Les avantages des SEW sont l'indépendance de l'enseignement et de l'apprentissage en ce qui concerne le temps et le lieu. Les matériaux pédagogiques installés et maintenus dans un endroit peuvent être employés par un grand nombre d'utilisateurs partout dans le monde. Un but de nos recherches est «*le développement des systèmes d'éducation adaptatifs et intelligents basés sur le Web (SEW-AI⁷)*», intégration d'une certaine quantité d'adaptabilité et d'intelligence.

4. RNIS : Réseau Numérique à Intégration de Service

5. Webcam : caméra vidéo numérique reliée directement à l'ordinateur

6. sigle anglais, WES : Web-based Education System

7. sigle anglais, AIWED : Adaptive and Intelligent Web-based Education Systems

L'adaptabilité en ce qui concerne la présentation et le support de la navigation aide les apprenants à visualiser et localiser l'univers d'apprentissage. Le parcours d'apprentissage de chaque apprenant est calculé dynamiquement et dépend de ses compétences [Gav91]. Les équipements intelligents d'un support d'explication des erreurs de résolution de problèmes soutiennent le travail des apprenants dans l'exécution de leurs tâches [Vas96]. Le support adaptatif de collaboration proposé dans PHelps [GMC⁺98] offre des outils qui permettent aux apprenants de rechercher les assistances pour résoudre les problèmes. Ce sont juste quelques exemples mis en application dans un certain nombre de systèmes de recherches qui montrent comment un apprenant peut tirer bénéfice des technologies adaptatives et intelligentes. Tandis que la plupart des systèmes commerciaux de «*courseware*»⁸ n'impliquent pas de technologies intelligentes ou adaptatives, un certain nombre de systèmes existants de recherches ont déjà formé une masse critique pour une discussion de la création de systèmes d'apprentissage adaptatifs et intelligents basés sur le Web.

Formations des enseignants en Thaïlande

Le département Informatique et Formation de la faculté d'Education Technique de KMITNB⁹, a été établi en 1991. Actuellement ce département offre les diplômes de «*B.S. Tech. Ed.*»¹⁰ (*Computer Technology*) (depuis 1991) et «*M.S. Tech.Ed.*»¹¹ (*Computer Technology*) (depuis 1997) dans différents secteurs, il développe l'enseignement de l'informatique pour professionnel, notamment information individuelle et à la demande. Pour remplir sa mission KMITNB focalise ses formations en Thaïlande en devenant un producteur de technologie plutôt qu'un simple utilisateur. Nous envisageons de former des éducateurs, des administrateurs professionnels du domaine éducatif et les enseignants d'éducation technique au niveau du doctorat (à partir de l'année scolaire 2002 option «*Computer Education*» et à partir de l'année scolaire 2004 option «*Curriculum and Instruction (Computer Technology)*») avec les recherches approfondies dans les deux domaines. Au même moment, nous envisageons également d'intégrer les formations «*hors-campus, niveau master*» en dehors de Bangkok (coopération entre notre département, KMITNB et le Département d'éducation professionnelle sous la direction du Ministère de l'éducation). Ces objectifs et mes intérêts personnels m'ont permis de définir mes études de niveau doctorat dans cette direction, c'est-à-dire l'enseignement à distance, le campus virtuel, la classe virtuelle et les environnements d'apprentissage.

Objectifs du travail

Le WWW¹², alors, devient l'un des vecteurs majeurs pour mettre en ligne les matériaux pédagogiques. Les principaux avantages de cet outil sont maintenant bien établis: l'indépendance de lieu et l'indépendance de plateforme. Les utilisateurs, n'importe où dans le monde, peuvent accéder en général au centre d'apprentissage avec toutes sortes d'ordinateurs connectés à l'Internet.

L'une des caractéristiques la plus importante des environnements d'apprentissage avancés est leur capacité à évaluer l'acquisition et la rétention de connaissance chez leurs utilisateurs et à s'adapter à leurs besoins. Il n'est pas surprenant que beaucoup de systèmes aient été adaptés sur

8. Terme français: *didacticiels* ou parfois *matériaux pédagogiques*

9. KMITNB: *King Mongkut's Institute of Technology North Bangkok, Bangkok, Thailand*

10. B.S. Tech. Ed.: *Bachelor of Science in Technical Education*

11. M.S. Tech.Ed.: *Master of Science in Technical Education*

12. sigle anglais de *World Wide Web*

le Web, cela signifie que leurs auteurs essaient de passer d'un système individuel à un système de distribution collective et de changer le mode de distribution des matériaux pédagogiques. Par exemple, PATAIgebratutor a donné naissance à PAT Online [Rit97], MANIC [Ste97] a été conçu pour permettre aux enseignants-auteurs de prendre des cours existants et de les convertir en cours en ligne.

La plupart des systèmes tutoriels intelligents ne permet pas aux enseignants-auteurs d'ajouter leurs connaissances dans le système. Dans certains cas la partie auteur est limitée aux concepteurs du système, dans d'autres cas il est difficile pour des enseignants d'assimiler les modèles et de parvenir à enrichir effectivement les contenus. Même si certains systèmes permettent aux enseignants-auteurs de modifier les contenus, il est presque impossible de mettre à jour immédiatement les contenus. C'est-à-dire, il faut attendre que les contenus ou nouveaux systèmes (recompiler si les systèmes sont assemblés en seul programme) soient remplacés.

Le manque de productivité en conception d'EIAO¹³ ouverts est un des problèmes auquel nous voulons tenter d'apporter une solution. La cause provient, en partie, de la complexité des systèmes comme nous l'avons indiqué au paragraphe précédent. Dans ce contexte, nos objectifs sont les suivants : d'abord produire un environnement de création de cours pour le Web, appelé ECSAIWEB¹⁴ à partir de l'environnement existant ECSAI¹⁵[Gav91]. Nous avons étudié le processus d'adaptation de l'environnement d'une part et des matériaux pédagogiques d'autre part, pour tirer partie des possibilités introduites par l'utilisation du Web. Nous souhaitons que le nouveau système soit plus flexible, plus intelligent et plus adaptatif. De nouvelles fonctionnalités étant fournies par de nouvelles technologies gérées en réseau, il y a maintenant le potentiel pour que les étudiants partagent leurs expériences d'étude les uns avec les autres et également interagissent plus dynamiquement avec leurs enseignants. Bien que la collaboration supportée par ordinateur de façon asynchrone ait été un sujet fréquent de recherches, le travail synchrone en ligne est un point qui a attiré l'attention dans le domaine de CSCL¹⁶. Puisque les interactions synchrones sont une partie essentielle de la formation scolaire, nous avons choisi ici de nous concentrer sur les outils synchrones, notamment l'interaction et la rétroaction actives immédiates entre les utilisateurs. Enfin, notre objectif est de déduire des recommandations utilisables pour tous les auteurs qui ont à adapter des cours ou des environnements existants.

Plus précisément

Notre travail porte sur les points suivants :

- produire un environnement de conception de systèmes d'apprentissage sur le Web,
- étudier le processus d'adaptation de l'environnement d'une part et des matériaux pédagogiques d'autre part pour tirer partie des possibilités introduites par l'utilisation du Web,
- produire un environnement auteur permettant aux enseignants-auteurs de créer facilement des matériaux pédagogiques,
- produire un environnement tuteur permettant aux apprenants d'apprendre des contenus avec les matériaux pédagogiques créés par les enseignants-auteurs,
- étudier la possibilité d'introduire les techniques adaptatives pour présenter les contenus au sujet et naviguer dans l'univers d'apprentissage,

13. EIAO : *Environnement Interactif d'Apprentissage avec Ordinateur*

14. ECSAIWEB : *Environnement de Conception de Système d'Apprentissage Intelligent sur le Web*

15. ECSAI : *Environnement de Conception de Systèmes d'Apprentissage Intelligent*

16. sigle anglais de *Computer Supported Collaborative Learning*

- étudier le générateur de parcours d'apprentissage pour chaque mode d'apprentissage : libre ou guidé,
- étudier les outils synchrones, notamment l'interaction et la rétroaction actives immédiates entre les utilisateurs.

Contenu du mémoire

Le premier chapitre de ce mémoire est consacré à la motivation et aux caractéristiques des SEW. Il décrit les études comparatives des systèmes existants. Ces études nous permettent de savoir quelles sont les fonctionnalités et les ressources pédagogiques fournies dans les systèmes, quels sont les points forts et faibles de chaque système. Cela nous permettra ensuite de définir des fonctionnalités pour notre système.

Le deuxième chapitre résume les principes du système ECSAI et étudie de façon approfondie chacun d'entre eux. Puis il introduit les évolutions des systèmes autonomes au Web : les architectures et les études sur les points forts et faibles de chaque architecture. Ensuite, il décrit le changement de la situation d'apprentissage plaçant l'apprenant au centre du système pédagogique, et le changement de la formation traditionnelle à la formation à distance. A partir de l'analyse de ce point de vue, nous pouvons mettre en place un premier cahier des charges de notre système que nous appelons «ECSAIWEB».

Le troisième chapitre décrit les techniques adaptatives d'hypermédia que nous voulons appliquer à ECSAIWEB. Ces techniques sont divisées en deux parties : la présentation adaptative et la navigation adaptative. L'une des idées décrites dans ce chapitre nous permet de définir la façon adaptative de présenter et de naviguer dans l'univers d'apprentissage.

Le quatrième chapitre décrit les modèles du domaine utilisés dans ECSAIWEB : le modèle des concepts, le modèle des unités logiques d'apprentissage, les modèles des unités physiques d'apprentissage et les descriptions des opérations associées à tous les modèles.

Le cinquième décrit un autre modèle : le modèle d'élève. Il faut noter que le parcours d'apprentissage de chaque apprenant dans ECSAIWEB est créé automatiquement en utilisant les informations stockées dans ce modèle. Par conséquent, les techniques adaptatives que nous voulons intégrer dépendent de ce modèle. Le chapitre décrit d'abord les rôles de ce modèle, puis la représentation du modèle, ensuite il détaille les modèles de recouvrement multi-points de vue, proposés par notre système, pour aider à annoter les concepts et à créer un parcours d'apprentissage.

Le sixième chapitre décrit, du point de vue des utilisateurs principaux (enseignants-auteur et apprenants), les fonctionnalités retenues pour l'ECSAIWEB. Nous y présentons les interfaces d'accès et les fonctionnalités mises en œuvre pour atteindre les objectifs fixés au logiciel. Nous présentons l'environnement de conception d'univers d'apprentissage, l'environnement élève, l'environnement partagé et les outils de système administratif.

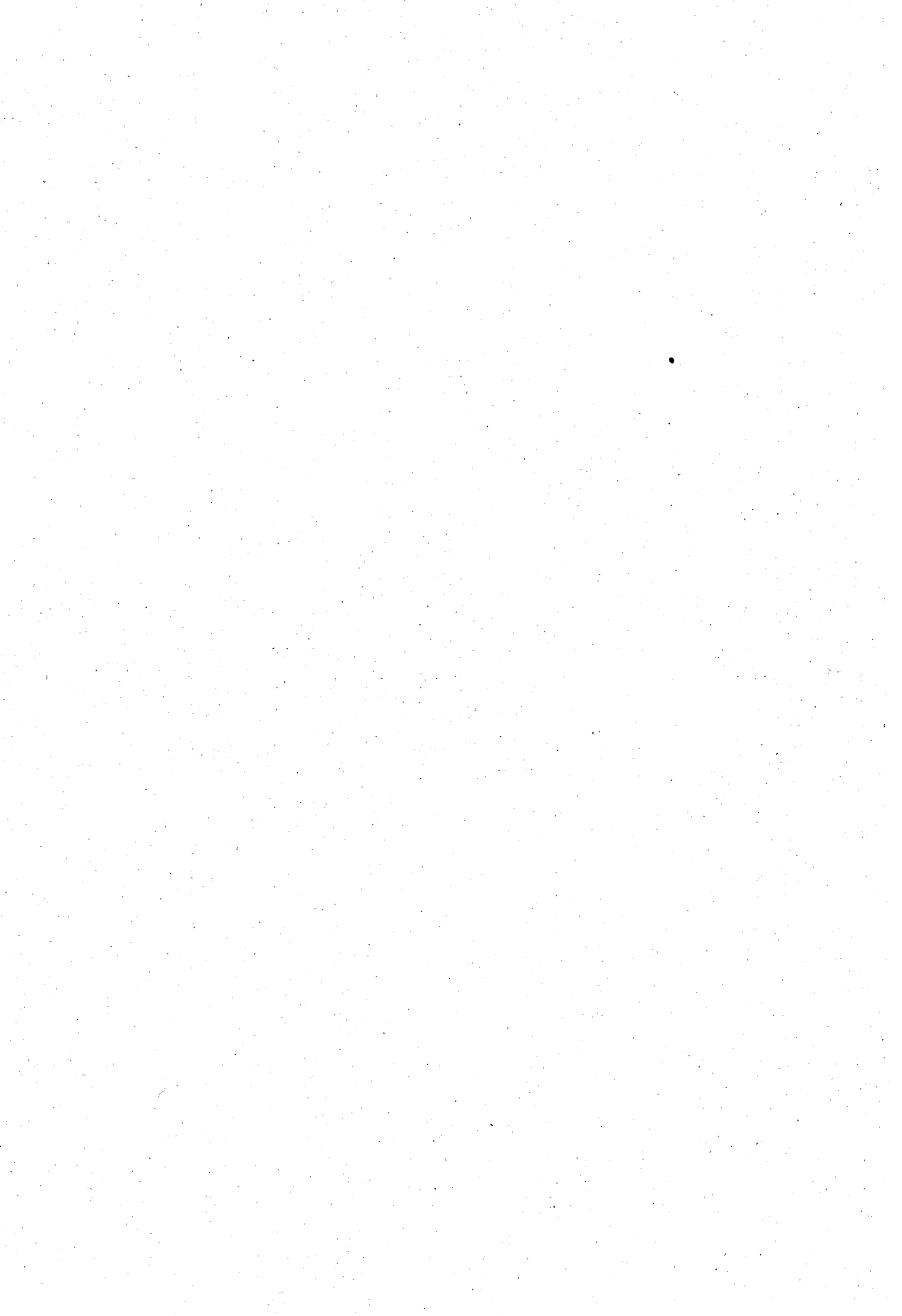
Le septième chapitre, plus technique, est consacré à l'implantation d'ECSAIWEB. Ce chapitre décrit, dans la première partie, les choix de langage de programmation et la visionneuse de documents Web. La partie suivante détaille les programmes principaux et les modules associés

aux concepts que nous décrivons dans les chapitres précédents.

Nous présentons en conclusion de cette thèse les moyens mis en œuvre pour expérimenter et valoriser notre système. Nous mettons en évidence les apports de ce travail et abordons les perspectives de recherche qu'elles permettent d'envisager.

Première partie

Etat de l'art et cahier des charges : d'ECSAI à ECSAIWEB



Chapitre 1

Motivations et Systèmes existants

1.1 Introduction

L'EAO ou CBL¹⁷ a été utilisé depuis le temps où les premiers ordinateurs ont été mis dans la salle de classe. Quelques enseignants l'ont vu comme une grande occasion d'augmenter leur efficacité. D'autres l'ont naturellement vu comme une menace et cette attitude règne toujours en quelque sorte aujourd'hui. Elle gêne les efforts de l'utilisation d'EAO. Comme avec n'importe quel outil cela fonctionne seulement s'il est utilisé et utilisé correctement.

Parmi les supports d'EAO proposés c'est le CDROM qui a stimulé l'imagination des enseignants. Ce support permettait une diffusion d'une information beaucoup plus riche avec une interface plus animée que dans les supports précédemment utilisés. Quelques domaines ont été bien couverts par le CDROM pour aider des enseignants tandis que d'autres n'étaient même jamais touchés. Les CDROM sont une grande voie pour l'usage d'EAO, mais des établissements éducatifs n'ont pas beaucoup de sources de financement et semblent toujours incapables de savoir quelles ressources sont disponibles à eux. Et une fois qu'un CDROM est fait, plus d'information ne peut être ajoutée. La nouvelle information est incorporée, seulement lorsque des versions de mise à jour sont publiées. Habituellement, ce module coûtant plus d'argent, les écoles n'ont pas la possibilité de l'acheter. Et cette nouvelle information sera démodée avant qu'elle soit utilisée réellement. C'est donc un support intéressant, mais il ne peut pas être employé pour tous. L'enseignant doit se contenter de ce qui est disponible et ne peut pas utiliser ses propres connaissances qui peuvent être plus détaillées que ce qui est disponible.

Un outil extrêmement utile en EAO était HyperCard¹⁸. Il utilise le concept des boutons combinant l'information textuelle avec des images et des sons. Il inclut des outils que les enseignants peuvent utiliser facilement. La création des aides d'enseignement personnel est aussi facilement faite avec le langage de programmation d'HyperCard, il suffit d'ajouter des «pointer» et «cliquer», utilisant l'hypertexte comme base. Beaucoup d'utilisations d'HyperCard peuvent être transférées directement aux utilisations du Web, c'est-à-dire, ceux qui peuvent écrire en HyperCard, peuvent également écrire en HTML¹⁹ sur le Web.

Durant les cinq dernières années, le Web a connu énormément de succès. En effet, pour beaucoup de gens, le Web est le seul système d'hypertexte qu'ils ont jamais utilisé ou dont ils ont jamais entendu parler. À cet effet, des chercheurs ont proposé plusieurs articles avec des

17. sigle anglais de *Computer Based Learning*

18. HyperCard est souvent décrit comme «paquet de cartes». Les cartes peuvent être liées les unes avec les autres. HyperCard est parfois décrit comme une sorte de base de données pour organiser des informations et les enchaîner ensemble.

19. sigle anglais de *HyperText Markup Language*

considérations particulières sur la façon dont leurs résultats s'appliquent au Web maintenant ou pourraient s'appliquer aux versions du Web qui apparaîtront à l'avenir. Comme avec tout ce qui est actuellement impliqué dans le Web, il y a trop d'informations en ce moment et obtenir une bonne prise en compte de ce qui est utilisable est une tâche utopique. Basés sur les réseaux, les idées d'infrastructure, et les scénarios de la formation et de l'éducation du futur ont déjà existé. Les écoles et les universités ont déjà eu des années d'expérience avec ces nouvelles technologies, pour les étudiants résidentiels et distants. Dans plusieurs cas les anciennes connaissances et/ou les anciens systèmes sont transférés sur ces canaux des nouvelles technologies. L'avantage principal de l'EAO sur le Web est que le Web est un support dynamique. L'information peut être ajoutée et changée immédiatement. Tous ceux qui utilisent cette information, utilisent l'état mis à jour. L'enseignant pourra utiliser l'information, car elle devient disponible, au lieu d'attendre longtemps que des processus bureaucratiques la lui fassent parvenir. Le Web est aussi universel que n'importe quel support d'ordinateur a été depuis la prolifération des ordinateurs au monde dans son ensemble et particulièrement dans le contexte des établissements d'étude. Une fois que des matériaux pédagogiques sont créés, ils peuvent être utilisés à peu près par n'importe qui, à une condition que les utilisateurs aient été formés à ces modes de travail.

Les capacités de ceux qui sont les plus adaptés à écrire le matériel pourraient être utilisées. En effet chaque sujet a des enseignants qui sont meilleurs à enseigner certaines parties du cours que d'autres. L'EAO sur le Web permet à ces enseignants, résidant n'importe où dans le monde, d'aider à créer une meilleure leçon. La base de données pourrait évoluer selon la volonté des enseignant-auteurs, évitant à des multitudes de personnes d'inventer la même chose à plusieurs reprises.

1.2 Motivations

Nous avons rassemblé un certain nombre de raisons et de motivations pour développer un environnement permettant de créer des cours sur le Web. Plusieurs des avantages d'utilisation du Web comme support de l'EAO sont les avantages du Web en général. Les avantages principaux de la technologie de Web sont résumés ci-dessous.

1.2.1 Plateforme indépendante

Les navigateurs graphiques sont disponibles pour toutes les plateformes principales, c'est-à-dire, Macintosh, PC ou station de travail quelque soit le système d'exploitation (Windows, UNIX par exemple). En effet quelques fournisseurs de navigateur de Web supportent chaque plateforme avec le même produit faisant l'interface aussi identique que possible sur chacun. Ceci augmente considérablement les assistances potentielles comparées au matériel préparé pour les environnements particulièrement configurés ou avec le logiciel disponible seulement sur une plateforme de matériel simple.

1.2.2 À prix réduit

Les serveurs et les navigateurs sont fournis par un certain nombre de fournisseurs, les principaux sont essentiellement libres pour les établissements éducatifs. En outre, l'utilisation de serveur n'a pas besoin de ressources exigeantes et n'exige ainsi pas en particulier de machines puissantes. En effet les programmes sur les machines des clients peuvent être exécutés en même temps que d'autres travaux personnels.

1.2.3 Facilité d'entretien

La connaissance et l'expertise nécessaires pour créer et mettre à jour un serveur sont minimales pour un éventail de services. Le serveur Macintosh est particulièrement élégant, car les diverses pièces peuvent être possédées et mises à jour par des individus avec le partage de fichiers d'AppleShare, mais Windows ou Linux conviennent aussi et sont plus utilisés. Il y a un éventail d'outils maintenant disponibles pour mettre en application les différentes fonctions d'un serveur, par exemple, rechercher, créer des hyper-images, et analyser des trafics. Les matériaux pédagogiques, les applications sont installées dans une seule machine (serveur) et sont donc faciles à maintenir.

1.2.4 Portabilité

Le contenu de serveur formaté comme documents d'HTML peut être transporté entre les serveurs ou même sur les environnements autonomes. Ce mode autonome peut être un avantage en prenant le matériel des environnements non gérés en réseau, par exemple le CDROM.

1.2.5 Distribué

Le Web est en soi un système de réseau distribué, ceci a deux avantages principaux. Premièrement, ceux qui sont les plus capables de mettre à jour leur information peuvent le faire localement avec leur propre contrôle. Deuxièmement, il n'y a pratiquement pas de limite à la quantité d'information à stocker.

1.2.6 Intégration Internet

Des transmissions d'informations non graphiques (fichiers ou textes) par l'Internet, plus anciennes, tels que le courrier électronique, le transfert de fichiers ou ftp²⁰, le Gopher et les forums de discussion (ou «news», en anglais) sont tous accessibles par la plupart des navigateurs de Web. Ces applications restent disponibles et peuvent être intégrées dans les applications nouvelles.

1.2.7 Médias riches

Le langage fondamental derrière le Web est HTML, il a été rendu relativement simple mais il semble toujours être suffisamment riche pour la majorité des applications. Les types fondamentaux de l'information incluent le texte formaté, les images, l'animation/vidéo, l'acoustique/sons. Les éléments interactifs de l'utilisateur incluent des liens hypertexte de textes ou de graphiques, des boutons radio et checkbox, des boutons pop-up avec des listes instantanées, et des zones d'enregistrement de texte. L'HTML évolue toujours avec des fonctionnalités supplémentaires et les nouveaux composants tels que DHTML²¹, VRML²², etc.

1.2.8 Serveur extensible.

Les fonctionnalités de serveur sont extensibles à l'aide de programmes de CGI [Tea]²³, ceux-ci sont des scripts ou des applications qui s'exécutent sur le serveur. Ces applications peuvent

20. sigle anglais de *file transfer protocol*

21. sigle anglais de *Dynamique HyperText Markup Language*

22. sigle anglais de *Virtual Reality Markup Language*

23. sigle anglais de *Common Gateway Interface*

exécuter ainsi n'importe quelle opération, elles communiquent généralement avec l'utilisateur par l'HTML normal.

1.3 Caractéristiques de SEW

Les termes suivants caractérisent parfaitement les avantages des formations que nous pouvons mettre en œuvre grâce aux SEW :

- accessibilité n'importe quand (*anytime*),
- depuis n'importe où (*from any place*),
- juste à temps (*just in time*) et non pas juste au cas où (*not just in case*),
- en quantité adéquate (*just enough*),
- sous un format approprié (*in the right form*),
- sans intermédiaire (*without intermediaries*).

1.4 Environnements intégrés

Nous nous sommes penchés plus particulièrement sur les systèmes de formation présentés comme des environnements intégrés d'apprentissage, c'est-à-dire des logiciels dans lesquels les divers outils offerts aux utilisateurs sont intégrés en un seul «produit» ou système (par opposition à des environnements d'apprentissage composés de manière ad-hoc à partir d'un ensemble de logiciels d'origines diverses). Nous adoptons l'appellation «WEDS²⁴» ou plus simplement encore, plateformes.

Les WEDS sont des environnements informatiques conçus spécialement à des fins de formation et d'éducation. Il ne faut pas confondre ces systèmes informatiques avec des sites et serveurs Web classiques : les WEDS exploitent les capacités de communication synchrones et asynchrones de l'Internet tout en reflétant une démarche pédagogique et un processus de formation. Il est donc important de relever que :

- leur but ne consiste pas uniquement à fournir du matériel pédagogique brut (un ensemble de pages de cours), mais bien plutôt à délivrer une formation à distance dans un contexte de classe de travail,
- ils permettent de mettre en œuvre un modèle d'apprentissage coopératif aussi bien qu'un modèle d'apprentissage individuel et autonome,
- ils peuvent être utilisés tant pour une formation à distance que pour un enseignement en salle de classe,
- ils offrent de nombreuses fonctionnalités de gestion et de logistique :
 - ◊ des cours et de leurs composants,
 - ◊ des programmes de formation,
 - ◊ des apprenants (inscriptions, suivi, etc.).

1.5 Systèmes WEDS existants

Un didacticiel est déjà beaucoup plus que l'assemblage de plusieurs hyperdocuments. Les systèmes d'éducation sur le Web sont pour objectif de fournir un environnement pédagogique

²⁴ sigle anglais de *Web-based Education Delivery System*

où les étudiants se sentent comme s'ils étaient dans une salle de classe virtuelle, guidés par une méthodologie pédagogique d'enseignement et étant aidés par leurs enseignants. Les éditeurs et les convertisseurs d'HTML ne fournissent pas un tel cadre éducatif et donc, plusieurs outils de création de cours en ligne sont apparus ces dernières années. Nous donnons dans la suite du chapitre un bref aperçu de quelques systèmes proposés pour le Web.

1.6 Études comparatifs des systèmes WEDS

Il existe actuellement de nombreuses plates-formes de développement et de distribution de cours sur le Web plus ou moins adaptatives. Parmi les systèmes peu adaptatifs et commerciaux, nous citons WebCT [GSS96], TopClass [Top] et WebMentor [Web]. Parmi les systèmes plus adaptatifs et non commerciaux, nous citons InterBook [BES98], ACE [Spe] et KBS-HyperBook [HN99]. ORAVEP²⁵ sur la direction du Ministère de l'Éducation Nationale et de la Recherche et de la Technologie en France²⁶[MENRT00] et les associations de professionnels : CEDITI [CED], C2T2 [C2T] et CIT [CIT], ont produit d'intéressantes études comparatives des systèmes appelés WEDS (Web-based Education Delivery System), les publient et les mettent à jour.

Il nous semble important, avant de définir des fonctionnalités à ECSAIWEB, de comparer ces systèmes. Les systèmes (indiqués dans le paragraphe précédent) ont été choisis selon les critères suivants :

- être opérationnels et avoir été utilisés par des étudiants dans le cadre d'enseignements,
- reposer sur les technologies de l'Internet,
- permettre de gérer plusieurs types d'activités pédagogiques (lecture, exercices, communication),
- ne pas exiger l'installation d'un logiciel particulier sur le poste étudiant.

Notez que l'étude comparative du ministère est faite depuis juillet 2000 et il n'y a pas d'étude plus récente disponible au moment où nous écrivons. Il n'existe pas d'étude comparative des systèmes plus adaptatives mais pas encore commercialisés tels que InterBook, ACE et KBS-HyperBook.

Le but de ces comparaisons est de nous permettre d'une part définir des fonctionnalités de ECSAIWEB et d'autre part d'avoir une vue globale comparée aux systèmes existants les plus utilisés : quelles sont des entités gérées par le système, quelles sont les fonctionnalités supportées pour chaque intervenant, quelles manières de présenter leurs matériaux pédagogiques et naviguer dans l'univers d'apprentissage : plus ou moins adaptée. En effet, cela nous permet de choisir les deux critères suivants parmi les autres (critères communs proposés par plusieurs études comparatives et nos intérêts) : les entités gérées par les plates-formes (l'objectif de la section 1.6.1) et les fonctionnalités offertes aux différents intervenants (décrites dans la section 1.6.2). Ces critères permettent de déterminer de manière relativement complète les fonctionnalités d'ordre technique supportées par un système d'aide à l'apprentissage.

1.6.1 Entités gérées par la plate-forme

Plusieurs types d'entités sont gérés par le WEDS. Nous distinguerons les entités jouant un rôle primordial dans les environnements répartis intégrés d'apprentissage : d'une part les intervenants

25. ORAVEP : *Observatoire des ressources pour la formation*

26. Une étude de l'ORAVEP réalisée avec l'appui de la Direction de la Technologie, sous-direction des technologies éducatives, des technologies de l'information et de la Communication (DT/SDTETIC) du ministère de la recherche et du Fonds Social Européen.

jouant un rôle dans l'usage, la mise en œuvre et l'exploitation du système, d'autre part, les objets qui constituent le matériel pédagogique et les ressources de communication.

a) Intervenants

Les différents types d'acteurs qui interviennent dans la mise en œuvre et l'exploitation d'un système peuvent être classés en différentes catégories selon le rôle d'usage : les apprenants, les tuteurs, les évaluateurs, les gestionnaires, les auteurs ou concepteurs de cours et les réalisateurs de cours, représentés dans le tableau 1.1. Chaque catégorie d'intervenants travaille au moyen d'interfaces et d'outils qui lui sont propres, leurs rôles étant distincts en général. Le tableau qui suit indique pour quelles catégories d'intervenants des outils sont disponibles. Pour ECSAI-WEB, nous avons regroupé les intervenants en deux groupes : apprenants et enseignants-auteurs (tuteurs, réalisateurs de cours, auteur ou concepteurs de cours et administrateur) et nous ne distinguerons pas d'autres intervenants.

Intervenants	1	2	3	4	5	6	7
Apprenants	✓	✓	✓	✓	✓	✓	✓
Tuteurs	✓	✓	✓	✓**	✓	✓	✓
Réalisateurs de cours	✓		✓	✓**			
Administrateurs	✓	✓	✓	✓			
Auteurs/concepteurs de cours				✓**			
Autres intervenants	✓*		✓				

✓ -- disponible dans le système
 * -- des évaluateurs ont pour le rôle d'accorder des notes aux apprenants
 vide -- n'est pas disponible dans le système
 ** -- les enseignants jouent ces rôles

1 - WebCT 2 - TopClass 3 - WebMentor 4 - proposé dans ECSAIWeb
 5 - InterBook 6 - ACE 7 - KBS-HyperBook

TAB. 1.1 - Intervenants

b) Matériaux pédagogiques

Tous les systèmes manipulent divers types d'objets en vue de constituer les matériaux pédagogiques mis à la disposition des apprenants. Ces objets pédagogiques sont donc la raison d'être des WEDS. Parmi les objets pédagogiques en général manipulés par ces systèmes, nous trouvons des cours, des unités d'apprentissage, des unités de matière, des travaux pratiques, des exercices et des contrôles de connaissances, des ressources et des synthèses, représentés dans le tableau 1.2. Nous trouvons également dans l'un de ces systèmes des objets complémentaires à la formation : un glossaire, un index, des aides et une bibliothèque. KBS-HyperBook ajoute une séquence pour produire le guide, des unités d'information appropriées pour un projet (exercice ou exemples de problèmes résolus). L'étude de ce support permet donc d'établir la nature des objets qui constituent les matériaux pédagogiques gérés par le WEDS, représentés dans le tableau 1.2.

Les objets sont concrétisés la plupart du temps par le contenu multimédia. De nombreux formats de support multimédia existent. Le support multimédia supporté par le WEDS est donc un critère important de comparaison : l'usage d'un format disponible via le Web tel que HTML, GIF, JPEG, VRML, etc. permet par exemple un échange aisé de matériaux pédagogiques entre un nombre important de plates-formes et la confection de matériaux pédagogiques à partir d'un grand nombre possible d'outils tandis que la réalisation d'un support multimédia correspondant à un format propriétaire implique en général l'usage d'un outil spécifique de développement.

Les objets gérés par la plate-forme	1	2	3	4	5	6	7
Cours ⁽¹⁾	✓	✓	✓	✓	✓	✓	✓
Unités d'apprentissage, modules, chapitres,...		✓		✓	✓	✓	
Unités de matière: concepts, thèmes, notions	✓		✓	✓	✓		✓
Exercices			✓	✓	✓	✓	
Contrôles de connaissances	✓	✓	✓	✓	✓	✓	✓
Ressources ⁽²⁾	✓		✓				
Glossaire	✓				✓	✓	
Index	✓				✓		
Aide	= *		= **	= **	= **	= **	

(1) – un cours se compose de cours, de pages et de contrôles de connaissances.

(2) – les ressources contiennent des références et des hyperliens menant à des informations relatives à la leçon mais non reprises dans le cours en tant que leçon ou faisant partie d'autres leçons.

=* – aide contextuelle disponible uniquement pour le réalisateur

=** – aide contextuelle sur les outils

vide – n'est pas disponible dans le système

1 – WebCT
5 – InterBook

2 – TopClass
6 – ACE

3 – WebMentor
7 – KBS-HyperBook

4 – proposé dans ECSAIWeb

TAB. 1.2 – Objets gérés par la plate-forme

Tous les formats exploitables par un navigateur Web peuvent être utilisés pour réaliser le matériel pédagogique: en effet, l'accès aux systèmes se fait au travers d'un navigateur Web standard. InterBook emploie la structure du document par l'utilisation des en-têtes pour indiquer au système quels concepts se tiennent derrière chaque section. ACE divise les matériaux pédagogiques en plusieurs catégories telles qu'une introduction, des textes, des cours de jeu, des exemples, des essais, et le sommaire. KBS-HyperBook exploite une unité d'introduction et des unités de projets. Des formats propriétaires peuvent donc être également utilisés pour autant que des plug-ins soient exploités par le navigateur utilisé sur le poste de l'apprenant. Dans ECSAI-WEB, nous ajoutons des balises spéciales pour marquer les unités physiques d'apprentissage tel que la balise PRESENTATION ou QUESTION, et d'autres balises pour classer des catégories de présentation ou de question.

c) Présentation des objets

Il s'agit ici de caractériser deux aspects complémentaires dans la présentation de matériaux pédagogiques: le contenu proprement dit qui a fait l'objet du paragraphe précédent et la navigation, c'est-à-dire l'ensemble de tous les chemins que peut emprunter un apprenant pour parcourir le contenu ou encore l'ensemble des activités à réaliser dans le cours pour maîtriser la matière.

Présentation du contenu

Tous les systèmes ont choisi la présentation adaptative du contenu. Le contenu d'un cours peut être adapté pour un étudiant ou un groupe d'étudiants appartenant à une classe. Dans WebCT, le contenu d'un cours peut être adapté par le tuteur pour un apprenant d'une classe: les unités d'apprentissage sont proposées aux apprenants sélectionnés par le tuteur-réalisateur. TopClass exploite le même processus que WebCT et d'ailleurs, l'information affichée à l'écran d'un apprenant peut varier dynamiquement et automatiquement en fonction du profil de l'apprenant et de l'évolution du temps. TopClass permet, en utilisant le profil de l'apprenant, l'ajout et le retrait automatique d'unités d'apprentissage d'après les résultats obtenus par un apprenant aux contrôles de connaissances. Les unités d'apprentissage sont accessibles ou non selon

des critères de validité dans le temps. Seules les unités valides sont affichées, des pages étant valides en fonction d'une date ou d'un laps de temps spécifié. Le WebMentor exploite des attributs (statiques) de l'apprenant pour construire le contenu qui convient au profil de l'apprenant ou aux capacités techniques de son poste de travail. InterBook emploie le modèle de domaine comme base pour structurer la teneur d'un manuel électronique ou ET²⁷ adaptatif, qui inclut deux parts : glossaire et manuel. Un ET adaptatif utilise le modèle d'étudiant pour adapter son comportement à chaque utilisateur particulier. Le contenu dans ACE peut être adapté selon des règles définies avec chaque stratégie d'enseignement sur chaque concept. ACE a une stratégie par défaut pour chaque concept ou une stratégie préférée définie par l'auteur. L'information affichée peut changer dynamiquement selon les comportements de l'apprenant. Par exemple si l'étudiant a fait quelques essais sur l'unité courante avec moins de succès et a déjà vu un certain matériel pédagogique alors le système modifie les stratégies par défaut à la séquence des matériaux pédagogiques manqués, les essais les plus faciles et la répétition des textes. Le contenu principal de KBS-HyperBook se compose d'unités sémantiques d'information et d'unités de projet ; des relations entre elles sont adaptées. Les unités d'information et de projet peuvent être adaptées selon la connaissance actuelle de l'apprenant. ECSAIWEB exploite la présentation statique correspondant à l'ensemble des pages d'une unité logique d'apprentissage fixée pour tous. La présentation dynamique (le contenu adapté) et adaptative d'une séquence d'unités d'apprentissage est faite selon les compétences de chaque apprenant.

Présentation du contenu	1	2	3	4	5	6	7
Présentation statique				✓ ⁽⁴⁾			
Présentation dynamique							
Contenu adaptatif (en fonction du passé)		✓ ⁽¹⁾					✓
Contenu adapté (en fonction du profil)	✓ ⁽²⁾	✓ ⁽²⁾	✓ ⁽³⁾	✓ ⁽⁵⁾	✓	✓	

vide – n'est pas disponible dans le système

- (1) – L'information affichée à l'écran d'un apprenant peut varier dynamiquement en fonction du profil de l'apprenant et de l'évaluation du temps.
- (2) – Le contenu d'un cours peut être adapté par un apprenant d'une classe : les unités d'apprentissage sont proposées aux apprenants sélectionnées par le tuteur-réalisateurs.
- (3) – Le contenu d'un cours adapté : WebMater exploite des attributs (statique) de l'apprenant pour construire le contenu qui convient au profil de l'apprenant ou aux capacités techniques de son poste de travail.
- (4) – ECSAIWEB emploie la présentation statique pour présenter des unités physiques d'apprentissage de type normal.

1 – WebCT 2 – TopClass 3 – WebMentor 4 – proposé dans ECSAIWeb
5 – InterBook 6 – ACE 7 – KBS-HyperBook

TAB. 1.3 – Présentation du contenu

Possibilités de navigation

WebCT et WebMentor exploitent la navigation ouverte, permettent, à tout moment à l'apprenant d'accéder librement à la table de matières. TopClass exploite la navigation semi-ouverte, c'est-à-dire que l'accès libre à n'importe quelle page de cours requiert cependant quelques manipulations parce que l'apprenant n'a jamais une vue complète et simultanée de toutes les unités d'apprentissage. InterBook, ACE et KBS-HyperBook exploitent la navigation semi-ouverte qui permet aux apprenants d'atteindre librement la table des matières. Naturellement, ils exigent quelques manipulations parce que les étudiants eux-mêmes n'auraient jamais une vue complète et simultanée de toutes les unités d'apprentissage. ECSAIWEB exploite deux types de navigation : la navigation statique pour indiquer la prochaine page dans l'ensemble des pages de l'unité

27. ET : *Electronic TextBook*

d'apprentissage et la navigation dynamique pour indiquer la prochaine unité d'apprentissage calculée par le système, plus précisément dans le mode guidé.

Possibilités de navigation	1	2	3	4	5	6	7
Navigation libre ou contrainte	✓ ⁽¹⁾	✓ ⁽²⁾	✓ ⁽¹⁾	✓ ⁽²⁾	✓ ⁽²⁾	✓ ⁽²⁾	✓ ⁽²⁾
Navigation statique				✓ ⁽³⁾			
Navigation dynamique							
Navigation adaptative		✓		✓ ⁽⁴⁾			
Navigation adaptée				✓ ⁽⁵⁾	✓	✓	✓

vide - n'est pas disponible dans le système

(1) - La navigation ouverte.

(2) - La navigation semi-ouverte.

(3) - ECSAIWEB exploite la navigation statique pour indiquer la prochaine page dans l'ensemble des pages de l'unité d'apprentissage.

(4) - ECSAIWEB exploite la navigation statique pour indiquer la prochaine page dans l'ensemble des pages de l'unité d'apprentissage.

(5) - ECSAIWEB calcule le parcours d'apprentissage dynamique en fonction de leurs besoins (mode libre) ou de leurs compétences (mode guidé).

1 - WebCT
5 - InterBook

2 - TopClass
6 - ACE

3 - WebMentor
7 - KBS-HyperBook

4 - proposé dans ECSAIWeb

TAB. 1.4 - Possibilités de navigation

d) Ressources de communication

Les ressources de communication sont la troisième dimension des WEDS. Celles-ci sont offertes par le WEDS pour établir des communications entre les différents acteurs (apprenants et tuteurs, par exemple). La plupart des outils de communication offerts par un WEDS sont des outils de communication asynchrones : courrier électronique et forum de discussion, et des outils de communication synchrones : chat room, échange d'information par tableau partagé et partage réparti d'applications et de documents. Tous les environnements exploitent les outils de communication asynchrones. Pour les outils de communication synchrones, le WebCT utilise le chat room et des tableaux partagés. ECSAIWEB exploite la salle de discussion en temps réel: ChatRoom.

1.6.2 Fonctionnalités offertes aux différents intervenants

Ce critère a trait aux différents services que l'on peut obtenir d'un WEDS. Nous avons choisi de les organiser en fonction des rôles joués par les différents intervenants. Dans cette partie nous ne détaillons que le point de vue de l'apprenant et le point de vue de l'enseignant-auteur.

a) Point de vue de l'apprenant

L'apprenant est du cœur de tout WEDS. Les critères de description et/ou de comparaison relative à la gamme des outils et des possibilités offertes par son environnement d'apprentissage sont examinés dans cette section. Les cours suivis au travers d'un WEDS laissent en, général un grand degré de liberté aux apprenants sur le plan horaire. Le moyen d'accès aux matériaux pédagogiques disponibles pour tout cours intégré aux quatre systèmes est une table de matières. De manière optionnelle, dans les pages du cours nous trouvons des boutons de navigation (suivant/précédent/remonter au niveau supérieur) et des hyperliens entre les pages du cours et des cartes des matières. Par exemple, la présentation de la table des matières, dans tous les systèmes, est dynamique dans le sens que ces tables se replient et se déplient automatiquement en fonction

Ressources de communication	1	2	3	4	5	6	7
Communication asynchrone							
Courrier électronique	✓ ⁽³⁾	✓ ⁽²⁾	✓ ⁽¹⁾	✓ ⁽¹⁾			
Forum de discussion	✓ ⁽⁶⁾	✓ ⁽⁴⁾	✓ ⁽⁴⁾	✓ ⁽⁶⁾			
Communication synchrone							
«Chat rooms»	✓						
Tableaux partagés («shared whiteboards»)	✓						
Partage d'applications et de documents	= ⁽⁷⁾	= ⁽⁷⁾					

vide - n'est pas disponible dans le système

(1) - boîtes de réception externes

(2) - Dans la version évaluée, seules les boîtes aux lettres intégrées à TopClass sont permises. Dans TopClass V3.1.0, le courrier électronique peut être envoyé soit vers une boîte aux lettres interne soit vers une boîte aux lettres externe à Top-Class.

(3) - Le courrier électronique envoyé est interne à WebCT. Les personnes déjà familiarisées avec un outil de courrier électronique doivent donc s'habituer à une seconde interface de courrier; les personnes possédant une boîte à lettres doivent par ailleurs vérifier l'arrivée de messages relatifs au cours suivi sous WebCT. Ces personnes doivent donc gérer deux interfaces. Par ailleurs, cette caractéristique permet aux personnes de travailler dans un environnement complet, cohérent et au travers d'une seule interface, que ce soit pour des activités d'apprentissage ou pour communiquer.

(4) - forums internes aux classes (WebMentor a des forums internes supplémentaires à l'entreprise)

(5) - WebCT propose deux types de listes de discussion : la publique est accessible à tous les intervenants d'un cours, les membres d'une liste privée sont déterminés par le cours. Le réalisateur peut créer autant de forums qu'il le désire.

(6) - ECSAIWEB propose deux types de listes de discussion, accessible à tous les intervenants d'un cours : groupes permanents (Student, Teacher et All) et les groupes associés à chaque matière. L'administrateur peut créer autant de forums qu'il le désire.

(7) - partage de documents via l'envoi de documents attachés à un courrier électronique

1 - WebCT

2 - TopClass

3 - WebMentor

4 - proposé dans ECSAIWeb

5 - InterBook

6 - ACE

7 - KBS-HyperBook

TAB. 1.5 - Ressources de communication

de la section du cours consulté ou encore au moyen d'un petit curseur mobile qui pointe sur la section consultée. En d'autres termes, le support à la navigation peut refléter ou non la progression de l'apprenant dans le cours ou donner une indication visuelle sur la localisation dans le cours de l'information consultée. De plus, ECSAIWEB permet de cacher des items et utilise la technique d'annotation adaptative [Bru97] pour montrer à l'apprenant quels objectifs ont déjà été atteints.

Il est rassurant pour l'apprenant de savoir où il en est dans le processus d'apprentissage : quelle partie de la matière a-t-il déjà parcouru et quand, quelles sont les sections qui lui ont pris le plus de temps etc. Les quatre systèmes choisissent des bilans pour visualiser la progression de l'apprenant qui sont les bilans personnels des objectifs pédagogiques, des travaux pratiques et des notes obtenues. De plus le WebCT et le WebMentor exploitent aussi des bilans personnels des outils utilisés sous la forme de la répartition d'utilisation des différents contenus ou outils (page d'accueil, contenu, articles postés, articles lus, etc.) et des bilans comparatifs des notes obtenues.

b) Point de vue de l'enseignant-auteur

La perception qu'a un enseignant-auteur d'un WEDS diffère forcément de celle des apprenants. Des outils spécifiques doivent être mis à sa disposition pour lui permettre de suivre, à distance, l'évolution d'un apprentissage et de guider cet apprentissage, et pour lui permettre de créer des cours.

Les moyens dont dispose l'enseignant pour évaluer la progression des apprenants dans le processus d'apprentissage sont de deux natures : le bilan des activités faites et restant à faire

dans le cours et la vérification de la maîtrise de la matière. Pour suivre et évaluer la progression d'apprenants, les quatre systèmes exploitent les bilans de la progression qui conservent la trace des objets pédagogiques exploités par un apprenant, puis des outils de synthèse de ces objets. De plus, le WebCT, le TopClass et le WebMentor font aussi des synthèses par groupe d'apprenants.

Étapes de création d'un cours	1	2	3	4	5	6	7
Interdépendance des étapes							
étapes distinctes, indépendantes			✓ ⁽³⁾	✓		✓	
étapes couplées	✓	✓					
situation intermédiaire	= ⁽¹⁾	= ⁽¹⁾		✓ ⁽¹⁾			
Dépendance par rapport au serveur							
forte	✓ ⁽²⁾	✓ ⁽²⁾		✓ ⁽³⁾		✓	
moyenne					✓		
faible							

vide - n'est pas disponible dans le système

- (1) - La construction du contenu, son intégration ainsi que la création d'un cours se font typiquement au cours d'une même phase. Néanmoins, le développement du contenu des objets pédagogiques «page» peut être réalisé en dehors du WEDS pour y être intégré peu après. Il nous recommande cette manière de procéder.
- (2) - Le réalisateur travaille au moyen de formulaires Web accessibles via un navigateur Web, en connexion avec le serveur. Il peut donc travailler à partir d'un poste distant du site serveur du type quelconque (Unix, PC, Mac).
- (3) - un cours peut donc être conçu de manière indépendante du système et livré (typiquement sous forme d'un fichier «zip») au gestionnaire du système pour être ensuite intégré à la base de donnée des cours.

1 - WebCT 2 - TopClass 3 - WebMentor 4 - proposé dans ECSAIWeb
5 - InterBook 6 - ACE 7 - KBS-HyperBook

TAB. 1.6 - Étapes de création d'un cours

Intégration du contenu : outils	1	2	3	4	5	6	7
Intégration des objets du contenu	✓	✓	✓	✓			
Prévisionneur	✓	✓	✓	✓			
Modification d'objets déjà intégrés	✓	✓	✓	✓			
Création de nouvelles versions d'objets intégrés		✓	✓	✓			
Intégration de formats multimédias particuliers :							
Intégration sans conversion			✓	✓			
Intégration par filtres de conversion	✓	✓					
Production de la navigation : table des matières	✓	✓	✓	✓			

vide - n'est pas disponible dans le système

1 - WebCT 2 - TopClass 3 - WebMentor 4 - proposé dans ECSAIWeb
5 - InterBook 6 - ACE 7 - KBS-HyperBook

TAB. 1.7 - Intégration du contenu : outils

Le rôle de l'auteur de cours concerne la création de contenu : il s'agit d'intégrer à un WEDS la version multimédia des objets qui constituent le matériau pédagogique. La construction du contenu, son intégration sous WEDS ainsi que la création d'un cours se font typiquement au cours d'une même phase. Néanmoins, le développement du contenu des objets pédagogiques «pages» peut être réalisé en dehors du WEDS pour y être intégré ensuite. Nous recommandons cette manière de procéder. L'auteur travaille au moyen de formulaires Web accessibles via un navigateur Web, en connexion avec le serveur. Il peut donc travailler à partir d'un poste distant du site serveur de type quelconque (station de travail UNIX, PC, Macintosh). Les outils existant dans tous les systèmes sont représentés dans le tableau suivant.

WebCT, TopClass et WebMentor offrent des outils pour développer le contenu tel que des

Développement du contenu	1	2	3	4	5	6	7
Outils d'édition d'objets pédagogiques	✓	✓	✓	✓		✓	
Outils d'édition pour les TP	✓	✓	✓				
Contenus annexes							
Outils d'édition de contenus : bibliothèque	✓						
Outils d'édition de contenus : glossaire	✓						
Outils d'édition de contenus : aides							
Navigation							
Outils d'édition de la table des matières			=*	=*			

vide - n'est pas disponible dans le système

=* - construction automatiquement

1 - WebCT
5 - InterBook

2 - TopClass
6 - ACE

3 - WebMentor
7 - KBS-HyperBook

4 - proposé dans ECSAIWeb

TAB. 1.8 - Développement du contenu

outils pour éditer les objectifs pédagogiques et travaux pratiques. En outre, WebCT offre des outils pour éditer le contenu de la bibliothèque et du glossaire. Ces trois systèmes offrent les outils additionnels pour importer les matériaux pédagogiques dans les systèmes. InterBook utilise de manière régulière la structure d'un dossier de MS Word. Il emploie un style «*en-tête*» prédéfini de textes, par exemple la section la plus élevée devrait avoir le style «*en-tête niveau 1*» comportant alors l'annotation concept-based de ET. Le dossier sortant doit être sauvegardé en format RTF et être converti en format d'InterBook. ACE offre un éditeur qui est une manière facile de modifier et créer le cours adaptatif pour le WWW. Avec cet éditeur, l'auteur peut définir une structure de programme d'études, enrichir les unités d'apprentissage avec des matériaux pédagogiques, indiquer des relations entre les unités et tous les paramètres nécessaires.

ECSAIWEB offre des outils pour définir les items d'évaluation, les objectifs pédagogiques, et les unités logiques d'apprentissage. L'éditeur d'ECSAIWEB permet à l'auteur de créer ou modifier le contenu d'une unité physique d'apprentissage. ECSAIWEB offre également des outils pour vérifier des cohérences dans l'univers d'apprentissage.

1.6.3 Conclusion

Il paraît donc au terme de cette étude comparative que les fonctionnalités de guidage et d'adaptation à l'apprenant sont encore peu développées au sien des plate-formes commercialisées.

ECSAIWEB offre un mode guidé qui permet un guidage assez fin de l'apprenant, c'est la différence majeure qu'il faut retenir en le comparant aux autres environnements. D'autres fonctionnalités seraient aussi souhaitables, mais elles n'auront pas encore été implantées dans cette version, nous les évoquons dans les perspectives de ce travail au chapitre 8.

1.7 Conclusion

Pendant la décennie nous avons vu le succès du WWW dans le secteur éducatif comme moyen pour distribuer les matériaux pédagogiques. Puisqu'il combine les avantages de l'éducation à distance et les possibilités de l'enseignement assisté par ordinateur (simulation, matériel de hypermédia, etc.), le Web est une plateforme appropriée où l'apprentissage peut être suivi avec succès.

Malheureusement, la plupart des systèmes télépédagogiques basés sur le Web sont liés aux sujets scientifiques ou technologiques. Notre sentiment est que la raison de ceci est le rapport

qu'entretiennent leurs auteurs avec le monde des nouvelles technologies de l'information. La plupart d'entre eux sont familiarisés avec les ordinateurs, Internet et le Web. Par conséquent, il est très facile pour eux de développer ces nouveaux contenus éducatifs sur le Web. Cependant, ceci n'est pas le cas dans la plupart des sujets, car les enseignants ne sont pas familiers de ces technologies. De plus il existe des éditeurs et des convertisseurs fantastiques de HTML, mais ceux-ci ne fournissent pas des outils pour définir une relation pédagogique appropriée entre les documents développés.

Évidemment, les enseignants ont changé leurs tableaux noirs pour des projecteurs de diapositives; le changement de leurs habitudes d'enseignement a été inévitable, par exemple, l'introduction de ressources informatiques et les systèmes d'éducation basés sur le WWW dans leurs établissements scolaires.

Pour adapter techniquement les enseignants, nous croyons que nous avons la responsabilité de fournir un ensemble d'outils appropriés pour créer des environnements éducatifs pour le Web. Nous présenterons dans les deux prochains chapitres (chapitre 2 et 3) comment nous adaptons notre système au Web, et la partie II, nous présenterons un outil pour développer de tels environnements en utilisant le Web comme cadre de travail. Ceux qui emploient notre outil ont besoin seulement de relier leur navigateur à ECSAIWEB pour créer leurs cours sur le Web.

Chapitre 2

Évolution d'ECSAI vers le Web

2.1 Introduction

La formation basée sur le Web est une excellente application des technologies avancées. Les navigateurs Web permettent de surmonter plusieurs des imperfections des EAOs ou des STIs²⁸ traditionnels puisqu'ils peuvent être exécutés avec les principaux systèmes d'exploitation (Windows, MacOS, UNIX, Linux) et permettent la mise à jour facilement des matériaux pédagogiques. Les EAOs existants (développés en utilisant un logiciel auteur commercial) fonctionnent souvent seulement dans un type d'environnement, par exemple sur Windows.

Plusieurs auteurs ont choisi d'adapter leurs systèmes au Web, par exemple de passer de PATALgebratutor à Pat Online [Rit97]. On a conçu MANIC [Ste97] et SQLT-Web [Mit00] pour permettre aux enseignants-auteurs de prendre des cours existants et de les convertir en cours en ligne. Dans notre cas nous voulons aussi adapter le système ECSAI [Gav91], un générateur pour la conception de systèmes d'apprentissage intelligents, au Web. Nous avons étudié la possibilité d'adapter ce système au Web d'une part et d'adapter les matériaux pédagogiques d'autre part.

Dans ce chapitre nous détaillerons tout d'abord le système ECSAI, puis nous expliquerons, cas par cas, les points forts et les points faibles du système. Ensuite, nous décrirons des évolutions en commençant par la section 2.3 avec le passage d'un système autonome au système sur le Web, puis la section 2.4 avec l'évolution des formations pilotées par des enseignants à des formations centrées sur des apprenants et la section 2.5 avec l'évolution des formations traditionnelles aux formations à distance. Enfin nous résumerons dans la section 2.6 l'application à l'évolution d'ECSAI.

2.2 Système ECSAI

L'objectif principal du système ECSAI est de modéliser des connaissances pédagogiques dans un environnement de conception de système d'apprentissage. ECSAI a proposé une possibilité de définir un cadre de travail accessible à l'enseignant pour permettre une description rapide en terme de contenus pédagogiques de l'apprentissage. Ce cadre prend en compte les trois étapes principales d'enseignement : présentation de connaissances (cours, rappels), mise en pratique de cette connaissance à travers des exercices plus ou moins complexes (entraînement ou exercice résolu) et évaluation des connaissances acquises par l'apprenant (test, problème). Pour faciliter l'instanciation de ce cadre et sa mise en œuvre par l'enseignant, ECSAI offre un certain nombre de fonctionnalités capables de générer des systèmes d'apprentissage dans des domaines variés.

28. STI : *Système Tutoriel Intelligent*

2.2.1 Prototypage du système ECSAI

Le système ECSAI se divise en deux environnements : environnement auteur et environnement élève. L'environnement auteur permet un prototypage rapide conduisant à réaliser des produits didactiques « ouverts ». En effet l'enseignant dispose d'un cadre prédéfini dans lequel il place les éléments constituant son enseignement. L'ensemble est automatiquement pris en charge par le système et le créateur peut, quand il le désire, effectuer des mises à jour.

Le système ECSAI est organisé en quatre modules (figure 2.1) : un module tutorial, un module apprenant, un module expert et une interface utilisateur/système. Le module tutorial est un organe central dans cet environnement. Il renferme la connaissance pédagogique (contenus à présenter à l'élève et stratégies pédagogiques) et a la responsabilité de la conduite de l'apprentissage. Le module apprenant doit fournir des informations nécessaires relatives à l'apprenant au module tutorial. Le module expert (non développé par E. Gavignet) détient l'expertise du domaine et assure la résolution des problèmes qui lui sont soumis. Il a en charge le suivi de la résolution par l'élève et par conséquent il lui apporte l'aide éventuelle dont il aurait besoin. L'interface utilisateur/système a en charge les échanges qui s'effectuent entre les utilisateurs potentiels (élèves, enseignants) et le module tutorial.

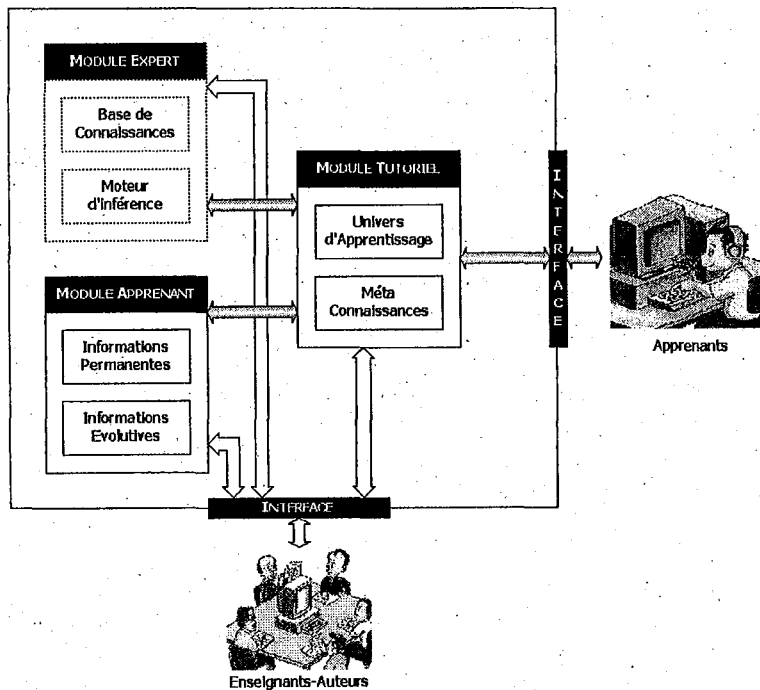


FIG. 2.1 – Diagramme du système ECSAI

2.2.2 Modèles d'ECSAI

Les facultés d'un didacticiel ouvert reposent principalement sur les modèles utilisés, notamment des modèles du domaine et les modèles utilisateurs. Nous décrivons dans cette section les modèles utilisés par ECSAI : les modèles du domaine et le modèle de l'apprenant.

a) Modèles du domaine : des concepts

L'enseignement traditionnel est basé sur la transmission d'un ensemble des connaissances dont le volume et le niveau de détail sont variables selon le public ciblé. Selon les objectifs pédagogiques à atteindre, l'enseignant construit son cours à partir d'un sous-ensemble des concepts du domaine. ECSAI propose l'idée «des items d'évaluation» [Gav91] [GG92] [Gra99] pour construire la liste des concepts. Ainsi la liste des concepts peut être associée facilement aux objectifs pédagogiques et aux processus d'évaluation. Naturellement, des unités d'apprentissage peuvent traiter de plusieurs concepts, mais seulement des items d'évaluation qui sont inclus dans le modèle du domaine. La figure 2.2 illustre le passage d'une représentation abstraite des concepts du domaine à un modèle exécutable pour enseigner un sous-ensemble de ces concepts.

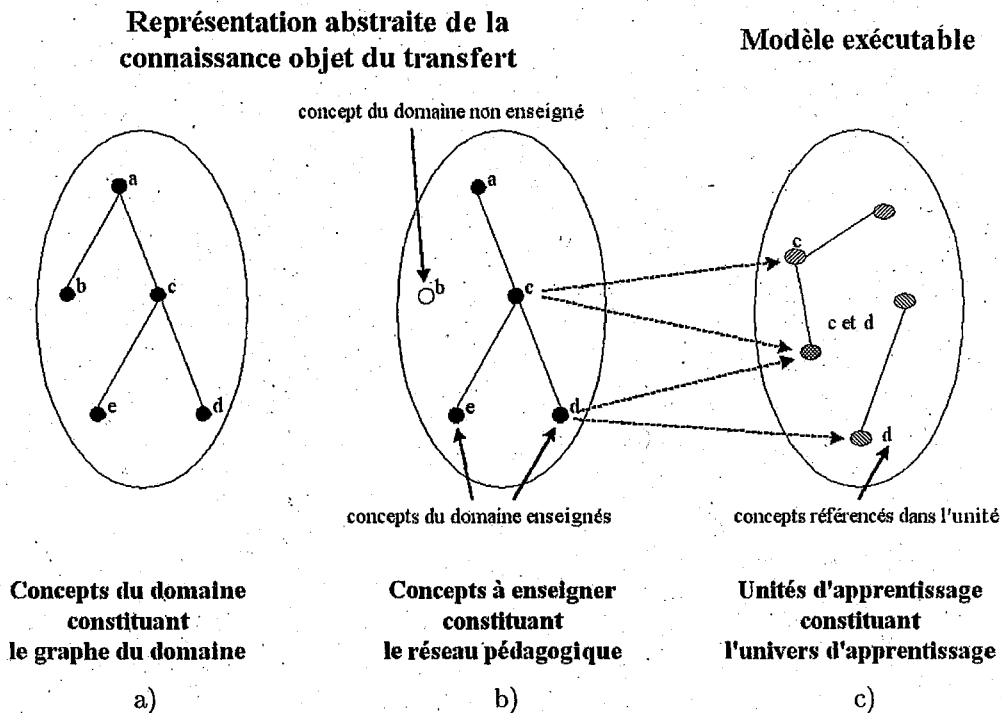


FIG. 2.2 – Représentation des concepts dans les différents contextes d'utilisation

Des relations sémantiques entre les items (figure 2.2.a) sont représentées par des liens sémantiques dans le réseau du domaine. Il y a deux types de liens. Le premier type représente l'organisation hiérarchique du domaine et le deuxième type représente des contraintes pédagogiques. Nous détaillons par la suite les items d'évaluation (la plus petite unité de connaissance sur laquelle l'enseignant-auteur porte un jugement). Un item est représenté par un libellé qui résume le concept sous-jacent du domaine. Etant sujet à évaluation, l'item est presque toujours accompagné de valeurs qui représentent le niveau de connaissances de chaque unité d'apprentissage et la compétence de chaque élève pendant tout le déroulement d'apprentissage. Chaque item d'évaluation porte deux valeurs : la mesure globale et le degré d'acquisition.

Mesure Globale

La première valeur que E. Gagnon [Gav91] a choisie est une valeur qualitative, appelée

« mesure globale ». Cette valeur permet de savoir si l'item a fait l'objet d'une présentation quelconque au cours de l'apprentissage et de situer un apport de connaissance dans le temps par rapport à la session courante. Elle peut prendre une des quatre valeurs décrites ci-dessous.

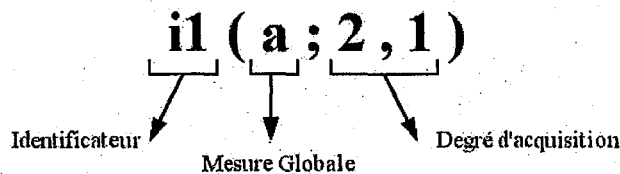
- **Inconnu (i)** : si l'item n'a jamais été présenté,
- **Abordé (a)** : si la présentation s'est déroulée pendant la session courante,
- **Vu(v)** : si la présentation de l'item remonte à une session antérieure à la session courante,
- **Acquis(q)** : si l'acquisition de cet item est antérieure au début de l'apprentissage.

La relation d'ordre sur les valeurs entre mesures globales, notées mg, est exprimée ci-dessous :

$$\text{Acquis (q)} > \text{Vu (v)} > \text{Abordé (a)} > \text{Inconnu (i)}$$

Degré d'acquisition

La deuxième est la valeur qualitative, appelée « degré d'acquisition ». Cette valeur précise la manière dont l'élève maîtrise le savoir-appliquer ou le savoir-faire relatif à l'item. Le degré d'acquisition est un couple d'entiers qui comptabilisent des jetons que E. Gavignet a attribués ou prélevés au niveau de connaissances de l'élève à la fin d'exécution d'un exercice. Le premier d'entre eux correspond à l'aspect application/restitution (savoir-appliquer) de l'exercice et l'autre à l'aspect manipulation/utilisation (savoir-faire). Voici un exemple d'item d'évaluation :



où

- il : identificateur associé à l'item concerné,
- a : sa mesure globale,
- 2 : correspond à l'aspect savoir-appliquer du degré d'acquisition,
- 1 : correspond à l'aspect savoir-faire du degré d'acquisition

il(a ; 2, 1) indique que l'item identifié par il a fait l'objet d'une présentation lors d'une session courante et qu'il a aussi fait l'objet d'évaluation lui attribuant respectivement 2 et 1 jeton.

La relation entre les degrés d'acquisition est représentée par la définition 1 ci-dessous.

Définition 1:

Soient D_1 et D_2 deux degrés d'acquisition. On pose $D_1 = (x_1, y_1)$ et $D_2 = (x_2, y_2)$ avec $x_1, x_2, y_1, y_2 \in \mathbb{N}$.

Si $\{(x_2 \geq x_1) \text{ et } (y_2 \geq y_1)\} \text{ ou } \{(x_1 \geq x_2) \text{ et } (y_2 \geq y_1)\}$

Alors $D_2 \geq D_1$

b) Représentation des niveaux de connaissances

Les objectifs pédagogiques représentent le niveau de connaissances dont l'élève doit se rapprocher. Ils regroupent les compétences relatives à chacun des items. Plus l'item a de l'importance

dans l'apprentissage et plus les nombres de jetons seront élevés. Un objectif pédagogique est dit «*atteint*» si la compétence de l'élève sur cet item est supérieure ou égale à celle exprimée dans les objectifs ou si chaque objectif pédagogique est lui-même atteint.

Exemple: considérons par exemple que le domaine soit restreint à trois items d'évaluation: i_1, i_2 et i_3 . Les objectifs pédagogiques pour un élève donné peuvent être les suivants: $\{i_1(v; 4, 4), i_2(v; 3, 6)$ et $i_3(a; 2, 7)\}$. Ils fixent les valeurs globales et les degrés d'acquisition qui doivent être approchés par l'élève pour chacun des trois items. Au cours de l'apprentissage le module tutoriel compare les compétences d'élève (niveaux de connaissances réels) et les objectifs pédagogiques (niveaux de connaissances souhaités) pour aider à décider de continuer ou arrêter. Nous détaillons maintenant la relation entre les compétences portant sur le même item.

Définition 2:

Soient C_{1k} et C_{2k} deux compétences portant sur le même item k avec $C_{1k} = \{i_k(mg_1, D_1)\}$ et $C_{2k} = \{i_k(mg_2, D_2)\}$

Si $(mg_2 > mg_1)$ **ou** $\{(mg_1 = mg_2)$ **et** $(D_2 > D_1)\}$

Alors $C_{2k} \geq C_{1k}$

Nous exprimons ensuite les niveaux de connaissances et leurs relations.

Définition 3:

Par extension soient N_1 et N_2 deux niveaux de connaissances portant sur n items au plus. On pose:

$$N_1 = \{C_{1i}\} \quad i \in 1..p, p \leq n$$

$$N_2 = \{C_{2j}\} \quad j \in 1..q, q \leq n$$

Si $\forall C_{1i} \in N_1, \exists C_{2j} \in N_2$ **tel que** $C_{2j} \geq C_{1i}$

Alors N_1 «*inclus dans*» N_2 (notée $N_1 \overset{C}{\text{dans}} N_2$)

c) Univers des unités d'apprentissage

Une fois que le réseau de domaine est complété, l'étape finale consiste à décrire chaque unité d'apprentissage dans le modèle représenté dans la figure 2.3. Une description d'unité d'apprentissage inclut trois parties: filtre, contenu et post-actions.

Filtre

Le filtre, appelé aussi pré-requis ou pré-condition, permet à l'auteur d'exprimer les conditions dans lesquelles l'apprenant peut accéder à une unité. Les conditions rendent compte des contraintes liées au passé de l'apprentissage, appelé antécédent, et le niveau des connaissances nécessaire pour exécuter l'unité. L'antécédent est l'ensemble des unités qui doit avoir été activées par l'élève durant l'apprentissage. Le niveau des connaissances est un ensemble des compétences qui doivent être acquis pour certains items d'évaluation.

Contenu

Le contenu regroupe l'ensemble des références aux informations qui seront présentées aux élèves, ainsi que les caractéristiques de l'unité dans l'univers d'apprentissage. Il a deux parties:

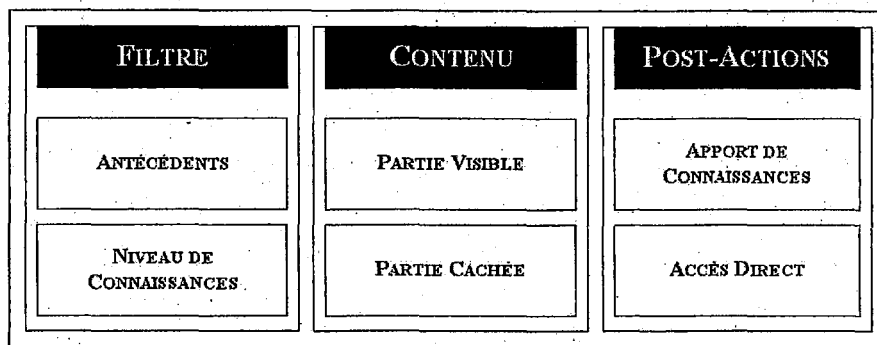


FIG. 2.3 – Composants d'une unité pédagogique.

identificateur	U12	
Nature	Application	Partie Cachée de Contenu
Items Référencés	17	
Antécédentes	aucun	Filtre
Pré-requis	17(a,0-0)	
Titre	Exercice pour équilibrer des équations	Partie visible de Contenu
<p>Ecran</p> <p>Équilibrer les équations-bilans des réactions suivantes :</p> $\text{Al} + \text{CO}_2 \rightarrow \text{Al}_2\text{O}_3$ $\text{SO}_2 + \text{O}_2 \rightarrow \text{SO}_3$ $\text{C}_2\text{H}_6 + \text{O}_2 \rightarrow \text{CO}_2 + \text{H}_2\text{O}$ $\text{Fe}_3\text{O}_4 + \text{Al} \rightarrow \text{Al}_2\text{O}_3 + \text{Fe}$ $\text{H}_2\text{SO}_4 + \text{NaCl} \rightarrow \text{HCl} + \text{Na}_2\text{SO}_4$ $\text{ZnS} + \text{O}_2 \rightarrow \text{ZnO} + \text{SO}_2$		Partie Visible de Contenu
Nombre d'écrans	1	Partie Cachée de Contenu
Accès direct à	aucun	
Apports	17(.,6-0)	Post - Actions

FIG. 2.4 – Exemple de description d'une unité d'apprentissage

la partie visible du contenu qui correspond à une action pédagogique présentée pour l'élève donc elle est composée d'un titre, d'une référence, d'un fichier exécutable et d'un nombre d'écrans. La deuxième partie est la partie cachée, ceci permet d'identifier l'unité (identificateur et nature) et de calculer les liens entre les unités. Cette information est importante pour choisir une unité plutôt qu'une autre et juger de son aptitude à satisfaire les objectifs pédagogiques.

Post-actions

Les post-actions sont des informations fournies par l'enseignant. Elles interviennent à la fin d'exécution d'une unité d'apprentissage et ont pour effet de modifier la valeur associée aux items abordés au cours de l'activité. Par exemple dans la partie «post-actions» de la figure 2.4, l'opération d'après l'exécution de cette unité est la suivante :

Si une seule réponse correcte sur 7
Alors modifier la compétence relative à l'item i7 en ajoutant 1 jeton au degré d'acquisition (application)
Sinon Si deux réponses correctes sur 7
Alors modifier la compétence relative à l'item i7 en ajoutant 2 jetons au degré d'acquisition (application)
Sinon Si trois réponses correctes sur 7
Alors modifier la compétence relative à l'item i7 en ajoutant 3 jetons au degré d'acquisition (application)
Sinon Si

d) Relations entre des unités d'apprentissage

L'univers des unités d'apprentissage est aussi organisé en réseau, représenté dans la figure 2.2.c. Les nœuds représentent des unités d'apprentissage avec les traits liés entre les nœuds représentant la relation entre eux. Certains liens sont statiques, pré-définis par l'auteur/enseignant. Mais le système ECSAI crée dynamiquement la plupart des liens en utilisant la partie «*filtre*» de l'unité et les informations stockées dans le module d'élève.

Liens Statiques

Les liens statiques existent indépendamment de l'élève en situation d'apprentissage. Ces liens s'appuient uniquement sur les informations contenues dans la partie filtre et post-conditions des unités. Ils sont divisés en trois catégories : «*accès-direct-à*», «*suivi-de*» et «*même-que*».

Les liens «*accès-direct-à*» expriment une relation privilégiée entre deux unités. Ces liens sont les seuls définis par l'enseignant-auteur au moment de la description des unités d'apprentissage (dans partie «*post-actions*»).

Les liens «*même-que*» expriment une relation de substitution entre deux unités d'apprentissage. Nous pouvons les définir de la manière suivante :

Définition 4:

$$\underline{U1 \text{ « même-que » } U2} \quad \text{si} \quad \left\{ \begin{array}{l} \text{Items référencés (U1)} = \text{Items référencés (U2)} \\ \text{et Filtre (U1)} = \text{Filtre (U2)} \\ \text{et Apport (U1)} = \text{Apport (U2)} \end{array} \right\}$$

Les liens «*suivi-de*» correspondent à une relation «*suivi immédiatement de*» entre deux

unités. Ces liens peuvent être exprimés de la façon suivante :

Définition 5:

Soit le filtre de l'unité U2 constitué de :

U_{2A} : l'ensemble des unités devant être activées avant l'unité U2

U_{2N} : le niveau de connaissances requis pour activer l'unité U2

Soit U1 une unité d'apprentissage dont :

U_{1F} : le prérequis.

U_{1P} : les antécédents

U_{1A} : l'apport de connaissances

L'unité U2 est dite « **accessible après** » l'unité U1 si $(U_{2A} \subset U_{1P})$ et $\{U_{2N} \overset{C}{\text{dans}} (U_{1F} + U_{1A})\}$

Autrement dit, U1 « **suivi-de** » U2 ou U2 « **accessible après** » U1 si les conditions exprimées dans le filtre de U2 sont couvertes à la fin de l'exécution de U1.

La relation « *suivi-de* » n'est pas symétrique. Par exemple U3 peut être activée après U1 mais l'inverse n'est pas possible. Cette relation n'est pas transitive non plus. On a, par exemple, « *textitU1 suivi-de U3* », « *U3 suivi-de U4* » et pourtant « *U1 suivi-de U4* » n'existe pas.

Liens dynamiques

Les liens dynamiques sont spécifiques à chaque élève et sont générés à la fin d'exécution d'une unité. Ils sont calculés en utilisant le niveau de connaissances de chaque élève, le passé de son apprentissage et les informations dans les unités d'apprentissage. Ils existent deux catégories de liens : « *équivalent-à* » et « *implique* ».

Les liens « *équivalent-à* » relient deux unités qui portent la même référence, le même apport de connaissances et qui sont toutes les deux accessibles pour l'apprenant. Nous pouvons les exprimer de manière suivante :

Définition 6:

Soit le filtre de l'unité U constitué de :

U_A : l'ensemble des unités devant être activées avant l'unité U

U_N : le niveau de connaissances requis pour activer l'unité U

Soit C des connaissances de l'apprenant X,

Soit P le passé de l'élève X.

L'unité U est dite « **accessible pour** » l'apprenant X si $(U_A \subset U_P)$ et $\{N \overset{C}{\text{dans}} C\}$

$U1 \llbracket \text{équivalent-à} \rrbracket U2$ si $\left\{ \begin{array}{l} \text{Items référencés (U1)} = \text{Items référencés (U2)} \\ \text{et Apport (U1)} = \text{Apport (U2)} \\ \text{et U1 « accessible pour » l'apprenant X} \\ \text{et U2 « accessible pour » l'apprenant X} \end{array} \right\}$

Les liens « *implique* » sont le dernier type de liens qui peuvent être considérés comme une extension du lien « *suivi-de* ». Les conditions portent sur les même éléments mais les connaissances sont élargies à celles contenues dans le module de l'apprenant. Par exemple, U1 « *implique* » U2, si le niveau de connaissances de l'élève après l'activation de U1 est suffisant par rapport à celui exprimé dans le filtre de U2 et si les éventuelles unités nécessaires avant l'activation de l'unité

U2 font parties de celles que l'élève a déjà activées. Ce qui peut traduire de la façon suivante :

Définition 7:

U1 « implique » U2 si { U2 « accessible pour » l'apprenant X }

Autrement dit, après l'exécution de l'unité U1, l'apprenant X peut accéder à l'unité U2.

2.2.3 Générateur du parcours d'apprentissage

La stratégie pédagogique retenue, en général, a pour fonction de choisir un objectif pédagogique en rapport avec l'acquis de l'apprenant, de lui suggérer une activité répondant le mieux à ses objectifs, puis de contrôler le comportement de l'élève pour évaluer sa progression.

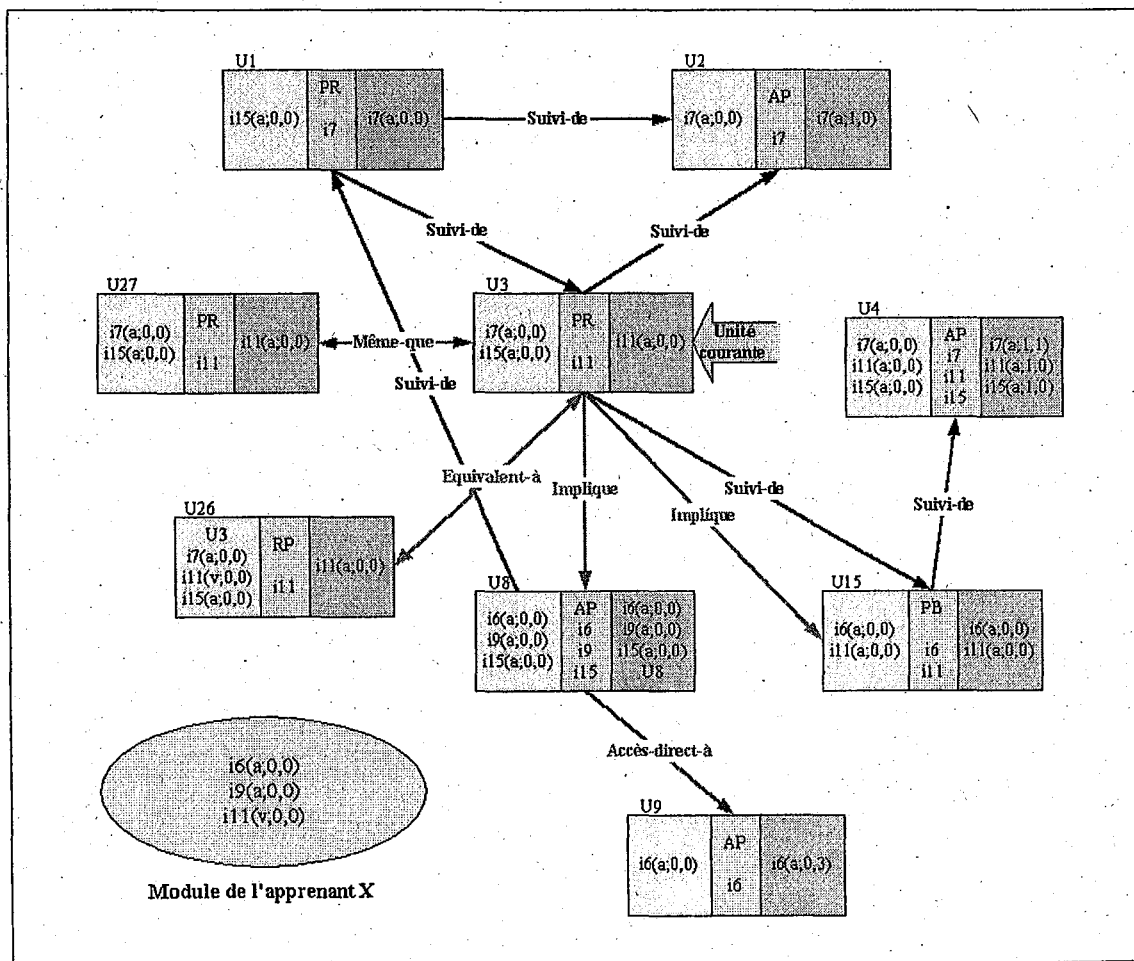


FIG. 2.5 – Liens pour l'apprenant X

La figure 2.5 met en évidence toutes les possibilités de liens qui sont offertes à l'élève X

à partir de l'unité U3. Le problème se pose maintenant de choisir l'une des unités candidates pour pouvoir continuer. ECSAI a proposé plusieurs étapes pour raffiner les unités candidates, montrées dans la figure 2.6.

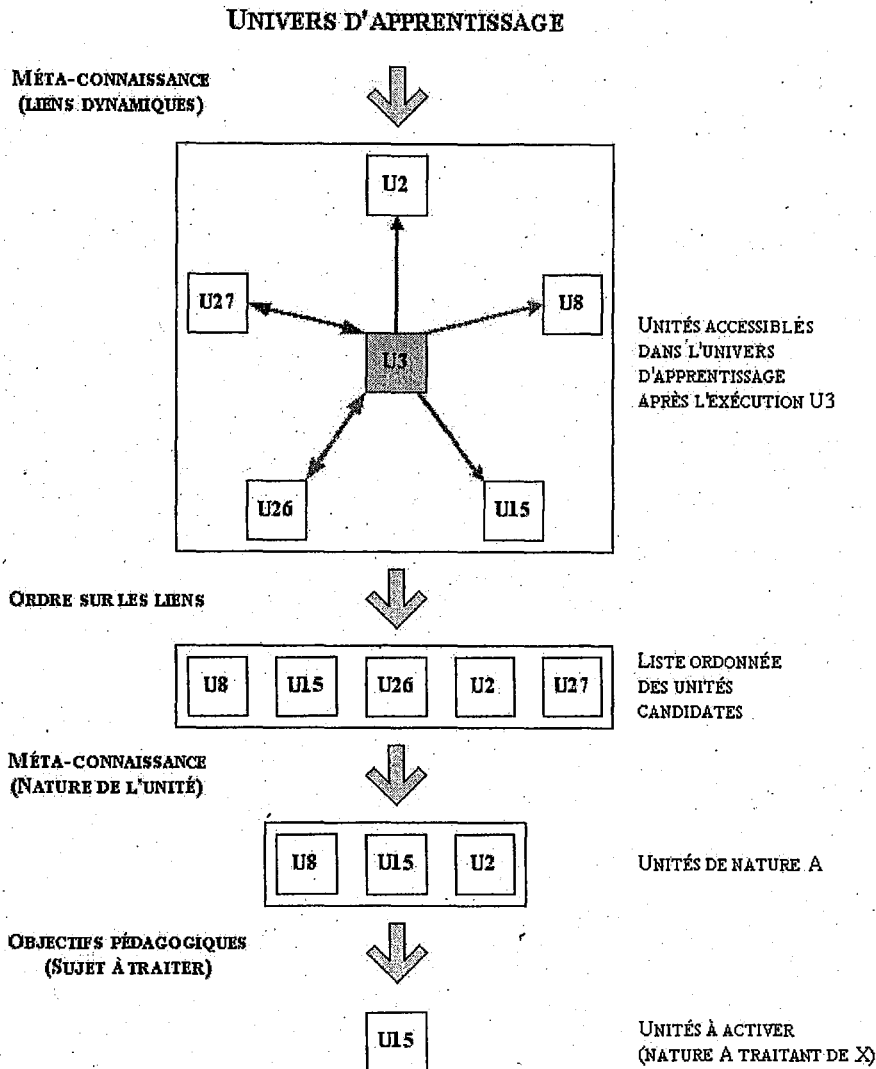


FIG. 2.6 – Processus de raffinement du choix de la prochaine unité d'apprentissage

La première étape détermine, à partir de l'unité courante, les unités accessibles en fonction des liens qui peuvent être établis. Le second étape consiste à choisir la nature de lien qui convient. Ce choix s'effectue à partir de l'ordre importance. Cet ordre privilégie d'abord les indications fournies par l'enseignant (lien « accès-direct-à »). Ensuite, pour que l'apprentissage reste le plus fidèle possible à l'élève, le système donne l'ordre importance aux liens dynamiques et puis aux liens statiques. En effet, l'ordre décroissant utilisé est le suivant : accès-direct-à, implique et suivi-de ou accès-direct-à, équivalent-à et même-que. La troisième étape permet d'ordonner les unités

candidates à l'intérieur de la liste sélectionnée. Le critère des choix dans cette étape est la nature de l'unité choisie. Les méthodes employées dans ECSAI sont présentation-présentation-application (présentation d'abord) ou présentation-application-application (suivi immédiatement d'une évaluation). Enfin, la dernière étape est de choisir les objectifs pédagogiques. Si l'unité choisie porte plusieurs items référencés alors le système choisit l'item qui est en tête de la liste.

2.2.4 Modes d'apprentissage

Le mode d'apprentissage s'exprime essentiellement par le choix des unités d'apprentissage. Pour naviguer dans l'univers d'apprentissage, ECSAI propose deux modes : libre et guidé. Dans le premier mode, mode libre, un ensemble d'activités est proposé à l'élève en fonction des items sur lesquels il souhaite travailler. Aucun contrôle de son parcours d'apprentissage n'est effectué. Le système joue le rôle d'esclave : il propose les activités et exécute celles qui sont choisies par l'apprenant. Dans le deuxième mode, mode guidé, le système effectue à chaque étape le choix de la prochaine unité à exécuter. Le choix des unités est représenté dans la figure 2.6. Les connaissances de l'élève sont mises à jour chaque fois que cela est nécessaire.

2.2.5 Points forts et faibles du système ECSAI

Avant d'expliquer l'adaptation du système ECSAI au Web, nous décrivons maintenant les points forts et les points faibles de ce système.

a) Points forts

Les points forts de ce système concernent l'environnement de conception (l'environnement auteur) et ses modèles. Cet environnement est un environnement auteur «ouvert» qui permet aux enseignants de créer le cours par eux-mêmes. Il offre une accessibilité accrue à l'enseignant, facilite la maintenance du produit didactique, vérifie certaines incohérences dans l'élaboration de l'univers d'apprentissage, reste portable à divers domaines d'enseignement. En effet, plusieurs outils d'aide à l'instanciation de modèles offrent à l'enseignant-auteur la possibilité de formaliser son propre enseignement pour une matière donnée selon ses besoins. C'est-à-dire, l'enseignant-auteur peut, à volonté, créer et modifier les contenus pédagogiques sans avoir à se soucier de leur agencement qui est à la charge du système. Nous avons trouvé dans plusieurs systèmes tutoriels intelligents que leurs modèles sont strictement dépendants de la matière d'étude. Au contraire, ECSAI propose les modèles ouverts qui sont indépendants de la matière. Avec ce modèle, nous pouvons facilement passer d'une matière à une autre.

b) Points faibles

L'auteur de ce système, E. Gavignet, a mis l'accent sur la modélisation des connaissances du domaine et pédagogiques. Donc nous avons trouvé plusieurs outils, cités dans la partie description du prototype de sa thèse, destinés aux utilisateurs principaux qui sont les enseignants. Mais nous en avons trouvé moins pour l'élève qui est le plus important utilisateur. L'élève ne peut choisir que les modes d'apprentissage : guidé ou libre. Nous avons trouvé également le problème de distribuer et mettre à jours les cours. Chaque fois que l'enseignant modifie ses cours il faut avoir le temps pour remplacer l'ancienne version par la nouvelle version. ECSAI est encore dépendant de la plateforme. Il est donc difficile de le transférer sur une autre plateforme.

2.3 Passage d'un système autonome à un système sur le Web

Selon nos besoins exprimés au début de cette thèse et après avoir étudié l'ancien système (ECSAI) de notre équipe, nous avons décidé de prendre les concepts principaux du système ECSAI. Nous souhaitons que notre système soit complètement ouvert pour tous types utilisateurs, facilite aux enseignants l'assimilation et des modèles pour enrichir effectivement des contenus, facilite la mise à jour des matériaux pédagogiques, améliore la façon de travailler pour les utilisateurs, soit complètement indépendant de la plateforme. Pour répondre à tous nos besoins, nous choisissons d'adapter l'ancien prototype ECSAI au Web. Cela signifie que nous devons entièrement récrire le système. Il se pose alors le problème de choisir l'architecture de notre système. Les architectures des systèmes basés sur le Web sont détaillées dans la section suivante. Le choix de l'architecture de notre système fait l'objet de la section 2.6.

2.3.1 Architecture des systèmes basés sur le Web

Plusieurs architectures des systèmes basés sur le Web ont, jusqu'à maintenant, émergé. Nous pouvons les regrouper en trois catégories selon l'endroit où les fonctions de système sont exécutées : centralisée, décentralisée et distribuée.

a) Architecture centralisée

Dans l'architecture centralisée, représenté dans la figure 2.7, les serveurs Web et des applications fonctionnent du côté de serveur, alors que l'interface des utilisateurs est montrée dans un navigateur Web sur la machine de client. Le serveur d'application exécute toutes les fonctions du système. L'utilisateur agit avec des formulaires de HTML, et l'information sur ces formulaires est envoyée au serveur Web, qui passe les demandes et les actions d'utilisateur vers le serveur. Le serveur et l'interface communiquent typiquement par l'intermédiaire des programmes de CGI. L'information envoyée par un navigateur Web est traitée par un programme CGI externe, et les résultats sont renvoyés sous forme de nouvelles pages de HTML. PAT-Online [Rit97], un tuteur d'algèbre, est exploité avec cette architecture. Une autre option pour créer les systèmes d'enseignement centralisés est d'employer les serveurs du Web programmables qui peuvent être agrandis avec coder l'application, puis éliminant le besoin pour les programmes CGI externes. C'est l'architecture que SQLT-Web [Mit00] et ELMART [Bru96] ont exploitée.

b) Architecture décentralisée

Dans l'architecture décentralisée, le système entier réside dans une applet Java qui doit être téléchargée et est exécutée sur la machine des utilisateurs. Toutes les fonctions de système sont donc exécutées sur la machine du client, alors que le serveur est seulement employé comme dépôt du logiciel à télécharger. L'exemple de cette architecture est le tutoriel développé par l'outil auteur DCG [Vas97], représenté dans la figure 2.8.

c) Architecture distribuée

Dans l'architecture distribuée, les fonctions du système sont distribuées entre le client et le serveur. La politique exacte pour distribuer les fonctions peut changer. Le plus souvent, le serveur d'application se compose d'un modèle d'étudiant, d'un module de domaine, et d'un module pédagogique. L'interface utilisateur habituellement basée sur Java peut exécuter quelques fonctions d'enseignement. La communication entre l'interface et le serveur d'application n'implique

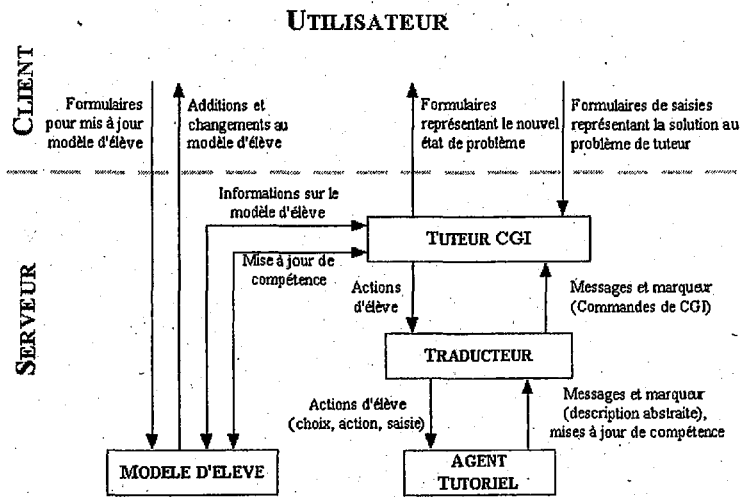


FIG. 2.7 – Architecture centralisée : PATOnline

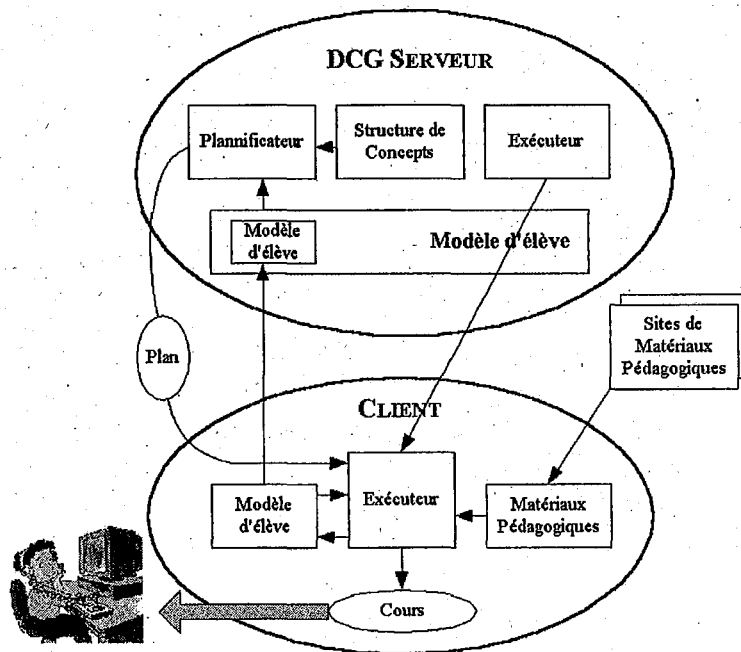


FIG. 2.8 – Architecture décentralisée : DCG

pas nécessairement le serveur Web; il est possible d'établir un raccordement direct de TCP/IP²⁹ entre l'applet et le serveur d'application afin d'accélérer le système. AlgeBrain [ASF99] et Belvedere [SJ97] sont basés sur cette architecture. AlgeBrain, représenté dans la figure 2.9, soutient des étudiants pour résoudre des équations algébriques pendant le parcours d'apprentissage. Un applet du module de téléchargement fournit une interface d'utilisateur impliquant un agent qui réagit à une action d'élève, et fournit la rétroaction immédiate sur chaque étape d'élève. Belvedere est un système pour apprendre à faire des enquêtes. Le langage Java est employée pour fournir l'interface d'utilisateur, alors que le serveur d'application offre une variété d'outils.

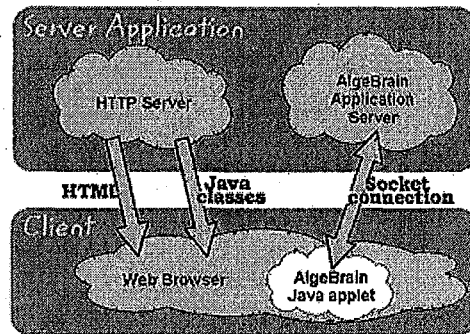


FIG. 2.9 – Architecture distribuée : AlgeBrain.

2.3.2 Avantages et désavantages des architectures

La quantité d'effort impliquée en construisant un système à architecture décentralisée est identique à celle pour établir un système autonome. Ces systèmes sont très rapides, parce que tout le traitement est fait dans la machine du client. Cependant, une limitation significative de cette architecture est le fait que le modèle d'élève doit être stocké sur la machine où le système a été exécuté. Par conséquent, l'élève doit toujours employer le système de la même machine s'il/elle veut tirer bénéfice des sommaires des sessions précédentes stockées dans le modèle d'élève. Une solution intéressante à ce problème peut être trouvée dans DCG [Vas97], représenté dans la figure 2.8, où des copies des modèles d'élève sont également gardées sur le serveur entre les sessions pour le stockage permanent. Bien que cette solution enlève la condition qu'un élève doit toujours utiliser le système dans la même machine, il y a aussi des problèmes si une erreur de réseau se produit avant que l'élève accomplisse une session, car les informations les plus récentes sur l'exécution de l'élève seront alors perdues.

Dans des architectures décentralisées et distribuées, il est nécessaire qu'un élève télécharge le logiciel afin de commencer à employer le système – une tâche que quelques élèves peuvent trouver frustrante. En outre, il est nécessaire de télécharger chaque nouvelle version d'un système pour tirer bénéfice des améliorations. Dans le cas d'une architecture centralisée, il n'y a pas de tel problème. Un avantage significatif des architectures centralisées et distribuées est le fait que les modèles d'élève soient maintenus dans un endroit (sur le serveur) et l'élève peut employer le système sur n'importe quelle machine. Un problème de cette architecture peut être la vitesse réduite, provoquée par des communications entre le client et le serveur et par l'interprétation des commandes envoyées par le poste de client. La situation pourrait être meilleure pour un système

²⁹ sigle anglais de *Transmission Control Protocol/Internet Protocol*

avec une architecture distribuée, où certaines des actions de système sont effectuées du côté du client et par conséquent le nombre de communications est réduit. Cependant, communiquer entre l'interface et le serveur dans une architecture distribuée peut exiger des techniques spéciales, qui présentent une complexité additionnelle pour le développement de système.

2.4 Evolution des «formations pilotées par des enseignants» à des «formations centrées sur des apprenants»

L'approche systématique dont nous partons pour modéliser la situation d'apprentissage place l'apprenant au centre du système pédagogique (voir figure 2.10): «... On voit que l'apprenant occupe ici la place centrale. Tout part est revient à lui, mais il ne présente pas un entité indépendante. ...»³⁰. Etant donné que ce système ne résulte pas de la somme des fonctionnalités qui le constituent, mais de leur interaction réciproque, chacune d'entre elles est nécessairement solidaire de toutes les autres, non seulement du point de vue formel, mais surtout du point de vue des contenus sémantiques. Chaque segment de la praxis pédagogique, véhicule des signifiés qui retentissent sur tout le processus.

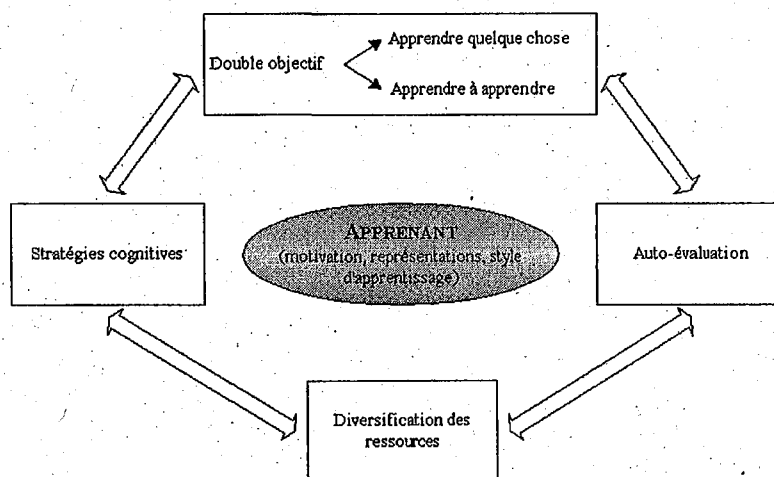


FIG. 2.10 – Système pédagogique

Mieux connaître l'apprenant et l'orienter vers sa propre connaissance constitue une opération délicate, car tout ce qui pourrait être vécu comme typologie et jugement aurait un effet pervers. Nous proposons cependant de prendre en compte un certain nombre de variables, qui peuvent s'appliquer à la notion de «profil de l'apprenant».

Barbot et Camatarri [BC99a] a indiqué que l'acte pédagogique est un acte social qui n'est jamais neutre. Par conséquent, qu'on le veuille ou non, l'innovation met en jeu autre chose que du pédagogique. C'est pour cela que le résultat de l'innovation ne peut venir de l'école seule, car en tant que lieu de convergence et de tension des composantes de toute la société, elle se trouve à la fois conservatrice et novatrice: «Peut-on de l'école, ou du moins de ses agents, la production de modèles culturels sinon révolutionnaires, du moins résolument évolutifs? Elle peut simplement y contribuer dans une société relativement ouverte et qui lui laisserait du champ.»³¹.

30. Approche systématique de Richterich [BC99a, page 25]

31. L'acte pédagogique est un l'acte social [BC99a, page 160]

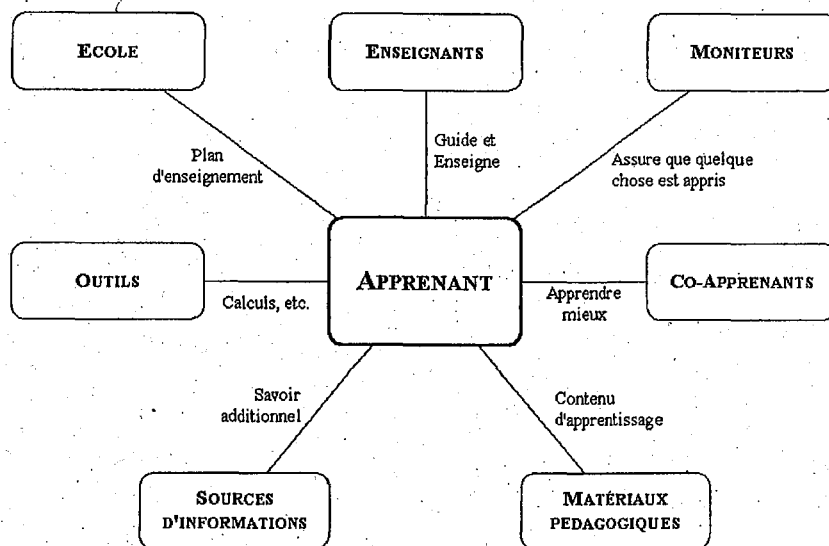


FIG. 2.11 – Pédagogie centrée sur l'apprenant

Par conséquent, les innovations pédagogiques proposées sont les plus souvent considérées comme un moyen privilégié – sinon le seul – d'atteindre une meilleure démocratisation des études, une diminution sensible de l'inégalité des chances devant l'enseignement et même parfois, et par voie de conséquence, une réduction des inégalités sociales.

La changement du système apprentissage d'autonomie au Web change aussi le mode de communication. L'apprenant situé au centre de l'apprentissage peut collaborer avec des autres utilisateurs : les enseignants, les moniteurs, les co-apprenants, et/ou accéder à toutes les informations nécessaires : plan d'enseignement, outils, cours et etc., présentés dans la figure 2.11. La seule façon d'obtenir ce type des fonctionnalités est d'utiliser l'Internet parce que les autres intermédiaires n'offrent pas toutes les fonctionnalités que nous avons souhaitées.

2.5 Évolution des formations traditionnelles aux formations à distance

En peu de temps, les technologies éducatives pour une éducation de niveau plus élevé et la formation à distance ont évolué de la production des multimédia et du courseware basé sur l'hypertexte dans le CDROM aux scénarios gérés en réseau facilitant l'interaction, les activités communes, et le partage des ressources. Nous avons déjà vu un ensemble de métaphores naissantes telles que le campus virtuel [Har99] pour conceptualiser ce nouveau cadre.

La situation est parfois exposée sur la forme d'un passage d'un modèle de formation individuelle au cours à distance véritablement interactif dans un scénario prolongé de salle de classe au moyen d'un ensemble d'outils fournissant la communication synchrone. La technologie offre un grand potentiel, mais comment adapter la technologie dans la pratique courante des établissements est une question controversée embrassant les points de vue didactiques, organisationnels, sociaux, et économiques. La situation est plus dure quand la grande échelle et les solutions robustes sont nécessaires pour manipuler des cours traitant un nombre élevé d'étudiants enregistrés. C'est le cas des universités ouvertes fonctionnant actuellement avec le modèle

industriel, caractérisé par une utilisation dominante des médias à sens unique. Avec cette approche, le coût d'éducation ne dépend pas fortement du nombre d'étudiants. En fait, le taux de personnel scolaire par étudiant est habituellement bas, dans l'ordre de 1/500³² pour l'étudiant préparant une licence. Les salles de classe virtuelles ne sont pas en tant que telles une solution de rechange pour l'éducation à distance. Tout d'abord, elles sont encore chères et non réalistes pour s'attendre à ce que chaque étudiant ait un raccordement RNIS et une WebCam à la maison. En outre, les étudiants à distance ont des restrictions d'établissement du programme et il est habituellement difficile qu'ils soient disponibles à l'horaire exigé pour un événement synchrone, et en conclusion, il est hors de question d'offrir un service «one to a few» quand le rapport dans votre établissement est d'un professeur pour plus de cent étudiants.

Des systèmes d'éducation supportés par ordinateur ne peuvent pas être mis en application avec succès s'ils sont conçus et projetés sans tenir compte soigneusement des contraintes, des demandes et des choix spéciaux de l'environnement : le choix des ressources, les techniques pédagogiques et les modèles pour faciliter la dérive d'apprentissage de l'environnement actuel.

Une stratégie pour guider le développement de la technologie gérée en réseau est d'ignorer l'utilisation courante et de concevoir un nouveau scénario didactique. Un exemple de cette approche est la nouvelle conception du cours de programmation à l'université ouverte britannique [Site M206], à l'occasion d'un changement d'un point de vue (passage d'un paradigme procédural à un paradigme orienté objet) sur les contenus et les objectifs du cours. Une stratégie pragmatique est de trouver des occasions d'explorer les nouveaux scénarios d'étude, de faire des petites expériences, développements et d'injecter progressivement des technologies gérées en réseau dans la pratique de l'éducation de distance.

2.6 Application à l'évolution d'ECSAI

Le système ECSAI basé sur le Web, appelé ECSAIWEB, devra être développé pour soutenir l'enseignement des étudiants résidentiels et à distance en utilisant le Web comme plateforme commune de la distribution des matériaux pédagogiques. Alors pour passer le système ECSAI au Web, nous étudierons la possibilité d'adapter les modèles utilisés dans ECSAI, d'une part et l'environnement pour le Web d'autre part. Le nouveau support, Internet, nous permet également d'avoir la possibilité de faire évoluer des formations pilotées par les enseignants vers des formations centrées sur les apprenants et de faire évoluer des formations traditionnelles vers des formations à distance. Pour appliquer ces évolutions à ECSAI, nous décrirons alors dans la section 2.6.1 la liste des conditions pour un système basé sur le Web et les premières contraintes d'implantation de notre système dans la section 2.6.2.

2.6.1 Liste des conditions pour un système sur le Web

A partir des études sur le système autonome ECSAI et les systèmes existants basés sur le Web, nous développons une liste de conditions pour un système basé sur le Web que nous voulons tenter. Ce sont :

- un dépôt centralisé des modèles d'élève,
- la nécessité de télécharger le logiciel,
- un support des multiples élèves simultanés,
- la liberté pour accéder au système à tout moment et à n'importe quel endroit,
- la facilité de mettre à jour les matériaux pédagogiques,

32. Le cas d'études du projet STEED (Sistema Teleático para el Estudio en Enseñanza a Distancia)

- la possibilité aux utilisateurs de passer du travail individuel au travail collaboratif.

Ces conditions nous permettent de décider d'employer l'architecture centralisée parce que cette architecture peut remplir toutes les conditions que nous voulons.

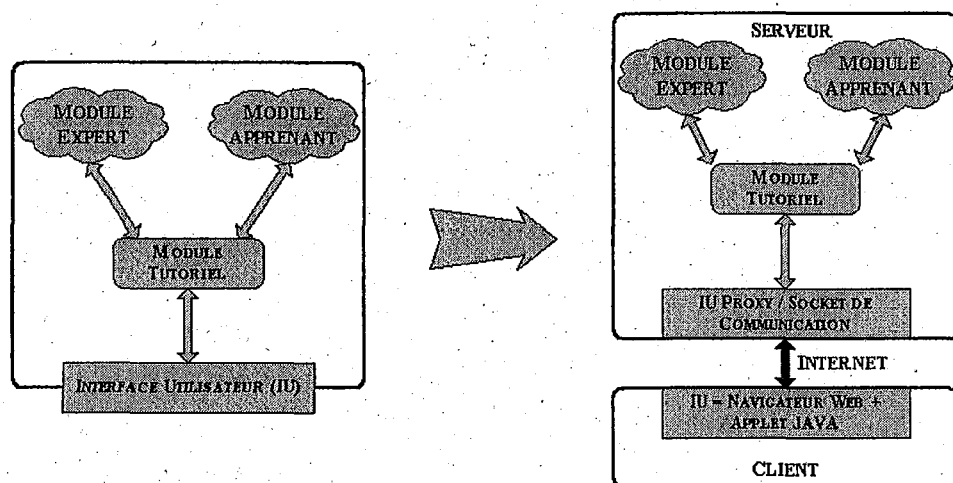


FIG. 2.12 – Evolution de système autonome via le Web

2.6.2 Contraintes d'implantation

A partir du paragraphe précédent, nous décrivons maintenant la première contrainte d'implantation de notre système. Selon l'architecture centralisée, tous les composants d'un environnement doivent résider dans un serveur. Alors la meilleure façon de les communiquer est de faire passer par le programme HTTPD³³ sur le serveur. Ce choix est fait en raison de la simplification de manipulation, entretien et extension des fonctionnalités. Le HTTPD soutient un développement d'application en prolongeant directement le serveur Web en utilisant des scripts de CGI. Les programmeurs peuvent définir des fonctions de CGI pour manipuler des demandes entrées dans les formulaires, et produire des pages «*on-the-fly*» de HTML et renvoyer ces pages aux navigateurs des postes de clients. Le HTTPD est basé sur la programmation multi-tâches, et crée un fils séparé pour répondre à chaque client. Pendant que plusieurs élèves qui emploient le système concurremment partagent quelques composants du système, il est nécessaire de présenter un dispositif de verrouillage pour assurer la non-intervention entre des diverses sessions. Le système doit également maintenir plusieurs élèves à la fois, il doit associer la demande de chaque élève au modèle d'élève correspondant. Nous discutons plus tard comment ECSAI basé sur le Web soutient les élèves dans le modèle d'élève (chapitre 5).

2.7 Conclusion

Nous reprenons les principes d'ECSAI pour construire ECSAIWEB. Nous conservons un ensemble d'unités indépendantes, des modèles du domaine et un générateur dynamique des liens entre unités d'apprentissage. Nous allons modifier quelques parties dans les modèles du domaine

³³ sigle anglais de *HyperText Transfer Protocol Daemon*

et les unités d'apprentissage pour qu'ils supportent l'hypermédia et soient adaptés à une utilisation sur le Web. Nous conservons deux environnements : environnement auteur (point fort du système ECSAI) et environnement élève.

Toutefois d'autres évolutions sont nécessaires :

- un support de communication asynchrone et synchrone destiné à favoriser l'interaction, la coopération et le travail de groupe entre les professeurs, les étudiants, et les professeurs et les étudiants,
- une meilleure adaptation aux besoins des élèves sur le Web.

Pour cela nous décrivons au chapitre suivant les techniques d'adaptation disponibles.

Chapitre 3

Adaptation d'Hypermédia

3.1 Introduction

Comme nous l'avons indiqué, un des buts de nos recherches est le développement des applications d'apprentissage avancées basées sur le Web qui peuvent offrir une certaine quantité d'interactivité et d'adaptabilité afin de soutenir les étudiants qui ont différentes connaissances de base et qualifications en début d'apprentissage. Il existe dans certains systèmes d'apprentissage en ligne, des adaptabilités pour les différents types de modèles d'utilisateur. Ce chapitre présente les techniques adaptatives existantes exploitées sur le Web.

3.2 Adaptation dans les systèmes d'apprentissage basés sur le Web

En général, un hypermédia se compose d'un ensemble des nœuds ou «pages» reliés par des liens. Chaque page contient de l'information locale et un certain nombre de liens aux pages relatives. Ces liens peuvent apparaître dans le contenu d'une page, dans un menu séparé ou sur un cadre local séparé. Les systèmes d'hypermédia peuvent également inclure un index et un cadre global qui fournissent des liens à toutes les pages accessibles. Ce qui peut être adapté dans l'hypermédia adaptatif sont les contenus des pages régulières (adaptation de niveau contenu) et les liens aux pages régulières, aux pages d'index et aux cadres (adaptation de niveau lien). En général, l'adaptation de niveau contenu est employée pour résoudre le problème des systèmes d'hypermédia employés par différentes classes d'utilisateurs, alors que l'adaptation de niveau lien est employée pour fournir le support de navigation et pour empêcher des utilisateurs d'être perdus dans l'hyperespace. Nous distinguons l'adaptation de niveau contenu et de niveau lien comme deux manières différentes d'adaptation de l'hypermédia et appelons la première *la présentation adaptative* et la dernière *le support adaptatif de navigation*. A l'avenir, le support adaptatif de collaboration, particulièrement conçu pour le contexte de l'éducation basé sur le Web, pourra s'ajouter à cette liste.

Pour l'enseignement, l'ensemble des pages est une partie des matériaux pédagogiques qui peuvent être présentés :

- **de manière statique**, la présentation du matériel pédagogique est fixée une fois pour toutes et est donc toujours la même,
- **de manière dynamique**, la présentation du matériel pédagogique peut varier et ce de manière automatique en fonction de certains éléments (caractéristiques de l'apprenant, état du système, etc.).

Lorsque la présentation des matériaux pédagogiques est dynamique, nous distinguerons les facettes suivantes relatives à ce point.

- **Contenu adaptatif** : le contenu est adaptatif quand l'information affichée à l'écran de l'utilisateur peut varier dynamiquement et automatiquement en fonction de critères prédéfinis comme par exemple le taux de réussite aux exercices, la partie de matière déjà lue, le temps passé sur une section, etc.,
- **Contenu adapté** : on peut aisément imaginer qu'un même cours sera suivi par des publics à profil différents (un gestionnaire de projet et un technicien, par exemple) pour obtenir des formations différentes. Il est clair que les intérêts des publics et leurs attentes respectives peuvent différer dans de tels cas. On parlera de contenu adapté lorsque la matière présentée à l'apprenant correspond au profil de l'apprenant et ce pour un même cours,
- **Navigation adaptative** : on parle de navigation adaptative lorsque l'ensemble des activités pouvant être effectuées à partir d'un point du cours est calculé dynamiquement en fonction de critères préétablis,
- **Navigation adaptée** : tout comme le contenu, le chemin à parcourir dans la matière peut varier en fonction du profil de l'apprenant. Par exemple, une personne intéressée par un aperçu se verra proposer les parties introductives du cours tandis qu'une personne souhaitant bénéficier d'une formation approfondie se verra présenter une matière beaucoup plus détaillée.

Par ailleurs, la navigation peut être de l'un des trois types décrits ci-dessous.

- **Navigation contrainte** : la navigation dans un cours peut être complètement contrôlée par le système ou non. Une navigation linéaire est un exemple typique de contrôle strict de la navigation : les éléments d'informations sont alors présentés les uns à la suite des autres à la manière des pages d'un livre que l'on tourne,
- **Navigation libre ou ouverte** : dans le cas d'une navigation totalement non imposée, l'apprenant est libre de passer d'un élément d'information à l'autre sans aucune contrainte venant du système. De même, un apprenant souhaitant obtenir une information sur un point bien précis de la matière pourra, dans un schéma de navigation ouverte, consulter directement les quelques écrans l'intéressant sans devoir pour autant consulter tous les préalables éventuels de cette section. Il faut toutefois remarquer qu'une navigation libre n'empêche pas les conseils de navigation,
- **Navigation semi-ouverte** : dans ce cas, l'apprenant peut à tout moment accéder à la table de matières de l'unité d'apprentissage courante ; il peut consulter librement les sous-unités de cette unité courante. L'accès libre à n'importe quelle page de cours (i.e. l'accès à des pages appartenant à des unités d'apprentissage d'autre que l'unité courante) requiert cependant quelques manipulations car l'apprenant n'a jamais une vue complète et simultanée de toutes les unités d'apprentissage ; c'est pourquoi nous qualifions la navigation semi-ouverte ou semi-libre.

Dans cette section nous détaillerons d'abord la présentation adaptative et puis le support adaptatif de navigation.

3.2.1 Présentation adaptative

L'idée des diverses techniques adaptatives de présentation est d'adapter un contenu d'une page consultée par un utilisateur particulier aux connaissances actuelles, aux buts, et à d'autres caractéristiques de l'utilisateur. Par exemple, un utilisateur qualifié peut être destinataire d'informations plus détaillées et plus approfondies tandis qu'un débutant peut recevoir des explications

additionnelles. Les techniques adaptatives de présentation existantes sont actuellement une adaptation des textes. L'adaptation des textes implique que les différents utilisateurs dans le temps peuvent obtenir différents textes comme contenu de la même page. Nous groupons toutes ces techniques dans une technologie appelée *la technique adaptative de présentation des textes*. Actuellement, cette technologie est la technologie la plus étudiée de l'adaptation d'hypermédia. La plupart des premiers travaux sur l'hypermédia adaptatif (HA) a été concentrée sur la présentation adaptative des textes [Bea94] [BE94] [Bru92]. Cette direction de recherche a été influencée par la recherche sur une explication adaptative et présentation adaptative dans les systèmes intelligents [MS89] [Par88]. Dans la section 3.3, nous décrivons un certain nombre de techniques pour la présentation adaptative des textes qui ont été développées.

3.2.2 Support adaptatif de Navigation

L'idée des techniques adaptatives de navigation a pour objectif d'aider des utilisateurs à trouver leurs chemins dans l'hyperespace en adaptant la présentation de liens aux buts, à la connaissance et à d'autres caractéristiques d'un utilisateur individuel. Bien que ce domaine de recherche soit nouveau, un certain nombre de techniques intéressantes ont été déjà suggérées et mises en application. Ces techniques peuvent être classifiées dans plusieurs groupes selon la manière retenue pour adapter la présentation des liens. Nous considérons ces groupes de techniques comme différentes technologies pour adapter la présentation de lien. Les technologies les plus connues sont les suivantes :

a) Guidages directs

Les guidages directs sont la technologie la plus simple du support adaptatif de navigation. Des conseils directs peuvent être appliqués dans n'importe quel système. Ils peuvent décider quel est le prochain «*meilleur*» nœud pour l'utilisateur selon son but de visite et les autres paramètres représentés dans le modèle d'utilisateur. Pour fournir des conseils directs, le système peut décrire visuellement le lien au «*meilleur*» nœud comme cela est fait dans «*WebWatcher*» [RFJM95] ou présenter un lien dynamique additionnel (habituellement appelé «*next ou suivant*») qui est relié au «*meilleur*» nœud comme dans «*ISIS-Tutor*» [BP94], «*HyperTutor*» [PGL95], «*Land User Tutor*» [KW94] ou bien la partie de générateur du parcours d'apprentissage dans notre ancien système ECSAI [Gav91]. Un problème des guidages directs est qu'ils ne fournissent aucun support pour les utilisateurs qui ne voudraient pas suivre les suggestions du système. Les guidages directs sont utiles mais ils devraient être employés avec une autre technologie «*plus adaptative*».

b) Ordonnement

L'idée de la technologie d'ordonnement adaptatif est de trier tous les liens d'une page particulière selon le modèle d'utilisateur et quelques critères d'utilisateur : plus près du but, le lien est plus approprié, par exemple. L'ordonnement adaptatif a une applicabilité limitée : elle peut être employée avec les liens non contextuels, mais elle peut à peine être employée pour les index et les pages de table des matières (qui ont habituellement un ordre stable des liens), et ne peut jamais être employée avec les liens et les cartes contextuelles. Un autre problème avec la commande adaptative est que cette technologie rend l'ordre des liens non-stable : il peut changer chaque fois que l'utilisateur entre dans la page. La recherche récente prouve que l'ordre stable des options dans les menus est important pour des débutants [Kap93]. Cependant, cette technologie est utile pour des applications de recherche information [RFJM95][KFC93]. La recherche expérimentale [KFC93] a prouvé que la commande adaptative peut de manière

significative réduire le temps de navigation dans des applications où chaque page peut avoir beaucoup de liens non-contextuels. Un domaine d'application relatif où la commande adaptative peut être employée est celui des systèmes de documentation en ligne [HBG96].

c) Technique Cachée

Cette technique est actuellement la plus généralement utilisée pour le support adaptatif de navigation. L'idée est la limitation de l'espace de navigation donc le masquage des liens aux pages non pertinentes. Une page peut être considérée comme non appropriée pour plusieurs raisons : par exemple, elle n'est pas liée au but courant de l'utilisateur [BP94][HKW⁺96][Vas96] ou elle présente des matériaux que l'utilisateur n'est pas encore prêt à comprendre [BP94][GH95][PGL95]. Cette technique protège les utilisateurs contre la complexité de l'hyperespace sans restriction et réduit leur surcharge cognitive. Elle a une applicabilité large. Elle peut être employée avec toutes sortes d'index non-contextuels, et liens de cadre en cachant des boutons ou des articles de menu [BP94] et avec les liens contextuels en remplaçant « *des mots-clés* » clicquables sur le texte normal [GH95][PGL95]. Elle est également plus transparente à l'utilisateur et plus stable que l'ordonnement adaptatif (des liens sont habituellement ajoutés incrémentalement, mais pas enlevés ou commandés à nouveau). Il y a, cependant, un autre problème, remarqué par quelques psychologues : la technique cachée peut provoquer la formation de modèles mentaux incorrects de l'hyperespace.

d) Annotation

L'idée de la technologie adaptative d'annotation est d'augmenter les liens avec une certaine forme de commentaires qui peuvent indiquer à l'utilisateur l'état actuel des nœuds derrière les liens annotés. Ces annotations peuvent être fournies sous forme textuelle [ZOF93] ou sous forme de sélections visuelles en utilisant, par exemple, différentes icônes [BSW96a], couleurs [BP94], tailles de police [HBG96] ou famille de police [BSW96a]. L'annotation de lien est connue pour être une manière efficace de support de navigation dans l'hypermédia. Le type d'annotation considéré dans l'hypermédia traditionnel est une annotation statique (indépendante de l'utilisateur). Le support adaptatif de navigation peut être fourni par le modèle d'utilisateur dynamique. L'annotation adaptative sous la forme simple basée sur son histoire (décrivant les liens aux nœuds précédemment visités) a été appliquée dans quelques systèmes d'hypermédia utilisant dans plusieurs navigateurs Web. Même si cette forme peut distinguer seulement deux états des liens (liens aux nœuds visités ou non visités), elle semble être tout à fait utile. Les systèmes adaptatifs d'hypermédia, indiqués dans [BP94] et [BSW96a], peuvent distinguer et annoter différemment jusqu'à six états basés sur le modèle d'utilisateur.

L'annotation semble être une bonne méthode de support adaptatif de navigation. L'annotation peut être naturellement employée avec toutes les formes possibles de liens. Cette technique maintient un ordre stable des liens et évite des problèmes avec les cartes mentales incorrectes. L'annotation est généralement une technologie plus puissante que la technique cachée. La technique cachée peut distinguer seulement deux états pour les nœuds relatifs appropriés et non appropriés tandis que l'annotation peut distinguer plusieurs états, par exemple, Hypadapter [HBG96] emploie six états d'annotations pour montrer des niveaux d'importance de chaque lien. Les annotations ne limitent pas la surcharge cognitive autant que la technique cachée, mais la technologie cachée peut être tout à fait bien simulée par la technologie d'annotation en utilisant une sorte « *d'obscurcir* » au lieu de cacher pour les liens « *non appropriés* ». L'obscurcissement peut diminuer la surcharge cognitive dans quelques connaissances (l'utilisateur peut apprendre

à ignorer les liens obscurcis). Les liens obscurcis sont encore évidents cependant (et traversable, s'il y a lieu) ce qui empêche l'utilisateur de former des cartes mentales incorrectes.

e) Conclusion

Les guidages directs, les ordonnancements, la technique cachée et l'annotation sont les technologies primaires pour le support adaptatif de navigation. Car nous verrons dans les sections suivantes, la plupart des applications existantes utilisent une des techniques d'adaptation pour fournir le support adaptatif de navigation. Cependant, ces technologies ne sont pas mutuellement exclusives et peuvent être combinées. Par exemple, les utilisations d'ISIS Tutor [BP94] dirigent des guidages, la technique cachée et annotation. Hypadapter [HBG96] emploie les ordonnancements, la technique cachée et l'annotation. En particulier, la technologie guidage direct peut être naturellement employée en combinaison avec n'importe laquelle de ces trois autres technologies.

3.3 Application des adaptations d'hypermédia

3.3.1 Techniques Adaptatives de Présentation

Adapter la présentation d'information dans une page est effectué souvent comme manipulation de fragments des textes. Le but de ces manipulations est de fournir les explications préalables, additionnelles ou comparatives aux utilisateurs. Ces explications additionnelles peuvent être manipulées avec des états de connaissances : la connaissance préalable absente, des détails additionnels, ou une comparaison un concept précédemment connu. Il existe plusieurs techniques pour le faire : textes conditionnels, stretchtext, framed-based, hybride.

a) Technique des textes conditionnels

Une technique simple, mais efficace pour l'adaptation de contenus est la technique «des textes conditionnels» qui est employée dans ITEM/IP [Bru92] et C-book [KK94]. Avec cette technique, toutes les informations sur un concept sont divisées en plusieurs morceaux de textes. Chaque morceau est associé à une condition au niveau de la connaissance d'utilisateur représentée dans le modèle d'utilisateur. Quand la présentation des informations sur le concept est activée, le système présente seulement les morceaux où la condition est vraie. Cette technique est une technique de bas niveau (on a besoin de la programmation de l'auteur pour mettre toutes les conditions exigées) mais elle est également très flexible. En choisissant des conditions appropriées au niveau de la connaissance du concept courant et des concepts relatifs représentés dans le modèle d'utilisateur, l'auteur peut appliquer différentes méthodes d'adaptation. Un exemple simple est de cacher les morceaux qui contiennent des explications non pertinentes si la connaissance de l'utilisateur du concept courant est assez bonne, ou la rotation sur un morceau avec des explications comparatives si le concept relié correspondant est déjà connu. Cette technique peut être facilement appliquée dans le contexte de WWW. Le travail de Kay et de Kummerfeld [KK94] présente plusieurs idées qui peuvent être employées pour mettre en application cette technique dans un système basé sur le WWW.

b) Technique «Stretchtext»

Une technique de niveau plus élevé qui peut également se cacher et s'afficher sur différentes parties du contenu selon le niveau de la connaissance de l'utilisateur est suggérée dans MetaDoc [BE94], développée plus loin dans KNAHS [KMN94] et ADAPTS [BC99b]. Cette technique est

basée sur le *stretchtext* qui est une sorte spéciale d'hypertexte. Un hypertexte régulier, un résultat de clic sur un mot-clé se déplace à une autre page avec le texte relatif. Dans le *stretchtext*, ce texte relatif peut simplement remplacer le mot-clé activé (ou une expression comprenant le mot-clé) en prolongeant le texte dans la page courante. Ce texte prolongé (le terme utilisé dans MetaDoc, «*uncollapsed* ou *extended text*») peut être réduit de nouveau à un mot-clé, si l'utilisateur le désire.

Dans MetaDoc, pour chaque fragment de l'information il y a des mots-clés. Le système détermine quels fragments devraient «être affichés» et quels fragments devraient «être cachés» (seulement le mot-clé est affiché). Cette décision détermine seulement la présentation initiale du fragment. L'utilisateur peut afficher ou cacher des fragments par des clics de souris. Le système emploie ces actions d'utilisateur pour prévoir mieux quels fragments à afficher ou cacher dans des demandes de la page suivante.

De plus, pour réaliser cet événement, l'auteur peut déclarer l'information textuelle cachée contenue dans un nœud : une explication additionnelle d'un concept et un détail de bas niveau d'un concept particulier. En option, l'utilisateur de MetaDoc qui a une bonne connaissance d'un concept ne voit pas les explications additionnelles de ce concept, mais voit les détails de bas niveaux. Au contraire, l'utilisateur qui a une faible connaissance du concept voit toujours des explications additionnelles mais pas les détails. L'utilisateur qui a une connaissance moyenne verra les deux sortes d'information, comme résumé dans le tableau 3.1.

Niveau de connaissances du concept.	Explications additionnelles	Détails de bas niveaux
faible	visibles	cachés
moyen	visibles	visibles
élevé	cachés	visibles

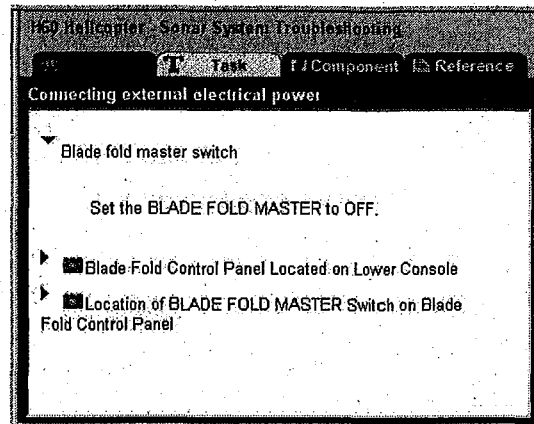
TAB. 3.1 – Présentation adaptative utilisée dans MetaDoc

Un dispositif important de la technique adaptative de *stretchtext* est qu'il laisse l'utilisateur et le système adapter le contenu d'une page particulière, tenant compte de la connaissance et des préférences de l'utilisateur. Après sa présentation initiale, la page de *stretchtext* peut être encore adaptée par l'utilisateur qui peut configurer à «afficher» ou «cacher» des explications et des détails de bas niveaux appropriés selon ses préférences. Le système met à jour le modèle d'utilisateur selon les préférences démontrées par l'utilisateur pour s'assurer que l'utilisateur verra toujours la combinaison préférée des informations cachées. Par exemple, si l'utilisateur met des explications additionnelles d'un concept particulier comme visible, le système montrera toujours les explications additionnelles jusqu'à ce que l'utilisateur change les préférences.

Un autre exemple de *stretchtext*, l'utilisation du *stretchtext* dans ADAPTS [BC99b] qui utilise des paragraphes procéduraux pour indiquer ou cacher des détails représentée dans la figure 3.1. Ce *stretchtext* est semblable à son utilisation dans des systèmes tels que MetaDoc. Un paragraphe particulier pourrait être affiché dans la fenêtre de la présentation si l'utilisateur est au courant de l'information présentée dans ce paragraphe ou si cette information n'est pas très appropriée au contexte courant.

c) Technique «frame-based»

La technique adaptative de présentation la plus puissante est la technique «*frame-based*» mise en application dans Hypadapter [HBG96] et EPIAIM [dRdCP94]. Avec cette technique toutes les informations sur un concept particulier sont représentées sous forme de cadre de fenêtre. Les

FIG. 3.1 – *Stretchtext adaptatif: fragment caché et affiché*

cadres peuvent contenir plusieurs explications différentes du concept, des liens à d'autres cadres, des exemples, etc. Des règles spéciales de présentation sont employées pour décider quels cadres devraient être présentés à un utilisateur particulier et dans quel ordre. Plus exactement, dans EPIAIM ces règles sont employées pour choisir un des schémas existants de présentation (chaque schéma est un sous-ensemble commandé de cadres) et le schéma est employé pour présenter le concept.

The Echidna

The Echidna, also known as the spiny Anteater, is a type of Monotreme that is covered in stiff, sharp spines mixed with long, coarse hairs.

The Echidna has the following subtypes:

- the short-beaked Echidna and
- the long-beaked Echidna.

The Echidna is about the same length as a domestic cat. It ranges from 2 kg to 7 kg in weight. It has a brownish black coat and paler-coloured spines. It has a small head. It has a prolonged, slender snout. It has no teeth. It uses its extensible, sticky tongue for catching ants, termites and other small insects. It is a carnivore and eats ants, termites and earthworms. It has powerful claws allowing for rapid digging of hard ground. It is found in Australia. It is active at dawn and dusk. It lives by itself. It has an average lifespan in captivity of 50 years.

This text is generated for the novice user level. If you would like the text for the expert user level click [here](#).




FIG. 3.2 – *Description de l'hérissonne est produite par PEBA-II*

Dans Hypadapter, les règles sont employées pour calculer la «*priorité de présentation*» pour chaque cadre. Alors un sous-ensemble de cadres avec la priorité élevée sont présentés par ordre de priorité décroissante. Dans leurs parties conditionnelles, ces règles peuvent se référer non seulement à la connaissance d'utilisateur d'un concept présenté, mais également à n'importe quel dispositif représenté dans le modèle d'utilisateur. En particulier, les deux systèmes qui emploient

cette technique tiennent compte du modèle de l'utilisateur.

Un bon exemple de l'implémentation de la technique frame-based sur WWW est le système PEBAIL [MO98] (figure 3.2). Ce système emploie des techniques frame-based de représentation de la connaissance et de génération de langage naturel pour produire des informations descriptives et comparatives sur différents animaux. Cette information est adaptée au niveau de l'utilisateur d'expertise.

d) Technique hybride

Une technique s'est développée récemment dans le projet de PUSH [HKW⁺96] et peut être considérée comme une combinaison «*de stretchtext*» et «*d'adaptation frame-based*». Une page d'hypermédia dans ce système fournit une description complète d'un objet particulier structuré comme une séquence ordonnée des entités dactylographiées de l'information. Chaque type d'objet dans le PUSH a ses propres assortiments des types d'entité de l'information qui sont employés pour décrire un objet de ce type.

Cette technique est semblable au modèle frame-based (des objets peuvent être considérés comme fenêtres et les entités de l'information comme cadres). La différence est que chaque entité de l'information est une partie d'hypertexte raisonnablement grande. La description complète d'un objet est habituellement elle-même grande, et prend plusieurs pages d'information. Pour protéger les utilisateurs contre le débordement de l'information et pour les aider à trouver l'information requise, le système utilise la technique cachée: il présente seulement les types d'informations sur l'objet courant qui sont appropriés à l'objectif courant de l'utilisateur (le but peut être fixé par l'utilisateur ou être déduit par le système). En même temps, pour maintenir l'adaptation transparente, le système maintient l'ordre stable de présentation de l'information et ne cache jamais complètement les entités non-appropriées: les titres des entités non-appropriées cachées sont toujours affichés. Si l'utilisateur n'est pas satisfait de la décision du système pour montrer ou cacher une entité particulière, le contenu de l'entité de l'information peut être rétréci ou élargi en cliquant sur une icône près de son titre. L'interface résultante ressemble à l'interface de stretchtext de MetaDoc: des morceaux non-appropriés ne sont pas présentés à l'utilisateur, montrant seulement le mot-clé (dans MetaDoc) ou le titre (dans le PUSH), mais l'utilisateur peut dépasser l'adaptation en ouvrant et en fermant n'importe quelle information désirée.

La figure 3.3 montre un exemple du système PUSH de mise en application du stretchtext comme technique adaptative de présentation dans le contexte de WWW. D'une manière générale, une technique stretchtext n'est pas la plus facile à mettre en application sur WWW. Pourtant, les développements actuels dans le domaine du WWW, telle que la technologie du Java 2 ouvrent la voie aux techniques adaptatives stretchtext de présentation sur WWW [EH96].

3.3.2 Techniques Adaptatives de Support de Navigation

a) Technique ordonnée

Une des techniques la plus fréquemment utilisée pour assortir des liens adaptatifs est la technique ordonnée. Elle a été mise en application dans HYPERFLEX [KFC93], un système d'information en ligne. HYPERFLEX fournit à l'utilisateur le support de navigation en montrant une liste ordonnée de nœuds liés au nœud courant. Les liens sont triés selon leur pertinence par rapport au nœud courant (le plus approprié d'abord). En outre, HYPERFLEX maintient une liste de buts possibles de recherche. Si l'utilisateur choisit un de ces buts de recherche, l'ordonnancement tient compte de la pertinence des liens montrés avec le but choisi. Les nouveaux buts peuvent être créés par les utilisateurs eux-mêmes. Le composant principal du modèle

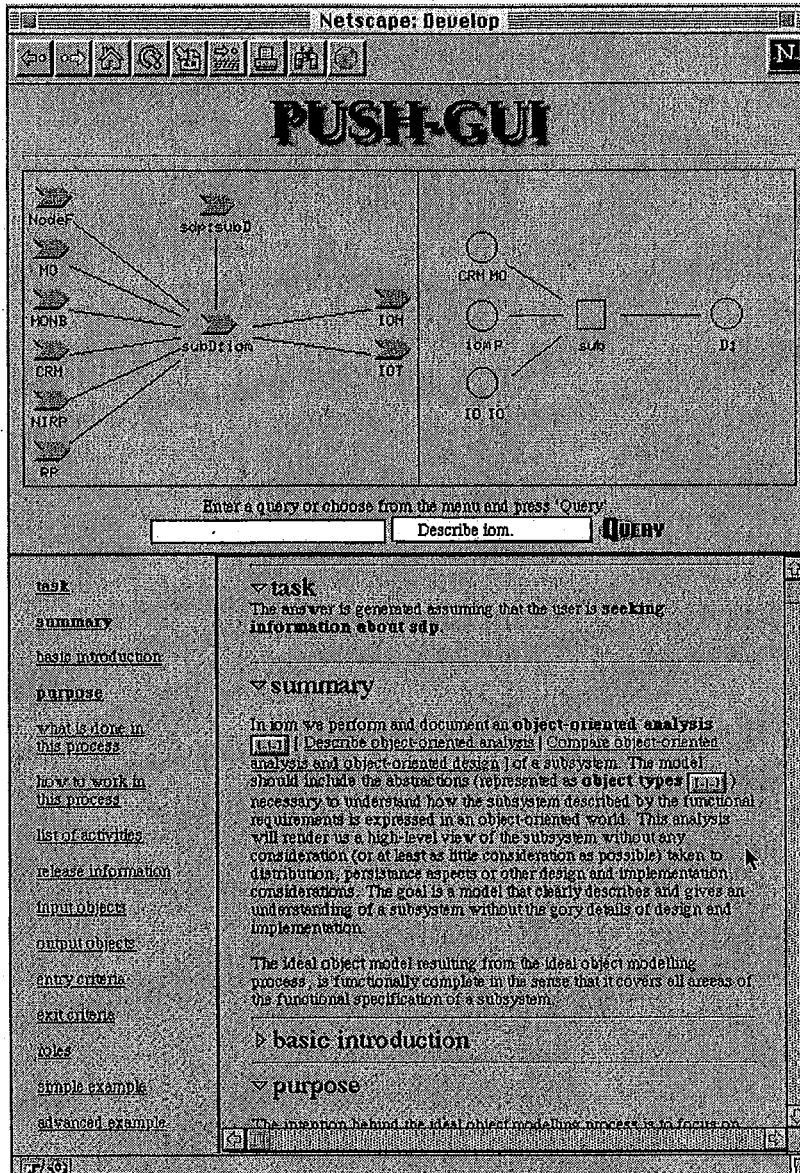


FIG. 3.3 – Copie d'écran du système PUSH.

d'utilisateur dans HYPERFLEX est une matrice de pertinence qui stocke des valeurs de pertinence entre chaque paire de documents et de chaque but à chaque document. Cette matrice reflète principalement des préférences d'utilisateur pour l'ordre des liens. Dans HYPERFLEX, l'utilisateur peut déplacer les liens pour dire au système directement ses préférences sur l'ordre relatif des liens (c.-à-d., quels liens appropriés et dans quel ordre il voudrait voir en regardant un document particulier ou en approchant un but particulier). Ces préférences sont traitées par le système pour mettre à jour le modèle d'utilisateur. Par conséquent les préférences choisies dans un contexte peuvent influencer l'adaptation dans un autre contexte.

b) Technique cachée

Un exemple de la technique cachée de support de navigation est la technique basée sur les règles mise en application dans les systèmes HyperTutor [PGL95] et SYPROS [GH95]. Les deux systèmes emploient des ensembles spéciaux de règles pédagogiques pour décider quels concepts et nœuds devraient être évidents à un moment donné et lesquels ne devraient pas. Ces règles tiennent compte du type du concept et de ses liens à d'autres concepts et à l'état actuel de la connaissance d'utilisateur reflétée dans le modèle d'utilisateur. Si le système décide de ne pas rendre un nœud visible, alors tous les liens contextuels à ce nœud sont changés de mots-clés en texte normal. La technique cachée basée sur les règles est très flexible. En manœuvrant le texte caché avec des règles, il est possible d'appliquer plusieurs méthodes de support de navigation basées sur cette technique, tel que cacher des nœuds que l'utilisateur n'est pas encore prêt à apprendre, ou cacher des nœuds qui appartiennent aux futures leçons.

La technique cachée utilisée dans Hynecosum [Vas96] est différente. Ce système soutient des hiérarchies des tâches pour des utilisateurs de différentes catégories. Chaque nœud de l'hyperespace dans le système est classé par les tâches (terminales) élémentaires qui exigent l'accès à ce nœud. Ainsi, parce que, chaque tâche terminale, le système peut compiler la liste associée aux nœuds concernant cette tâche. Par définition, une liste de nœuds appropriés pour des tâches de niveau plus élevé inclut tous les nœuds appropriés au moins à un de ses tâches secondaires. Dans Hynecosum l'utilisateur peut indiquer directement au système quelle tâche lui ou elle accomplit actuellement en choisissant une des tâches à partir d'une hiérarchie personnelle. Après ceci, le système montre à l'utilisateur seulement les nœuds qui sont appropriés à la tâche courante. Il rend l'ensemble des nœuds évidents maniables.

c) Technique d'annotation

Un exemple de technique adaptative du support de navigation basée sur l'annotation peut être fourni par les systèmes ISISTutor et ITEM/PG. L'organisation de l'hyperespace dans ces systèmes est constituée par un réseau conceptuel qui représente la structure pédagogique du sujet enseigné. Chaque concept du réseau conceptuel est représenté par un nœud de l'hyperespace. Des nœuds de concept du réseau sont reliés par différents types de relations formés comme «*is-a*», «*part-of*» et «*prerequisite*». Une autre sorte de nœuds dans l'hyperespace est des unités d'apprentissage (telles que des unités de présentation, des problèmes, et des exemples). Chaque unité d'apprentissage est reliée à tous les nœuds de concept qui sont exigés pour fonctionner avec cette unité. Le modèle d'élève par recouvrement multi-couches représente séparément le niveau de la connaissance de l'élève de chacun des concepts. En utilisant ce modèle et des liens prérequis, les systèmes peuvent distinguer quatre états éducatifs pour chaque nœud représenté par une page d'hypermédia : «*pas-prêt-à-être-appris*» (c.-à-d., des prérequis ne sont pas suffisants), «*prêt-à-être-appris*», «*être-en-train-de-travailler*» (commencer à apprendre) et «*appris*» (l'élève a

résolu un certain nombre de problèmes pour le concept). L'idée est que les concepts ayant des différents états éducatifs ont différentes significations pour l'étudiant et en rendant les états éducatifs visibles, on aidera l'étudiant dans la navigation d'hyperespace. Pour rendre les états éducatifs visibles, des liens aux concepts avec différents états éducatifs sont annotés différemment en utilisant différentes couleurs et quelques caractères spéciaux.

Le même cadre de travail a été employé dans ISIS-Tutor pour implémenter une technique adaptative de support de navigation basée sur le but d'apprentissage. Dans ce système, le professeur peut régler pour chaque étudiant un ordre des buts d'apprentissage. Un but est un ensemble de nœuds de concept du réseau qui doit être appris. Des concepts qui appartiennent au même but doivent être appris ensemble et alors maîtrisés en résolvant un certain nombre de problèmes avant qu'un étudiant se déplace au prochain but. ISIS-Tutor emploie deux méthodes pour s'adapter au but d'apprentissage: premièrement, pour attirer l'attention de l'étudiant, il peut décrire des liens aux concepts appartenant au but courant, deuxièmement (en option), pour diminuer la charge cognitive de l'étudiant, il peut cacher les concepts qui appartiennent aux futurs buts d'apprentissage.

L'annotation est une technique très appropriée pour implémenter le support adaptatif de navigation sur WWW. Un exemple d'annotation adaptative sur WWW est fourni par le système ELMART [BSW96a] et par le système d'auteur InterBook [BSW96b]. Ces systèmes emploient la même technique qu'ISIS-Tutor pour l'annotation de liens basée sur la connaissance et sur le but d'apprentissage. Les différences sont les sélections visuelles qui sont employées par ELM-ART et InterBook: plutôt que changer la couleur des mots clé elles-mêmes (comme a été fait dans ISIS-Tutor et ITEM/PG), ces systèmes changent la couleur des icônes et de police de caractères (pour des utilisateurs sans contrôles de couleur). En particulier, pour les liens aux nœuds non-visités, ELM-ART emploie la métaphore de feu de circulation (figure 3.4): des liens aux nœuds non-prêt-à-être-appris sont annotés par une icône de boule rouge et une police italique, des liens aux nœuds prêt-à-être-appris appartenant au but courant sont annotés par une icône de boule verte et de caractères gras et des liens aux nœuds prêt-à-être-appris n'appartenant pas au but courant sont annotés par une icône de boule jaune et une police normale. ELM-ART et InterBook mettent en application l'annotation adaptative des liens en utilisant CGI, CLHTTP [Mal94] et génération de «on-the-fly» page.

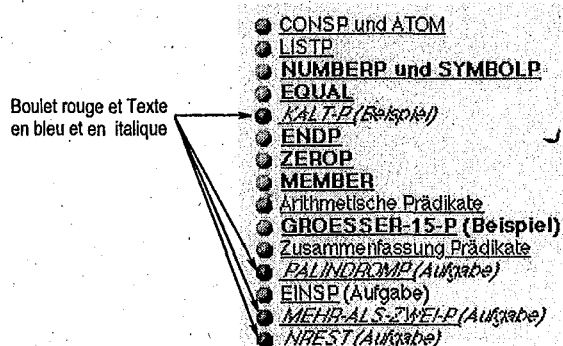


FIG. 3.4 – Exemple d'annotation adaptative des liens exploitant l'interface WWW dans ELM-ART.

d) Conclusion

Actuellement, toutes les technologies adaptatives principales de support de navigation sont mises en application sur WWW. ELM-ART et InterBook fournissent un exemple de support adaptatif de navigation basé sur l'annotation et sur l'ordonnement. WebWatcher [RFJM95] implémente la technologie de guidage directe dans le contexte de recherche documentaire (Web-Watcher n'est pas complètement adaptatif encore : l'adaptation est actuellement basée sur un modèle d'utilisateur de groupe plutôt que sur le modèle individuel d'utilisateur). Quelques idées comment mettre en application le support adaptatif de navigation basé sur la technique cachée dans le contexte de WWW peuvent être également trouvées dans [LCY95]. Ce système a toute la connaissance exigée pour cacher des liens aux pages non-prêt-à-être-appriées, mais plutôt pour cacher de tels liens, le système modifie leur comportement. Si une page derrière un lien est non-prêt-à-être-apprié, c.-à-d., il y a des pages pré-requises que l'utilisateur n'a pas encore visitées, le lien ne conduit pas l'utilisateur à la page demandée, mais à la liste de pages qui doivent être appriées avant d'étudier la page demandée.

3.4 Conclusion

A partir des techniques adaptatives existantes, nous choisissons les techniques les plus générales qui sont utilisées dans plusieurs systèmes pour présenter et naviguer dans l'univers d'apprentissage. Les techniques adaptatives de présentation que nous voulons employer sont les techniques hybrides : la combinaison entre la technique «*stretchtext*» et «*ordonnement*». Ces techniques sont utilisées pour les unités physiques d'apprentissage. L'ensemble des contenus dans l'unité physique est présenté aux apprenants selon leurs compétences ou leurs besoins (aider à résoudre le problème) et la stratégie pédagogique. Les techniques adaptatives de navigation sont aussi les techniques hybrides. Ceci dépend du mode d'apprentissage choisi par les utilisateurs.

Supposons que des enseignants veulent présenter un exemple de résolution de problème. Les enseignants doivent d'abord présenter un énoncé puis présenter comment résoudre ce problème étape par étape. Il est vrai que l'enseignement dans la classe de cette façon est trop simple, les enseignants peuvent mettre tous les contenus dans un seul transparent, puis cacher la partie qu'ils ne veulent pas présenter à l'apprenant. Mais pour mettre en évidence cette technique sur le Web, c'est plus difficile car on ne peut pas créer directement ce type de présentation en format HTML et la mettre dans un seul fichier. L'enseignant doit créer plusieurs fichiers HTML représentant chaque étape de présentation et incluant les liens dans chaque fichier. Ces travaux diminuent le courage des enseignants parce qu'il faut avoir le temps pour préparer et mettre à jour. Nous décidons alors de proposer des modèles bien structurés et des outils pour simplifier les travaux des enseignants. Ces modèles doivent supporter la présentation adaptative d'une part et la navigation adaptative d'autre part.

Nous détaillerons dans les chapitres «*Modélisation des ressources pédagogiques*» (cf. chapitre 4) et «*Modèle des élèves*» (cf. chapitre 5) les techniques adaptatives que nous choisirons et les exploiterons pour présenter les contenus dans chaque nature d'une unité physique et pour calculer le parcours dans l'univers d'apprentissage de chaque apprenant.

Conclusion vers un «*cahier des charges*» pour ECSAIWEB

L'étude des environnements d'apprentissage existants, présentée dans l'évolution du système ECSAI vers le Web (chapitre 2) et les techniques adaptatives d'hypermédia (chapitre 3) nous permettent de disposer d'une base de réflexion pour l'élaboration d'un premier cahier des charges pour le développement d'un prototype d'un environnement intégré (cf. section 1.4), c'est-à-dire une première phase d'expression des besoins déjà décrits dans le chapitre 1. En particulier, nous pouvons mettre en évidence le besoin d'un certain nombre de catégories de fonctionnalités dans chaque environnement.

La première concerne les fonctionnalités dans l'environnement auteur. Ce sont les fonctionnalités de «*construction ou description*» des matériaux pédagogiques. En conservant les idées initiales de l'ECSAI, nous pouvons conserver un certain nombre des modèles représentant les matériaux pédagogiques: les items d'évaluation et des unités d'apprentissage (appelée ULA: unité logique d'apprentissage dans ECSAIWEB). La différence entre ECSAI et ECSAIWEB est que ECSAIWEB contient des outils pour préparer les cours sur l'Internet. Cet environnement permet aux enseignants-auteurs de créer ou d'importer des documents disponibles et de fournir un ensemble d'unités d'apprentissage qui sont des morceaux séparés de logiciel. Cet ensemble est la base de données d'unités physiques [SG00a] qui seront invoquées à l'exécution. Nous voulons modifier les deux modèles de base, items d'évaluation et les unités logiques d'apprentissage, et ajouter le nouveau modèle, les unités physiques d'apprentissage, pour qu'ils soient adaptables et réutilisables au Web.

L'enseignant-auteur crée le modèle du domaine; le domaine d'étude doit être décrit par un réseau de sujets élémentaires. Nous voulons proposer ensuite des outils pour vérifier des cohérences entre les trois modèles. Il n'est cependant pas assuré que les apprenants pourront atteindre tous les objectifs pédagogiques parce que les enseignants-auteurs ne définissent pas directement le parcours d'apprentissage. Alors nous devons offrir un autre outil pour simuler un parcours d'apprentissage, c'est-à-dire, les enseignants-auteurs peuvent tester l'enchaînement des unités logiques d'apprentissage en situation idéale. Mais les enseignants-auteurs peuvent aussi tester le parcours d'apprentissage en situation réelle en jouant le rôle des apprenants et ils perdent alors des pouvoirs d'auteur.

La liste suivante est un ensemble des outils proposés dans l'environnement auteur et les fonctionnalités de chaque outil.

- Outil de Création d'un univers d'apprentissage qui permet de définir et créer les ressources pédagogiques. Les fonctionnalités que nous voulons proposer sont les suivantes,
 - définition d'un item d'évaluation: ajouter, lister; supprimer et trier,
 - détermination des objectifs pédagogiques: saisir les valeurs de compétences (mesure

- globale, degré d'acquisition : savoir–appliquer et savoir–faire) et lister,
- description d'une unité logique d'apprentissage : ajouter, charger ou modifier, lister, supprimer et trier,
- addition d'une unité physique d'apprentissage (le contenu d'une unité logique d'apprentissage) : ajouter, éditer ou modifier.
- Outil de Contrôle et Simulation du parcours d'apprentissage,
 - vérification des cohérences d'un univers d'apprentissage :
 - ◊ pertinence des items d'évaluation,
 - ◊ validité des objectifs pédagogiques,
 - ◊ accessibilité des unités logiques d'apprentissage.
 - simulation d'un parcours d'apprentissage : simuler, afficher et effacer un parcours d'apprentissage,
- Outil de Visualisation de l'univers d'apprentissage : visualiser un graphe statique présentant des liens entre unité logique choisie et des unités logiques dans univers d'apprentissage et des liens entre toutes les unités logiques dans l'univers d'apprentissage,
- Outil de Création de l'examen : créer un pré– ou post–examen d'un sujet choisi,
- Outil de Gestion des fichiers : mettre à jours des fichiers à partir du fichier local au serveur, modifier les fichiers, convertir les fichiers HTML au format PRESENTATION de notre système,
- Outil de Gestion de profil d'élève : visualiser et effacer les contenus dans le profil d'élève.

Une deuxième catégorie regroupe les fonctionnalités dans l'environnement élève. Ce sont les fonctionnalités de «*visualisation et exécution*» des matériaux pédagogiques permettant d'aider l'apprenant à visualiser et naviguer dans les cours et à évaluer ses connaissances. Dans cet environnement, nous voulons conserver les modes d'apprentissage de ECSAI : libre et guidé. Nous appliquons ici les techniques adaptatives : la présentation adaptative et la navigation adaptative. La présentation adaptative est appliquée à la présentation des contenus d'une unité physique, celle ci donne un point de vue différent de la même unité logique pour chaque apprenant. La navigation adaptative est appliquée aux modes d'apprentissage pour aider l'apprenant à naviguer dans l'univers d'apprentissage. En utilisant la technique adaptative d'annotation des concepts, l'apprenant peut savoir où il en est dans l'univers d'apprentissage et quel est le propos de la suite pour atteindre son but. L'apprenant peut également, selon son besoin, passer de l'un à l'autre mode.

Les fonctionnalités proposées doivent permettre librement aux apprenants de :

- choisir une des matières proposées à apprendre,
- choisir le mode d'apprentissage : libre ou guidé,
- tester des connaissances avant et après l'étude,
- visualiser les résultats.

Une troisième catégorie concerne les fonctionnalités dans l'environnement partagé. Elle regroupe deux parties des fonctionnalités : contrôle d'accès des utilisateurs et des outils de communication. Tous les utilisateurs doivent passer dans cet environnement pour entrer dans autre environnement.

Les fonctionnalités de contrôle d'accès des utilisateurs, ayant des droits d'accès nécessaires, doivent permettre aux utilisateurs de :

- s'inscrire dans le système,

- se connecter au système,
- modifier les informations personnelles,
- modifier leur profil,
- tester des connaissances (élèves) ou créer des examens globaux (enseignants-auteurs).

Les outils de communication sont un espace pour échanger les informations entre les utilisateurs : les enseignants, les apprenants, les enseignants-auteurs et les apprenants. Ce sont des outils de «*communication*» synchrone et asynchrone :

- forum de discussion,
- courrier électronique,
- chatroom.

Pour le système administratif, il offre le gestionnaire du système qui permet de :

- stocker l'historique des connexions,
- générer et supprimer les comptes des utilisateurs,
- générer et modifier les messages d'interface,
- générer et modifier les matières.

Deuxième partie

Conception et Réalisation
d'ECSAIWEB



Chapitre 4

Modélisation des ressources pédagogiques

4.1 Introduction

L'objectif de ce chapitre est de représenter des modèles permettant de décrire les ressources pédagogiques : l'ensemble des unités et l'ensemble des concepts du domaine qui seront transmis à l'apprenant. Les concepts du domaine constituent généralement un graphe hiérarchique, complet et consistant relativement à la sémantique du domaine. Ils doivent permettre une utilisation efficace des connaissances par un générateur du parcours d'apprentissage. Le réseau pédagogique qui doit être exploré par l'apprenant ne comprend que les concepts ayant un intérêt pédagogique. Par conséquent le réseau pédagogique n'est pas une copie conforme du graphe de domaine. Les liens entre les nœuds du graphe de domaine sont calculés selon l'intérêt pédagogique.

Nous décrivons dans la section 4.2 les items d'évaluation. Ils sont des éléments pour exprimer les concepts à enseigner d'une part et pour décrire les connaissances de l'apprenant et les unités d'apprentissage d'autre part. Les unités logiques d'apprentissage, utilisées pour calculer le parcours d'apprentissage, font l'objet de la section 4.3. Enfin, pour présenter les contenus associés à chaque unité logique d'apprentissage, nous proposons alors des unités physiques d'apprentissage, présentées dans la section 4.4. Ces unités sont structurées selon la nature des unités logiques d'apprentissage et la stratégie pédagogique. Dans chaque section, nous indiquons explicitement ce que nous avons conservé du modèle initial et ce que nous avons ajouté, retranché ou modifié.

4.2 Items d'évaluation

4.2.1 Rôle des items d'évaluation

La notion d'item d'évaluation a été décrite dans le modèle ECSAI [Gav91]. Il représente la plus petite unité de connaissance sur laquelle l'enseignant-auteur souhaite porter un jugement. Les items dépendent autant du domaine enseigné que de l'enseignant qui les a définis.

Chaque item est défini par un libellé qui résume le concept sous-jacent du domaine. Quand il est utilisé par le système il est toujours accompagné de valeurs qui représentent les aptitudes de l'élève à l'égard de celui-ci.

Les items associés à des valeurs permettent de spécifier les acquis de l'élève ainsi que les objectifs pédagogiques qu'il doit essayer d'atteindre.

Les valeurs associées aux items d'évaluation sont la « *mesure globale* » et le « *degré d'acquisition* ». Par rapport au modèle d'ECSAI nous n'utilisons plus la mesure globale parce que nous utilisons d'autres paramètres dans le modèle d'élève pour exprimer cette notion. Nous avons cependant conservé le modèle initial, dans la perspective d'une extension du système, et nous avons choisi donc de mettre la valeur « *a - acquis* » dans toutes les mesures globales des items. Les comportements d'élève alors sont exprimés par les valeurs de degré d'acquisition qui portent sur « *savoir-faire* » et « *savoir-appliquer* ».

4.2.2 Expression des objectifs pédagogiques

Les objectifs pédagogiques peuvent, en général, se décomposer en 3 parties (figure 4.1) : activité ou comportement de l'élève (souligné sur la figure); concepts du domaine (*italique* sur la figure) et autres caractères (**gras** sur la figure). L'exemple dans la figure 4.1 présente comment nous définissons les objectifs pédagogiques de chaque sujet. Il faut noter que la troisième partie de l'objectif parfois est absente. Dans ce cas, l'apprenant doit répondre correctement à toutes les questions dans les exercices liés à cet objectif à la fin du sujet choisi. Sinon l'enseignant doit indiquer les notes minimales de chaque objectif dans la troisième partie.

Sujet: Le circuit électrique en série

Après étude de ce sujet, les élèves peuvent :

1. Décrire les caractéristiques d'un circuit électrique en série au moins trois caractères.
2. Calculer la résistance équivalente d'un circuit électrique en série.
3. Calculer les chutes des tensions de chaque composant d'un circuit électrique en série.
4. Calculer le courant circulant dans chaque composant d'un circuit électrique en série.

...

FIG. 4.1 – Exemple d'objectifs pédagogiques

Ces objectifs pédagogiques sont exprimables avec la modélisation choisie. La mesure de l'activité (verbe souligné dans la figure 4.1) de l'apprenant va être représentée par les valeurs fournies pour « *savoir-faire* » et « *savoir-appliquer* ». Le concept du domaine sera explicité dans le libellé associé à l'item.

Le schéma 4.2 montrera la transcription d'un objectif pédagogique par une compétence sur un item d'évaluation telle qu'elle sera utilisée dans le système ECSAIWEB.

Les valeurs attribuées pour savoir-faire et savoir-appliquer pourront être plus ou moins élevées (dans la figure 4.2 (2, 4)) selon l'importance pédagogique de l'objectif pédagogique et le nombre d'exercice portant sur ce concept.

Par concept nous entendons les portions élémentaires pour le domaine donné. Leur nombre peut être plus ou moins grand. Selon le domaine et le champ d'application, les concepts peuvent représenter de plus grands ou plus petits morceaux de la connaissance du domaine. Par exemple, en calcul algébrique « *factoriser une expression* » est un item de plus grosse granularité qu'utiliser « *la différence de deux carrés* ».

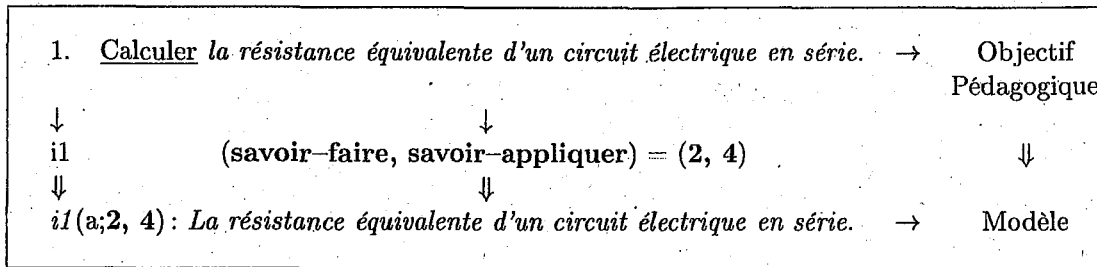


FIG. 4.2 – Transformation d'un objectif pédagogique pour ECSAIWEB

Une forme plus avancée du modèle de domaine est un réseau avec des nœuds correspondant aux concepts de domaine et avec des liens reflétant deux types de relations entre les concepts. Le premier type représente l'organisation hiérarchique du domaine. Le deuxième type représente des contraintes pédagogiques par exemple: *l'item i1 doit être appris avant l'item i2* (la figure 4.3, dans la partie des modèles formels à gauche). Un réseau de ce type représente la structure du domaine à enseigner dans ECSAIWEB (figure 4.3). Ce même modèle de domaine fournit aussi une structure pour la représentation de la connaissance que l'utilisateur a du sujet enseigné. Pour chaque concept du modèle de domaine, le modèle de la connaissance d'un utilisateur individuel enregistre une certaine valeur qui est une évaluation du niveau de la connaissance de l'utilisateur de ce concept. Ce type de modèle qui s'appelle un «*modèle de recouvrement*» est puissant et flexible; il peut mesurer indépendamment la connaissance de l'utilisateur de différents sujets, nous l'utilisons pour la construction des parcours d'apprentissage de l'élève.

Dans le réseau des ULAs (modèles formels à droite), le lien entre U3 et U6 représente une contrainte de succession entre les deux unités. D'autres liens seront calculés entre les ULAs de façon dynamique.

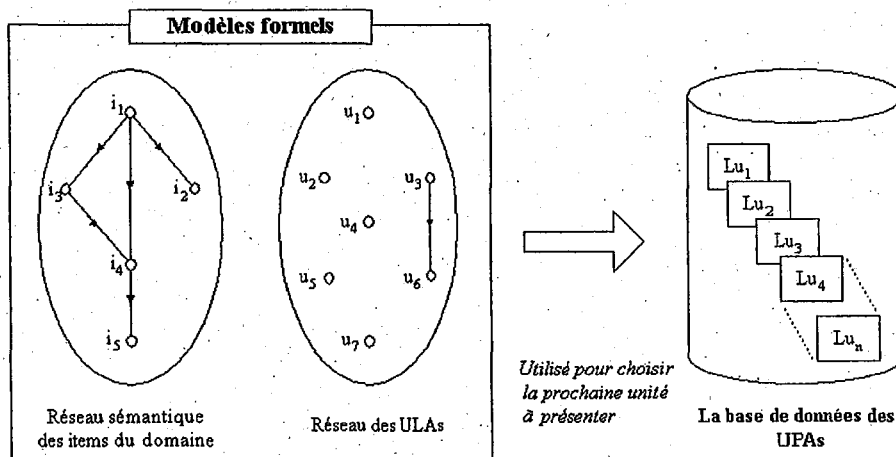


FIG. 4.3 – Modèles formels du système ECSAIWEB

4.2.3 Opérations sur les items

A partir du modèle d'un item d'évaluation représenté dans la figure 4.2, nous proposons maintenant la définition des opérations sur les items, les ensembles des items et le niveau de connaissances des unités logiques d'apprentissage et des apprenants. En effet, ces opérations sont utilisées pour construire le parcours d'apprentissage.

a) Valeur associée à un item

Cette opération nous permet de calculer un nombre entier à partir des valeurs associées à un item donné. Nous pouvons la définir de la manière suivante :

$$\text{Valeur : } Mg \times Sa \times Sf \rightarrow \mathbb{N}$$

Valeur quantitative :

Mg : mesure globale avec $i < a < v < q$
($q = \text{acquis}$, $v = \text{vu}$, $a = \text{abordé}$, $i = \text{inconnu}$)

Valeur qualitative :

Sa : Degré d'acquisition du savoir-appliquer

Sf : Degré d'acquisition du savoir-faire

Cette fonction est de la forme :

$$\text{Valeur}(i_k) = a \times Mg + b \times Sa + c \times Sf \quad \text{où } a, b, c \in \mathbb{N} \quad (4.1)$$

L'équation 4.1 donne une mesure quantitative des connaissances sur un item donné. Les valeurs de a , b et c , actuellement, sont égales à 1. Les valeurs de Mg sont $q = 4$, $v = 3$, $a = 2$, et $i = 1$. On utilise cette valeur pour calculer par exemple la relation d'ordre entre des items (cf. section 4.2.3.d).

b) Ecart entre valeurs d'un item

Cette opération permet pour un item donné de calculer l'écart entre les valeurs présentes dans l'objectif pédagogique et celles présentes dans les connaissances d'un élève à l'instant donné. Nous pouvons définir de manière générale cet écart comme suit :

$$\text{Ecart : item} \times \text{item} \rightarrow \text{item}$$

$$\begin{aligned} \text{Ecart}(i_{k_1}, i_{k_2}) &= i_{k_1}(A, B, C) - i_{k_2}(D, E, F) \\ &= i_k(\text{Min}(A, D), (B - E), (C - F)) \end{aligned} \quad (4.2)$$

Exemple : L'écart entre les valeurs de l'objectif et celles de l'élève peut être définie de la façon suivante :

$$\begin{aligned} \text{Soit } i_{1obj}(a, 2, 2) &: \text{l'objectif pédagogique sur l'item } i_1 \\ i_{1act}(v, 1, 2) &: \text{les valeurs actuelles des connaissances de l'élève sur l'item } i_1 \\ \text{Ecart}(i_{1obj}, i_{1act}) &= i_{1obj}(a, 2, 2) - i_{1act}(v, 1, 2) \\ &= i_1(\text{Min}(a, v), (2 - 1), (2 - 2)) \\ &= i_1(a, 1, 0) \end{aligned}$$

c) Somme des valeurs d'un item

L'objectif de cette opération est d'ajouter à un item d'autres valeurs lui afférant. Nous pouvons définir de manière générale cette somme comme suit :

$$\text{Somme : item} \times \text{item} \rightarrow \text{item}$$

$$\begin{aligned} \text{Somme}(i_{k-1}, i_{k-2}) &= i_{k-1}(A, B, C) + i_{k-2}(D, E, F) \\ &= i_k(\text{Max}(A, D), (B + E), (C + F)) \end{aligned} \quad (4.3)$$

Exemple: Supposons que les connaissances de l'élève X soient: $i_1(a, 1, 1)$. Supposons que l'élève réponde correctement à toutes les questions dans l'exercice, si l'apport de cet exercice est de $i_1(a, 2, 2)$, alors nous pouvons calculer les connaissances actuelles de cet élève de la façon suivante :

$$\begin{aligned} \text{les connaissances actuelles} &= \text{les anciennes connaissances} + \text{l'apport de l'exercice} \\ &= i_1(a, 1, 1) + i_1(a, 2, 2) \\ &= i_1(\text{Max}(a, a), (1 + 2), (1 + 2)) \\ &= i_1(a, 3, 3) \end{aligned}$$

d) Relation de proximité entre valeurs d'item

La relation de proximité, R_{prox} , exprime quel est l'item le plus proche d'un item donné parmi plusieurs (même item avec des valuations différentes):

$$R_{prox} : \text{item} \times \text{item} \times \text{item} \rightarrow \text{item}$$

$$R_{prox}(i_{k-1}, i_{k-2}, i_{k-3}) = \begin{cases} i_{k-2} & \text{si } \text{Valeur}(\text{Ecart}(i_{k-1}, i_{k-2})) < \text{Valeur}(\text{Ecart}(i_{k-1}, i_{k-3})) \\ i_{k-3} & \text{sinon} \end{cases} \quad (4.4)$$

Exemple:

Soit $i_{1obj}(a, 3, 2)$: un objectif pédagogique,
 $i_{1_a}(a, 2, 1)$: les valeurs associés à l'item i_1 à l'instant a,
 $i_{1_b}(a, 0, 2)$: les valeurs associés à l'item i_1 à l'instant b,

Nous pouvons calculer quel est l'item le plus proche de l'item i_{1obj} :

$$\begin{aligned} \text{Valeur}(\text{Ecart}(i_{1obj}, i_{1_a})) &= \text{Valeur}(i_1(\text{Min}(a,a), (3-2), (2-1))) \\ &= \text{Valeur}(i_1(a,1,1)) \\ &= 4 \end{aligned}$$

$$\begin{aligned} \text{Valeur}(\text{Ecart}(i_{1obj}, i_{1_b})) &= \text{Valeur}(i_1(\text{Min}(a,a), (3-0), (2-2))) \\ &= \text{Valeur}(i_{1obj}(a,3,0)) \\ &= 5 \end{aligned}$$

donc,

$$R_{prox}(i_{1obj}, i_{1_a}, i_{1_b}) = i_{1_a}$$

e) Compétence relative à un item

L'abus de langage, qui consistait à utiliser un même terme, à savoir « *item* », pour référencer deux choses distinctes, prend fin ici. En effet cette confusion était possible tant que les opérations portaient sur valeurs différentes du même concept (unité de connaissance choisie par l'enseignant). Maintenant que nous voulons développer les opérations portant sur un ensemble de concepts, il devient impératif de distinguer des deux.

Dorénavant nous emploierons le terme de compétence relative à un item pour désigner le concept du domaine enseigné et les valeurs qui lui sont rattachées. La compétence relative à l'item i_k sera défini par :

$$i_k(mg_1, sa_1, sf_1) \quad \text{notée} \quad C_{1k}$$

où mg_1 représente la valeur de la *mesure globale* pour la compétence C_{1k}
 sa_1 représente la valeur du *savoir-appliquer* pour la compétence C_{1k}
 sf_1 représente la valeur du *savoir-faire* pour la compétence C_{1k}

Relation d'ordre sur les compétences :

Les opérations définies précédemment vont permettre de définir la relation d'ordre \leq entre des compétences relatives à un même item.

$$\text{TestComp} : \text{Comp} \times \text{Comp} \rightarrow \text{Booléen}$$

Cette opération est une comparaison entre des compétences portées sur le même item avec une valuation différente. Elle permet ensuite de définir le niveau des connaissances sur un sujet.

Soit les compétences $C_{1k} = \{i_k(mg_1, sa_1, sf_1)\}$ et $C_{2k} = \{i_k(mg_2, sa_2, sf_2)\}$ portant sur le même item i_k .

$$\text{TestComp}(C_{1k}, C_{2k}) = \begin{cases} 1 & \text{Si} \quad \text{Valeur}(C_{1k}) \leq \text{Valeur}(C_{2k}) \\ 0 & \text{Sinon} \end{cases} \quad (4.5)$$

4.2.4 Définition d'opérations sur un ensemble d'items

Pour modéliser le module tutoriel nous avons besoin de calculs sur l'ensemble des compétences relatives aux items afin de choisir les activités qui seront proposées dans le parcours d'apprentissage.

C'est l'ensemble des compétences relatives à chacun des items d'évaluation définis par l'enseignant-auteur que l'on nomme le niveau de connaissance quand il s'agit de l'élève et objectifs pédagogiques lorsqu'il s'agit du but à atteindre.

Soit n le nombre d'items définis pour sujet donné.

Soit C_{1k} la compétence relative à l'item i_k pour l'objet d'étude numroté 1.

Deux niveau de connaissances de ce même objet d'étude sont notés N_1 et N_2 et définis comme suit :

$$\begin{aligned} N_1 &= \{C_{1k}\} & k \in 1 \dots p, \text{ avec } p \leq n \\ N_2 &= \{C_{2k}\} & k \in 1 \dots q, \text{ avec } q \leq n \end{aligned}$$

Des opérations similaires à celles définies sur un item existent sur l'ensemble des items.

Valeur d'un ensemble d'items :

$$EValeur(N_1) = \sum_{j=1}^n Valeur(C_{1j}) \quad (4.6)$$

Somme entre deux ensembles d'items :

$$ESomme(E_{jN}, E_{kM}) = \text{ensemble de } \begin{cases} Somme(i_j, i_k) & j = k \\ i_j, i_k & j \neq k \end{cases} \quad (4.7)$$

Inclusion de deux ensembles :

Inclus : $N_1 \subset N_2$

Cette opération est utilisée pour tester si le niveau de connaissances N_1 est un sous-ensemble de N_2 . Un exemple d'utilisation de cette comparaison est présenté dans la section (cf. 4.3.2.a). Nous pouvons écrire cette opération de la façon suivante :

Inclus : Niveau \times Niveau \rightarrow Booléen

$$Inclus(N_1, N_2) = \begin{cases} 1 & \underline{Si} \quad \forall C_{1i} \in N_1, \exists C_{2j} \in N_2 \quad \text{avec } i = j \\ 0 & \underline{Sinon} \end{cases} \quad (4.8)$$

InclusDans : $N_1 \overset{\subset}{\text{dans}} N_2$

A partir de la définition 3 (cf. page 27), nous pouvons écrire cette opération comme suit :

InclusDans : Niveau \times Niveau \rightarrow Booléen

$$InclusDans(N_1, N_2) = \begin{cases} 1 & \underline{Si} \quad Inclus(N_1, N_2) \text{ et } \forall C_{1i} \in N_1, TestComp(C_{1i}, C_{2i}) \\ 0 & \underline{Sinon} \end{cases} \quad (4.9)$$

4.3 Unités logiques d'apprentissage

4.3.1 Structure des unités logiques d'apprentissage

Dans le modèle ECSAI chaque unité logique d'apprentissage est décrite par un bloc comprenant trois parties représentées sur la figure 4.4.

Les pré-conditions expriment les conditions préalables requises afin de pouvoir exécuter cette unité. Celles-ci se composent d'une liste d'unités, éventuellement vide et d'un prérequis. La liste d'unités, appelée antécédent, contient les unités qui doivent avoir été exécutées avant l'unité actuelle.

La partie contenu se compose de plusieurs paramètres : identificateur, type, item référencés et titre. Un identificateur est utilisé pour référencer de manière unique chaque unité d'apprentissage. Le type de l'unité est soit une présentation, soit un exercice. Les items référencés regroupent les concepts qui sont enseignés dans l'unité.

Les post-actions décrivent les changements à effectuer dans le modèle d'élève quand l'unité est terminée et d'autres actions que le système exécutera. Ces descriptions d'unités permettent la construction du réseau des unités d'apprentissage (partie droite des modèles formels de la figure 4.3). Avec les modèles fournis par l'enseignant dans notre exemple, l'unité U3 précède obligatoirement l'unité U6.

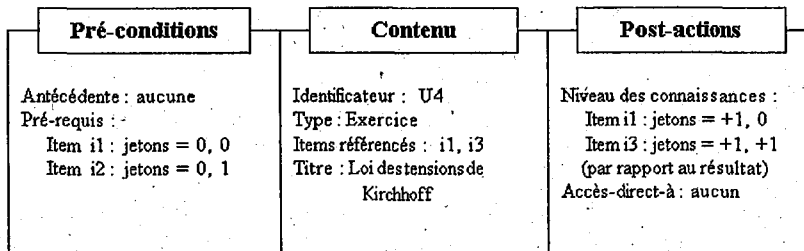


FIG. 4.4 – Modèle d'une unité logique d'apprentissage

4.3.2 Définition d'opérations sur les unités logiques

Rappelons que nous utilisons les informations dans ULAs pour calculer le parcours d'apprentissage pour chaque apprenant. Pour simplifier les calculs et les implantations, nous écrivons maintenant les opérations sur les ULAs.

Pour la suite il sera utile de faire référence aux différents éléments rentrant dans la définition d'une unité. Pour simplifier l'écriture nous adopterons les conventions suivantes :

Soit U_i l'unité d'apprentissage numéro i .

$\text{Filtre}(U_i)$ est l'ensemble des éléments contenant dans la partie pré-conditions.

$\text{ItemsReferences}(U_i)$ est l'ensemble des items références dans cette unité.

$\text{Apport}(U_i)$ est l'ensemble des compétences susceptibles d'être acquises pendant l'exécution de l'unité i .

a) Accès aux unités

A partir de la structure d'unité logique, nous pouvons définir les opérations sur les accessibilités d'unité des deux façons suivantes.

«*Accessible après*»

Cette comparaison permet de savoir quelles sont les ULAs qui peuvent être accessibles après l'unité actuelle. L'opération retourne un booléen qui.

$$\text{AccessibleAprès : Unité} \times \text{Unité} \rightarrow \text{Booléen}$$

Soit le filtre de l'unité logique U2 constitué de :

A_2 : l'ensemble des unités logiques devant être activées avant l'unité U2

N_2 : le niveau de connaissances requis pour activer l'unité U2

Soit U1 une unité logique dont :

N_1 : le prérequis,

A_1 : les antécédents,

U_{1A} : l'apport de connaissances.

L'unité U2 est accessible après l'unité U1 si $\text{AccessibleAprès}(U1, U2)$ retourne vrai.

$$\text{AccessibleAprès}(U2, U1) = \begin{cases} 1 & \text{Si } (A_2 \subset A_1) \text{ et } \text{InclusDans}(N_2, \text{Somme}(N_1 + U_{1A})) \\ 0 & \text{Sinon} \end{cases} \quad (4.10)$$

«*Accessible pour*»

Cette comparaison prend en compte le niveau des connaissances d'un élève donné. Alors elle nous permet de calculer les liens dynamiques pour chaque élève. Nous pouvons écrire cette opération en utilisant la définition présentée dans la page 30 de façon suivante :

$$\text{AccessiblePour : Unité} \times \text{Apprenant} \rightarrow \text{Booléen}$$

Soit le filtre de l'unité logique U1 constitué de :

A_1 : l'ensemble des unités logiques devant être activées avant l'unité U1

N_1 : le niveau de connaissances requis pour activer l'unité U1

Soit l'apprenant X dont :

P : l'ensemble des unités activées par cet apprenant

N_X : le niveau de ses connaissances

Par la définition 6, nous obtenons

$$\text{AccessiblePour}(U1, X) = \begin{cases} 1 & \text{Si } (A_1 \subset P) \text{ et } \text{InclusDans}(N_1, N_X) \\ 0 & \text{Sinon} \end{cases} \quad (4.11)$$

b) Liens statiques

Les liens statiques existent indépendamment d'un élève. Cette opération détermine la nature du lien qui relie deux unités.

«*LienStatique*»

LienStatique: Unité × Unité → Lien (Texte)

$$\text{LienStatique}(U1, U2) = \begin{cases} \text{même-que} & \text{si} & \begin{cases} \text{Items référencés (U1)} = \text{Items référencés (U2)} \\ \text{Filtre (U1)} = \text{Filtre (U2)} \\ \text{Apport (U1)} = \text{Apport (U2)} \end{cases} \\ \text{suivi-de} & \text{si} & \text{AccessibleAprès}(U2, U1) \\ \text{vide} & \text{sinon} & \end{cases} \quad (4.12)$$

«*estSymétrique*»

estSymétrique: Unité × Unité → Booléen

$$\text{estSymétrique}(U1, U2) = \begin{cases} 1 & \text{si} & \begin{cases} \text{LienStatique}(U1, U2) = \text{suivi-de} \\ \text{LienStatique}(U2, U1) = \text{suivi-de} \end{cases} \\ 0 & \text{sinon} & \end{cases} \quad (4.13)$$

«*estTransitive*»

estTransitive: Unité × Unité × Unité → Booléen

$$\text{estTransitive}(U1, U2, U3) = \begin{cases} 1 & \text{si} & \begin{cases} \text{LienStatique}(U1, U2) = \text{suivi-de} \\ \text{LienStatique}(U2, U3) = \text{suivi-de} \\ \text{LienStatique}(U1, U3) = \text{suivi-de} \end{cases} \\ 0 & \text{sinon} & \end{cases} \quad (4.14)$$

c) Liens dynamiques

A l'inverse des liens statiques, les liens dynamiques sont créés pendant l'apprentissage en tenant compte des progrès réalisés par l'élève.

LienDynamique: Unité × Unité × Apprenant → Lien (Texte)

$$\text{LienDynamique}(U1, U2, X) = \begin{cases} \text{équivalent-à} & \text{si} & \begin{cases} \text{Items référencés (U1)} = \text{Items référencés}(U2) \\ \text{Apport}(U1) = \text{Apport}(U2) \\ \text{AccessiblePour}(U1, X) \\ \text{AccessiblePour}(U2, X) \end{cases} \\ \text{implique} & \text{si} & \text{AccessiblePour}(U2, X) \\ \text{vide} & \text{sinon} & \end{cases} \quad (4.15)$$

4.4 Unités physiques d'apprentissage

Nous passons maintenant des modèles utilisés dans ECSAIWEB à la structure d'une unité physique d'apprentissage (UPA³⁴). Ceci est un service supplémentaire. Cette unité a une structure complexe susceptible par son exécution d'enrichir les connaissances de l'élève. Il existe différents types d'unités physiques correspondant aux différentes formes pédagogiques généralement proposées dans l'enseignement traditionnel : présentation, rappel, exercice et test. Par définition, les unités physiques d'apprentissage contiennent les actions pédagogiques qui doivent être mises en œuvre pour faire acquérir à l'élève une part des connaissances du domaine.

L'ensemble des unités physiques est groupé, selon la nature des unités d'apprentissage, en deux bases : la base des présentations et la base des exercices. Pour faciliter le changement du contenu des bases des unités, les unités peuvent incorporer tous les éléments des documents de HTML comprenant des graphiques et du son entièrement composés.

4.4.1 Unités physiques de type «PRESENTATION»

La structure de la base des unités physique du type «PRESENTATION» contient trois catégories : normal, exemple et rappel. La méthode de visualisation de chaque catégorie dépend de la stratégie pédagogique, par exemple lors de la présentation d'un exemple, après présentation d'énoncé aux élèves, les enseignants en général expliquent comment résoudre le problème étape par étape. Avec les balises HTML existantes, nous ne pouvons pas créer directement les pages comme nous l'avons décrit dans la phase précédente. Alors nous avons décidé d'ajouter des balises supplémentaires pour former des unités physiques à plusieurs catégories comme présenté dans les sections suivantes.

a) Structure générale

La structure générale d'une unité physique d'apprentissage du type PRESENTATION est fournie dans la figure 4.5.

Les balises <PRESENTATION ...> au début et </PRESENTATION> à la fin représentent le cadre de chaque présentation. Les deux balises sont les balises spéciales utilisées pour identifier la nature et les catégories d'une unité physique d'apprentissage. Les ... au milieu représentent les balises spéciales et les balises HTML normales. Les informations entre les balises <PRESENTATION ..> et </PRESENTATION> sont manipulées différemment selon le type de présentation. Le TYPE de présentation est la catégorie : normale, exemple et résumé ou rappel.

b) La présentation de la catégorie NORMAL

Dans la présentation de la catégorie Normal, toutes les informations entre les balises <PRESENTATION ..> et </PRESENTATION> doivent être présentées une seule fois (voir la figure 4.7.b). Nous avons utilisé le modèle indiqué dans la structure générale (figure 4.5) en changeant de *TYPE=xxx* à *TYPE=Info* pour identifier le type de la présentation. Nous l'appelons "*la présentation de la catégorie NORMAL*". Le modèle de cette catégorie est représentée dans la figure 4.6.

L'exemple du contenu d'une présentation de cette catégorie et la page que nous attendons sont représentés dans la figure 4.7.

34. UPA : Unité Physique d'Apprentissage

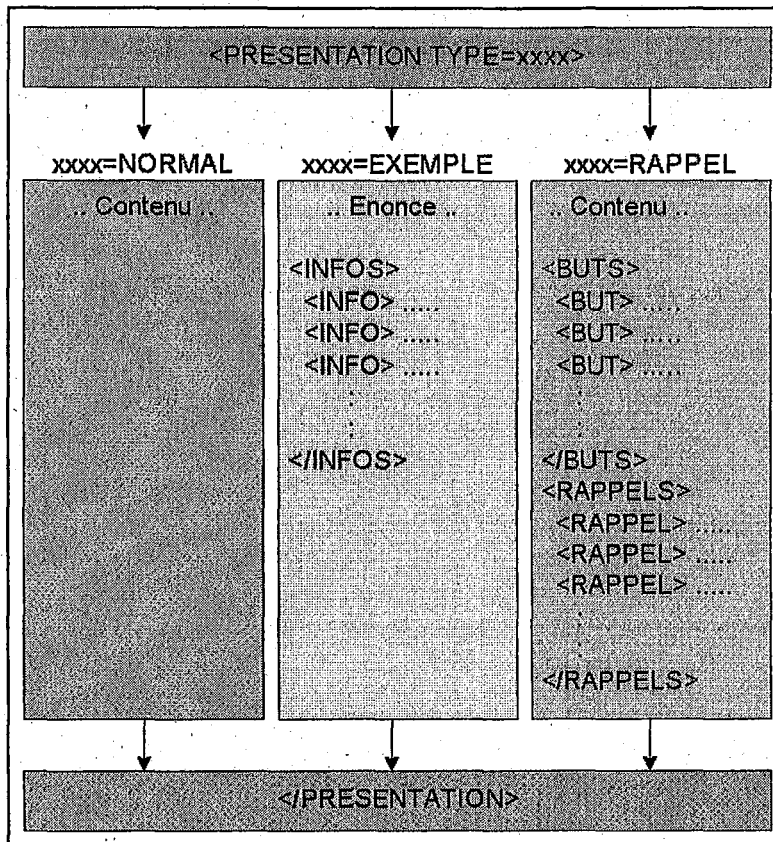


FIG. 4.5 – Structure générale de la présentation d'une unité physique

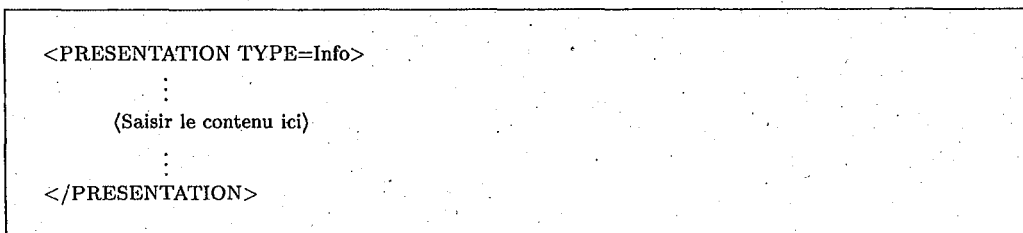


FIG. 4.6 – Modèle de la présentation du type NORMAL

```

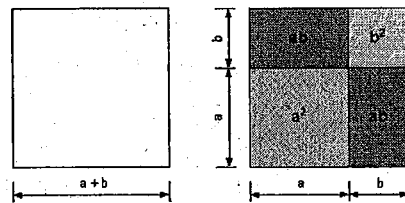
<PRESENTATION TYPE=Info>
<H1>La différence de deux carrés</H1>
<H2>Introduction:</H2>
<CENTER><IMG SRC="http://www.loria.fr/~sanrach/docs/Images/B28-2.gif"></CENTER>
<UL>
<LI>Observez le dessin ci-dessus et exprimez de deux façons différentes l'aire totale du carré.
<BR>En déduire que:  $(a + b)^2 = a^2 + 2ab + b^2$ 
<LI>En observant que  $45 = 40 + 5$ , calculez  $45^2$  en utilisant le résultat ci-dessus.</LI>
</UL>
<H2>Propriétés:</H2>
<UL>
<LI>Quels que soient les nombres relatifs a et b
<PRE>
       $(a + b)^2 = a^2 + 2ab + b^2$ 
</PRE>
</UL>
<H2>Démonstration:</H2>
<PRE>
       $(a + b)^2 = (a + b)(a + b)$ 
      or  $(a + b)(a + b) = a*a + a*b + a*b + b*b = a^2 + 2ab + b^2$ 
      d'où  $(a + b)^2 = a^2 + 2ab + b^2$ 
</PRE>
</PRESENTATION>

```

a) Le contenu

La différence de deux carrés

Introduction :



- Observez le dessin ci-dessus et exprimez de deux façons différentes l'aire totale du carré. En déduire que: $(a + b)^2 = a^2 + 2ab + b^2$
- En observant que $45 = 40 + 5$, calculez 45^2 en utilisant le résultat ci-dessus.

Propriétés :

- Quels que soient les nombres relatifs a et b

$$(a + b)^2 = a^2 + 2ab + b^2$$

Démonstration :

$$(a + b)^2 = (a + b)(a + b)$$

or

$$(a + b)(a + b) = a*a + a*b + a*b + b*b = a^2 + 2ab + b^2$$

d'où

$$(a + b)^2 = a^2 + 2ab + b^2$$

b) La page obtenue

FIG. 4.7 – Exemple de contenu que nous voulons présenter dans la page entière

c) La présentation de la catégorie EXEMPLE

Pour que la transmission d'un exemple représenté dans la figure 4.8 aux élèves ne soit pas trop chargée, nous devons présenter comment résoudre ce problème **ETAPE par ETAPE**. C'est-à-dire, nous présenterons d'abord l'énoncé du problème, puis expliquerons chaque étape de la solution en affichant le contenu associé à l'étape demandée.

Pour mettre en application directement cette méthode sur le Web, nous devons créer plusieurs pages avec les liens liés à chaque page de chaque étape. Imaginons que nous ayons plusieurs présentations de ce type, il est difficile pour l'enseignant de les faire parce qu'il doit prendre beaucoup du temps et bien s'organiser. Nous avons décidé de mettre l'ensemble des étapes en une seule page en ajoutant des balises spéciales (INFO) dans le premier modèle (figure 4.6) pour identifier la nature de la présentation et les étapes. Nous avons appliqué ici la technique adaptative de présentation «stretchtext». Nous l'appelons «*la présentation de la catégorie EXEMPLE*» (figure 4.9).

L'exemple de la présentation du type EXEMPLE et la page obtenue sont présentés dans la figure 4.10.

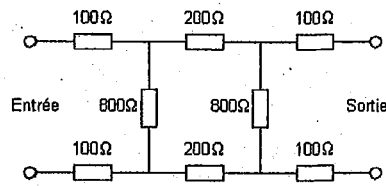
d) La présentation de la catégorie RAPPEL ou RESUME

La catégorie «*Rappel*» est utilisée pour aider des élèves en leur rappelant ou en leur résumant les connaissances qu'ils sont déjà étudiées ou en leur transmettant des informations de façon différente. Ce type contient deux balises dépendantes: <BUTS> et <RAPPELS>. Les éléments <BUT> indiquent le numéro correspondant à l'ordre de celui de <RAPPEL>. La balise RAPPEL n'est pas réservée à cette catégorie comme le nom indiqué mais nous pouvons l'utiliser aussi pour présenter un exercice qui a plusieurs méthodes de résolution ou bien intégrer plusieurs exemples dans une seule page (figure 4.12).

4.4.2 Unités physiques de type «EXERCICE»

Ramsden [Ram92] a décrit au sujet des essais de choix multiples que «*les questions à choix multiples fournissent une autre excellente occasion d'offrir la rétroaction sous une forme efficace. La rétroaction sur les questions à choix multiples, si elle est donnée à toutes, est habituellement limitée à des points indiquant la proportion de bonnes réponses obtenues. Les étudiants ne savent pas à quelles questions ils ont mal répondu, pourquoi ils ont tort, ou ce que seraient les réponses correctes. Pourtant une question est relativement simple pour fournir à des étudiants la clef pour un tel essai et pour fournir des explications principales pour la réponse correcte.*» Le rôle de la rétroaction dans n'importe quel environnement d'apprentissage est réellement important. Au cours des années une variété de mécanismes d'essai ont été utilisés dans des épreuves d'enseignement allant des essais classiques dans la classe aux essais réguliers à l'emploi des systèmes d'apprentissage contrôlés par ordinateur. Plusieurs de ces mécanismes de rétroaction emploient des méthodes d'essai de choix multiples en particulier dans des domaines d'étude professionnels et scientifiques [BDG95]. Le rôle de rétroaction est très important mais pas seulement la question à choix multiple, il est valable aussi pour tous types des questions.

Le temps pris pour fournir la rétroaction indiqué par Ramsden est une question importante [Rek83]. Avec une population d'étudiants qui est géographiquement distribuée, comme cela est le cas pour la plupart des environnements d'apprentissage ouverts, il n'y a pas le retour des corrections d'exercice ou d'examen et d'autres solutions doivent être considérées. Les outils d'impression, audiovisuels etc. utilisés dans l'étude ouverte peuvent fournir la rétroaction structurée à l'étudiant mais sont chargés des difficultés.

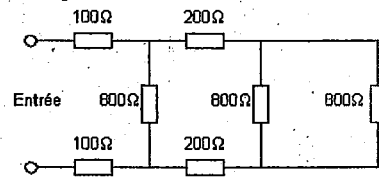
Exemple:**Figure a)**

Calculer la résistance entre les bornes d'entrée du réseau en échelle représenté à la figure a) lorsqu'une résistance de charge de 600 ohms est connectée entre les bornes de sortie.

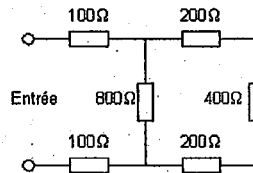
Solution:

Etape 1. La résistance de charge de 600 ohms branchée entre les bornes de sortie forme un circuit série avec les deux résistances de droite de 100 ohms. D'où :

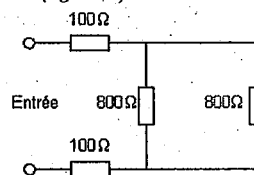
$$R_{eq} = 100 + 600 + 100 = 800 \text{ ohms.}$$

**Figure b)**

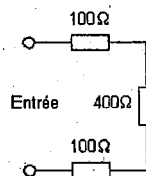
Etape 2. Remplaçons les deux résistances de 800 ohms en parallèle du circuit équivalent de la figure b) par une résistance de 400 ohms (figure c).

**Figure c)**

Etape 3. Remplaçons les deux résistances de 200 ohms et la résistance de 400 ohms en série du circuit de la figure c) par une résistance équivalente de 800 ohms (figure d)

**Figure d)**

Etape 4. Dans ce cas aussi, remplaçons les deux résistances en parallèle de 800 ohms par leur résistance équivalente de 400 ohms (figure e)

**Figure e)**

Etape 5. Enfin, la résistance totale du circuit série simple de la figure e) est

$$R_i = 100 + 400 + 100 = 600 \text{ ohms.}$$

FIG. 4.8 -- Exemple d'une présentation de l'exemple

```

<PRESENTATION TYPE=Info>
  ( Zone d'énoncé ou de contenu )
<INFOS>
  <INFO>.....
</INFOS>
</PRESENTATION>

```

FIG. 4.9 – Modèle de présentation de la catégorie EXEMPLE

```

<PRESENTATION TYPE=Info>
<H2>Exemple:</H2>
<CENTER><IMG
SRC="http://www.loria.fr/~sanrach/docs
/Image/Img7-5.gif"><BR>
<B>Figure a</B>
</CENTER>
Calculer la résistance entre les bornes d'entrée du réseau
<b>en échelle</b> représenté à la figure a) lorsqu'une
résistance de charge de 600 ohms est connectée entre les
bornes de sortie.
<h2>Solution:</H2>
<INFOS>
<INFO>La résistance de charge de 600 ohms branchée
entre les bornes de sortie forme un circuit série avec les
deux résistances de droite de 100 ohms. D'où : <BR>
<CENTER>Réq = 100 + 600 + 100 = 800 ohms.<BR>
<IMG SRC="http://www.loria.fr/~sanrach/docs
/Image/Img7-7a.gif"><BR>
<B>Figure b)</B>
</CENTER>

```

...

```

<INFO>Enfin, la résistance totale du circuit série simple
de la figure e) est <BR>
<CENTER>Ri = 100 + 400 + 100 = 600
ohms.</CENTER>
</INFOS>
</PRESENTATION>

```

a) Le contenu dans le modèle

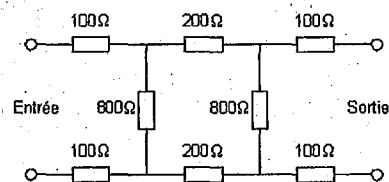
Exemple:

Figure a)

Calculer la résistance entre les bornes d'entrée du réseau en échelle représenté à la figure a). Lorsqu'une résistance de charge de 600 ohms est connectée entre les bornes de sortie.

Solution:

Etape 1. La résistance de charge de 600 ohms branchée entre les bornes de sortie forme un circuit série avec les deux résistances de droite de 100 ohms. D'où :

$$R_{\text{éq}} = 100 + 600 + 100 = 800 \text{ ohms.}$$

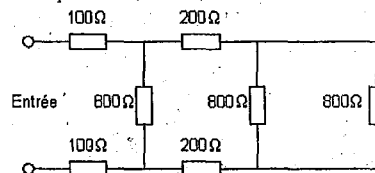


Figure b)

Etape suivante?

b) La page obtenue

FIG. 4.10 – Exemple d'une présentation du type EXEMPLE


```

<PRESENTATION Type = Rappel>
  ( Le contenu commun)
<BUTS>
  <BUT>...
  :
</BUTS>
<RAPPELS>
  <RAPPEL>...
  :
</RAPPELS>
</PRESENTATION>

```

FIG. 4.11 – Modèle de présentation de la catégorie RAPPEL

<pre> <PRESENTATION TYPE=Rappel> <h2>Rappels :</h2> Quels que soient les nombres a et b:
 : (a-b)<sup>2</sup>=a<sup>2</sup>- 2 ab + b<sup>2</sup> : <h2>Exemple:</h2> <BUTS> : <BUT>(2 - (3/5)x)<sup>2</sup> : </BUTS> <RAPPELS> : <RAPPEL>(2-(3/5)x)<sup>2</sup>= 2<sup>2</sup>- 2*(3/5)x + ((3/5)x)<sup>2</sup>
 : </RAPPELS> </PRESENTATION> </pre> <p>a) Le contenu</p>	<p>Rappels :</p> <p>Quels que soient les nombres a et b :</p> <ul style="list-style-type: none"> • $(a + b)^2 = a^2 + 2ab + b^2$ • $(a - b)^2 = a^2 - 2ab + b^2$ • $(a + b)(a - b) = a^2 - b^2$ <p>Exemple:</p> <ol style="list-style-type: none"> 1. $(x + (3/2))^2$ 2. $(2 - (3/5)x)^2$ ← Choisi ici. 3. $(2x - (1/2))(2x + (1/2))$ <p>RAPPEL: 2:</p> $(2 - (3/5)x)^2 = 2^2 - 2 * 2 * (3/5)x + ((3/5)x)^2$ $= 4 - (12/5)x + (9/25)x^2$ <p>b) La page obtenue</p>
---	---

FIG. 4.12 – Exemple d'une présentation de la catégorie RAPPEL que nous avons souhaitée

La nature d'hypertexte de WWW permet la capacité d'incorporer l'exercice ou l'examen dans le milieu des matériaux de WWW. Un étudiant pourra avoir des hyper-liens de matériaux pédagogiques à un exercice qui lui donne la «*rétroaction immédiate*» et les sommaires statistiques à l'enseignant. Les questions peuvent incorporer tous les éléments des documents de WWW (HTML) comprenant les graphiques et le son entièrement composés. De même la réponse aux étudiants peut fournir des liens d'hypertexte à d'autres matériaux.

La conception de modélisation d'un exercice est fortement importante comme celle de la présentation. Elle doit aussi fournir des extensions du langage HTML pour faciliter la création d'une séquence d'essai. Ces extensions non standard, cependant, ne gêne pas le client ou le serveur de WWW, ils sont traités avant de retourner les réponses au client. Ce modèle fournit les types de questions suivants :

- la question à choix multiples (QCM) [BDG95],
- la numérique simple,
- le vrai/faux [BDG95],
- le texte simple.

Ce modèle doit également fournir :

- les conseils par le choix des utilisateurs,
- les rétroactions dépendantes du choix de la réponse avec la commande substantielle sur la réponse à fournir,
- une interface cohérente aux questions du même type.

Il est clair qu'il y a eu beaucoup à faire pour que l'enseignant puisse mettre en application facilement cela. Ce modèle a été développé en se concentrant sur le soulagement du développement des *questions renforcées* qui seront incluses dans les unités physiques d'apprentissage ou les examens. Le modèle a été conçu pour fournir un développement d'essai basé sur les formes en ligne. Les enseignants-auteurs devront pouvoir entrer dans cet environnement pour développer et augmenter la question. L'environnement doit mener l'enseignant-auteur par les étapes de développement de la question, à partir de la question aux conseils, aux rétroactions et aux réponses fausses.

Notez que le programme HTTPD ne met pas en application un programme d'analyse syntaxique de HTML. Les éléments, présentés dans la section 4.4.2, s'ils semblent et doivent être pertinents, doivent être les premiers sur leurs lignes. Notez également que quoiqu'une extension de HTML soit utilisée, le programme HTTPD ne passe aucun des éléments étendus vers le client, donc les clients ne doivent pas être modifiés pour utiliser ces documents. En général, les balises utilisées ne sont pas différentes du HTML standard, sauf que des lignes commençant par n'importe lequel de ces éléments sont manipulées de manière particulière par le serveur Web avant de les transférer aux postes des clients.

a) Structure Générale

Chaque question doit contenir en général selon son type de chaque question les éléments suivantes :

- le numéro de la question,
- l'objectif pédagogique lié à la question,
- le type de question (question à choix multiple, numérique, texte et vrai/faux),
- le degré d'acquisition (savoir-faire ou savoir-appliquer),
- la zone d'énoncé,

- la réponse,
- les rétroactions associées aux réponses (options),
- les aides (options).

A partir de contenu de chaque question, il nous permet de définir les nouvelles balises spéciales pour identifier chaque question les caractères que nous avons indiqués au dessus. Pour former une question, nous avons défini la structure générale représentée dans la figure 4.13.

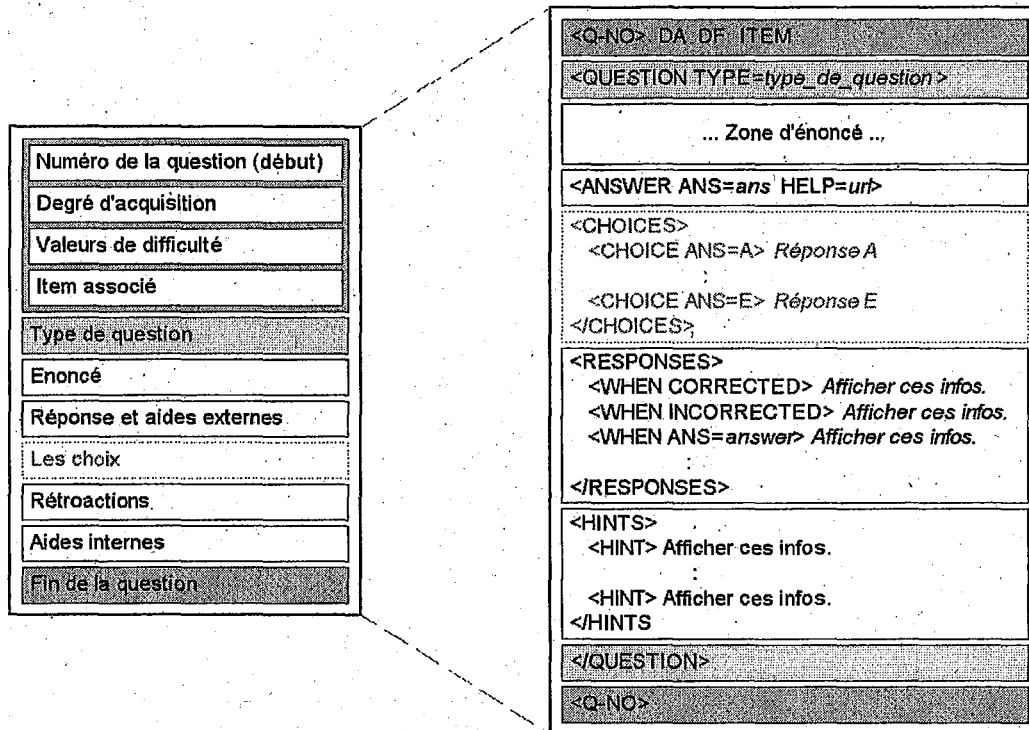


FIG. 4.13 – Contenu (à gauche) de chaque question et son modèle (à droite).

Les balises <Q-NO> au début et </Q-NO> à la fin représentent le cadre et numéro de chaque question. Nous avons ajouté aussi les valeurs qui ont été utilisées pour exprimer la valeur de degré d'acquisition (Da), la difficulté (Df) et l'item d'évaluation de chaque question. Les ... représentent les paramètres spéciaux de la question. Pour faciliter aux enseignants-auteurs l'évaluation des connaissances des élèves, nous permettrons à chaque question dans l'exercice de n'être lié qu'à un seul item d'évaluation, mais plusieurs questions peuvent porter sur le même item. Les informations entre les balises <QUESTION ..> et </QUESTION> sont manipulées différemment selon le type de question.

"xxx" identifie le type de question. Les types permis sont décrits ci-dessous.

- **Multiple-Choice** : pour dénoter une question à choix multiple (QCM),
- **Single-Numeric** : pour dénoter une question qui doit être répondue avec une valeur numérique simple,
- **True-False** : pour dénoter une question vrai/faux,

- **Text** : pour dénoter une question qui doit être répondue avec des textes.

ANSWER

La forme de la <ANSWER ...> est comme la suivante :

<ANSWER ANS=*x* HELP=*url*>

L'attribut de ANS=*x* indique que *x* est l'étiquette de la réponse correcte.

L'attribut de HELP=*url* est une option. S'il est donné, il indique le URL d'un document d'aide à l'extérieur d'une unité physique, et le lien est automatiquement inclus dans l'exercice.

RESPONSES

La section de <RESPONSES> est une option, elle fournit les rétroactions qui devraient être affichées après qu'une réponse ait été sélectionnée. Le format de la section de <RESPONSES> est représenté dans le paragraphe suivant.

Les éléments <WHEN ...> indiquent les conditions dans lesquelles le texte suivant doit être affiché comme la rétroaction.

<WHEN ANS=*x*> indique que le texte qui suit doit être affiché toutes les fois que l'utilisateur choisit l'étiquette *x* comme réponse.

<WHEN CORRECT> indique que le texte qui suit doit être affiché toutes les fois que l'utilisateur choisit la réponse **correcte**. Aucun autre WHEN doit être affiché sauf si la réponse ANS=*x* est donnée et correcte.

<WHEN INCORRECT> indique que le texte qui suit doit être affiché toutes les fois que l'utilisateur choisit une réponse **incorrecte**. Aucun autre WHEN doit être affiché sauf si la réponse ANS=*x* est donnée et incorrecte.

Tous les éléments WHEN sont des options, si aucun n'est donné alors la section de RESPONSES n'a pas besoin d'être donnée non plus. Dans l'un ou l'autre cas, une réponse appropriée par défaut est fournie, donnant la rétroaction si la réponse choisie était correcte ou pas.

HINTS

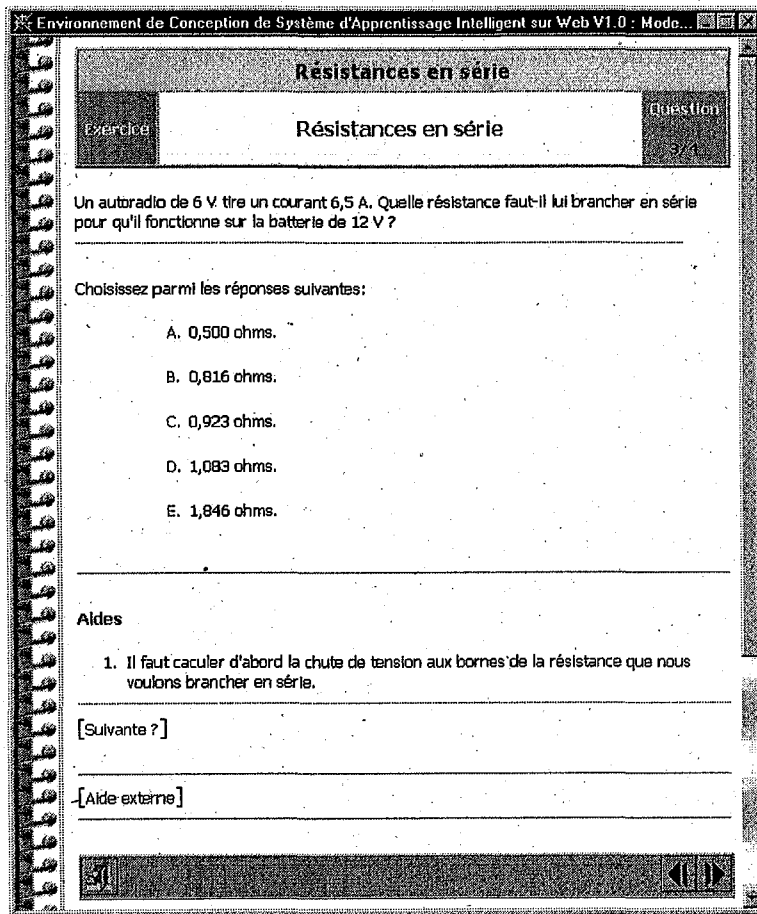
Cette section est utilisée pour aider les apprenants à résoudre un problème précis. Cette aide est différente de celle indiquée dans la section ANSWER. Elle n'est utilisée que pour résoudre la question. Cette partie doit être affichée ETAPE PAR ETAPE en cliquant sur le mot «Aide suivante». Mais HINT indiqué dans la partie ANSWER est un lien vers une aide située dans un fichier séparé et peut être lié à plusieurs questions.

b) La question à choix multiples

Les questions du type «*choix multiples*» sont celles dans lesquelles l'élève choisit une réponse à partir d'une liste d'un petit nombre des choix. Ce type de question a des balises supplémentaires, les choix, la liste des réponses possibles à la question. Même s'il y a plusieurs réponses, l'élève ne peut choisir qu'une seule réponse. La structure de QCM est présentée dans la figure 4.15.

CHOICES

La section <CHOICES> énumère les réponses possibles. L'attribut "ANS=..." est facultatif, et, s'il existe, fournit une étiquette pour le choix, sinon une étiquette numérique sera assignée automatiquement. Les étiquettes sont des liens hyper-texte que l'utilisateur choisit pour indiquer



Environnement de Conception de Système d'Apprentissage Intelligent sur Web V1.0 - Mode...

Resistances en série

Exercice Question

Un autoradio de 6 V tire un courant 6,5 A. Quelle résistance faut-il lui brancher en série pour qu'il fonctionne sur la batterie de 12 V ?

Choisissez parmi les réponses suivantes:

- A. 0,500 ohms.
- B. 0,816 ohms.
- C. 0,923 ohms.
- D. 1,083 ohms.
- E. 1,846 ohms.

Aides

- Il faut calculer d'abord la chute de tension aux bornes de la résistance que nous voulons brancher en série.

[Sulvante ?]

[Aide externe]

FIG. 4.14 – Exemple de la présentation d'un exercice

```

<QUESTION Type=Multiple-choice>
  (Zone d'énoncé)
<CHOICES>
  <CHOICE ANS=A>...
  <CHOICE ANS=B>...
  <CHOICE ANS=C>...
  <CHOICE ANS=D>...
  <CHOICE ANS=E>...
</CHOICES>
<ANSWER ANS=x HELP=url>
<RESPONSES>
  <WHEN ANS=A>...
  <WHEN ANS=B>...
  <WHEN ANS=C>...
  <WHEN ANS=D>...
  <WHEN ANS=E>...
  <WHEN CORRECT>...
  <WHEN INCORRECT>...
</RESPONSES><HINTS>
  :
  <HINT> ...
  :
</HINTS>
</QUESTION>

```

FIG. 4.15 – Modèle de question à choix multiples

son choix de réponse. L'étiquette est affichée automatiquement devant chaque choix.

Exemple

Un exemple de question du type QCM est présenté dans la figure 4.16.

c) La question du type «numérique simple»

Les questions du type numérique simple sont ceux dans lesquelles l'apprenant doit fournir un nombre simple comme la réponse à la question. L'auteur de la question peut exiger l'égalité exacte ou l'égalité approximative avec une tolérance pré-définie. Un navigateur avec la capacité de formes est exigé pour ce type de question. Les trois sections d'une question numérique simple sont représentées dans la figure 4.17.

Les descriptions principales de question sont présentées dans la section 4.4.2.a. Les descriptions supplémentaires sont décrites comme suit :

ANSWER

La forme de l'élément de <ANSWER...> est comme suit :

```
<ANSWER ANS=l TOL=t HELP=url>
```

La tolérance (TOL) t est la quantité relative dont la réponse fournie par l'apprenant peut être différente de la réponse correcte pour être jugée comme correcte. En d'autres termes, une réponse fournie, ANS, est jugée comme correcte si et seulement si $ABS(ANS - l) \leq ABS(t * l)$.

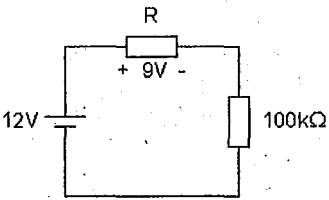
<pre> <1> sa 0.5 il <QUESTION TYPE=Multiple-Choice> <center></center> <p>Dans ce circuit, quelle est la résistance R?</p> <CHOICES> <CHOICE ANS=A>33 kohms. <CHOICE ANS=B>50 kohms. <CHOICE ANS=C>63 kohms. <CHOICE ANS=D>75 kohms. <CHOICE ANS=E>100 kohms. </CHOICES> <ANSWER ANS=A> <REPONSES> <WHEN CORRECT>Bravo!! votre réponse est juste. <WHEN INCORRECT>Il y a des faux dans votre calcul, corrigez-les!! </REPONSES> </QUESTION> </1> </pre> <p>a) Le contenu d'une question QCM</p>	<p>1.</p>  <p>Dans ce circuit, quelle est la résistance R?</p> <hr/> <p>Choisissez parmi les réponses suivantes:</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> A. 33 kohms. B. 50 kohms. C. 63 kohms. D. 75 kohms. E. 100 kohms. <hr/> <p>Bravo!! votre réponse est correcte.</p> <p>b) La présentation de QCM obtenue.</p>
--	--

FIG. 4.16 – Exemple d'exercice du type QCM

```

<QUESTION Type=Single-Numeric>
:
  (Zone d'énoncé)
:
<ANSWER .....>
<RESPONSES>
:
  <WHEN ...>...
:
</RESPONSES>
<HINTS>
:
  <HINT>...
:
</HINTS>
</QUESTION>

```

FIG. 4.17 – Modèle de question du type NUMERIQUE SIMPLE

RESPONSES

La section <RESPONSES> est une option; elle fournit les réponses qui devraient être affichées après qu'une réponse ait été donnée. Le format de la section <RESPONSES> est représenté ci-dessous :

```
<RESPONSES >
  <WHEN ANS=x>.....
  <WHEN EXPR=e>.....
  <WHEN CORRECT>.....
  <WHEN INCORRECT>.....
</RESPONSES>
```

L'attribut que nous avons ajouté est la balise <WHEN EXPR=e>. Elle indique le texte ayant été affiché comme rétroaction toutes les fois que l'expression, *e*, évalue pour rectifier. L'expression *e* peut contenir l'ANS comme variable, l'ANS sera la réponse numérique fournie par l'apprenant. Par exemple: <WHEN EXPR="ans<2.7"> signifie que le texte qui suit sera affiché comme la rétroaction si l'élève fournit une réponse inférieure à 2,7.

Exemple

La figure 4.18.a représente le contenu d'une question du type «Numérique Simple» et la figure 4.18.b représente la page que nous souhaitons présenter.

<pre><2>sf 0.5 i2 <QUESTION TYPE=Single-Numeric> Un vecteur bidimensionnel est écrit en utilisant la notation $3i + 4j$. Quelle est sa grandeur? <ANSWER ans=5.0> <HINTS> <HINT>La grandeur d'un vecteur est la racine carrée de la somme de la grandeur de ses composants. <HINT>Les deux composants d'un vecteur écrit dans la notation de vecteur (i.e., <VAR>Xi+Yj</VAR> </VAR>) sont <VAR>Xi</VAR>and <VAR>Yj</VAR>. <HINT>La grandeur d'un composant le long de l'axe i, tel <VAR>X i</VAR>, est simplement la valeur <VAR>X</VAR>. <HINT>La grandeur de <VAR>Xi+Yj</VAR>est ainsi la racine carrée de <VAR>X <SUP>2</SUP></VAR>+ <VAR>Y<SUP>2</SUP></VAR>. <HINT>La grandeur de <VAR>3i+4j</VAR>est donc la racine carrée de 3<SUP> 2</SUP>+ 4<SUP>2</SUP>. <HINT>Vous avez besoin vraiment plus de conseils? </HINTS> </QUESTION> </2></pre> <p>a) Le contenu</p>	<p>2. Un vecteur bidimensionnel est écrit en utilisant la notation $3i + 4j$. Quelle est sa grandeur?</p> <p>Votre réponse: <input type="text"/> <input type="button" value="Envoyer"/></p> <hr/> <p>Aides: 1. La grandeur d'un vecteur est la racine carrée de la somme de la grandeur de ses composants.</p> <p>Aide suivante?</p> <p>b) La page obtenue</p>
---	--

FIG. 4.18 – Exemple de la question du type numérique simple

d) La question du type «vrai ou faux»

Les questions vraies/fausses sont celles qui peuvent être répondues par une des deux réponses possibles : «*vraie*» (oui), ou «*fausse*» (non). Les caractéristiques principales de ce type de question ont déjà été présentées dans une section précédente (cf. section 4.4.2.a). L'exemple de la question du type vrai/faux et sa page sont présentés dans la figure 4.20.

```

<QUESTION TYPE=True-False>
:
<Zone d'énoncé>
:
<ANSWER .....>
<RESPONSES>
:
</RESPONSES>
<HINTS>
:
</HINTS>
</QUESTION>

```

FIG. 4.19 – Modèle de question du type vrai/faux

e) La question du type «texte»

Les questions de ce type sont celles qui peuvent être répondues par un mot, une phrase ou une valeur numérique, mais cette valeur devra être traitée comme texte. La structure de cette question, représentée dans le schéma 4.21, est la même que celle de la question du type numérique simple. L'exemple de la question du type "texte" et sa page sont présentés dans la figure 4.22.

4.5 Base de tests

Nous proposons en plus dans ECSAIWEB une base de tests. Cette base contient une base de pré-examen et une base de post-examen de chaque sujet. Chaque fichier de cette base contient le texte et l'image avec plusieurs questions. En général, les balises utilisées ne sont pas différentes de celles des exercices, sauf qu'il n'y a aucune aide à chaque question et que chaque question peut être liée à plusieurs items d'évaluation, par exemple la question 3 est liée aux items i1, i2 et i4 (voir figure 4.23). L'exemple suivant présente le contenu d'un examen et la page obtenue à chaque question est présentée dans la figure 4.24.

4.6 Conclusion

En résumé, le modèle du domaine que nous avons décrit est assez complexe. Donc il paraît difficile pour les enseignants-auteurs de manipuler de tels modèles pour enrichir les matériaux pédagogiques. Le problème se pose maintenant ; comment peut-on aider ces enseignants-auteurs

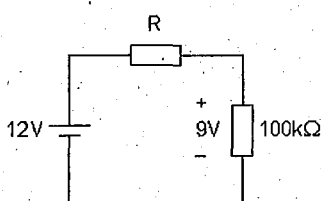
<pre> <3>sf 0.5 i2 <QUESTION TYPE=True-False> <center> </center> <p>Si l'on augmente les valeurs de la résistance R, les tentions aux bornes de la résistance 100 ohms diminuent : </p> <ANSWER ANS=true> </QUESTION> </3> </pre> <p>a) Le contenu</p>	<p>3.</p>  <p>Si l'on augmente les valeurs de la résistance R, les tentions aux bornes de la résistance 100 ohms diminuent :</p> <p>Choisissez parmi les réponses suivantes:</p> <p><input checked="" type="checkbox"/> Vrai, ← Choisir cette réponse</p> <p><input type="checkbox"/> Faux.</p> <hr/> <p>Bravo !!</p> <p>b) La page obtenue.</p>
--	--

FIG. 4.20 – Exemple de la question du type Vrai/Faux

```

<QUESTION Type = Single-Text> (Zone d'énoncé) <ANSWER .....>
<RESPONSES>
:
</RESPONSES>
<HINTS>
:
</HINTS>
</QUESTION>

```

FIG. 4.21 – Modèle de question du type TEXTE

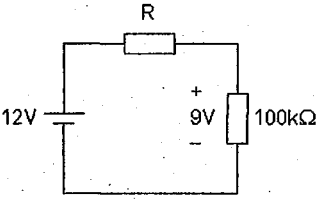
<pre> <4> sa 0.5 i3 <QUESTION TYPE=Single-Text> <center> </center> <p>Si l'on augmente les valeurs de la résistance R, les tentions aux bornes de la résistance 100 ohms diminuent ou augmentent? </p> <ANSWER ANS=diminuent HELP=http://www.loria.fr/~sanrach/helps/ node3.html> </QUESTION> </4> </pre> <p>a) Le contenu</p>	<p>4.</p>  <p>Si l'on augmente les valeurs de la résistance R, les tentions aux bornes de la résistance 100 ohms diminuent ou augmentent?</p> <p>Votre réponse : <input type="text" value="diminuent"/> <input type="button" value="Envoyer"/></p> <p>Bravo !!</p> <p>b) La page obtenue.</p>
--	--

FIG. 4.22 – Exemple de la question du type Texte

```

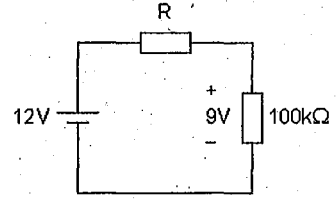
<3>sa 0.7 i1 i2 i4
<QUESTION TYPE=Multiple-Choice>
<center></center>
<p>Si on enlève la résistance R<sub>L</sub>, quelle est la chute de tension aux bornes de la résistance
5 ohms? </p>
<CHOICES>
<CHOICE ANS=A>16,67 V.
<CHOICE ANS=B>25 V.
<CHOICE ANS=C>33,33 V.
<CHOICE ANS=D>35 V.
<CHOICE ANS=E>36,67 V.
</CHOICES>
<ANSWER ANS=A>
</QUESTION>
</3>
<4>sf 0.5 i2 i3
<QUESTION TYPE=True-False>
<center></center>
<p>Dans le circuit, l'un des éléments AB représente une source de tension et l'autre une résistance. La
voltmètre et l'ampèremètre lisent correctement lorsqu'on connecte leurs bornes selon les signes de polarité
indiqués. </p>
<p><i><b>Est-ce que la boîte A représente la résistance? </b></i></p>
<ANSWER ANS=true>
</QUESTION>
</4>

```

FIG. 4.23 – Exemple d'une base de test

TEST

Circuits série et Circuits parallèle



Dans le circuit, quelle est la résistance R ?

Choisissez parmi les réponses suivantes:

- A. 33 kohms.
- B. 50 kohms.
- C. 63 kohms.
- D. 75 kohms.
- E. 100 kohms.

31

FIG. 4.24 – Présentation d'un contenu du test

et quels outils faut-il leur offrir? Ce point sera développé au chapitre 6 : des outils intégrés dans ECSAIWEB. Le chapitre suivant est consacré aux modèles d'élève.

Chapitre 5

Modèle des élèves

5.1 Introduction

Comme nous l'avons indiqué dans les chapitres précédents, le modèle d'élève est un composant d'un système tutoriel intelligent qui stocke des informations sur les élèves. Ces informations comprennent en particulier l'état des connaissances actuelles de l'élève. Il y a des difficultés pour obtenir une représentation précise des capacités de l'élève. Par conséquent, le modèle de l'élève ne peut pas être parfaitement précis et des mesures doivent être prises pour s'assurer que les actions de système sur la base de ces informations imprécises ne sont pas inadéquates.

Étant donné les difficultés ci-dessus, une question évidente au sujet des modèles d'élève est : faut-il en avoir un ? Simplement, le modèle d'élève est nécessaire afin de concevoir, l'instruction en fonction des caractéristiques d'un étudiant et des besoins d'apprentissage. Sans cette connaissance, le composant pédagogique du tuteur n'a aucune base sur laquelle prendre des décisions, et est forcé de traiter tous les étudiants de la même façon.

Dans le chapitre 3, nous avons montré également que les facultés d'adaptation et d'intelligence sont basées sur le modèle d'utilisateur, plus précisément «*le modèle des élèves*». Ce modèle joue un rôle très important dans un système tutoriel adaptatif et intelligent. En utilisant les informations stockées dans ce modèle, le STI peut générer le parcours d'apprentissage approprié pour chaque apprenant. Nous expliquerons dans la section 5.2 le rôle du modèle des élèves en général. La section 5.3 décrit généralement la représentation de ce modèle et ses contenus. Et le modèle recouvrement multi-points de vue que nous proposons fait l'objet de la section 5.4.

5.2 Rôle du modèle des élèves

L'individualisation de l'information et des liens-ancres (présentation adaptative) fournissant à l'utilisateur le support de navigation (navigation adaptative) est effectuée dans un système sur la base de l'information maintenue dans un modèle d'utilisateur : la représentation du système des préférences de l'utilisateur, la connaissance, la croyance ou les buts de la recherche d'information. Paiva, Self et Hartley [PSH95] ont succinctement défini la nature et le but d'un modèle d'élève ou UM³⁵) comme «... *des représentations de certaines caractéristiques et attitudes des étudiants, qui sont utiles pour réaliser l'interaction proportionnée et individualisée établie entre les environnements informatiques et les étudiants. Ils sont constitués par des descriptions de la connaissance supposée d'un étudiant à propos du sujet étudié et fournissent des informations à l'environnement d'apprentissage pour s'adapter à chaque étudiant.*» [PSH95]. Par

35. sigle anglais de *User Modeling*

exemple, l'ISIS-Tutor [BP94] est un système d'hypermédia adaptatif consacré à l'enseignement de quelques aspects de programmation qui fournit le support adaptatif de navigation. Le modèle d'élève pour chaque étudiant inclut des nœuds appris, prêts à être appris ou non-appris. Dans ce système, cinq éléments relatifs à l'utilisateur sont représentés à l'intérieur de ses modèles, seuls ou en association. Ce sont :

- le but courant de l'utilisateur (ou la tâche) : un but de recherche dans des tâches de recherche documentaire ou un but d'apprentissage dans des applications éducatives,
- la connaissance de l'utilisateur sur le domaine présenté dans l'hypermédia,
- l'expérience professionnelle de l'utilisateur (profession, travail dans des secteurs relatifs, point de vue),
- l'expérience de l'utilisateur (la connaissance sur l'utilisation d'un hyperespace),
- les préférences de l'utilisateur.

Différentes techniques ont été employées pour acquérir des informations sur l'utilisateur. Celles-ci incluent :

- les stéréotypes (classification des sous-groupes d'utilisateurs en utilisant des stéréotypes) sont incorporés dans le système par le concepteur de système au moment de la conception, alors le système à l'exécution doit déterminer quel stéréotype est applicable à l'utilisateur courant,
- les préférences écrites par l'utilisateur (demandées à l'utilisateur par le système au moment de l'exécution),
- l'analyse des actions de l'utilisateur et l'identification ou inférence de plan (calculée par le système à l'exécution) ; des mouvements entre les pages, la manière de choisir la page (navigation/query) ou le choix parfois explicite des tâches peuvent être employés par le système pour déterminer l'intention de l'utilisateur.

L'efficacité de certaines de ces techniques demeure controversée. J. Kay a noté que «*le modèle ne peut pas être quelque chose mais une conjecture s'il essaye de modéliser la connaissance de l'utilisateur*» [KK94]. Elle et d'autres [HKW⁺96] préconisent qu'un modèle d'utilisateur devrait être de type «*glass box*», c'est-à-dire inspectable par des utilisateurs et devrait travailler sous leur commande.

Dans le scénario éducatif de système de corpus fermé, le tuteur a une occasion sensiblement plus grande d'obtenir des informations sur l'utilisateur par des activités dirigées. En revanche, dans un scénario plus libre comme le parcours d'un hypermédia le recueil de telles informations est plus difficile. L'hypermédia peut être organisé comme une hiérarchie des nœuds correspondant à la hiérarchie des concepts du domaine. Chaque nœud peut se composer de pages multiples du matériel comprenant des explications, remédiation, prolongation, révision et essais. Les informations obtenues au cours des différentes activités pratiquées dans l'hypermédia augmentent les données contenues dans le modèle de l'utilisateur. Cette approche a été utilisée avec succès dans le système PUSH [EH96].

En résumé, dans les systèmes combinant un parcours hypermédia assez libre et un tutorat pédagogique, le modèle de l'élève doit apporter à la fois un support à la navigation dans l'hypermédia et les informations permettant les décisions pédagogiques du tuteur. Pour ECSAIWEB, il s'agit donc d'intégrer les techniques différentes utilisées dans ces deux domaines. De plus, nous pensons comme Chan, Lin et Kuo [CLK93] Kaplan, Fenwick et Chen [KFC93] Kobsa, Mümller et Nill [KMN94] Linard et Zeiliger [LZ95] et Brusilovsky [Bru96] que les modèles d'élèves doivent servir à montrer à l'élève où il en est de son apprentissage, ce qui est encore trop peu fréquent. Les recherches au sujet de la nature de l'utilisateur et de l'efficacité de l'environnement sont

basées sur le modèle d'utilisateur. Une intégration réussie d'un certain nombre de ces secteurs est exigée pour former des principes de base pour guider les recherches d'un SHA.

5.3 Représentation et Contenu du modèle d'élève

Il y a plusieurs méthodes pour représenter des informations sur l'utilisateur. Deux techniques généralement utilisées sont des *modèles de recouvrement* et des *réseaux bayésiens*.

5.3.1 Modèle de recouvrement

Le paradigme standard pour représenter un modèle d'étudiant est le modèle de recouvrement [CG77] dans lequel la connaissance de l'étudiant est considérée un sous-ensemble de la connaissance experte (figure 5.1.a). Avec cette représentation, un STI présente le matériel à l'étudiant avec l'objectif que sa connaissance s'assorte exactement à celle de l'expert. Les types de connaissance qui peuvent être représentés dans un modèle d'élève par recouvrement incluent les «*sujets*», qui correspondent aux éléments de la connaissance de domaine, et les règles de production qui correspondent à un savoir faire.

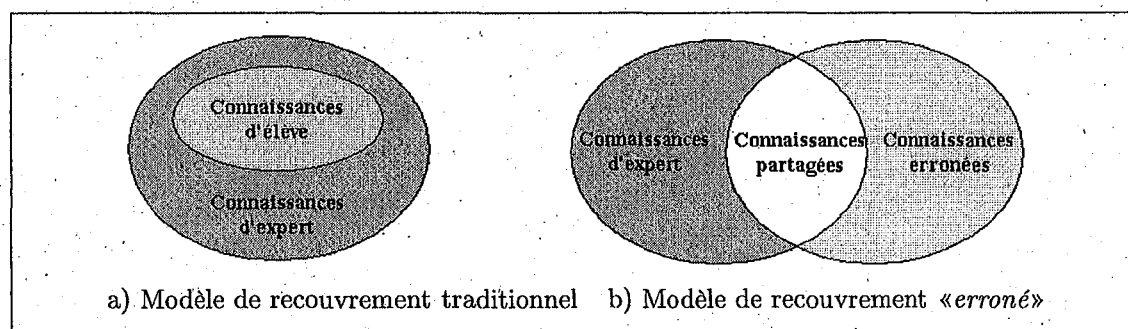


FIG. 5.1 - Modèles de recouvrement d'élèves

Un inconvénient de cette approche est qu'elle ne reconnaît pas que les étudiants peuvent avoir des croyances qui ne sont pas une partie de la base de connaissance de l'expert. Par exemple, les étudiants ont fréquemment des idées fausses au sujet d'un domaine. Par conséquent une extension du modèle de recouvrement représente explicitement *la connaissance erronée* que l'étudiant peut avoir (figure 5.1.b) [HDJG93]. Cette extension permet une meilleure remédiation des erreurs d'étudiant, puisque le fait qu'un étudiant croie quelque chose qui est incorrect est pédagogiquement significatif.

5.3.2 Réseaux bayésiens

Un autre mécanisme pour enregistrer la connaissance d'un étudiant est celui des réseaux bayésiens [MV95]. Ces réseaux font du raisonnement probabiliste sur l'état de la connaissance d'un étudiant, basé sur ses interactions avec le tuteur. Chaque nœud dans le réseau a une probabilité indiquant la probabilité de la connaissance d'étudiant sur le morceau de la connaissance. Nous n'avons pas étudié ce mécanisme parce que nous ne l'utilisons pas.

5.3.3 Ce qui est inclus dans le modèle d'élève

Dans ce paragraphe, nous analysons de façon plus précise que dans la section 5.3.1 et la section 5.3.2 ce que contient un modèle d'élève. Remarquons tout d'abord que les développeurs de STI ont une tendance à inclure plus d'information dans le modèle d'étudiant que le module pédagogique ne peut en employer [SB82]. Il faut donc préciser comment le modèle d'élève peut être utilisé par un système :

- pour créer un parcours d'apprentissage personnalisé,
- éventuellement pour tester l'utilité des quelques informations supplémentaires à des fins de recherche.

Donnant cette restriction, *quelles données un modèle d'étudiant devrait-il contenir ?* Clairement, il doit enregistrer la compréhension de l'étudiant du domaine. Cependant, *à quel niveau de granularité cette compréhension devrait-elle être représentée ?* À une extrémité, le tuteur pourrait simplement dire «*l'étudiant sait ce domaine*» ou «*l'étudiant ne sait pas ce domaine*». À l'autre extrémité, le modèle d'élève a pu enregistrer chaque action d'étudiant. La plupart des modèles d'étudiant se situent entre ces deux extrémités et essaient de modéliser l'étudiant au même niveau de granularité auquel le domaine est représenté. Puisque la plupart des domaines sont représentés en termes de concept de la matière, c'est la granularité la plus commune pour des modèles d'étudiant.

En plus d'enregistrer la compréhension d'un étudiant du domaine, un modèle d'élève peut également inclure des informations pédagogiques plus générales sur l'apprenant. Ce genre d'information pourrait exprimer les préférences générales d'un apprenant, comme s'il aime regarder des exemples avant d'essayer de répondre à n'importe quelle question ou s'il aime des exemples au sujet des «*calculs des résistances en série*» mais pas au sujet des «*capacitance*» etc.

D'autres informations générales de l'apprenant peuvent être incluses, comme l'acquisition et la conservation. L'*acquisition* mesure à quelle vitesse les apprenants apprennent de nouvelles matières, et la *conservation* mesure à quel point, ils se rappellent le matériel dans le temps. La recherche antérieure suggère que les facteurs généraux tels que l'acquisition et la conservation puissent être utiles pour modéliser d'étudiant. Les travaux de *LISP-Tutor* [BSW96a] et de *Stat Lady* [Shu95] indiquent que les facteurs généraux extraits à partir des données d'étude d'apprenant sont prédictifs de toutes les connaissances de l'apprenant et tiennent compte d'une réponse plus précise aux caractéristiques de l'étudiant.

Les informations stockées dans le modèle d'élève proposées dans ECSAI[Gav91] peuvent être séparées en deux catégories de connaissances : les connaissances permanentes et les connaissances évolutives. Les connaissances permanentes représentent un ensemble d'informations qui sont données au cours de la première session avec l'élève et restent valides pendant toute la durée d'apprentissage sans modifications. Elles sont les informations personnelles et les objectifs pédagogiques. Les connaissances évolutives représentent une synthèse, exploitable par le module tutoriel, du résultat de l'interaction avec l'apprenant. Elles sont divisées en deux groupes : l'acquis de l'apprenant à l'instant donné et l'historique de son apprentissage. L'acquis de l'apprenant exprime les compétences associées à chaque item d'évaluation. L'historique de l'apprentissage stocke des informations sur l'ordre dans lequel les différents contenus pédagogiques ont été proposés ainsi que sur la manière dont les enchaînements ont pu être réalisés.

5.4 ECSAIWEB : Modèles de recouvrement multi-points de vue

5.4.1 Généralités

En conservant la présentation de niveau des connaissances proposée dans ECSAI (cf. section 2.2.2), nous utilisons dans ECSAIWEB alors un ensemble des items d'évaluation pour représenter le niveau des connaissances du domaine de l'apprenant. Les unités d'apprentissage sont décrites à l'aide de ces items, elles contiennent les informations qui vont permettre d'une part d'accéder aux fichiers physiques et d'autre part de calculer la prochaine unité à présenter. Le modèle d'élève doit être lié à ces connaissances conceptuelles déclaratives de domaine.

Nous proposons maintenant un modèle d'élève, reposant sur ce modèle des connaissances et présenté comme «un modèle de recouvrement multi-points de vue». Ses composants sont représentés dans la figure 5.2. Remarquons tout d'abord que nous avons juxtaposé deux modèles, l'un pour la navigation adaptative, l'autre pour le calcul d'un parcours guidé. Dans l'état actuel du système, ces deux modèles ne communiquent pas, mais leurs fondements communs permettent d'assurer une telle communication dans le futur. La partie commune concerne les informations de nature administrative et les objectifs pédagogiques.

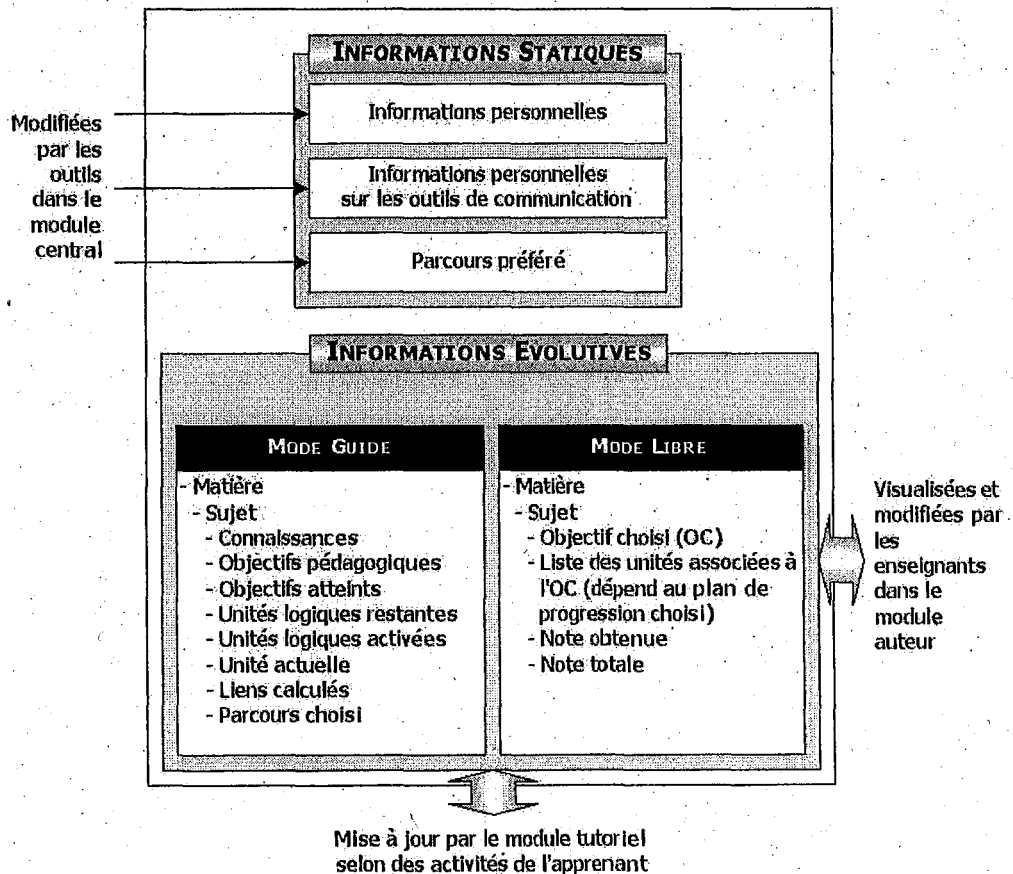


FIG. 5.2 – Modèle d'élève de ECSAIWEB

Les informations stockées dans notre modèle sont divisées en deux catégories : des informations statiques et des informations évolutives. Les informations que nous souhaitons stocker dans la partie statique sont les renseignements personnels. Ils correspondent à ce que l'enseignant souhaite connaître sur l'élève : nom, prénom, adresse, adresse électronique et etc. Ces renseignements sont fournis par l'élève au moment de l'inscription sur le système. Le système les utilise pour identifier l'apprenant lorsque l'apprenant veut entrer dans le système. Les informations évolutives sont séparées en deux parties, une pour chaque mode d'apprentissage proposé : guidé et libre [SG00c]. Ces informations sont stockées sous la forme de connaissances évolutives. Ces connaissances sont l'acquis de l'apprenant et l'historique de l'apprentissage. L'acquis de l'apprenant correspond à une évaluation portée sur le niveau de connaissances de l'élève qui est exprimé en terme de compétences associées à chacun des items d'évaluation. L'historique de l'apprentissage correspond, selon la matière donnée et le mode d'apprentissage choisi, à une évolution portée sur les objectifs pédagogiques devant être atteints, les unités logiques déjà activées, les parcours proposés et le parcours choisi représentés dans la figure 5.2.

De plus notre modèle d'élève permettra à l'apprenant au fur et à mesure de son parcours de savoir quels sont les objectifs pédagogiques déjà atteints, aussi bien en mode libre qu'en mode guidé. Ceci est réalisé par un système de présentation multi-points de vue (la partie «*visualisation et adaptation*» représentée dans la figure 5.3 et 5.5). Ce modèle soutient l'annotation adaptative des liens et l'ordonnement individuel de parcours d'apprentissage. Des liens qui sont montrés dans une vue d'ensemble à chaque page ou dans la table des matières sont visuellement annotés selon le bilan sur les items d'évaluation correspondant pour chaque élève.

5.4.2 Modèle pour le mode guidé

En mode guidé, le système propose un menu représentant une liste de sujets. Après le choix du sujet, le système prend la main et génère le parcours d'apprentissage pour chaque élève. Supposons que l'élève X choisisse un sujet à apprendre représenté dans la figure 5.3. Ce sujet est décrit par un objectif représenté par un niveaux sur deux items : $i1(a;1,2)$ et $i2(a;1,0)$ et fait appel aux unités logiques d'apprentissage (ULA) : U1, U2, U3 U4, U5 et U6. Les contenus des ULAs sont présentés dans la figure 5.4. Supposons que l'unité actuelle est U1. A la fin d'exécution de cette unité, le système sauvegarde les connaissances actuelles $i1(a;0,0)$, [«*la connaissance sur l'unité U1*»], sur le modèle de cet élève et calcule les prochaines unités candidates en utilisant les équations 4.10, 4.11, 4.12 et 4.15. Les unités candidates obtenues sont U2 (*lien «suivi-de(U1, U2)»*) et U4 (*lien «suivi-de(U1, U4)»*). Le système propose U2 comme prochaine unité (les algorithmes de cette décision seront présentés dans la section 6.5.2). Le système utilise l'écart entre les objectifs pédagogiques et les connaissances de cet apprenant pour aider à décider de continuer ou d'arrêter.

Le système, au fur et à mesure des activités accomplies par l'élève, décide de la prochaine unité à présenter en utilisant les paramètres de l'unité courante, le modèle d'élève et des critères additionnels. Les informations stockées dans ce modèle sont le sujet choisi, les connaissances représentées par l'ensemble des items, les liens générés pendant le parcours d'apprentissage, la branche choisie parmi les branches candidates, les objectifs pédagogiques.

5.4.3 Modèle pour le mode libre

En mode libre, le système permet à l'élève de choisir librement les items à apprendre. Les items sont présentés sur un menu adaptatif annotant le niveau choisi de la compétence sur ces items. Avec l'item choisi par exemple l'item $i1$, le système détermine des unités d'apprentissage à

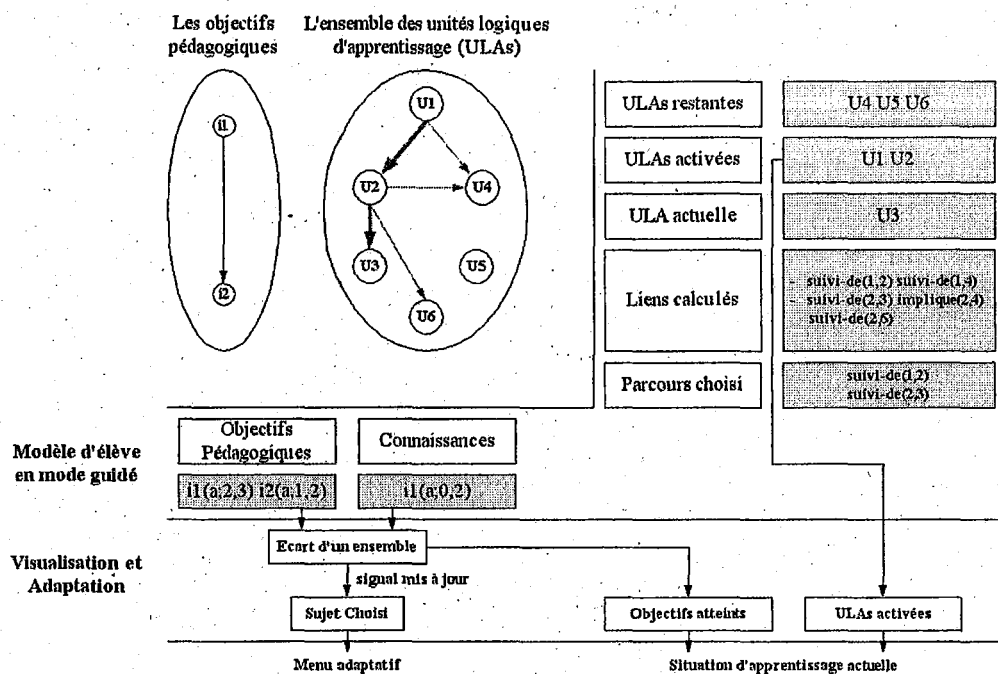


FIG. 5.3 – Modèle, Visualisation et Annotation du mode guidé.

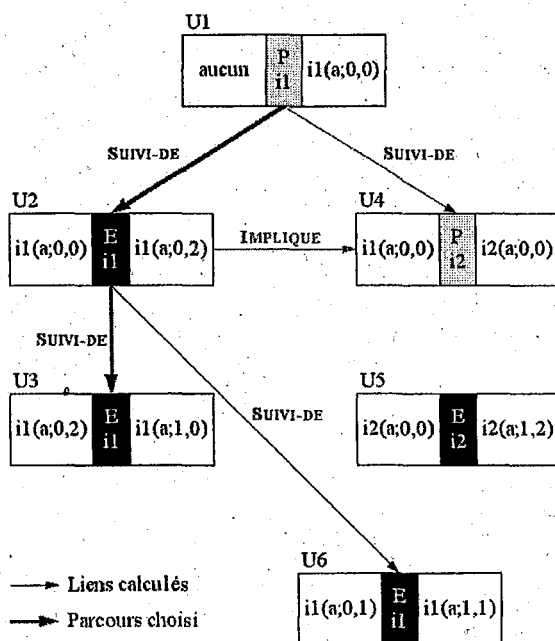


FIG. 5.4 – Exemple d'un parcours d'apprentissage en mode guidé de l'élève X.

présenter. Dans ce cas, nous obtenons la liste suivante : U1, U2, U3, et U9. Le système détaille la liste d'unités sur le plan progressif, selon les critères suivants : la présentation d'abord (U1, U2, U3, U6), l'ordre des liens (U1, U2, U3, U6 ou U1, U2, U6, U3), présentation seule (U1), exercice seul (U2, U3, U6). Le choix de ces unités est fait selon les items choisis par l'apprenant. A la fin du plan choisi, le système met à jour la table de matières et annote la table de matières pour montrer à l'élève l'état d'étude de chaque item (présenté dans la figure 5.5).

Ce modèle contient les objectifs pédagogiques choisis (représentés par les items, leur description et la note totale³⁶), la liste des unités logiques d'apprentissage associées à l'objectif choisi et les connaissances (représentées sur les notes obtenues sur chaque objectif).

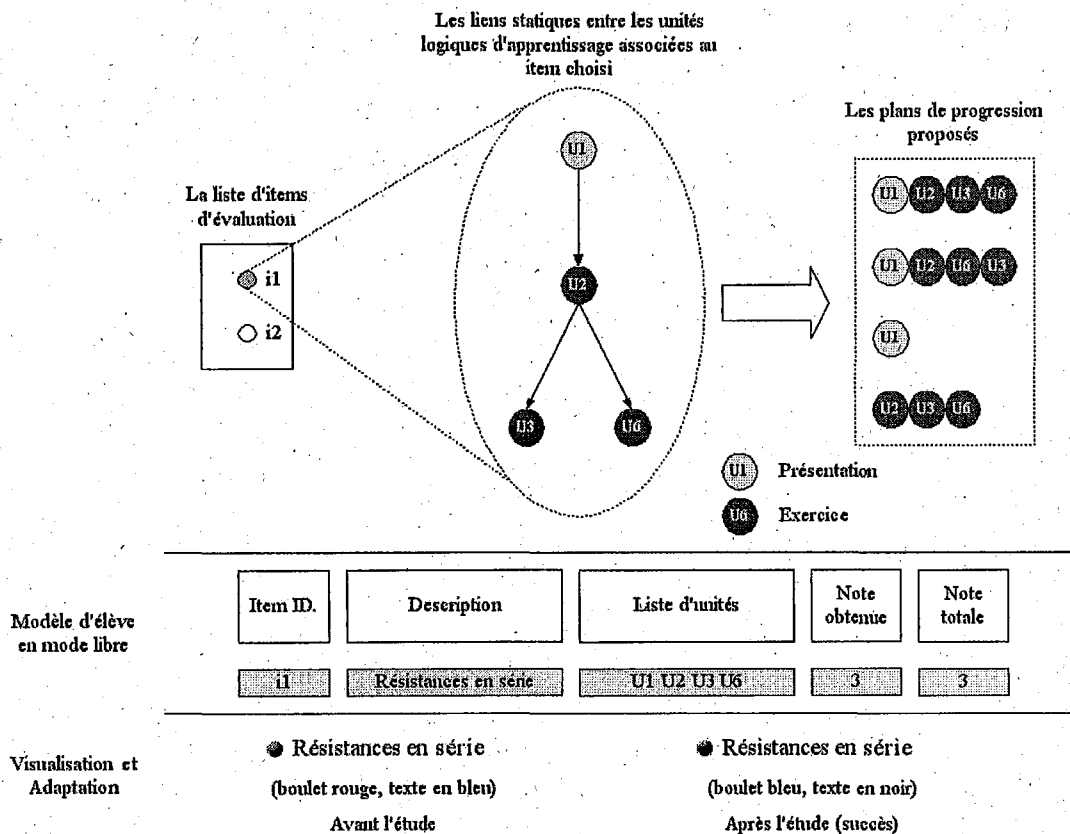


FIG. 5.5 – Modèle, Visualisation et Annotation des concepts en mode libre.

5.4.4 Visualisation et Adaptation

Dans ce paragraphe nous montrons comment les éléments du modèle d'élève permettent une présentation adaptative. Pour réaliser des adaptations différentes, nous proposons plusieurs points de vue sur les connaissances de l'apprenant.

- **Adaptation des menus en mode guidé :** Ceci se fait sur un ensemble d'objectifs pédagogiques de chaque sujet. Si tous les objectifs d'un sujet choisi sont atteints alors le système

36. La note totale est une somme entre la valeur de savoir-appliquer et de savoir-faire d'un item choisi.

désactive le lien-ancree sur le menu associé à ce sujet (représenté dans la figure 5.3). De plus, nous proposons le menu «*Résultat*» qui permet à l'apprenant de savoir; quels objectifs sont déjà atteints, quelles unités sont déjà activées. La technique d'annotation que nous voulons employer dans ce mode est «*désactivé*» ou «*activé*» d'hyperlien de la liste de sujets. Les sujets que l'élève a déjà passés sont «*désactivés*» donc l'élève ne peut pas choisir ce sujet ou même retravailler.

- **Adaptation des menus en mode libre:** Ceci correspond à un bilan sur chaque item de chaque apprenant. Ce sont les rapport entre les notes obtenues et les notes totales de chaque item. Les valeurs par défaut envisagées sont:

-	80% ≤	Notes obtenues		→	Très Bien
-	60% ≤	Notes obtenues	< 80%	→	Bien
-	40% ≤	Notes obtenues	< 60%	→	Passable
-	20% ≤	Notes obtenues	< 40%	→	Faible
-	0% ≤	Notes obtenues	< 20%	→	Très Faible
		(% par rapport à note totale)			

Ces points permettent à l'apprenant de savoir: quel est l'état actuel de son apprentissage, quels sont les objectifs satisfaits, quels objectifs doivent être retravaillés etc. Nous envisageons en plus que ce jugement peut être fait de la part de son propre enseignant.

5.5 Conclusion et Evolution

Notre modèle apporte donc les informations dont nous avons besoin sur des apprenants. Ces informations visualisent les états d'apprentissage de chaque apprenant d'une part et servent à calculer le parcours d'apprentissage d'autre part. Elles sont séparées en deux parties: statique et dynamique. Nous pensons évidemment à une fusion des informations utilisées en mode libre et en mode guidé. Cependant une telle fusion impliquerait un calcul beaucoup plus complexe du guidage et il n'est pas évident qu'on y gagnerait sur la plan pédagogique. Avant d'envisager un tel travail, il importe de recueillir des données en expérimentant le système dans sa version actuelle, en recueillant des avis des apprenants et des formateurs.

Chapitre 6

Fonctionnalités proposées pour ECSAIWEB

6.1 Introduction

Dans les chapitres précédents nous avons spécifié notre système d'apprentissage et choisi les modèles pour représenter les informations nécessaires. Dans ce chapitre, nous présenterons les principes retenus pour les interfaces (section 6.2) et les fonctionnalités du logiciel qui permettent d'atteindre les objectifs fixés, à savoir : permettre la construction de matériaux pédagogiques dans la matière choisie (environnement auteur, section 6.4), permettre l'exploration de ces matériaux pédagogiques (environnement élève, section 6.5) et favoriser l'observation, l'utilisation de ces matériaux qui permettent aux enseignants-auteurs de les améliorer (gestion des profils des élèves, section 6.4.9), et de partager les savoirs entre les utilisateurs (environnement partagé, section 6.6).

Rappelons que ce logiciel est de type client-serveur et indépendant d'une plateforme. Nous avons essayé de respecter les directives des navigateurs Web concernant la création d'une interface graphique : conserver une homogénéité dans les différents éléments de l'interface facilite la prise en main de l'application par tout nouvel utilisateur.

6.2 Interfaces graphiques adaptatives d'ECSAIWEB

Toutes les fonctionnalités seront disponibles via une interface graphique que nous voulons présenter. Quoique WWW soit souvent désigné sous le nom de son visionneur, il peut mieux être caractérisé comme combinaison d'un protocole de client-serveur (le protocole de transfert d'HyperText - HTTP) et d'un langage HTML. Le protocole indique comment les programmes aux deux extrémités d'une ligne de communication (un client et un serveur) devraient communiquer l'un avec l'autre, et le langage HTML indique l'ensemble des primitives d'affichage que le navigateur (client) devrait pouvoir manipuler (voir la figure 6.1). Parmi les logiciels disponibles le communicateur de Netscape et l'explorateur de Microsoft sont probablement les plus connus en raison des supports multimédia et gratuit. Mais nous avons choisi plutôt le navigateur de WWW chez Netscape parce que, pour le moment, il nous semble que ce navigateur peut être exploité dans plusieurs systèmes.

Pour utiliser ce logiciel, il faut avoir un navigateur Web, précisément le navigateur Communicateur version 4 ou plus. La compatibilité entre les navigateurs du Web pose un grand nombre de difficultés de l'implantation. Nous avons donc décidé de prendre en compte Communicateur

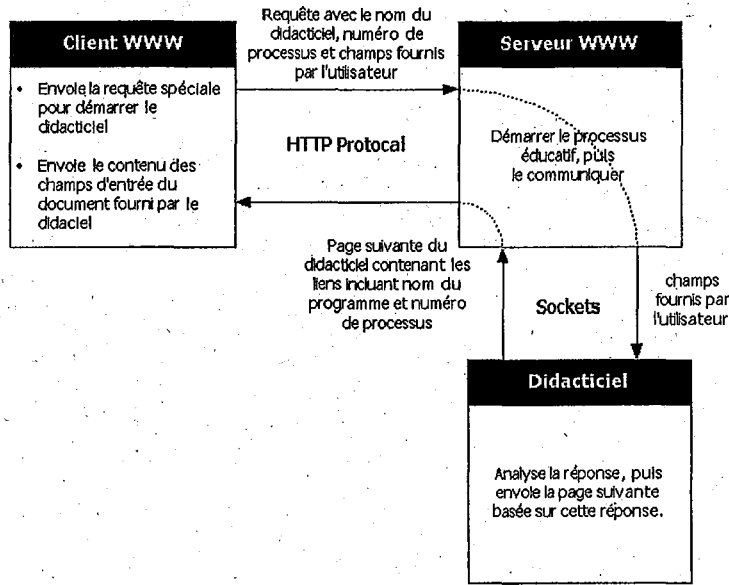


FIG. 6.1 – Schéma d'interaction entre les clients et le didacticiel par l'intermédiaire du WWW



FIG. 6.2 – Page d'accueil du système ECSAIWEB

comme le navigateur standard de notre système. Toutes les interfaces et les fonctionnalités sont basées sur ce type de navigateur.

ECSAIWEB fournit à l'utilisateur (enseignant ou enseignant-auteur ou élève) des outils lui permettant la liberté d'exploration de l'univers qui lui convient. Il y a deux choix quant au lancement du logiciel : il faut choisir le langage d'interface et puis le système demande le nom d'utilisateur et le mot de passe pour accéder à l'environnement choisi. Les langues que nous avons choisies sont la langue française, thaïe et anglaise (situées dans la partie haute de la figure 6.2). Chaque langue a les mêmes fonctionnalités y compris pour les matériaux pédagogiques qui sont créés dans chaque langue.

6.2.1 Type d'interfaces

Les interfaces, selon la nature de navigateur de Web, sont présentées sous la forme de barre de menu (figure 6.3.a), de menus POP-UP (figure 6.3.b), de liens d'hypertexte (figure 6.3.c), de boutons (figure 6.3.d), et de formulaires (figure 6.3.d). Selon la capacité du navigateur à exécuter les scripts java et les applets de Java, nous avons essayé de créer quelques interfaces utilisant ce type de programmation. De plus, nous avons pu utiliser les couleurs, les polices en appliquant les feuilles de style ou CSS³⁷ disponibles depuis la version 4 de communicateur pour changer la taille et la couleur de la police, la couleur du fond et etc. sans ajouter aucun élément dans le source original. Par exemple, si le source original utilise la balise en tête niveau 1 (<H1>..</H1>), sa taille, sa couleur et sa mise en forme devront être changées automatiquement, selon la description dans la feuille de style (cf. section 6.2.3.0). Pour chaque type d'interface nous présentons des principes, un exemple et la mise en œuvre de l'adaptation.

Liens d'hypertexte

Une des fonctionnalités la plus utile du langage HTML est la possibilité de créer des liens vers d'autres documents. Ces documents peuvent être des documents HTML, des images, du son et/ou des films, disponibles au poste de travail ou accédés via des serveurs telnet³⁸, WAIS³⁹, FTP et GOPHER⁴⁰. Le logiciel client WWW présente ce lien sous forme en général de mots soulignés ou d'image encadrée que l'on appelle **ancres** ou **liens** (zone d'image de ECSAIWEBV1.0, figure 6.4). Ce lien est soit un fichier local, soit une URL⁴¹. Une URL est le moyen de nommer un objet dans le monde WWW. La syntaxe d'une URL est la suivante :

`type://serveur:port/chemin_d'accès/fichier#etiquette?parametres`

Le type de serveur peut être :

- http : pour les URL provenant de serveurs WWW,
- gopher : pour les serveurs GOPHER,
- ftp : pour les fichiers à transférer,
- telnet : pour ouvrir une session interactive,

37. sigle anglais de *Cascading Style Sheets*

38. telnet - Protocole d'émulation de terminal qui permet à l'internaute d'entrer en communication avec un hôte Internet à partir de son propre ordinateur pour utiliser des programmes ou consulter des données qui y sont stockées.

39. sigle anglais de *Wide Area Information System*

40. Gopher - Système basé sur une structure de menus, qui permet la recherche d'information dans Internet, l'accès à cette information et sa visualisation.

41. sigle anglais de *Unified Resource Locator*

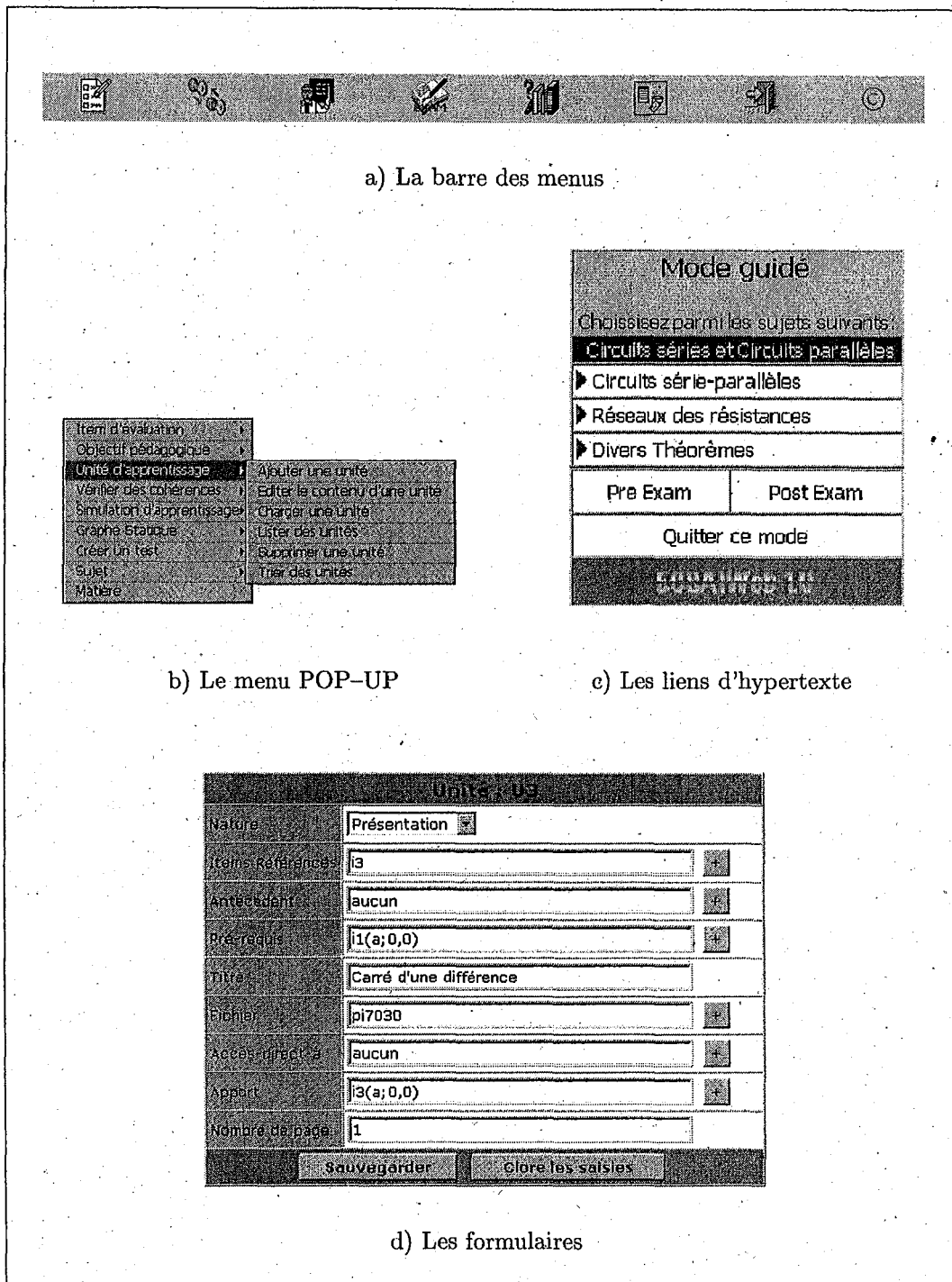


FIG. 6.3 – Exemples d'interfaces utilisées dans ECSAIWEB

- wais : pour interroger une base WAIS,
- news : pour accéder aux forums.

Dans notre cas, nous l'utilisons que le type HTTP. Les hyperliens peuvent exister dans certains documents fournis à l'élève. Nous distinguons 3 catégories dans hyperliens. La première catégorie, ces liens sont créés par l'auteur de cours incluant dans les documents qui ne seront pas modifier. La deuxième catégorie est automatiquement créée pendant le parcours d'apprentissage. Ces liens sont créés et mis dans les documents qui permettent à l'élève de suivre son parcours d'apprentissage proposé d'une part. D'autre part, ils permettent de faire apparaître de partie cachée dans les documents fournis à l'élève comme la présentation d'un exemple, après avoir présenté un énoncé. Le lien vers l'étape suivante est automatiquement créée (voir la description de la structure d'UPA du type Présentation exemple, section 4.4.1.c). La troisième catégorie d'hyperliens est créée par le système pour créer les menus incluant les commandes, le lien vers d'autre outils comme représenté dans la figure 6.4.

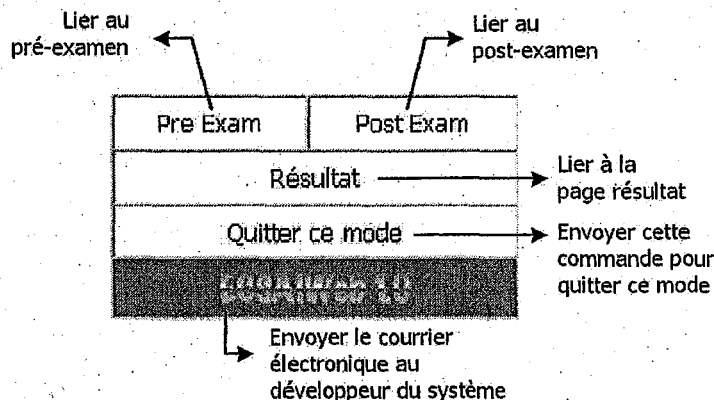


FIG. 6.4 – Liens d'hypertexte

De plus en appliquant la technique d'annotation, nous pouvons augmenter la possibilité de présentation de ces liens en plusieurs étapes, selon les comportements d'un élève. Par exemple, si l'élève choisit un sujet à apprendre et puis le suit jusqu'à la fin le parcours d'apprentissage, le système peut alors évaluer l'adaptation à cet élève en utilisant son résultat sur le sujet, et annoter les liens (appelés le menu adaptatif) sur ce sujet, représenté dans la figure 6.5. Donc nous pouvons juger chaque élève sur chaque objectif pédagogique à plusieurs niveaux (valeur par défaut) :

- | | | |
|-----------------|------------------------|--|
| • Très bien | : 80% ≤ résultat | → texte en noir et cliquable, boulet bleu |
| • Bien | : 80% < résultat ≤ 60% | → texte en bleu et cliquable, boulet cyan |
| • Passable | : 60% < résultat ≤ 40% | → texte en bleu et cliquable, boulet verre |
| • Faible | : 40% < résultat ≤ 20% | → texte en bleu et cliquable, boulet violet |
| • Très faible | : 20% < résultat | → texte en bleu et cliquable, boulet rouge |
| • Sans exercice | : - | → texte en noir et in cliquable, boulet noir |

Nous pensons également que ce jugement peut gêner certains enseignants alors il faut obligatoirement permettre à ces enseignants de juger eux-mêmes leurs élèves. Donc au début de la création de chaque matière, le système ECSAIWEB offre un outil administratif pour déclarer les niveaux de jugement de chaque matière.

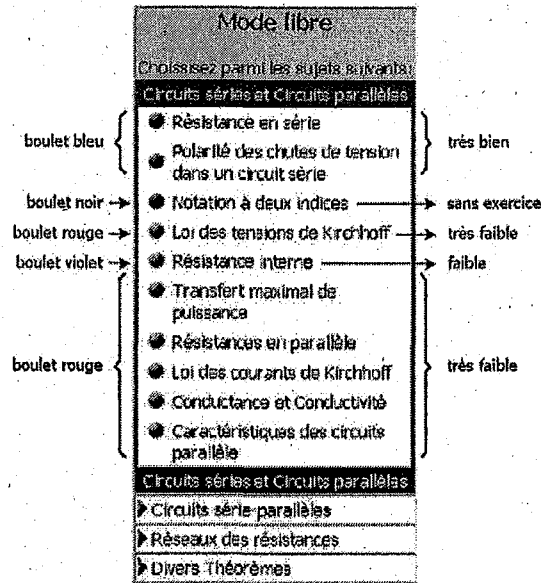


FIG. 6.5 – Menu adaptatif

6.2.2 Formulaires

Nous utilisons beaucoup ces supports pour la même raison que des liens d'hypertextes, de plus les utilisateurs peuvent ajouter eux-mêmes certains paramètres en utilisant un formulaire. Un formulaire est une fiche que l'utilisateur doit remplir, les informations ainsi saisies sont traitées par le serveur WWW à l'aide d'un script CGI [Tea]. La méthode que nous voulons utiliser est présentée dans les paragraphes suivants. L'exemple de l'utilisation de formulaires est présentée dans la figure 6.3.d.

Un formulaire commence et finit par les balises `<FORM ACTION METHOD>` et `</FORM>` respectivement. Le paramètre ACTION doit être un programme exécutable ou shell que le HTTPD doit appeler et exécuter sur le serveur. Le paramètre METHOD est une façon de passer des requêtes spéciales pour démarrer le didacticiel et des champs fournis par les utilisateurs au serveur WWW. Il existe donc deux méthodes de le faire: GET ou POST. Dans notre cas, nous utilisons plutôt la méthode POST parce que le nombre des informations en octet passant au serveur est illimité tandis qu'avec la méthode GET le nombre est limité à 1024 octets. Nous utilisons les champs suivants dans formulaire:

- **Bouton :**
 - RADIO pour les boutons multiples
 - CHECKBOX pour les cases à cocher
 - SUBMIT pour envoyer le formulaire une fois rempli
 - RESET pour effacer le contenu du formulaire
- **Texte libre sur une ligne du type :**
 - TEXT pour faire saisir de texte
 - PASSWORD les caractères saisis sont représentés par des '*'
 - HIDDEN le champ est caché: utile pour transmettre des informations non visibles.

- Texte libre sur plusieurs lignes : TEXTAREA
- Menu : SELECT

Les formulaires ne font pas l'objet d'adaptation, seuls les liens qu'ils contiennent peuvent être adaptés.

Barre des menus

Cette technique est une combinaison entre l'hypertexte et Javascript⁴². Nous créons le lien d'hypertexte et puis utilisons le Javascript pour détecter le mouvement d'une souris : les actions «*onMouseOver*», passant la souris sur le lien, et «*onMouseOut*», ayant la souris hors le lien. Avec ces actions, nous pouvons manipuler les images pour présenter les informations supplémentaires sur les liens (représenté dans la figure 6.6). Ces menus peuvent être adaptés selon le groupe d'utilisateurs : étudiants ou enseignants-auteurs.

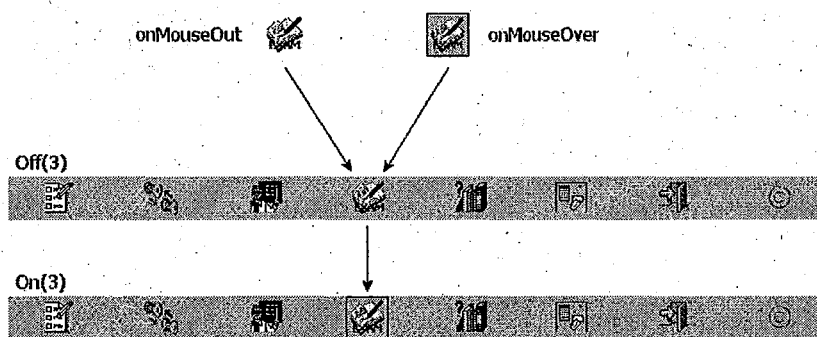


FIG. 6.6 – Processus de la barre des menus

6.2.3 Outils de gestion d'interfaces adaptatives

Nous utilisons deux outils pour gérer les interfaces adaptatives : les feuilles de style et la HTML dynamique.

Feuilles de style – CSS

Les feuilles de style CSS [W3Ca] offrent un nouvel outil puissant aux développeurs de Web. Il aide à simplifier la tâche complexe de maintien des sites de Web. Il fournit un grand nouveau tracé et conçoit caractéristiques pour des pages de Web sans le besoin de plug-ins, de long durée de transfert et d'outils coûteux [San99].

Les feuilles de style font partie des nouveautés importantes du langage HTML. Elles permettront :

- de changer l'aspect d'une page HTML sans modifier son contenu,
- de disposer plusieurs présentations différentes pour une même page Web,
- des modifications étendues de tout un site Web puisque leur principal intérêt réside dans le fait qu'elles peuvent être déclarées à l'extérieur d'un document HTML.

42. Exemple de langage de script chez Sun micro system

Nous employons cette technique pour que l'ensemble de nos pages reste identique : la taille de police, la couleur de polices, les couleurs de chaque ligne du tableau, et etc. Les enseignants-auteurs n'ont plus besoin d'ajouter les couleurs, la taille de police et etc. Cela nous assure que la nouvelle page venant d'être ajoutée sera identique celles existantes, autrement dit homogène. L'exemple dans la figure 6.7 représente une partie du style que nous souhaitons utiliser. Si l'enseignant-auteur utilise la balise `<H1>... </H1>`, le navigateur prend en compte immédiatement en changeant le style de H1 avec celui représenté dans la figure 6.7.

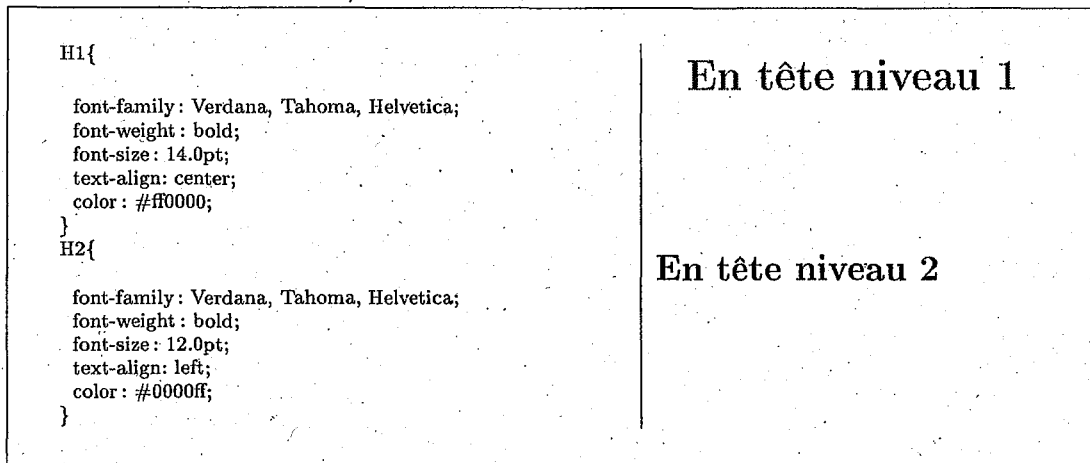


FIG. 6.7 – Exemple de feuille de styles utilisé dans ECSAIWEB

HTML Dynamique

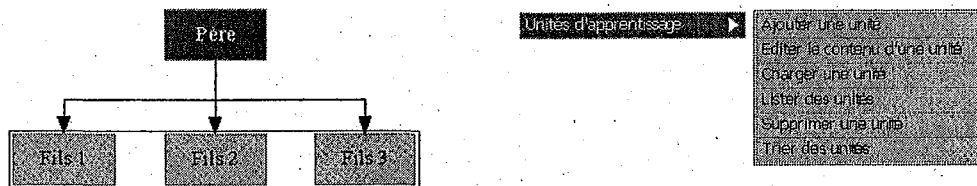
Depuis la version 4.0, le communicateur de Netscape et l'explorateur de Microsoft ont intégrés le DHTML. En fait, c'est un grand mot pour des choses que nous avons déjà. Il s'agit du regroupement de trois éléments : le HTML[W3Cc], la possibilité d'utiliser les feuilles de styles[W3Ca], et ce que nous appelons le DOM⁴³ [W3Cb]. Ce dernier correspond à la représentation du document HTML en une hiérarchie d'objets (au sens de la programmation orientée objet). Pour manipuler ce DOM, il nous faut donc un langage adapté : Javascript.

Le positionnement dans un document peut se faire à l'aide des définitions de feuilles de style comme nous l'avons vu (cf. section 6.2.3.0). Il peut en outre être mis en œuvre par une nouvelle balise nommée «*LAYER*». Cette balise s'utilise dans le corps d'une page HTML, c'est-à-dire, entre le couple de balise `<BODY>` et `</BODY>`. Cette balise permet de positionner une partie de code HTML et elle accepte en conséquence des arguments suivants [Mai] :

- **ID** est l'identifiant de la balise *LAYER*
- **TOP** est son positionnement en pixels par rapport au haut de la fenêtre
- **LEFT** est son positionnement en pixels par rapport au bord gauche de la fenêtre
- **WIDTH** est sa largeur en pixels (`WIDTH="10"`) ou en pourcentage (`WIDTH="10%"`)
- **HEIGHT** est sa hauteur en pixels (`HEIGHT="10"`) ou en pourcentage (`HEIGHT="10%"`)
- **POSITION** s'applique uniquement au *LAYERS* et non aux feuilles de style. La valeur de **POSITION** peut alors être relative ou absolue.

43. sigle anglais de *Document Object Model*

- **PAGEX** séparation en pixels entre le layer et le reste du document (horizontalement)
- **PAGEY** séparation en pixels entre le layer et le reste du document (verticalement)
- **SRC** donne le nom du fichier qui contient le code HTML à insérer dans le LAYER. Ceci permet par exemple de concaténer plusieurs fichiers HTML les uns à la suite des autres.



a) La hiérarchie du menu

b) L'exemple de menu

FIG. 6.8 – Menu POP-UP utilisant le DHTML

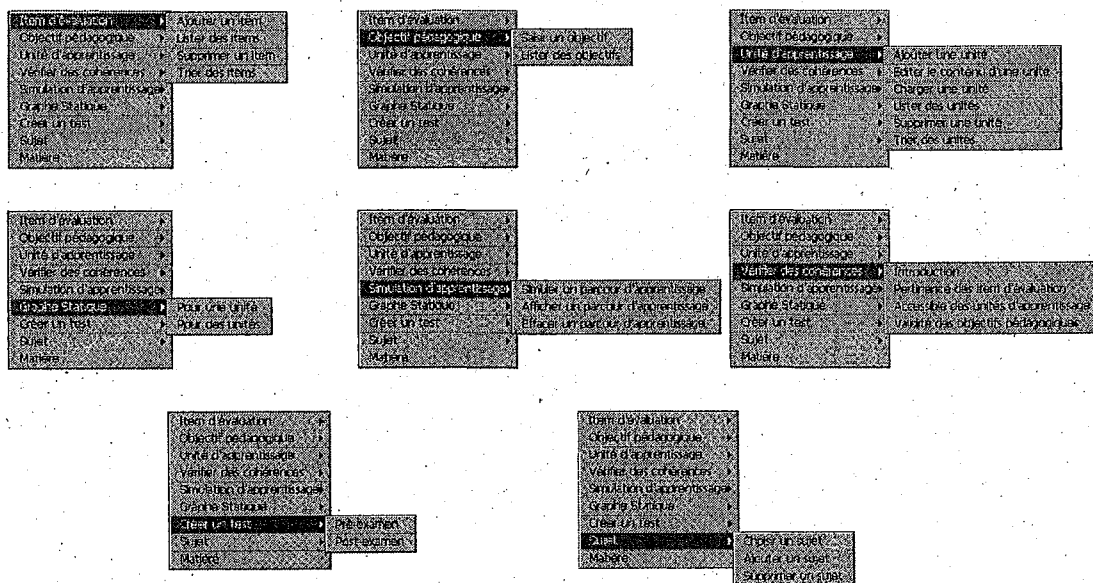


FIG. 6.9 – L'ensemble des menus pop-up

- **Z-INDEX** définit la position de la couche sur la pile des couches, autrement dit son ordre. Sa valeur est un entier positif.
- **ABOVE** donne le nom du layer précédent dans la pile des couches.
- **BELOW** donne le nom du layer suivant dans la pile des couches.
- **CLIP="x1,y1,x2,y2"** définit le rectangle en pixel servant à afficher le code HTML. Le code HTML dépassant la taille de ce rectangle n'est pas affiché. Si les deux premières valeurs sont omises, elles sont remplacées par 0. Les valeurs ne doivent pas contenir de caractère blanc. Si aucune valeur n'est spécifiée les valeurs HEIGHT et WITDTH sont prises en compte. Quand l'argument CLIP n'est pas présent, le code HTML est affiché dans tout le rectangle, si CLIP est spécifié le code HTML correspondant à la partie visible est affiché.

- **VISIBILITY=HIDE|SHOW|INHERIT** signifie que le layer est masqué, visible ou qu'il hérite de l'attribut de son layer parent.
- **BGCOLOR** est sa couleur de fond exprimée en nom de couleur (bleu) ou en valeur hexadécimale.
- **BACKGROUND** donne l'image utilisée pour la définition de la couleur du fond de page du Layer
- **OnMouseOver** active les commandes du Javascript quand on passe le souris au-dessus.
- **OnMouseOut** active les commandes du Javascript quand, pour le moment le souris est sur le lien, on passe le souris d'ailleurs.
- **OnFocus** : l'utilisateur clique sur une zone.
- **OnBlur** : l'utilisateur après avoir cliqué sur une zone clique à l'extérieur de la zone active.
- **OnLoad** : l'utilisateur charge la page dans le navigateur.

Ce type du modèle nous permet de représenter un menu POP-UP en une hiérarchie d'objets: objet père et sous-objets fils, comme dans l'exemple représenté dans la figure 6.3.b. Chaque père ou fils associé à un objet correspond aux différents niveaux de présentation. Par exemple, le père 2 a 3 fils (figure 6.8.a). Nous les regroupons en deux parties, pour faciliter la présentation. En utilisant la technique du DHTML, nous choisissons de présenter chaque objet par une couche dans la page Web et puis de détecter les événements de souris pour cacher ou afficher les couches. Par exemple, la figure 6.8.b représente l'ensemble des couches fils s'affichant quand nous avons passé la souris sur leur couche père. La figure 6.9 présente l'ensemble des menus POP-UP utilisés dans l'environnement auteur.

6.3 Description fonctionnelle de l'environnement

ECSAIWEB se compose d'un univers d'apprentissage qui réside sur une machine serveur. Il est accessible par l'intermédiaire d'un client du type navigateur Web. Lors de la connexion, le système identifie chaque utilisateur par son nom et son mot de passe, et lui fournit automatiquement un environnement adapté à son statut : celui d'enseignant, dont l'objectif est de transmettre de la connaissance ou celui d'étudiant de l'acquérir. ECSAIWEB se compose de trois parties (figure 6.10), chaque partie est associée aux environnements suivants : l'environnement auteur (module d'auteur), l'environnement élève (module tutoriel), et l'environnement partagé (page d'accueil, module central, module de communication et module de administration). Nous décrivons dans les sections suivantes les principales fonctionnalités dans chaque environnement.

Pour une description exhaustive, nous renvoyons le lecteur au mode d'emploi du logiciel [San99].

6.4 Fonctionnalités de l'environnement enseignant-auteur

L'activité d'enseignement renvoie au problème de la transmission de contenus dans une discipline donnée, à la méthode de transmission pour mettre en œuvre ces contenus et à l'évaluation de l'élève sur ses facultés à mémoriser, appliquer et maîtriser les connaissances qui lui ont été transmises. La figure 6.11 présente l'espace de travail pour le groupe des enseignants-auteurs.

ECSAIWEB, grâce aux fonctionnalités d'édition, permet aux enseignants-auteurs d'ajouter ou d'améliorer le vivier des connaissances : le sujet, les items d'évaluation, les objectifs pédagogiques et les unités. Autour de l'existence d'une base d'unités, organisée par domaine et à travers

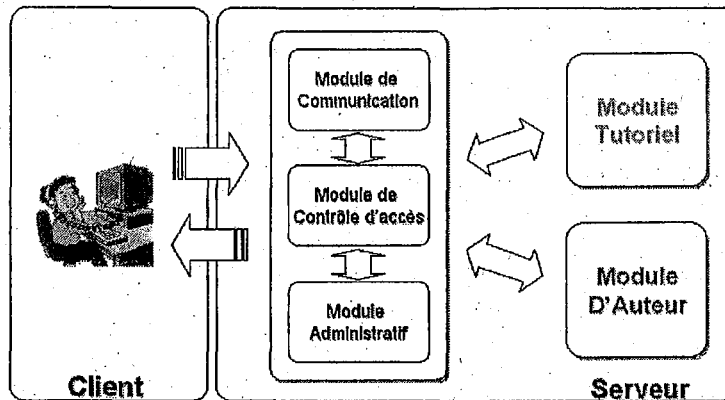


FIG. 6.10 - Organigramme du système ECSAIWEB

L'utilisation de formulaires appropriés, l'enseignant-auteur peut stocker des documents (texte, graphique et son). Dès lorsqu'un document est stocké dans la base d'unités, il peut être consulté et utilisé par tous les enseignants ayant les droits d'accès nécessaires. Cet environnement est employé par les enseignants-auteurs pour définir des objectifs pédagogiques, l'univers des unités d'apprentissage (matériaux pédagogiques), et offre des outils pour vérifier la cohérence de l'univers d'apprentissage et pour simuler le parcours d'apprentissage. Pour développer les matériaux pédagogiques, nous avons choisi «*Analyse de circuit électrique pour le courant continu*» comme domaine pour expérimenter. Nous proposons un processus de développement de cours en 5 étapes :

- définition des items d'évaluation,
- détermination des objectifs pédagogiques,
- description de l'univers des unités d'apprentissage,
- vérification de la cohérence de l'univers d'apprentissage, et
- création des examens (pré-examen et post-examen).

A propos des états de développement d'un cours, l'environnement doit offrir des outils pour aider les enseignants-auteurs à créer facilement leur cours sur le Web. Les outils que nous proposons dans cet environnement sont divisés en six catégories (représentées dans la figure 6.12) : l'outil de création d'un univers d'apprentissage, l'outil de visualisation de l'univers d'apprentissage, l'outil de contrôle et simulation, l'outil de création de l'examen, l'outil de gestion de fichier et l'outil de visualisation et gestion de profils d'élève. Les quatre premiers outils vous permettent d'intégrer ces deux aspects : présentation et évaluation. Les deux aspects peuvent être séparés en trois activités d'enseignement : présentation des contenus, exercices et examens. Les deux premières activités sont générées par l'outil de création d'un univers d'apprentissage, l'outil de visualisation de l'univers d'apprentissage et l'outil de contrôle et simulation. La troisième activité est générée par l'outil de création des examens. De plus le système propose un outil qui permet aux enseignants-auteurs, qui ont déjà créé leurs propres cours en page HTML, de convertir ces fichiers en format de présentation de ce système ou dupliquer ou modifier les fichiers. Enfin le dernier outil est celui de la visualisation et gestion des profils d'élève. Les fonctionnalités de tous les outils sont représentées dans la figure 6.13.

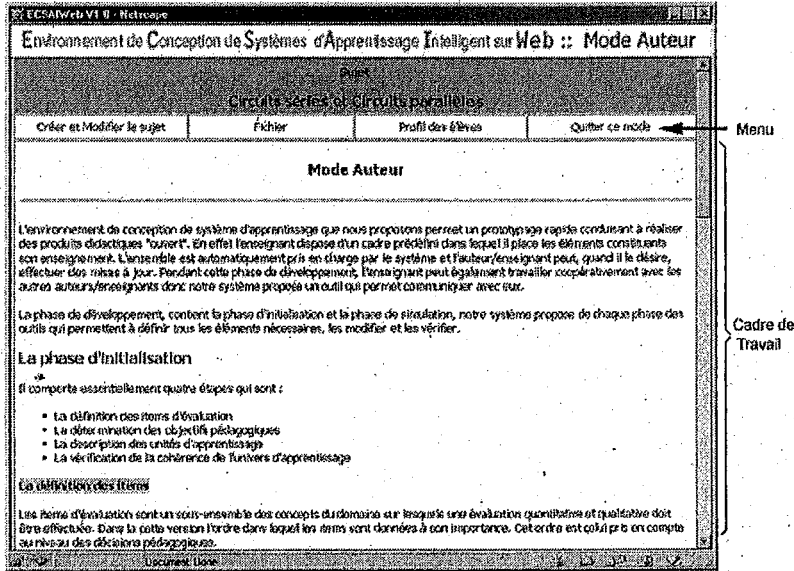


FIG. 6.11 – Espace de travail d'enseignants-auteurs

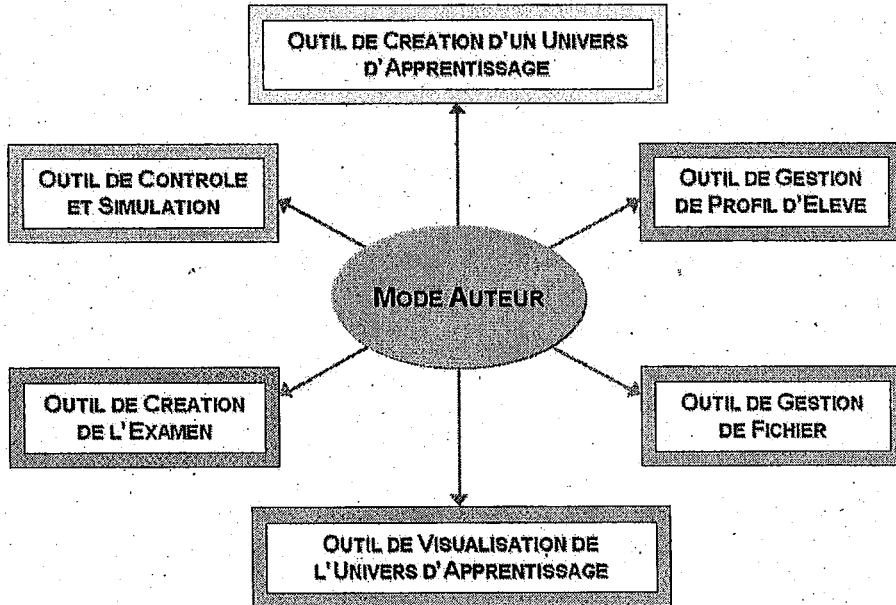


FIG. 6.12 – Six catégories d'outils dans l'environnement auteur

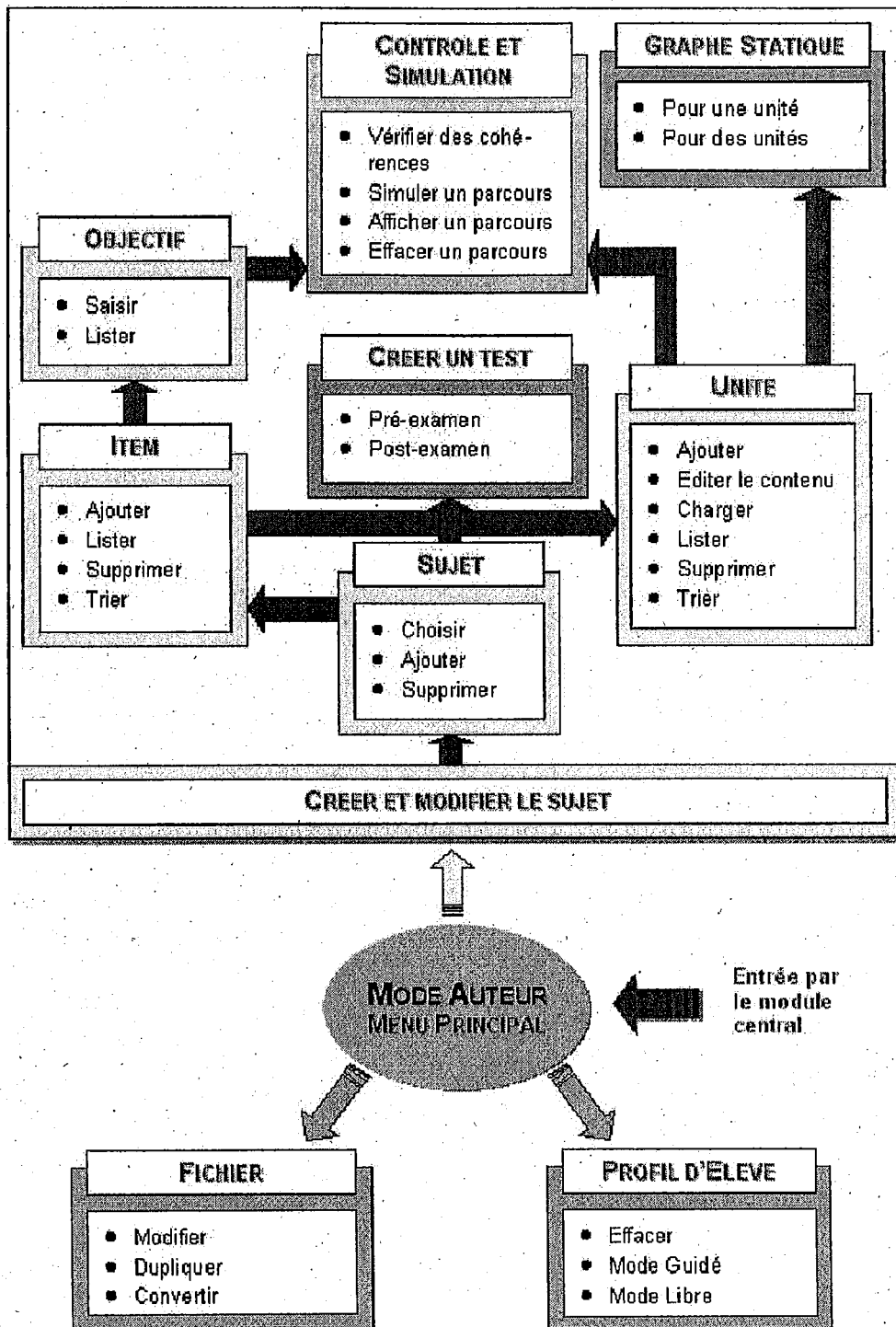


FIG. 6.13 – Organigramme des fonctionnalités dans l'environnement auteur

6.4.1 Définition des items d'évaluation

La première phase du développement est de définir les items d'évaluation. Ils représentent les concepts du domaine. Dans la version actuelle, l'ordre des items dans cette liste est important parce que l'environnement d'élève emploiera ce critère parmi d'autres pour choisir la prochaine unité à présenter. Ces items sont toujours accompagnés de deux valeurs : mesure globale et degré d'acquisition [Gav91][SG00a] qui présentent les aptitudes de l'élève et qui s'appelleront «*la compétence*» de cet item d'évaluation. La première valeur est une valeur quantitative, appelée mesure globale. Cette valeur permet de savoir si l'item a fait l'objet d'une présentation quelconque au cours de l'apprentissage et de situer un apport de connaissance dans le temps par rapport à la session courante. La deuxième est la valeur qualitative, appelée degré d'acquisition. Cette valeur précise la manière dont l'élève maîtrise le savoir–appliquer ou le savoir–faire relatif à l'item. Le degré d'acquisition est un couple d'entiers qui comptabilisent des jetons que nous avons attribués ou prélevés au niveau de connaissances de l'élève à la fin d'exécution d'un exercice. Le premier d'entre eux correspond à l'aspect application/restitution (savoir–appliquer) de l'exercice et l'autre à l'aspect manipulation/utilisation (savoir–faire). Pour définir les items d'évaluation, le système offre un cadre de définition, représenté dans la figure 6.14 à gauche.

Figure 6.14 est composée de deux interfaces utilisateur. À gauche, une interface pour saisir des items d'évaluation. Elle contient un tableau avec deux colonnes : 'Item' et 'Définition'. Les données sont : Item 1, Définition 'Différence de deux carrés'; Item 3, Définition 'Carré d'une différence'. En dessous, il y a des champs pour saisir un item (contenant '2') et une définition (contenant 'Carré d'une somme'). Des boutons 'Sauvegarder' et 'Fermer les saisies' sont également présents.

À droite, une interface pour saisir des objectifs pédagogiques. Elle commence par 'La liste des objectifs:' suivie de boutons '+', '-', et 'OK'. En dessous, un tableau résume les objectifs :

Objectif (n)	Définition	Mesure Globale	Degré d'acquisition	
			Savoir Appliquer	Savoir Faire
1	Différence de deux carrés	Abordé	2	0

Un menu contextuel est ouvert sur le bouton 'Abordé', montrant les options : 'Aquis', 'Vu', 'Inconnu', et 'None'. Des boutons 'Sauvegarder' et 'OK' sont situés en bas.


FIG. 6.14 – Saisie des items d'évaluation (à gauche) et des objectifs pédagogiques (à droite)

6.4.2 Détermination des objectifs pédagogiques

Les objectifs pédagogiques décrivent le niveau de connaissances dont l'élève doit se rapprocher. Chaque objectif est représenté par un ensemble d'items et regroupe les compétences relatives à chacun des items. Plus un item a de l'importance dans l'apprentissage et plus le nombre de jetons à affecter à cet item sera élevé. Les objectifs pédagogiques sont utilisés par le module tutoriel pour sélectionner la prochaine activité à suggérer à l'apprenant. Un objectif pédagogique est dit «*atteint*» si la compétence de l'élève sur les items concernés est au moins égale à celle exprimée dans les objectifs. Les objectifs pédagogiques sont atteints si chaque objectif pédagogique est lui-même atteint. La détermination des objectifs pédagogiques constitue une partie centrale de la définition du système d'apprentissage et elle se fait de façon interactive comme représentée sur la figure 6.14 à droite.

6.4.3 Description des unités logiques d'apprentissage

La saisie des divers éléments composant une unité d'apprentissage (pré-conditions, contenu et post-conditions) est également contrôlée par le système. La figure 6.15 montre comment les enseignants–auteurs ou les concepteurs de cours peuvent régler des paramètres pour l'unité U3 (la partie gauche de la figure 6.15). Cette unité est de type présentation, elle traite du concept

i3 et le fichier contenant l'unité physique d'apprentissage est pi7030. Il paraît difficile de remplir tous les paramètres nécessaires dans ce cadre. Alors nous avons dû ajouter d'autres fonctions pour aider les enseignants-auteurs à remplir ces paramètres facilement. En cliquant sur le bouton , le système ouvre la nouvelle fenêtre, représentée dans la figure 6.15 à droite, et fournit un cadre pour définir chaque paramètre.

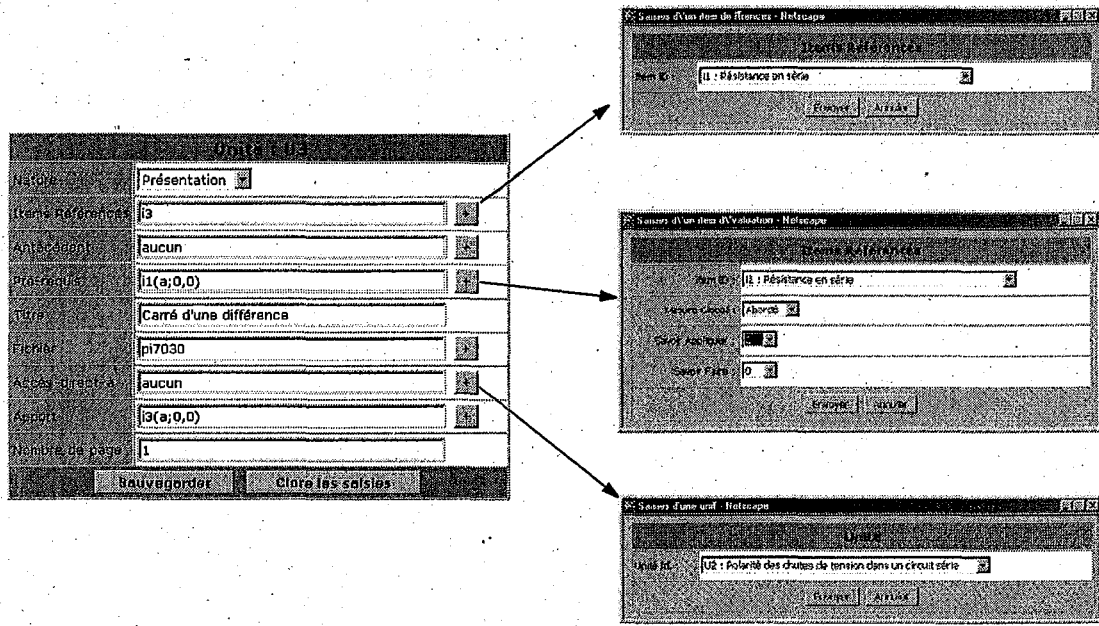


FIG. 6.15 – Description de l'unité logique d'apprentissage U3.

6.4.4 Edition des unités physiques d'apprentissage

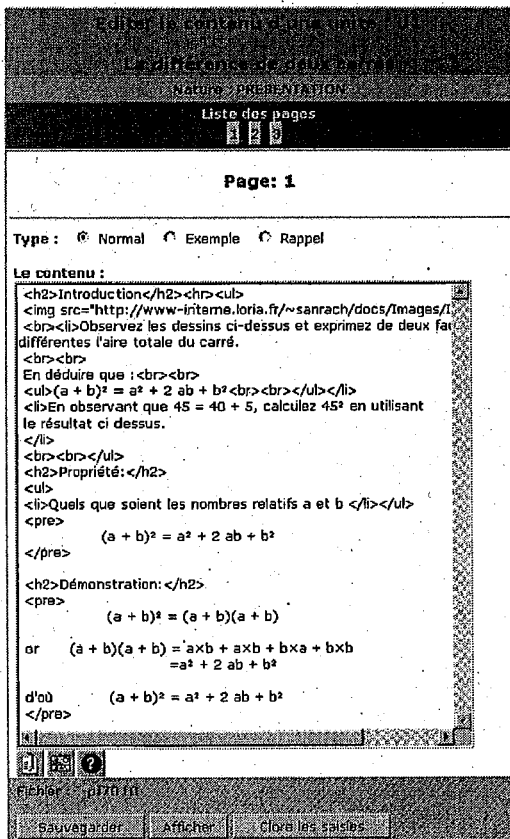
Après la description des unités logiques d'apprentissage, nous devons construire les unités physiques qui doivent être présentées à l'élève. Ces unités sont des fichiers reliés à la partie contenue de la description d'une unité logique, plus précisément la rubrique «*fichier*» de la figure 6.15.

La fonction d'éditeur d'un contenu d'une unité dépend de la nature d'unité, soit la présentation ou soit l'exercice et chaque nature est liée à une activité pédagogique [San99]: donner les informations normales, donner les exemples, rappeler les connaissances ou bien donner les exercices. Chaque activité est différente l'une de l'autre, donc les informations stockées dans chaque fichier devront être différentes. Nous commençons par l'unité de type présentation.

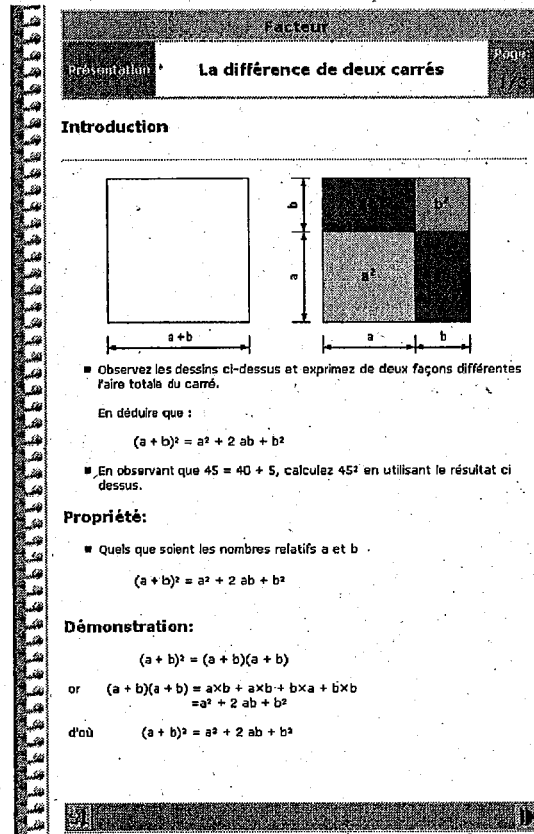
La présentation

L'éditeur d'une unité de présentation se divise en trois types: présentation normale, présentation d'un exemple et rappel. Dans le type normal, le contenu entier doit être présenté en même temps. Dans le type d'exemple et de rappel, le contenu doit être présenté étape par étape. L'éditeur de l'unité de présentation contient trois zones: une liste des pages correspondant au nombre de pages dans la partie de description d'une unité ULA (figure 6.15), le type et le contenu, et la zone de boutons de commandes (figure 6.16). La différence entre les trois types concerne la partie

de contenu. La présentation normale a un seul cadre de contenu (figure 6.16.a). La présentation de l'exemple a des cadres supplémentaires associés au nombre d'étapes pour résoudre le problème (figure 6.17). La présentation du rappel doit normalement résumer ou rappeler les connaissances d'élève. En effet, nous devons signaler au système quel est le but de rappel et quelle est l'information liée à ce but. Donc pour créer ce type de présentation, le système propose à la fois deux rubriques : une pour le but et l'autre pour l'information comme présenté dans la figure 6.18. Pour faciliter aux enseignants-auteur, le système propose également un éditeur HTML représenté dans la figure 6.19. Les enseignants-auteurs peuvent accéder à cet éditeur en cliquant sur le premier bouton situé au dessous de la partie «Contenu».



a)



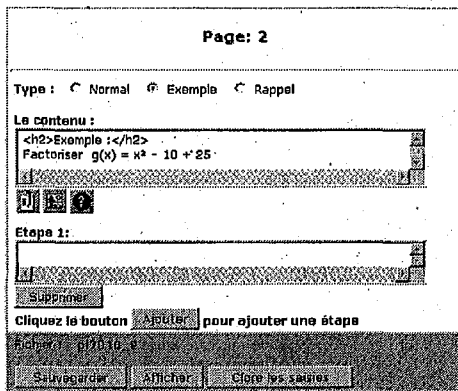
b)

FIG. 6.16 – Editeur de l'unité de présentation normale (a) et sa page obtenue (b)

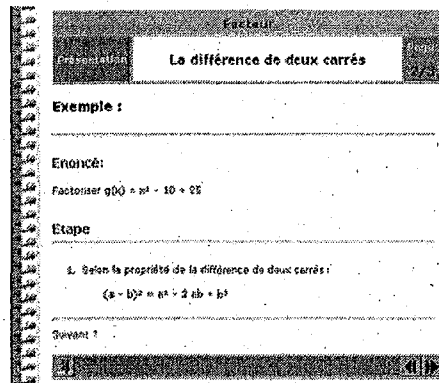
Exercice

L'éditeur d'une unité d'exercice est divisé en quatre catégories selon le type de question : les choix multiples, numérique, texte et vrai-faux. L'éditeur de chaque catégorie se divise en quatre parties : la liste des questions, insérer la question, le contenu et les commandes, représentées dans la figure 6.20.

La première partie est celle de la liste des questions représentées sur la forme des boutons,

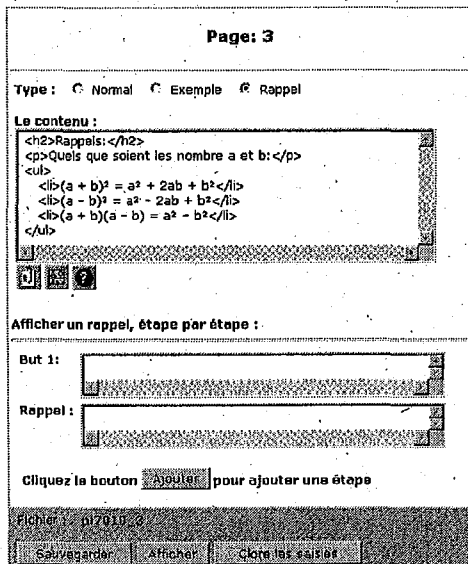


a)

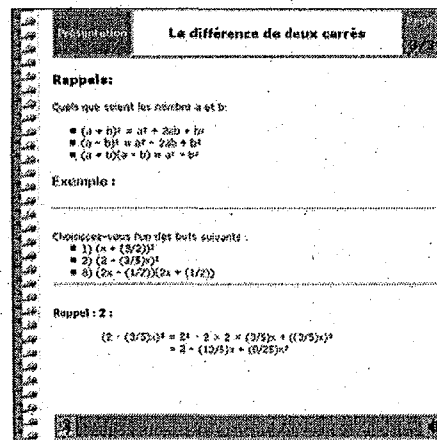


b)

FIG. 6.17 – Editeur de l'unité de présentation exemple (a) et sa page obtenue (b)



a)



b)

FIG. 6.18 – Editeur de l'unité de présentation rappel (a) et sa page obtenue (b)

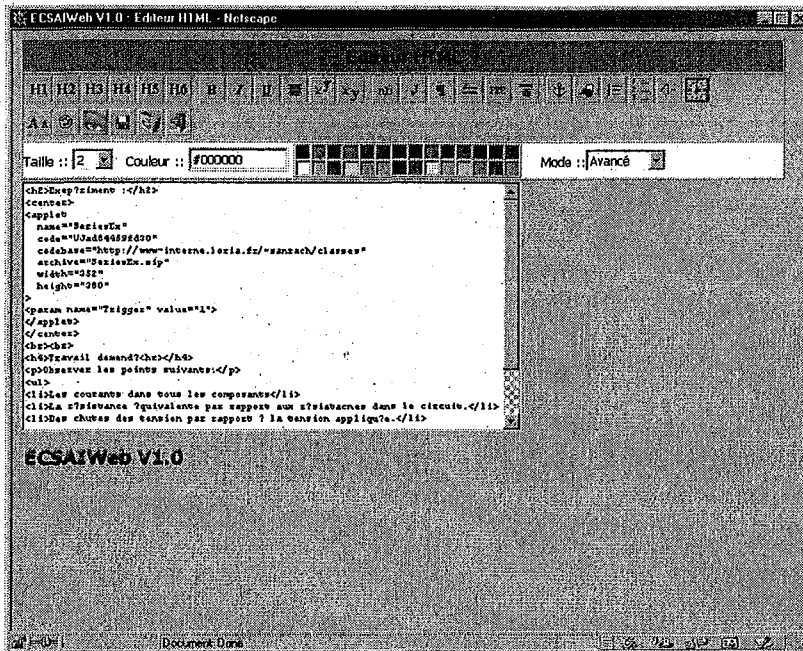


FIG. 6.19 – Editeur HTML

accompagné du numéro de question (figure 6.20). Ce bouton permet de changer la question pour modifier.

La deuxième partie permet d'insérer la question. Selon le processus d'insertion, nous devons saisir le numéro de la question, choisir la catégorie de cette question et puis envoyer la commande d'insertion.

Après avoir reçu la commande d'insertion de la question ou après le choix d'une question dans la liste, la troisième partie doit être affichée avec ses contenus. Cette partie doit contenir les informations nécessaires pour les exercices, par exemple, l'objectif de cette question, le type de question, la difficulté, la catégorie, l'énoncé, les choix et la réponse. Pour éditer ou modifier, nous devons :

- choisir l'un des objectifs en cliquant le bouton radio,
- choisir le degré d'acquisition (savoir-appliquer ou savoir-faire),
- choisir ou taper la difficulté (df). Nous proposons cinq niveaux décrits ci-dessous.
 - Très facile : $0.0 \leq df \leq 0.2$
 - Facile : $0.2 < df \leq 0.4$
 - Moyenne : $0.4 < df \leq 0.6$
 - Difficile : $0.6 < df \leq 0.8$
 - Très difficile : $0.8 < df \leq 1.0$

La valeur par défaut est de 0.5 (moyenne),

- choisir la catégorie de la question, normalement, celle-ci correspond à celle de votre choix à l'insertion d'une question.
- taper l'énoncé de cette question,
- taper les choix dans toutes les rubriques puis cliquez le bouton radio pour choisir la réponse correcte, selon le choix de l'état d'insertion question,

Liste des questions

Insérer la question

Entrez le numéro de la question :

Choisissez votre la catégorie suivante :

Rubrique(s) :

Question: 2

Objectifs :

- Résistance en série
- Polarité des chutes de tension dans un circuit série
- Notation à deux indices
- Loi des tensions de Kirchhoff
- Résistance interne
- Transfert maximal de puissance
- Résistances en parallèle
- Loi des courants de Kirchhoff
- Conductance et Conductivité
- Caractéristiques des circuits parallèle

Le degré d'acquisition : Savoir Appliquer Savoir Faire

La difficulté : Moyenne ou entrer la valeur difficulté :

Note : la valeur de difficulté est entre 0 et 1

Catégorie :

Énoncé :

On branche la circuit série de trois résistances, soit $R_1 = 5$ ohms, $R_2 = 10$ ohms et $R_3 = 15$ ohms, à une source de 120 V.

Les choix :

- A
- B
- C
- D
- E

Note : Si vous voulez lister à l'aide supplémentaire à l'extérieur du serveur alors entrez-vous l'URL ici : (ex. <http://www.toto.fr>)

Si la réponse est :

Note : si vous ne remplissez pas les rubriques suivantes, le système prendra les valeurs par défaut

- A alors afficher ce texte :
- B alors afficher ce texte :
- C alors afficher ce texte :
- D alors afficher ce texte :
- E alors afficher ce texte :
- CORRECT alors afficher ce texte :
- INCORRECT alors afficher ce texte :

Ajouter Aide

Fichier :

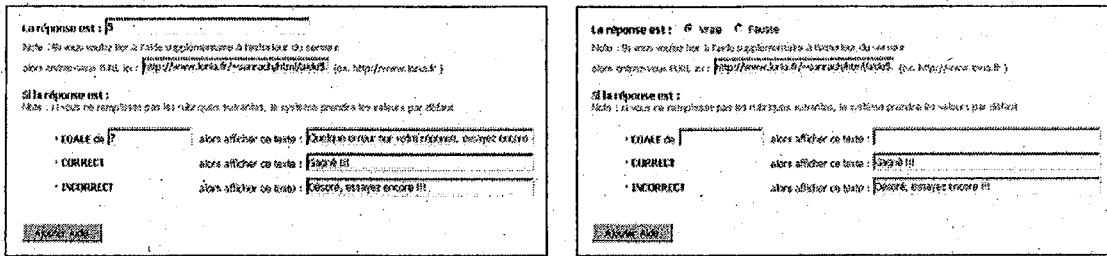
La liste des questions

La partie d'insertion

Contenu

Commandes

FIG. 6.20 – Editeur d'une unité exercice



a) numérique, expression et texte.

b) vrai-faux

FIG. 6.21 – Partie différente de chaque type de question

- éventuellement utiliser le bouton **Ajouter Aide** pour ajouter une aide.

L'édition d'autres catégories (numérique simple, texte et vrai-faux) de question reste inchangée sauf la partie concernant le choix et la réponse, représentée dans la figure 6.21.

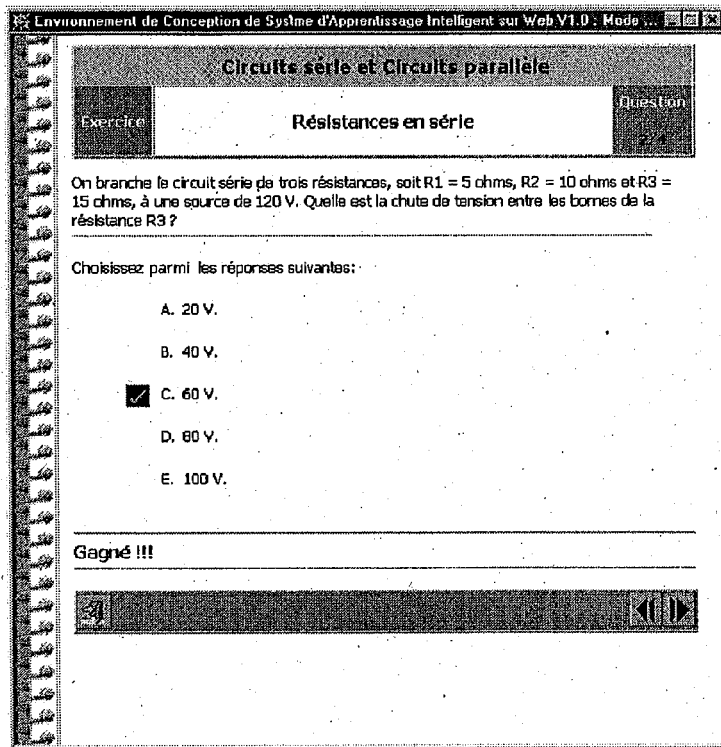


FIG. 6.22 – Affichage d'un exercice

La quatrième zone a pour rôle de sauvegarder les contenus, et de visualiser le contenu de la question (figure 6.22).

6.4.5 Vérification de cohérences

Encore une fois, dans notre environnement, les unités d'apprentissage sont décrites indépendamment les unes et les autres. Donc l'enseignant ne peut pas contrôler directement les cohérences de l'ensemble. Pour l'accessibilité à ces unités par les élèves, le système propose des outils de contrôle et de simulation d'un parcours d'apprentissage. Avec ces outils, l'enseignant peut vérifier les pertinences des items d'évaluation, la validité des objectifs pédagogiques et l'accessibilité des unités d'apprentissage. Ensuite, si les vérifications précédentes sont satisfaites, l'enseignant dispose des outils nécessaires à la simulation d'un parcours d'apprentissage. Il peut essayer le système d'apprentissage en situation d'apprenant et observer si le parcours proposé est conforme à ses prédictions. Ces outils sont indispensables à la mise au point du système d'apprentissage et permettent à l'enseignant de se rendre compte d'insuffisances au niveau de la définition de l'univers. Les vérifications portent essentiellement sur :

- la pertinence des items d'évaluation,
- la validité des objectifs pédagogiques,
- l'accessibilité de chaque unité,

par rapport à l'univers.

Pertinence des items d'évaluation

Tous les items doivent faire l'objet d'au moins une activité pédagogique quelle qu'elle soit (Présentation ou Exercice). Le système reprend chaque item d'évaluation et regarde dans l'univers d'apprentissage s'il existe au moins une unité faisant référence à cet item, sinon il faut ajouter au moins une unité correspondant à cet item ou l'enlever. La figure 6.23 représente un exemple de vérification des items d'évaluation.

Résumé		
Objectifs	Définition	Remarque
i1	Différence de deux carrés	Complete
i2	Carré d'une somme	Complete
i3	Carré d'une différence	Complete

FIG. 6.23 – Vérifier la pertinence des items d'évaluation.

Validité des objectifs

Cette vérification a pour but de s'assurer que l'activation de certaines unités logiques constituant l'univers d'apprentissage permet d'atteindre les objectifs pédagogiques (figure 6.24).

Accessibilité des unités d'apprentissage

Cette vérification consiste à s'assurer que dans l'univers d'apprentissage toutes les unités logiques définies par les enseignants-auteurs peuvent être activées par un élève. Il s'agit de contrôler si les filtres ne contiennent pas des niveaux de connaissances trop élevés par rapport à ceux qui peuvent être atteints au travers de l'univers (figure 6.25).

Le problème se pose pour les unités logiques qui n'ont aucun lien statique entre elles et pour lesquelles il faut envisager divers enchaînements d'unités logiques avant de pouvoir être certain qu'elles pourront être activées. Nous dirons que l'univers créé est cohérent si les trois vérifications

Résumé		
Unité	Apport	Résultat
1	i1(a;0,0)	i1(a;0,0)
2	i2(a;0,0)	i2(a;0,0) i1(a;0,0)
3	i3(a;0,0)	i3(a;0,0) i2(a;0,0) i1(a;0,0)
4	i1(a;1,0)	i1(a;1,0) i3(a;0,0) i2(a;0,0)
5	i3(a;0,1)	i3(a;0,1) i1(a;1,0) i2(a;0,0)
6	i2(a;0,2)	i2(a;0,2) i3(a;0,1) i1(a;1,0)
7	i1(a;1,0)	i1(a;2,0) i2(a;0,2) i3(a;0,1)
Objectifs Pédagogiques :		i1(a;2,0) i2(a;0,2) i3(a;0,1)
Objectifs calculer :		i1(a;2,0) i2(a;0,2) i3(a;0,1)
Résumé :		
Avec l'ensemble des unités existantes, vous <i>pouvez</i> atteindre tous les objectifs pédagogiques.		

FIG. 6.24 – Vérifier la validité des objectifs pédagogiques.

Résumé				
Unité	Pré-requis	Titre	Apport	Remarque
1	aucun	La différence de deux carrés	i1(a;0,0)	Accessible
2	i1(a;0,0)	Carré d'une somme	i2(a;0,0)	Accessible
3	i1(a;0,0)	Carré d'une différence	i3(a;0,0)	Accessible
4	i1(a;0,0)	La différence de deux carrés	i1(a;1,0)	Accessible
5	i3(a;0,0)	Carré d'une différence	i3(a;0,1)	Accessible
6	i2(a;0,0)	Carré d'un somme	i2(a;0,2)	Accessible
7	i1(a;1,0)	La différence de deux carrés	i1(a;1,0)	Accessible

FIG. 6.25 – Vérifier l'accessibilité des unités d'apprentissage

ci-dessus ont été faites. La vérification implantée à ce stade n'est pas complète, c'est pourquoi les outils de simulation sont proposés.

6.4.6 Simulation du parcours d'apprentissage

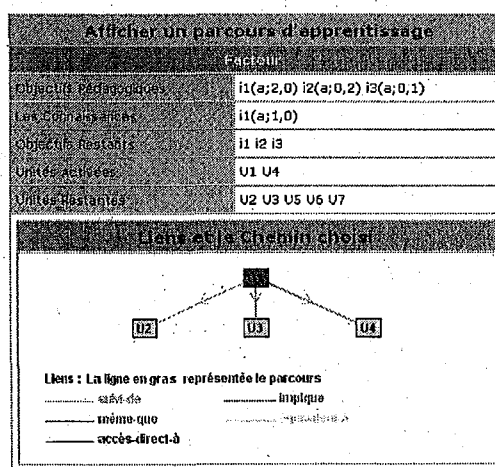
Si l'univers d'apprentissage est cohérent, le système propose des outils nécessaires à la simulation. Ces outils permettent aux enseignants-auteurs d'essayer de simuler le système d'apprentissage en situation d'apprenant. Ils vont avoir besoin simplement de regarder si le parcours proposé est conforme à leurs prédictions. Les enseignants peuvent également simuler le parcours d'apprentissage en situation réelle en utilisant le mode d'apprentissage mais ils perdent leurs droits sur le mode d'auteur. Pour la simulation d'un parcours d'apprentissage, le système nous propose trois fonctions suivantes : simulation, visualisation et suppression d'un parcours d'apprentissage.

Simuler le parcours d'apprentissage

Pour «*simuler*» le parcours d'apprentissage, si l'auteur n'a pas encore utilisé cette fonction, le système calcule et choisit l'état initial. Cet état est calculé d'une part en regardant dans la partie filtre des unités logiques s'il existe au moins une unité logique qui n'a pas besoin de pré-requis et d'autre part en prenant la première unité sur la liste des unités logiques. Sinon il reprend la dernière étape, puis calcule l'état suivant. L'état suivant est calculé en utilisant les informations stockées dans l'unité elle-même et dans le modèle d'utilisateur, puis en appliquant la méthode de raffinement pour obtenir la meilleure unité à suivre. Cette fonction est activée à l'aide du bouton «Continue» présenté dans la figure 6.26.a. Si tous les objectifs pédagogiques sont atteints, le système nous alerte et s'arrête.

Simulation d'apprentissage	
Unité choisie	U1 U7
Unité choisie	U1
Nature d'oreilles	Présentation
Titre	pi1010
Nombre de Page	1 / 3
Objectifs pédagogiques	i1(a;2,2) i2(a;0,1) i3(a;0,0) i4(a;1,1) i5(a;2,1) i6(a;2,1) i7(a;2,1) i8(a;1,0) i9(a;1,0) i10(a;1,0)
Les commentaires	i1(a;0,0)
Objectifs non atteints	i1 i2 i3 i4 i5 i6 i7 i8 i9 i10
Unités déjà activées	U1
Unités restantes	U2 U3 U4 U5 U6 U7 U8 U9 U10 U11 U12 U13 U14 U15 U16 U17 U18 U19
Liens entre les unités	• Dynamique : implique(1,7) • Statique : suivi-de(1,2) suivi-de(1,11)
Méthode choisie	Item d'abord
<input type="button" value="Continue"/>	
Unité suivante	U11
Unité actuelle	i1

a) Simuler un parcours d'apprentissage.



b) Afficher un parcours d'apprentissage.

FIG. 6.26 – Simulation du parcours d'apprentissage

Visualiser le parcours d'apprentissage

Pour «afficher» le parcours d'apprentissage, il faut simplement choisir le sous-menu «afficher un parcours d'apprentissage». L'état courant s'affiche dans le cadre. Cette afficheage contient tous les objectifs pédagogiques, le niveau des connaissances, les objectifs non-atteints, les unités activées, les unités restantes et les liens entre les unités et le chemin choisi représenté dans la figure 6.26.b.

6.4.7 Création des examens

Notre système propose pour chaque sujet des examens (s'il existe) à l'élève, appelé le pré-examen et/ou le post-examen pour évaluer les connaissances avant et/ou après l'étude respectivement. Cet examen dépend des objectifs pédagogiques que l'enseignant a donnés au début de la phase d'initialisation.

La table 6.1 représente les objectifs pédagogiques d'un sujet donné. Ces objectifs indiquent les comportements de savoir-appliquer et savoir-faire des élèves que l'enseignant a besoin d'évaluer. Chaque question de l'examen peut être liée à plusieurs items tandis que dans l'exercice nous pouvons choisir un seul item pour chaque question. Les catégories d'examen restent identiques à celles d'exercice, la différence est que dans l'examen il n'y a plus de rétroactions et d'aides.

Item.	Définition	Mesure globale	Degré d'acquisition	
			Savoir-Appliquer	Savoir-Faire
1	Résistance en série	a	2	2
2	Polarité des chutes de tension dans un circuit série	a	0	1
3	Notation à deux indices	a	0	0
4	Loi des tensions de Kirchhoff	a	1	1
5	Résistance interne	a	2	1
6	Transfert maximal de puissance	a	2	1
7	Résistances en parallèle	a	2	1
8	Loi des courants de Kirchhoff	a	1	0
9	Conductance et Conductivité	a	1	0
10	Caractéristiques des circuits parallèles	a	1	0

TAB. 6.1 – Liste des objectifs pédagogiques.

Création de l'examen

Pour créer les examens, pré-examen ou post-examen, notre système propose les outils et interfaces identiques à ceux des exercices. Chaque question contient 4 parties principales (figure 6.27):

- la liste des questions (les mêmes que pour l'exercice),
- L'insertion d'une question (les mêmes que pour l'exercice),
- les contenus qui contiennent :
 - le degré d'acquisition (savoir-appliquer et savoir-faire),
 - la difficulté a 5 niveaux (très facile, facile, moyenne, difficile et très difficile). Cette valeur doit être sélectionnée par l'enseignant, sinon elle prend la valeur par défaut <0.5> ou moyenne,
 - La catégorie de la question : le choix multiple, numérique, expression, texte et vrai-faux,
 - L'énoncé,

Menu de l'interface de l'examinateur

Menu de l'examinateur

Liste des questions

Inserer la question

Entrez le numéro de la question

Choisissez votre réponse suivante: Choix-Multiple

Cliquez sur Insérer

Question: 1

Objectifs :

- Résistance en série
- Polarité des chutes de tension dans un circuit série
- Notation à deux indices
- Loi des tensions de Kirchhoff
- Résistance interne
- Transfert maximal de puissance
- Résistances en parallèle
- Loi des courants de Kirchhoff
- Conductance et Conductivité
- Caractéristiques des circuits parallèles

Le degré d'acquisition: Savoir Appliquer Savoir Faire

La difficulté:

ou entrer la valeur difficulté:

Noté: la valeur de difficulté est entre 0 et 1

Catégorie:

Énoncé:

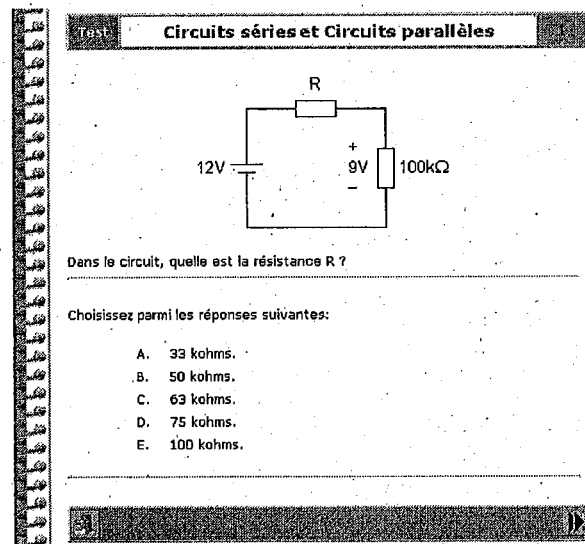
`<center><img src="http://www.loria.fr/~sanrach/docs/Images/`
`<p>Dans le circuit, quelle est la résistance R ?`

Les choix:

- A 33 kohms.
- B 50 kohms.
- C 63 kohms.
- D 75 kohms.
- E 100 kohms.

Fichiers associés:

Sauvegarder Afficher Clôturer les saisies



a) Editeur de chaque question

b) La présentation d'un examen

FIG. 6.27 – Interface pour créer un examen

- Le choix et la réponse.
- Les boutons de commandes : Sauvegarder, Afficher et Clore les saisies

Visualisation de l'examen

Cette fonction a pour effet de visualiser l'examen à partir de la question courante. Pour afficher la question, nous divisons en quatre zones : le titre, l'énoncé, le choix ou la saisie de réponse (selon le type de la question) et la rétroaction, et la barre de navigation (figure 6.27.b). Le titre a pour objectif d'indiquer le test, le sujet et le numéro de question. L'énoncé présente la question. Le choix et la saisie de réponse permettent de répondre à la question. La rétroaction permet de juger la réponse et de donner les informations à l'élève. La dernière, la barre de navigation permet de retourner à la question précédente ou d'aller à la question suivante ou simplement de fermer la fenêtre de visualisation de l'examen.

6.4.8 Importation et Gestion des fichiers

Après la première expérimentation, au département informatique et formation de KMITNB et sur la demande de plusieurs enseignants que nous avons rencontrés dans plusieurs conférences, nous avons décidé d'ajouter cette fonction pour aider les enseignants à transmettre leurs matériaux pédagogiques existants sur le Web. Cette fonction est importante pour l'auteur qui a déjà créé des matériaux pédagogiques en format HTML ou bien d'autres formats qu'ils peuvent être convertis au format HTML, par exemple Word, PowerPoint, Latex et etc. Nous avons ajouté aussi les fonctions de dupliquer les fichiers et de convertir les fichiers formats HTML à notre format (fichier de la présentation de type normal). La figure 6.28 présente la procédure de transformation des fichiers HTML au format NORMAL de la présentation de notre système.

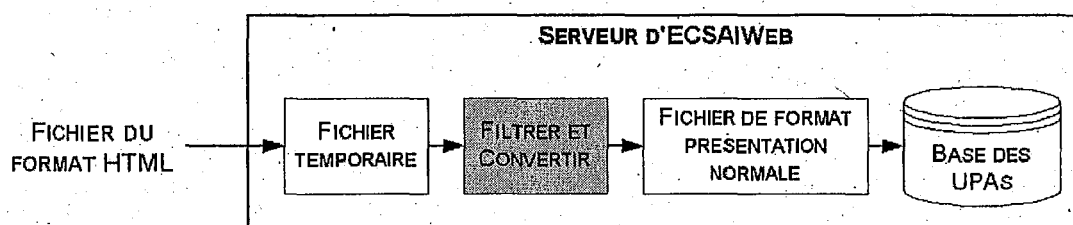


FIG. 6.28 – Processus de transformation des fichiers du système ECSAIWEB

6.4.9 Gestion des profils d'élèves

Le dernier outil de cet environnement est celui de gestion du profil des élèves. Cet outil permet à l'enseignant-auteur de suivre le parcours d'apprentissage des élèves et de gérer les contenus dans le module d'élève. Il faut noter que même si nous avons proposé l'outil de validation de parcours d'apprentissage à l'enseignant-auteur, celui-ci assure que l'élève pourra en général atteindre les objectifs pédagogiques, mais il ne nous offre aucune information sur la situation réelle de l'apprentissage. Alors cet outil permet aux enseignants-auteurs de visualiser la création d'univers d'apprentissage, puis ils peuvent utiliser ces informations pour améliorer leurs cours. L'ensemble des informations de chaque élève contient des informations personnelles : nom, prénom, et niveau d'étude (figure 6.30) et puis des informations évolutives. En mode libre, ce sont des notes (figure

6.31). En mode guidé (figure 6.32.a), il inclut le parcours d'apprentissage, les connaissances (figure 6.32.b) et le résumé sur les unités d'apprentissage (figure 6.32.c).

Gestion des élèves

Choisissez vous le nom d'élève suivant :

Choisir :

Langage :

Effacer des contenus de mode guidé dans un module d'élève

FIG. 6.29 – Commandes de gestion des profils d'élèves

Nom : SANRACH	Prénom : Charun
Class : Université	Niveau : 2ème année
Progression: <input type="text" value="Circuits séries et Circuits parallèles"/>	
Mode d'apprentissage : Mode guidé Mode libre	
Examen	
Pré-examen [Notes = -]	
Post-examen [Notes = 6/15]	

FIG. 6.30 – Informations personnelles

Progression: <input type="text" value="Circuits séries et Circuits parallèles"/>		
Mode d'apprentissage : Mode guidé Mode libre		
Mode Libre		
Exercice	Note	Remarque
Résistance en série	3 / 4	Bien
Polarité des chutes de tension dans un circuit série	1 / 1	Très Bien
Caractéristiques des circuits parallèles	0 / 1	Très faible

FIG. 6.31 – Connaissances en mode libre

6.5 Fonctionnalités de l'environnement élève

Suite à la création d'un univers d'apprentissage, le système est prêt à être utilisé donc nous passons dès maintenant au mode tutoriel. Le champ d'application de la première maquette est celui d'analyse du circuit électrique en régime continu pour les étudiants de fin secondaire et de premier cycle universitaire.

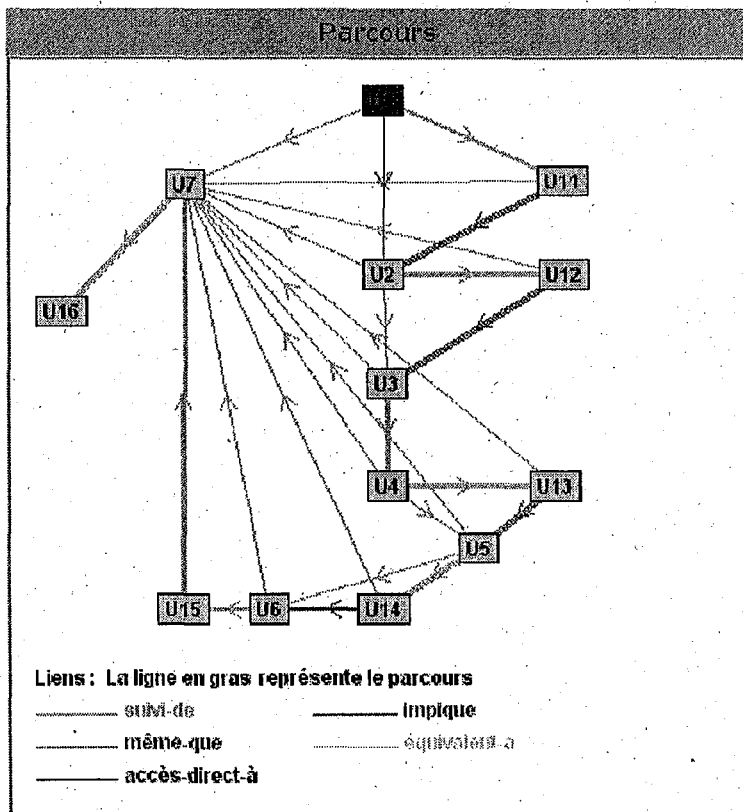
La nouvelle organisation d'apprentissage que nous proposons partage beaucoup de caractéristiques avec le nouveau mode d'apprentissage : «**COACHED BY TEACHER**» où des étudiants apprennent à appliquer la connaissance pour résoudre des problèmes. Ils apprennent aussi à travailler en équipes parce qu'ils font partie d'un réseau de transmission dans lequel la connaissance et l'expérience sont partagées par des compagnons et des experts. Pour supporter les actions

Les unités d'apprentissage			
Id	Définition	Nature	Remarque
1	Résistance en série	Présentation	Activée
2	Polarité des chutes de tension dans un circuit série	Présentation	Activée
3	Notation à deux indices	Présentation	Activée
4	Loi des tensions de Kirchhoff	Exercice	Activée
15	Transfert maximale de puissance	Exercice	Activée
16	Résistance en parallèle	Exercice	-
17	Loi des courants de Kirchhoff	Exercice	-
18	Conductance et Conductivité	Exercice	-
19	Caractéristiques des circuits parallèles	Exercice	-

Définition des liens	Les connaissances					
	Objet	sa	sa	sa	sa	sa
Résistance en série	a	2	2	a	0	1
Polarité des chutes de tension dans un circuit série	a	0	1	a	0	0
Notation à deux indices	a	0	0	-	-	-
Loi des tensions de Kirchhoff	a	1	1	-	-	-
Résistance interne	a	2	1	-	-	-
Transfert maximal de puissance	a	2	1	-	-	-
Résistances en parallèle	a	2	1	-	-	-
Loi des courants de Kirchhoff	a	1	0	-	-	-
Conductance et Conductivité	a	1	0	-	-	-
Caractéristiques des circuits parallèles	a	1	0	-	-	-

a) Le résumé sur les unités d'apprentissage

b) Les connaissances



c) Le parcours d'apprentissage

FIG. 6.32 – Informations en mode guidé

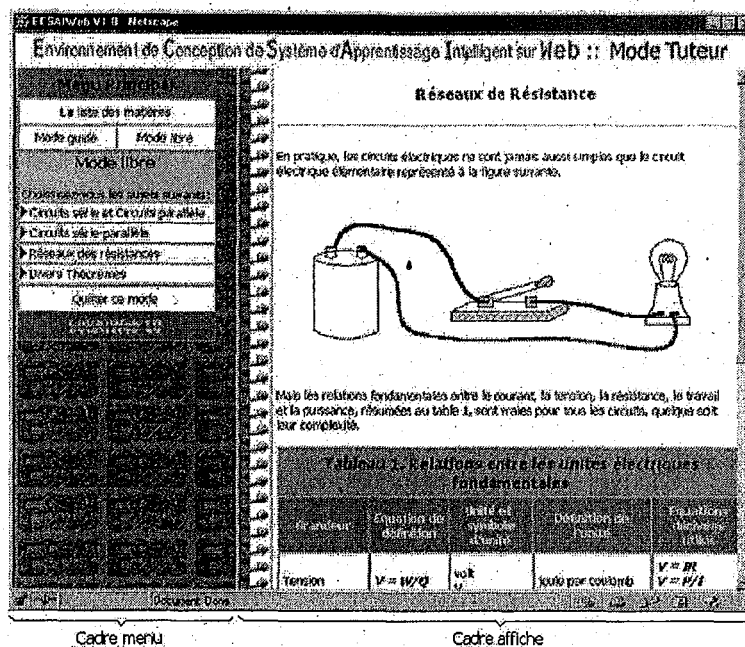


FIG. 6.33 – Espace de travail pour les élèves

des étudiants, l'environnement d'apprentissage doit offrir les outils génériques et adaptés à leurs besoins. Les outils que nous devons offrir aux étudiants sont l'outil de navigation dans l'univers d'apprentissage, l'outil de visualisation de contenu et l'outil de communication. La figure 6.34 représente la vue globale de l'environnement élève qui permet à l'élève d'accéder directement au vivier de connaissances en sélectionnant le mode d'apprentissage : libre ou guidé [San00].

6.5.1 Mode libre

En mode libre, le système permet à l'élève de choisir librement les items à apprendre. Les objectifs pédagogiques sont présentés sur un menu adaptatif en terme des items définis dans le modèle de domaine et de niveaux des compétences sur les items. La figure 6.35 présente un générateur du parcours d'apprentissage en mode libre. Avec l'item choisi (item i1), le système détermine des unités logiques d'apprentissage à présenter. Pour chaque unité logique, une liste de concepts associés avec cette unité est donnée mais le système proposera seulement les unités logiques qui contiennent au moins le concept choisi (unité U1 et U4) et les concepts connus (ou les concepts pré-requis, unité U2). Le système détaille la liste d'unités à présenter sur le plan progressif (figure 6.37.a) en utilisant la procédure représentée dans la figure 6.36. Le choix de ces unités est fait selon la nature d'unité, la présentation d'abord puis l'exercice ou bien selon le besoin de l'apprenant. Donc l'enchaînement des unités peut être : la présentation seule, l'exercice seul, la présentation d'abord (présenter toutes les unités de présentation, puis suivre par toutes les unités d'exercice) ou bien en utilisant l'ordre importance sur des liens et les items. L'exemple d'enchaînement est présenté dans la figure 6.37.a. Après le choix d'un plan parmi ceux qui sont proposés, l'apprenant doit le suivre jusqu'à la fin de ce parcours. A la fin de

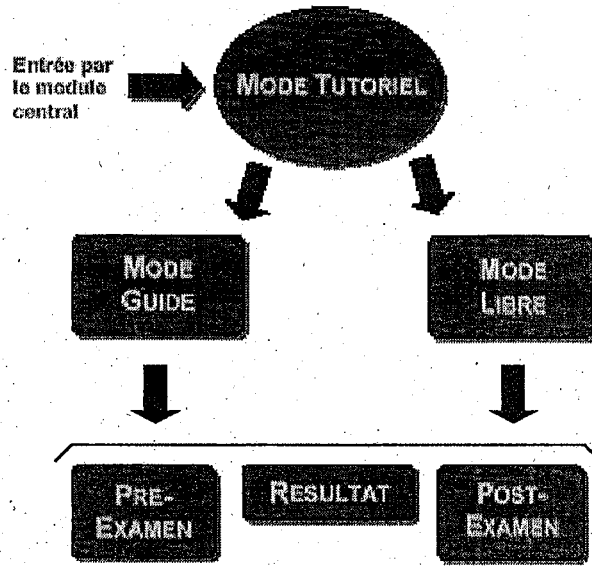


FIG. 6.34 – Organigramme de l'environnement élève

chaque parcours d'apprentissage (automatiquement) ou le besoin de l'apprenant (par son choix), le système propose le résumé global sur le résultat obtenu de chaque item choisis. En appliquant la technique d'annotation adaptative, nous pouvons annoter les états pédagogiques (cf. section 5.4.4) sur chaque item (figure 6.37.b). Avec cette annotation, l'apprenant sait toute de suite les objectifs pédagogiques sont atteints ou non.

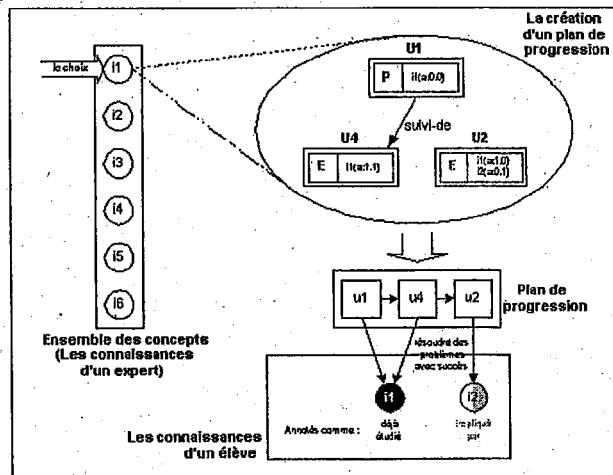


FIG. 6.35 – Exemple du générateur d'un parcours d'apprentissage en mode libre

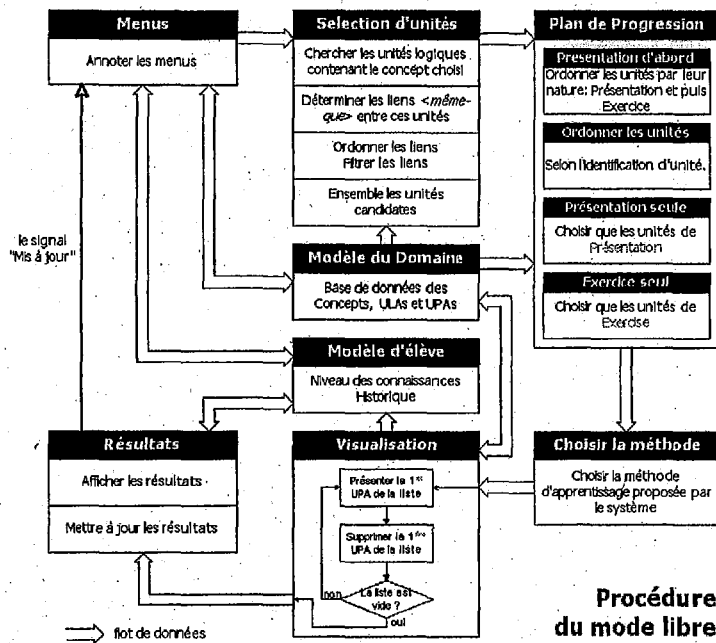


FIG. 6.36 – Procédure du mode libre.

Plan de la progression de

Diviseur de tension

titre de leçon	Type de leçon
Diviseur de tension à plusieurs tension de sortie	Présentation
Diviseur de tension	Exercice[1]

nombre total d'heures

Cliquez le bouton pour continuer

- Présentation / Exercice
- Présentation
- Exercice

Vos résultats sur

Circuit série et circuits parallèle

Exercice

Exercice	Notes	Bonifications
Résistance en série	4/4	Très Bien
Polarité des chutes de tension dans un circuit série	1/1	Très Bien
Notation à deux indices	-	Sans-exercice
Loi des tensions de Kirchhoff	0/2	Très Faible
Résistance interne	1/3	Faible
Transfert maximal de puissance	0/3	Très Faible

Total

Pré-examen	2/12	Très Faible
Post-examen	4/15	Faible

a) Le plan de progression en mode libre

b) Le résumé des compétences de l'apprenant de chaque objectif pédagogique en mode libre

FIG. 6.37 – Plan de progression et le résultat en mode libre

6.5.2 Mode guidé

En mode guidé, le système propose un menu qui est la liste de sujets (figure 6.38, partie gauche). Après le choix de sujet, le système prend la main, et propose le plan de progression adaptatif (figure 6.38, partie droite). Ce plan est l'ensemble des connaissances de l'apprenant sur le sujet choisi, détaillant les objectifs pédagogiques déjà atteints ou non. Puis le système génère le parcours d'apprentissage pour chaque élève (figure 6.39). Le système, au fur et à mesure des activités accomplies par l'élève, décide la prochaine unité à présenter en utilisant les paramètres de l'unité courante, le modèle d'élève et des critères additionnels. L'une des unités dont les conditions préalables ont été satisfaites par le modèle d'élève est choisie en fonction des objectifs pédagogiques. Après que l'unité est accomplie, le modèle d'élève est mis à jour selon les post-actions de l'unité et la compétence sur cette unité, et la prochaine unité est calculée. La figure 6.40 représente le parcours d'apprentissage obtenu de l'apprenant X.

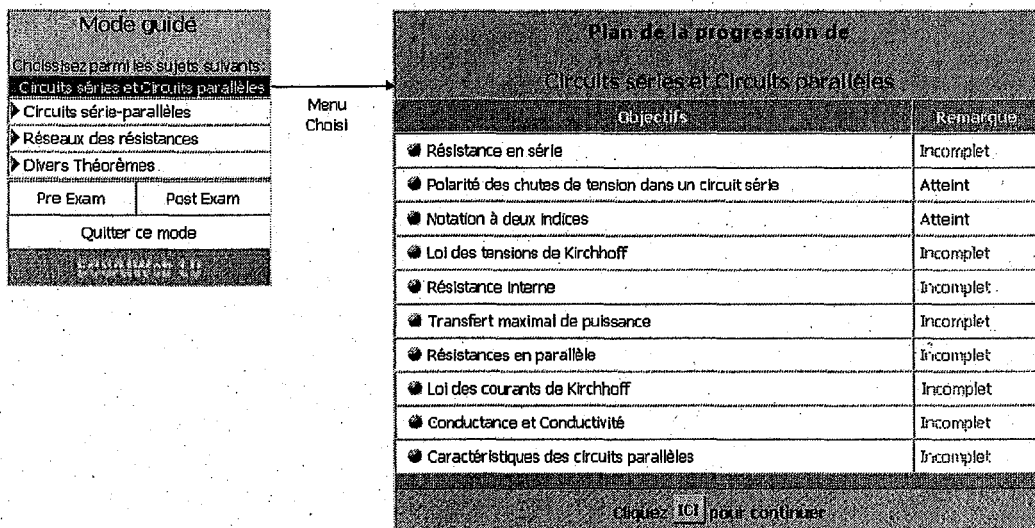


FIG. 6.38 – Plan de progression du mode guidé

Quand et comment choisir une unité d'apprentissage?

Les unités logiques d'apprentissage constituent les nœuds d'un réseau sémantique dont certains arcs sont créés pendant le parcours d'apprentissage (figure 6.40) pour compléter le réseau et former l'univers d'apprentissage. Cette dernière caractéristique confère à l'apprentissage un aspect dynamique puisque l'orientation de l'apprenant est réalisée en temps réel en fonction de sa progression. Le système crée des liens entre l'unité courante et les unités candidates. Puis il choisit l'une des unités candidates en appliquant des critères supplémentaires : les fonctions heuristiques de raffinement des liens. Le système s'arrête quand tous les objectifs pédagogiques sont atteints ou quand aucune unité ne peut être proposée. La figure 6.40 montre un ensemble d'unités candidates après exécution d'unité $U1$. Plusieurs types de liens sont employés, *suivi-de*($U1, U2$), *implique*($U1, U7$) et *suivi-de*($U1, U11$), pour lier l'unité courante ($U1$) à d'autres unités ($U2, U7$ et $U11$) qui peuvent être choisies comme prochaine unité.

La description plus détaillée de la procédure de création des liens dynamique des unités a déjà

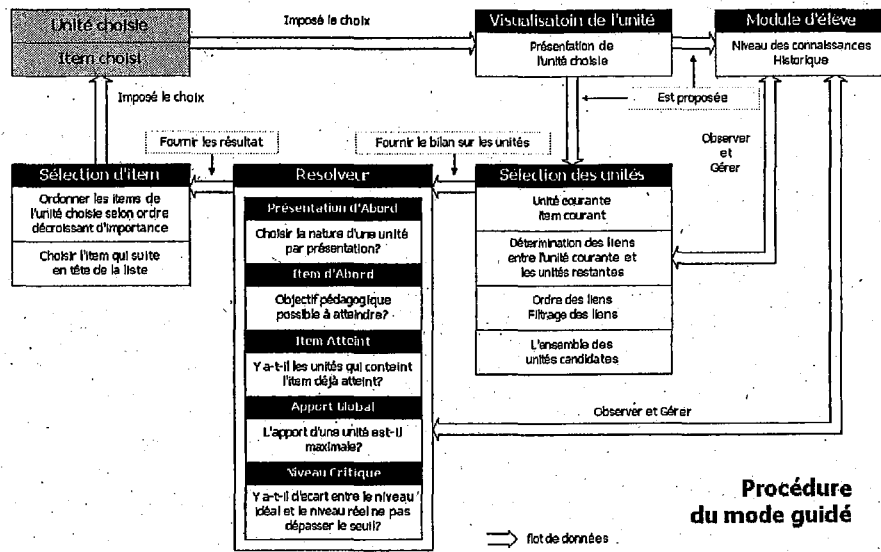


FIG. 6.39 – Procédure du mode guidé

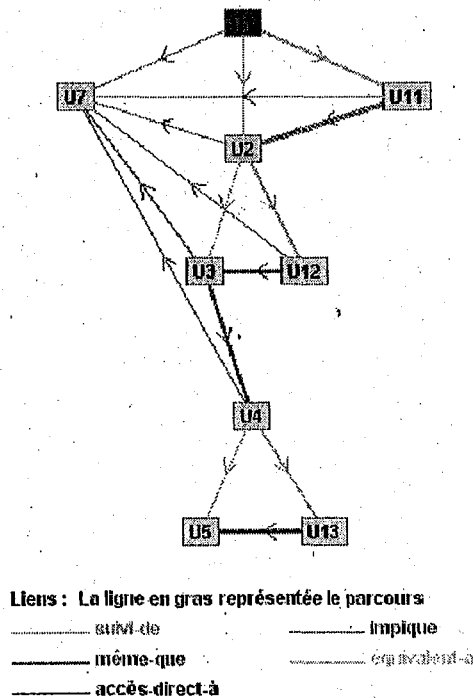


FIG. 6.40 – Parcours d'apprentissage de l'apprenant X en mode guidé

été proposée dans les sections 2.2.2.d et 4.3.2. Le choix va être guidé par une liste de fonctions heuristiques [SG00b]. Cette liste à deux niveaux. Le premier niveau s'exécute avant le deuxième, dont nous choisissons une méthode du «*profondeur d'abord*».

Premier niveau

Nous avons employé le choix des stratégies d'importance de liens, comme l'ancien système ECSAI (cf. [Gav91, page 73]). Les liens les plus importants sont les liens «*accès-direct-à*», parce que ces liens expriment une relation privilégiée entre les unités définies par les enseignants-auteurs au moment de la description des unités logiques d'apprentissage. S'il existe les liens «*accès-direct-à*» dans les liens obtenus, nous supprimons tous les autres liens et puis nous renvoyons directement cette liste au deuxième niveau. Sinon, nous trions les liens par l'ordre suivant : «*implique*», «*suivi-de*», «*équivalent-à*» et «*même-que*» avant de renvoyer au deuxième niveau.

Deuxième niveau

Nous cherchons l'ensemble des unités à partir des unités candidates. Plusieurs critères sont mis en évidence décrits ci-dessous.

- **Présentation d'abord** : Deux critères sont employés dans cette méthode : choisir les unités parmi les unités candidates qui ont la nature d'une unité de type présentation et puis nous appliquons les étapes suivantes pour raffiner les unités candidates. Cette méthode n'est choisie par le système que l'élève l'avait choisie avant entrer dans l'environnement élève. L'exemple suivant nous montre qu'à la fin de l'exécution d'unité U1, le système génère les liens entre les unités, puis les liens obtenus sont présentés dans la figure 6.41. En appliquant cette méthode, nous obtenons l'unité U7 à suivre.

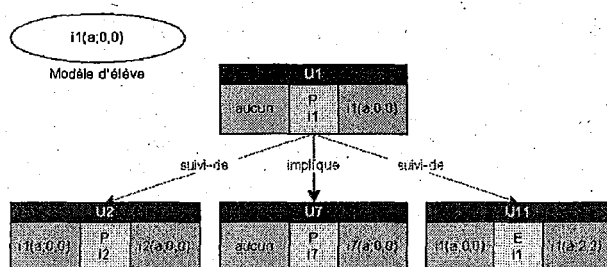


FIG. 6.41 – Méthode raffinée : Présentation d'abord

- **Item d'abord** : Soit un objectif porté sur un item i_k que l'apprenant est en train d'atteindre. Les compétences de l'apprenant, vis à vis de i_k , sont très proches de l'objectif pédagogique sur i_k . Nous essayons d'atteindre cet objectif le plus vite possible. Nous prenons donc une unité qui donnera le maximum d'apport sur i_k (parcours en profondeur sur les items). A partir des liens obtenus à la fin de l'exécution d'unité U1, représenté dans la figure 6.42, si l'apprenant n'avait pas choisi la méthode «*Présentation d'abord*» alors le système nous propose de suivre l'unité U11 parce qu'il n'y a que l'unité U11 associée au même item de l'unité U1 (sans regarder sur les ordres d'importance des liens).
- **Item atteint** : Lorsque les objectifs pédagogiques portant sur un item i_j sont atteints, nous privilégions les unités qui contiennent i_j et nous appliquerons «*item d'abord*». L'exemple

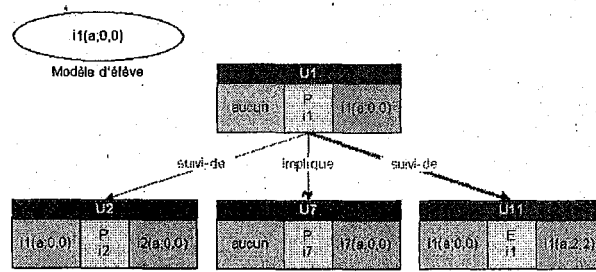


FIG. 6.42 – Méthode raffinée : Item d'abord

représenté dans la figure 6.43 exprime le choix de prochaine unité. Supposons que l'objectif i_1 est déjà atteint, alors l'unité U13 est sélectionnée parmi les unités candidates. En appliquant la méthode «*Item d'abord*», nous obtenons l'unité U13 comme la prochaine unité.

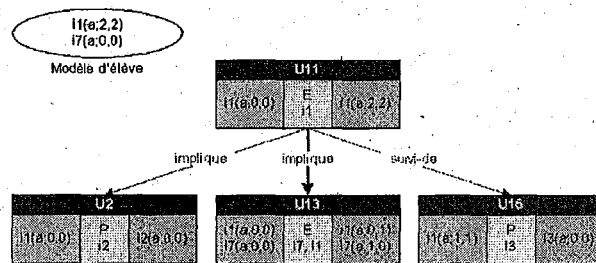


FIG. 6.43 – Méthode raffinée : Item atteint

- **Apport global** : Nous souhaitons que l'élève se rapproche le plus vite possible des objectifs pédagogiques, mais en considérant tous les items à la fois. Alors que «*item d'abord*» s'attache à un item, nous essayons de considérer le maximum d'apport de connaissance qui permettent d'atteindre ces objectifs. Dont il faut mesurer cet apport maximum pour chaque unité candidate (parcours en largeur sur les items). L'exemple dans la figure 6.44 nous montre que la prochaine unité doit être l'unité U16 parce que l'apport global de cette unité est maximal.
- **Niveau critique** : Nous connaissons le niveau de l'élève et son niveau idéal (objectifs pédagogiques) pour chaque item. L'écart entre le niveau idéal et le niveau réel donnera une valeur pour chaque item. Dès règles qui expriment le seuil (50% des compétences de chaque item au niveau passable) à ne pas dépasser pour un item donné, permettront d'exprimer la méconnaissance sur un item et réagiront en sélectionnant par exemple une unité qui mettra en évidence cet item.

Exemple: Soit $i_{6obj}(a, 5, 7)$ et à l'instant donné b , la connaissance d'un apprenant sur l'item i_6 est de $i_{6_b}(a, 1, 2)$, nous l'exprimons comme celle-ci :

si $\text{Ecart}(i_{6obj}, i_{6_b}) > \text{seuil de } i_6$ (seuil_{sa} de $i_6 = 2.5$ et seuil_{sf} de $i_6 = 3.5$)
 alors il faut choisir une unité qui met en valeur i_6 .

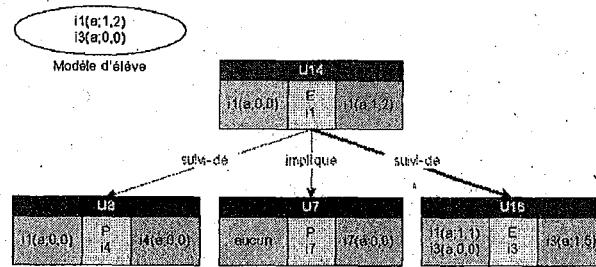


FIG. 6.44 – Méthode raffinée : Apport global

Conclusion via l'algorithme

En résumé, nous pouvons écrire l'algorithme pour le choix l'une parmi des unités candidates représenté dans la figure 6.45.

```
#####
# Algorithme pour raffiner les unités candidates
#####
# Profil choisi :
# 1 : Présentation - Présentation - Exercice
# 2 : Présentation - Application - Application
# Objectif → Objectif courant
# Lien → Liens calculés entre l'unité courante et les unités restantes
Début
  UniteChoisie ← vide
  Si ProfilChoisi == 1
    Alors
      UniteChoisie ← PresentationAbord(Objectif, Lien)
    Si UniteChoisie est vide Alors UniteChoisie ← ItemAbord(Objectif, Lien) Fin Si
  Sinon
    UniteChoisie ← ItemAbord(Objectif, Lien)
  Fin Si
  Si UniteChoisie est vide Alors UniteChoisie ← ItemAttient(Objectif, Lien) Fin Si
  Si UniteChoisie est vide Alors UniteChoisie ← ApportGlobal(Objectif, Lien) Fin Si
  Si UniteChoisie est vide Alors UniteChoisie ← NiveauCritique(Objectif, Lien) Fin Si
  retour UniteChoisie
Fin
```

FIG. 6.45 – Algorithme de raffinement du parcours d'apprentissage

Remédiation

Lorsque les objectifs pédagogiques ne peuvent plus être satisfaits avec les unités restantes (l'élève est trop faible), il faut revenir en arrière sur des unités déjà vues pour pouvoir continuer. Nous détectons donc les items défaillants par rapport à leur niveau idéal respectifs. Nous reprenons des unités mettant en œuvre un ou plusieurs de ces items et nous utilisons une des méthodes précédentes. Nous pouvons également raffiner cette détection défaillante par item en conservant pour un item donné le nombre d'essais et le nombre d'erreurs effectués pour celui-ci. Nous en déduisons le nombre de réponses justes et le taux d'erreur commis. Le système s'arrête après 3 fois d'essai et si l'élève ne peut pas atteindre ces items.

6.5.3 Navigation adaptative

Notre approche présente beaucoup plus de moyens de passer les matériaux pédagogiques en ligne. Le côté négatif, c'est qu'il y a un gros risque pour que l'utilisateur se perde dans un hyperespace complexe. Pour soutenir l'utilisateur naviguant dans le cours, le système emploie l'annotation adaptative, une technique adaptative hypermédia [BSW96a]. L'annotation adaptative signifie que le système emploie des sélections visuelles comme des icônes, polices, et couleurs pour montrer le type et le rôle pédagogique de chaque lien.

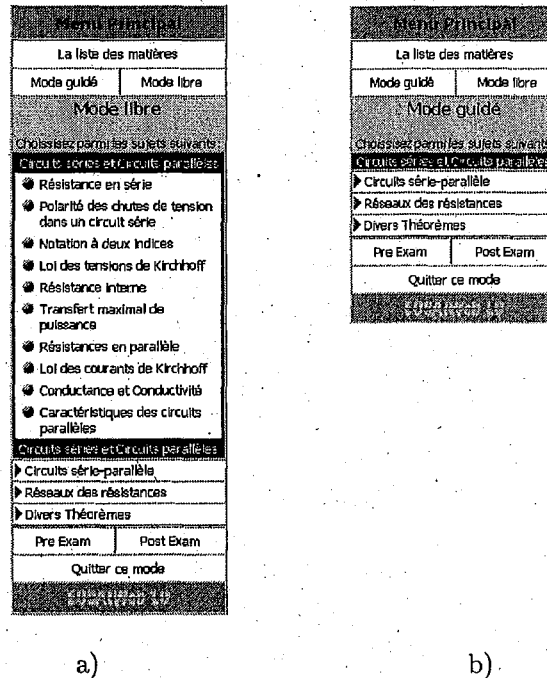


FIG. 6.46 – Menus adaptatifs a) en mode libre, et b) en mode guidé

En utilisant le modèle d'utilisateur, le système peut distinguer plusieurs états pédagogiques pour chaque page de cours, représenté dans la figure 6.46. Le contenu de la page peut être connu par l'utilisateur ou prête à être apprise. L'icône et la police de chaque lien présentées à l'utilisateur sont calculées dynamiquement à partir du modèle individuel d'utilisateur. Elles informent toujours l'utilisateur sur le type et l'état pédagogique du nœud derrière le lien. L'utilisation du modèle de recouvrement multi-points de vue (cf. section 5.4) permet d'annoter et ordonner les liens pour chaque élève.

6.5.4 Barres de navigation

Après le choix de suivre le parcours d'apprentissage proposé par le système, l'apprenant doit avoir dans chaque page les barres de navigation et les informations supplémentaires pour lui dire où il en est. Dans la partie haute de la page, l'apprenant doit avoir les indications sur le sujet, le titre d'unité, la nature d'unité et la page courante et le nom de page pour la présentation ou la question courante et le nombre de questions pour l'exercice (figure 6.47). Dans la partie basse de la page, l'apprenant doit avoir la barre de navigation pour aller à la page ou question

suivante ou bien retourner à la page ou question précédente. Dans le cas de l'exercice, la barre de navigation n'apparaît que lorsque l'apprenant a déjà répondu à la question, si l'apprenant n'a pas encore répondu, alors la barre de navigation est remplacée par la barre signalant «*Attente votre réponse*», comme présenter dans la figure 6.48.

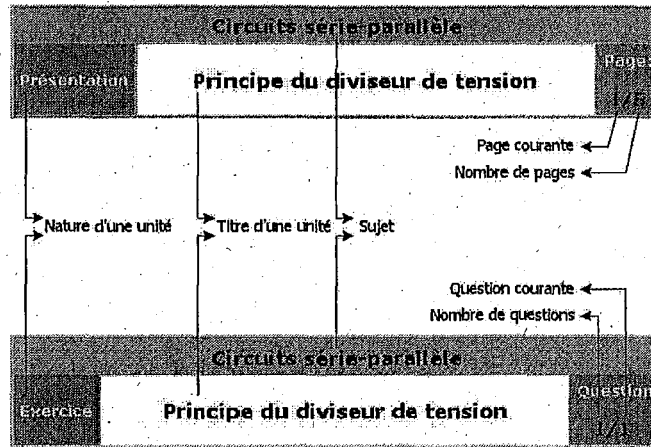


FIG. 6.47 – Informations dans la partie haute de la page de présentation

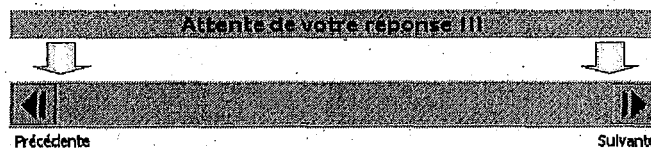


FIG. 6.48 – Barres de navigation

6.6 Fonctionnalités de l'environnement partagé

Un support de communication asynchrone ou synchrone est destiné à favoriser l'interaction, la coopération et le travail de groupe entre les enseignants, les étudiants et les enseignants et étudiants. Pour ce fait, chaque groupe dispose de trois outils de communication (figure 6.49) : un forum de discussion (destiné à faciliter le travail coopératif), une messagerie (l'instrument qui est utilisé pour envoyer les courriers électroniques entre les utilisateurs) et enfin un Chatroom pour réunir et changer des informations en ligne au sein du groupe.

6.6.1 Courrier électronique

Le courrier électronique⁴⁴ est l'outil le plus répandu pour des entreprises ainsi que pour le particulier. Il permet d'acheminer des courriers entre des personnes éloignées. L'appellation

44. Le nom anglais e-mail est resté dans le langage, et les utilisateurs parlent de leur adresse e-mail.

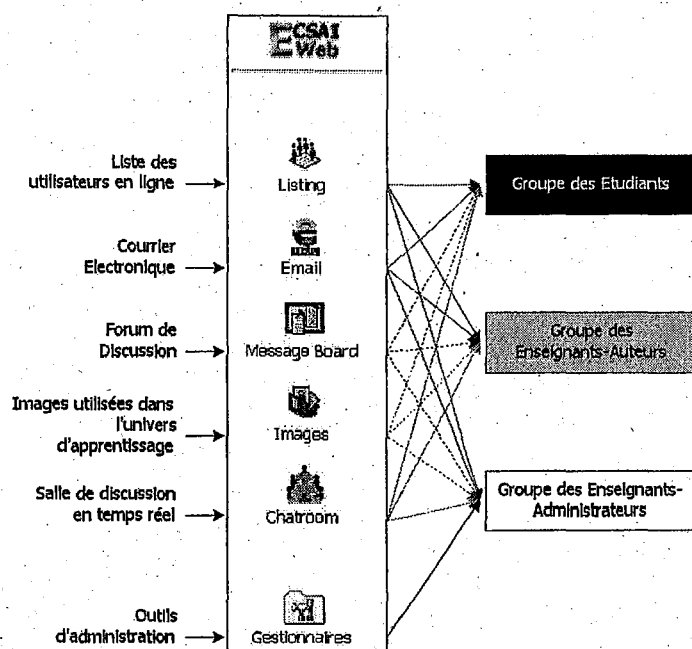


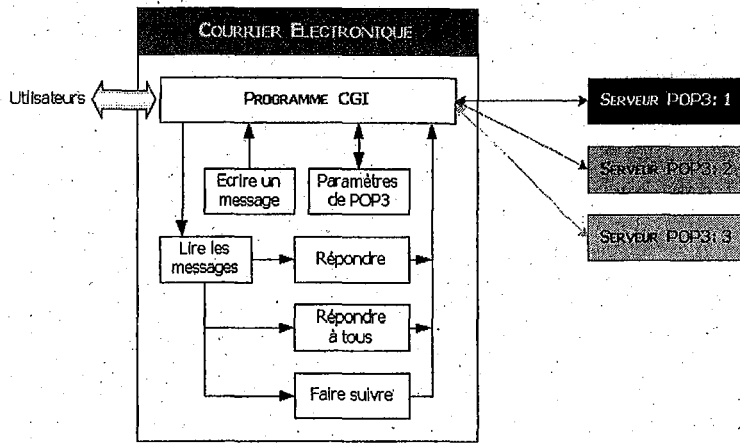
FIG. 6.49 – Organigramme des outils de l'environnement partagé

«*mél*» est le terme officiel mais encore peu utilisé. Nous essaierons d'employer le mot courrier électronique tout au long de cette section.

L'avantage du courrier électronique sur le téléphone ou sur le fax est considérable. En effet, il permet de joindre un correspondant avec des informations écrites tout comme le fax, mais qui peuvent être recopiées dans un document en mode texte. Par rapport au téléphone, les courriers électroniques permettent d'aller droit à l'essentiel, évitent les aléas des répondeurs et permettent de laisser des traces écrites. En outre nous pouvons lire notre courrier de n'importe où dans le monde lors de nos déplacements ce qui en fait un des meilleurs moyens de joindre son correspondant.

Les outils de messagerie électronique que nous essayons d'intégrer dans notre système permettent d'envoyer principalement des messages entre les utilisateurs de notre système mais sont valables en dehors du système. Le seul outil utilisé, en ce moment, est le protocole POP3⁴⁵ qui permet à un utilisateur connecté sur une ligne intermittente d'interroger son courrier situé dans la boîte aux lettres de son serveur ou de son fournisseur (figure 6.50.a). Notre système ne fournit pas d'adresse électronique, ni de boîte aux lettres donc avant de s'inscrire dans notre système les utilisateurs doivent avoir au moins une adresse e-mail. La page d'accueil (figure 6.50.b) représente l'entrée principale du courrier électronique. Si l'utilisateur a déjà saisi les paramètres POP3 de sa boîte aux lettres, il peut consulter tout de suite les courriers sinon le système demandera ses paramètres.

45. sigle anglais de *Post Office Protocol version 3*



a) L'organigramme des fonctionnalités du courrier électronique

Les boutons de la barre des outils sont : Lire les messages, Ecrire un message, Répondre, Répondre à tous, Faire suivre, et Régler les paramètres liés au serveurs POP3.

Sender	Subject	Size (Bytes)
Nadine =?iso-8859-1?Q?Beurn=E9?=<Nadine.Beurn@loria.fr>	Programmes =?iso-8859-1?Q?bilat=E9raux=20d=27=E9tudes?= et de recherche	2649
kcn@kmitnb.ac.th	=?TIS-620?Q?=A2=E8=D2=C7_=A1.=BE.44?= =	2125000
Helene Kirchner <Helene.Kirchner@loria.fr>	doc	880
"Defisc-immo" <edumont@angoosta.fr>	INVESTISSEZ VOS IMPOTS	2250

Note: mailhost.loria.fr = 4 message(s).

b) La barre des outils du courrier électronique

FIG. 6.50 – Outils de communication : courrier électronique

6.6.2 Forum de discussion

Alors que les documents électroniques entre individus ou au travers de groupes de diffusion sont stockés dans les boîtes aux lettres de chacun des correspondants, les contenus de forums ne sont pas envoyés à tous les utilisateurs, ils résident sur le serveur. Ils sont consultés par ceux qui sont intéressés par leur sujet (leur matière).

Une question ou un article comprend un titre, et un corps. Les réponses à une question ou à un article reprennent le même titre précédé par le mot Re: (pour réponse ou encore Re: Re: ..., représentée dans la figure 6.51).

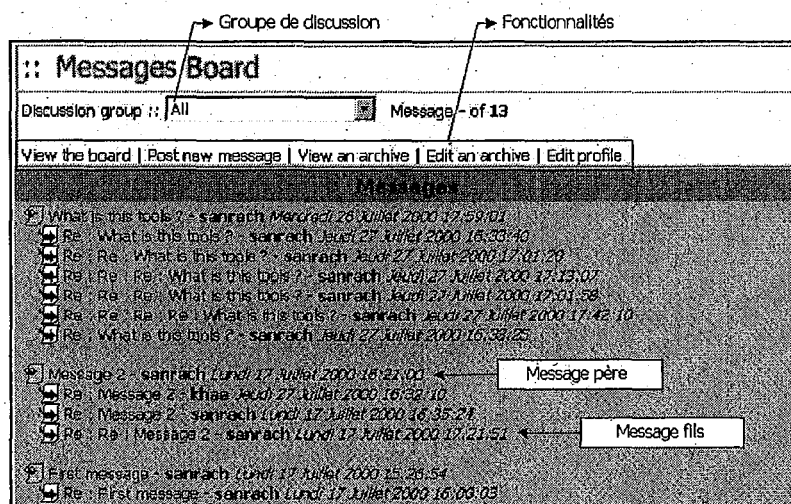


FIG. 6.51 – Outils de communication : forum de discussion

Un ensemble de « Article + Réponse » est appelé un *fil de discussion* (ou Thread en anglais) et regroupé pour faciliter la lecture suivie d'une discussion. En premier niveau, la fonction de lecture de forum nous présente les titres de ses fils de discussion, sans visualiser le corps du message. En sélectionnant un message, le corps du message et la partie réponse du message sont affichés (figure 6.52)

Chaque forum est appelé en anglais *Newsgroup*, chaque article d'un newsgroup est appelé une *News*. Les termes français les plus proches sont *forum et article*, mais il est probable que si nous employons ces termes peu de gens, sur Internet, nous comprennent. La tendance actuelle est d'utiliser le nouveau mot mais le terme n'est pas encore entré dans les moteurs et tout le monde parle de News.

Certains groupes sont dits modérés, lorsque les articles qui sont envoyés sont contrôlés par un ou plusieurs responsables (appelés modérateurs) qui pourront accepter ou refuser la publication de l'article.

De notre part, il y a trois groupes principaux: *All* (groupe général), *Students* et *Teacher*, et puis un groupe associé à chaque matière. Les utilisateurs d'ECSAIWEB doivent s'inscrire à chaque groupe, c.-à.-d., les utilisateurs ne peuvent lire et poser les questions que dans les groupes où ils sont inscrits. Mais les utilisateurs peuvent consulter les messages de tous les groupes en utilisant le nom d'utilisateur «*guest*».

Messages Board	
Discussion group :: All [v] << < Message 6 of 13 >> >>	
View the board Post new message View an archive Edit an archive Edit profile	
What is this tools ?	
Posted By:	sanrach
Email:	sanrach@loria.fr
Date:	Mercredi 26 Juillet 2000 17:59:01
Archive:	Available
Message:	Hi all !!!, This tools is used for discussing between us. Jer.
Responses:	
1. Re : What is this tools ? -- sanrach	
2. Re : What is this tools ? -- sanrach	
Name: [sanrach v]	
Subject: [Re : What is this tools ?]	

FIG. 6.52 – Visualisation des messages

6.6.3 ChatRoom

ECSAIWEB propose des salles de discussion réservées pour assurer l'interaction en temps réel entre les utilisateurs d'un groupe. Les utilisateurs peuvent dialoguer à deux ou à plusieurs avec tous les utilisateurs du groupe. Les salles virtuelles que nous avons préparées sont la salle pour tous (*All*), la salle pour les étudiants (*Students*), la salle pour les enseignants (*Teachers*) et une salle associée à chaque matière.

Dans chaque salle, l'administrateur peut contrôler les utilisateurs : admettre ou refuser qu'une personne utilise la salle, effacer tous les messages écrits dans la salle. Les commandes sont les suivantes :

- **Post/Reload** – poser un message dans la zone entrée et recharger les messages,
- **Clear** – effacer le message de la zone entrée,
- **Help** – afficher une aide sur les commandes utilisées,
- **Rooms** – passer une salle à l'autre,
- **Size et Color** – changer la taille et la couleur de police,
- **Exit** – quitter la salle de discussion.

Les commandes pour *tous les utilisateurs* sont :

- **/color [HEX ou nom de couleur or BOLD, ITALIC, BLINK] [message]** – Afficher le message dans le couleur choisie, gras, italique et clignotante respectivement,
- **/d [message]** – montre les informations sur la personne,
- **/nick [newnick]** – changer le nom,
- **/quit [quit message]** – fermer la session,
- **/cons** – afficher une galerie d'émoticons,

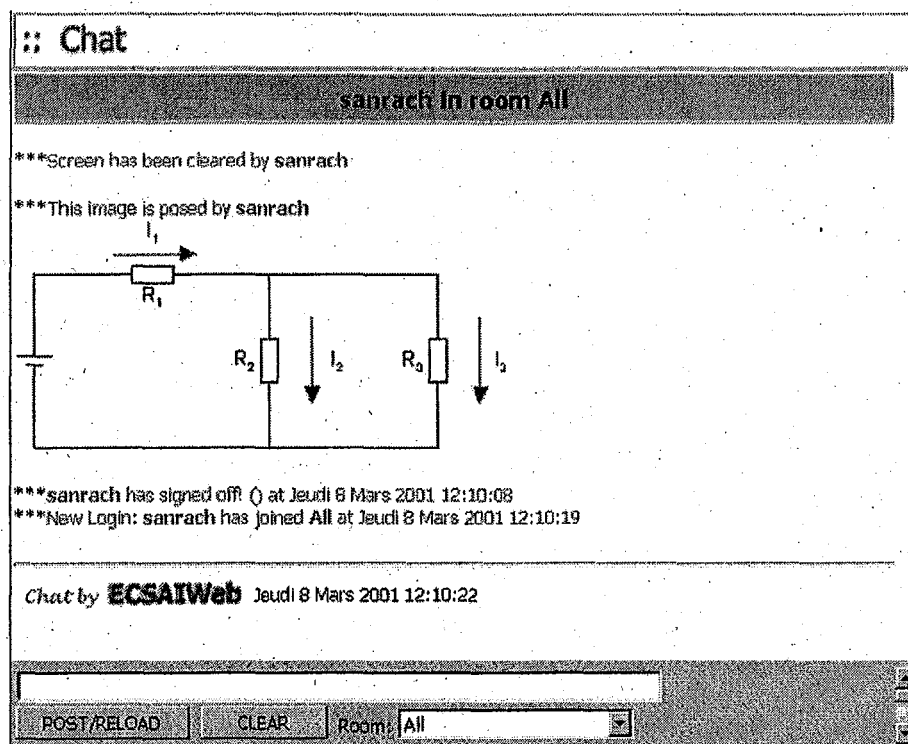


FIG. 6.53 – Outils de communication : salle de discussion en temps réel (Chatroom)

- **/fl** – rincer les messages privés,
- **/msg [nickname] [message]** – envoyer un message privé à un utilisateur,
- **/rules** – afficher les règles pour l'utilisation de la salle de discussion,
- **/whoall** – afficher tous les utilisateurs entrés dans toutes les salles.

Les commandes supplémentaires pour l'*administrateur* des salles sont :

- **/cls** – effacer tous les messages dans la salle de discussion,
- **/kick [nickname]** – interdire un utilisateur spécifique d'entrer dans la salle de discussion,
- **/ban [nickname]** – interdire une adresse IP spécifique d'entrer dans la session,
- **/send [message]** – envoyer un énoncé à toutes les salles de discussion,

6.7 Fonctionnalités de l'outil administratif

Les fonctionnalités mises en œuvre dans ECSAIWEB demandent de l'expertise technique pour administrer le système: création de comptes pour les intervenants, gestion du contrôle d'accès au système, installation du système, configuration des moyens de communications électroniques, maintenance, prises des copies de sauvegarde, création et modification des interfaces, des feuilles de styles et des profils des matières. La structure des fonctionnalités de cet environnement est présentée dans la figure 6.54.

La gamme d'outils autres que ceux fournissant un support à la communication et fournis par l'ECSAIWEB à l'intention des gestionnaires du système constitue une nouvelle catégorie

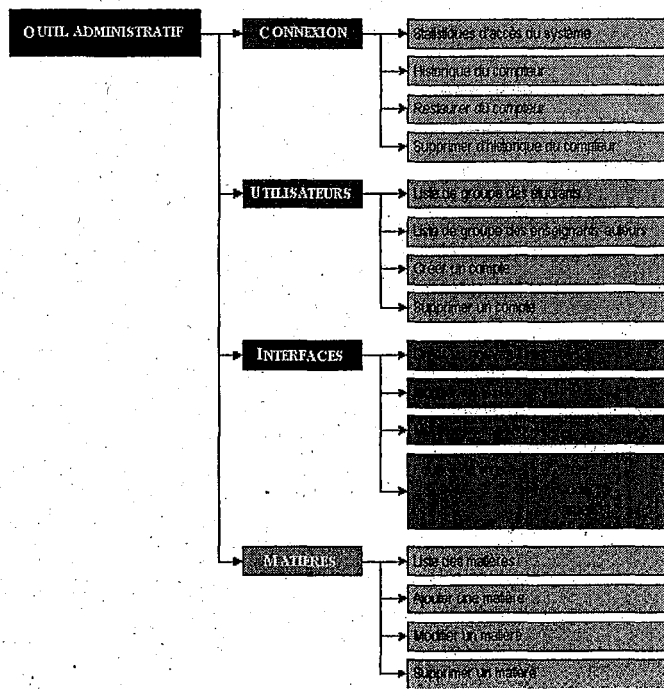


FIG. 6.54 – Outil administratif : diagramme des fonctionnalités

:: Gestionnaires

Server's name : **www-internet.loria.fr**
and system author name : **sanrach**

Histories	Users	Interfaces	Subject
Modify a subject			
Identifier :: <input type="text" value="220102"/>			
Description :: <input type="text" value="Analyse du circuit électrique en régime continu"/>			
Language :: <input type="text" value="Française"/>			
Author/Teacher (s) :: <input type="text" value="sanrach"/> <small>[separated by comma (,)]</small>			
Integrated Tools In this subject :: <ul style="list-style-type: none"> <input type="checkbox"/> Email <input checked="" type="checkbox"/> Chat IRC <input checked="" type="checkbox"/> Forum <input checked="" type="checkbox"/> Homework Distribution <input checked="" type="checkbox"/> Shared Tables <input checked="" type="checkbox"/> Others 			
<input type="button" value="Save"/>		<input type="button" value="Reset All"/>	

FIG. 6.55 – Exemple de la fonction «modifier une matière»

de fonctionnalités que nous avons intégrées. Ces outils peuvent être de natures distinctes et de degrés divers de sophistication en vue de permettre :

- la gestion des intervenants (disponible pour les enseignants-auteurs),
- la gestion des classes (pas encore réaliser),
- la gestion des historiques (consultation seulement),
- la gestion de la base de données de cours ou d'autres bases de données maintenues sur le site serveur de la plateforme (ne pas encore réaliser),
- la gestion des interfaces : multilinguisme (disponible dans la version actuelle, cf. figure 6.54), personnalisation,
- des interventions ou des diagnostics pouvant être utilisés à même le serveur ou à partir d'un poste distant (disponible dans la version actuelle en utilisant le navigateur Web),
- le monitoring des ressources consommées par ses utilisateurs (disponible dans la version actuelle).

6.8 Conclusion

Comme nous l'avons écrit au début de ce chapitre, notre système est de type client-serveur. Nous avons décidé d'employer l'architecture centralisée. L'exécution de tous nos programmes est faite au sein du serveur avant de renvoyer le résultat obtenu au poste de client et donc toutes les informations doivent être stockées sur le serveur. Chaque requête de commandes et les paramètres fournis par l'utilisateur doit être passés par le serveur Web, puis le serveur cherche le programme à exécuter en offrant tous les paramètres fournis par les utilisateurs.

Après avoir défini toutes les fonctionnalités de notre environnement, les étapes suivantes sont d'implanter les programmes que nous présenterons dans le chapitre suivant.

Chapitre 7

Architecture et Implantation réalisée

7.1 Introduction

Après la définition de toutes les fonctionnalités (cf. chapitre 6) et de la présentation de tous les modèles (cf. chapitre 4 et chapitre 5), nous décrivons dans ce chapitre l'architecture et l'implantation de notre système.

Pour avoir des environnements utilisables qui nous permettent de valider les fonctionnalités et les modèles d'une part et nos idées d'autre part, se sont posés de nouveaux problèmes: le choix du langage de programmation adapté au client-serveur et le choix du navigateur de documents Web. Nous décrivons d'abord les contraintes de mise en œuvre de notre environnement, les choix des outils d'implantation (section 7.2). Ensuite, nous présentons l'architecture complète de l'ECSAIWEB (section 7.3). Puisque notre système est séparé en plusieurs environnements, chaque environnement est associé, au moins, à un programme principal. Chaque programme est aussi lié à plusieurs modules ou sous-programmes. Alors il est important de présenter toutes les organisations de nos programmes (section 7.4) et de les détailler. Les programmes principaux et les sous-programmes seront donc présentés dans la section 7.5 et dans la section 7.6 respectivement. Le générateur des menus adaptatifs est fait l'objet de la section 7.8. A la fin de ce chapitre, nous décrivons l'implantation du générateur d'un parcours d'apprentissage (section 7.9).

7.2 Contraintes de mise en œuvre

Il faut noter que nous récrivons entièrement le programme parce que l'ancienne version est écrite en C sur un poste PC autonome. La nature de l'exécution de programme lui-même est changée comme nous avons indiqué dans la section conclusion du chapitre précédent. Cela nous permet de réorganiser le système et l'adapter au nouveau mode d'exploitation. Le langage de programmation choisi est le langage Perl⁴⁶ [WCS91] [WCS96][Sta]. Ce langage nous permet d'écrire facilement le prototype du type client-serveur. L'exécution générale des programmes ECSAIWEB est faite sur le serveur et puis les résultats obtenus sont envoyés au navigateur Web de la poste de client.

Il existe, dès aujourd'hui, plusieurs navigateurs de documents WWW, le plus récent est le plus performant. Mais ils ne sont pas entièrement compatibles. Il n'y a que les fonctionnalités de base qui sont compatibles. Par exemple, si nous utilisons les balises HTML simples, cela ne pose aucun problème. Contrairement, si nous voulions utiliser l'HTML dynamique (DHTML), il

46. sigle anglais de *Practical Extraction and Report Language*

nous pose tout de suite des problèmes car Exploreur et ses descendants, et Netscape ou Communicateur ne les traitent pas de la même façon. Par exemple, pour créer le document WWW multicouche, Explorateur utilise l'objet «*document.all*» tandis que Netscape utilise les objets «*document.layer*», ceux-ci ne comptent pas encore leurs sous-objets ou classes. Alors les critères des choix de navigateur ne restent que les plateformes parce qu'au début de cette thèse nous avons souhaité que notre système soit des plateformes indépendantes. Donc nous avons décidé de choisir le navigateur chez Netscape parce qu'il peut exploiter dans plusieurs systèmes opérateurs : Windows, Mac, Unix et etc.,

En résumé, ECSAIWEB est développé majoritairement en Perl. Il groupe 5 programmes principaux : *ecsaiweb.pl*, *auteur.pl*, *tuteur.pl*, *tools.pl* et *gestion.pl*, et 4 modules liés à chaque modèle : *Domain.pm*, *Student.pm*, *Presentation.pm* et *LireParam.pm*. Pour que notre système soit plus interactif, nous avons décidé d'ajouter quelques fonctionnalités en script du Java : le menu pop-up, et en applets de Java : la visualisation d'un parcours d'apprentissage. En totalité, il représente environ 20500 lignes de codes dont 65% sont consacrées au noyau fonctionnel, 35% à la gestion de l'interface.

Nous avons présenté dans le chapitre 2 section 2.2, chapitre 4 et chapitre 5 des organisations des modèles formels de notre système. Ces modèles ont ses propres modules qui, en effet, nous permettent de mettre en œuvre de créer rapidement le prototype.

7.3 Architecture du système ECSAIWEB

Nous nous sommes attachés à développer un environnement pédagogique multiple plateformes accessibles depuis tout ordinateur connecté au réseau Internet et dont l'utilisation se devrait d'être aussi intuitive que possible, tant pour les enseignants-auteurs que pour les étudiants. Suivi de l'étude approfondie sur les systèmes d'apprentissage intelligent, les hypermédias adaptatifs et l'ECSAI, nous proposons ensuite l'architecture du système ECSAIWEB, représenté dans la figure 7.1. Nous décidons donc de séparer l'ECSAIWEB en trois environnements : l'environnement auteur, l'environnement élève et l'environnement partagé. Les environnements bien que similaires par leur terminologie n'en sont pas moins différents par leurs contenus. Nous décrivons le résumé le rôle attribué à chacun des environnements.

7.3.1 Environnement auteur

L'environnement auteur, grâce à des fonctionnalités d'édition, permet aux enseignants-auteurs de créer, modifier et gérer leurs cours. Chaque cours peut comprendre plusieurs unités d'apprentissage selon qu'il s'agit de textes de présentation (*apprendre en lisant*) ou d'exercice : questions à choix multiple, questions vraie/fausse ou questions à compléter (*apprendre en faisant*). Autour de l'existence des bases de données (base des items, base des unités logiques, base des unités physiques et base des utilisateurs), et à travers l'utilisation de formulaires appropriés, l'enseignant-auteur peut indexer, stocker, créer et modifier tout type de document. Dès qu'un document est intégré dans les bases, il peut être consulté et utilisé par tous les utilisateurs ayant les droits d'accès nécessaires.

7.3.2 Environnement élève

L'environnement élève diffère de celui de l'enseignant par le fait qu'il ne dispose pas des fonctionnalités d'édition, pas plus qu'il n'a accès directement aux bases de données. Pour sélectionner un cours, il peut choisir un mode d'apprentissage, soit mode libre soit mode guidé. Puis

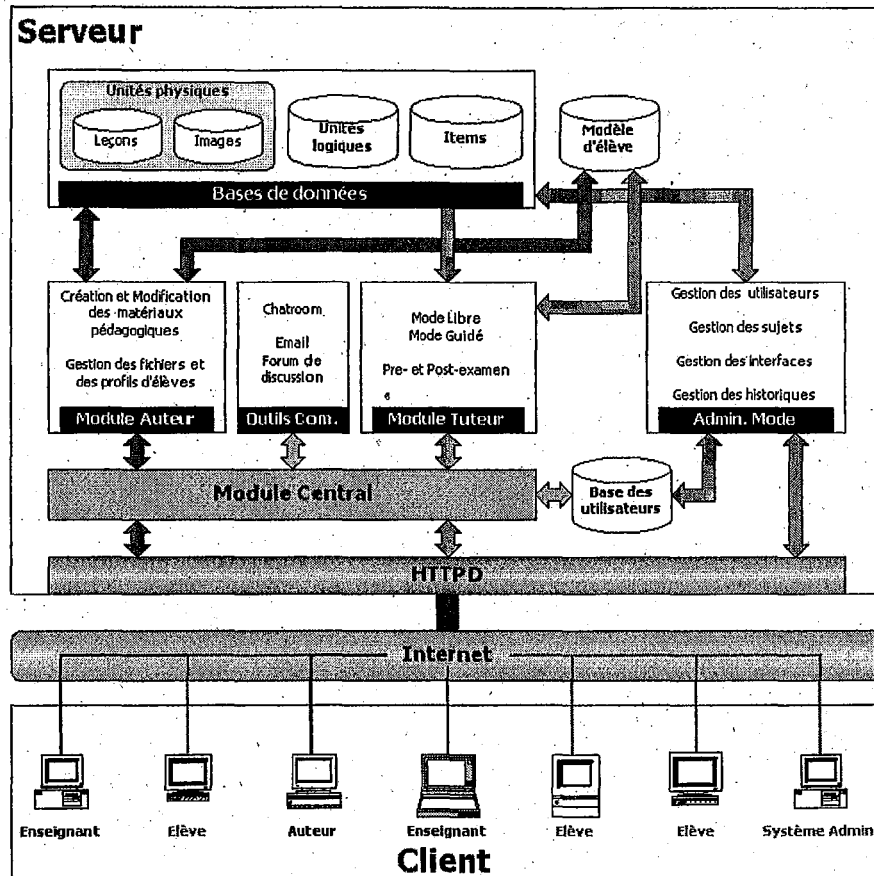


FIG. 7.1 – Architecture d'ECSAIWEB

il doit suivre le parcours d'apprentissage proposé par le système dans chaque mode. Avec les technologies hypermédias adaptative, l'élève peut savoir où il en est dans le cours, quels sont des objectifs pédagogiques déjà atteint.

7.3.3 Environnement partagé

L'environnement partagé, en outre, un support à la communication asynchrone et synchrone est destiné à favoriser l'interaction, la coopération et le travail de groupe entre les professeurs, les étudiants et les professeurs et les étudiants. Pour ce faire, chaque groupe dispose de trois outils de communication : un forum de discussion, une sorte de « *Newsgroup* » et un courrier électronique destiné à faciliter le travail coopératif, et des salles de discussion (Chatroom) destiné à discuter en temps réel entre utilisateurs.

Nous intégrons aussi le module central qui contrôle l'accès des utilisateurs et fournit l'environnement associé à chaque utilisateur dans cet environnement.

7.4 Organisation des programmes

Selon nos besoins, nous décrivons dès maintenant l'organisation de nos programmes présentés dans la figure 7.2. Schématiquement, ECSAIWEB est constitué en trois parties : l'ensemble des programmes principaux exécutés au sien de serveur (cf. section 7.5), l'ensemble des modules ou sous-programmes associés au modèle du domaine et modèle des élèves (cf. section 7.6) , et l'ensemble des bases de données utilisées dans le système (cf. section 7.7).

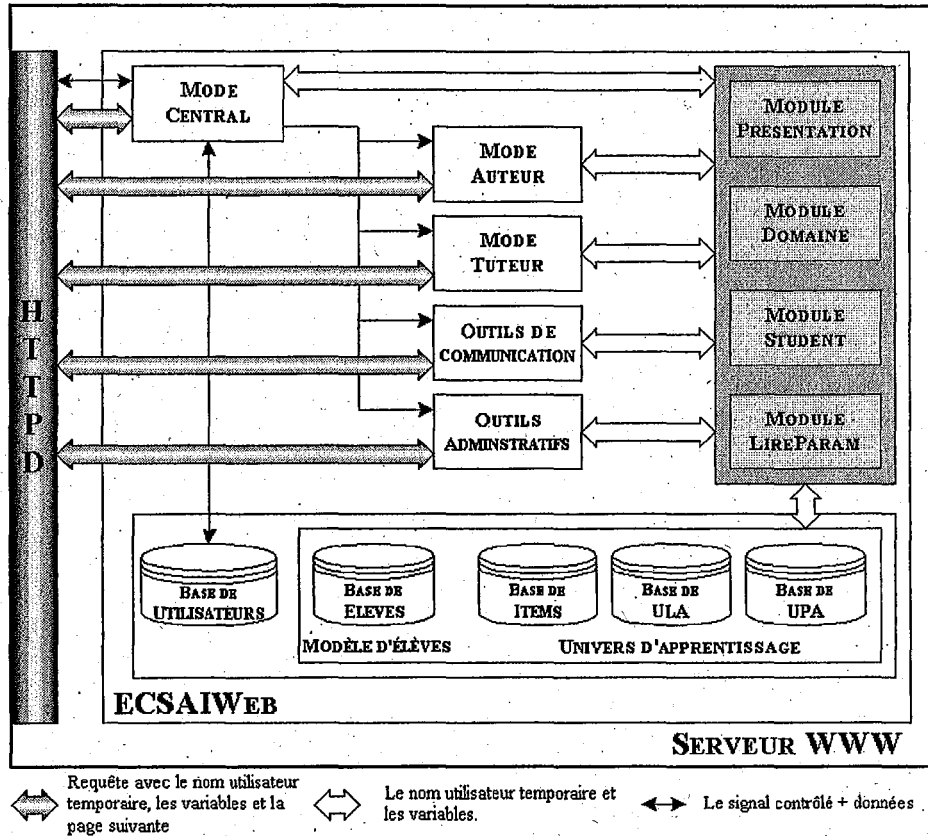


FIG. 7.2 – Organisation des programmes du système ECSAIWEB

7.5 Programmes principaux

Les programmes principaux (première couche, au niveau interface pour communiquer avec les utilisateurs), selon leurs fonctionnalités : gérer les interfaces et communiquer avec les utilisateurs, sont distingués en cinq programmes donc sont regroupés en 3 environnements :

- **Environnement auteur** – le programme principale associé est de «*Auteur.pl*»,
- **Environnement élève** – le programme principale associé est de «*Tuteur.pl*»,
- **Environnement partagé** – les programmes principaux associés sont des programmes de contrôle d'accès des utilisateurs «*ECSAIWeb.pl*», des outils de communication («*Tools.pl*») et la gestion du système («*Gestion.pl*»).

7.6 Modules intermédiaires

Les sous-programmes sont les modules séparés des programmes principaux. Ces modules sont les intermédiaires entre les programmes principaux et les bases de données (voir la figure 7.2). Selon leurs fonctionnalités, nous les séparons en quatre modules : module de paramètres (ECSAIWeb::LireParam, section 7.6.1), module de domaine (ECSAIWeb::Domain, section 7.6.2), module de présentations (ECSAIWeb::Presentation, section 7.6.3) et module de gestion des profils de l'apprenant (ECSAIWeb::Student, section 7.7).

7.6.1 Module LireParam

Le but de cette section est de présenter un module d'aide à la gestion de tous les paramètres nécessaires : les variables d'interfaces multi-langues, la statistique de la connexion au système, le filtre de requête des commandes et etc. Pour l'interface multi-langues, nous utilisons les tableaux associatifs de scalaires (les variables commençant par %) pour définir les messages apparaissant aux utilisateurs. Par exemple si l'utilisateur choisit la langue française pour les interfaces, le module LireParam doit charger tous les messages et stocker dans les tableaux des variables associées avant de passer aux autres programmes ou modules. Les sous-programmes principaux existants dans ces modules sont le sous programme LireSystemParam (lire les paramètres du système), CgiError (afficher les messages erreurs), UpLoad (filtre de requête des commandes), et etc. L'exemple du script représenté dans la figure 7.3 est un sous-programme pour filtre et traiter la requête des commandes dans les formulaires, cette partie est la définition de toutes les variables de formulaire. La figure 7.4 traite le formulaire du type «GET». La figure 7.5 représente le script pour traiter la formulaire du type «*multipart/form-data*». De plus, avec le type «*multipart/form-data*», nous pouvons attacher les fichiers en tant les commandes pour envoyer au serveur Web. Le script représenté dans la figure 7.5 présente la manipulation de toutes les informations sur les formulaires de ce type.

```
sub Upload {
    local ($perlwarn);
    $perlwarn = $^W;
    $^W = 0;
    local (*in) = shift if _; # CGI input
    local (*incfn, # Client's filename (may not be provided)
          *inct, # Client's content-type (may not be provided)
          *insfn) = _; # Server's filename (for spooled files)
    local ($len, $type, $meth, $errflag, $cmdflag, $got, $name);
    binmode(STDIN); # we need these for DOS-based systems
    binmode(STDOUT); # and they shouldn't hurt anything else
    binmode(STDERR);
    # Get several useful env variables
    $type = $ENV{'CONTENT_TYPE'};
    $len = $ENV{'CONTENT_LENGTH'};
    $meth = $ENV{'REQUEST_METHOD'};
    if ($len > $lireparam::maxdata) {
        &CgiDie("LireParam.pm: Request to receive too much data: $len bytes\n");
    }
}
```

FIG. 7.3 — Sous-programme pour filtrer des requêtes de commandes : partie de définition des variables

```

if (!defined $meth || $meth eq " || $meth eq 'GET' ||
    $meth eq 'HEAD' ||
    $type eq 'application/x-www-form-urlencoded') {
    local ($key, $val, $i);

    # Read in text
    if (!defined $meth || $meth eq " ) {
        $in = $ENV{'QUERY_STRING'};
        $cmdflag = 1; # also use command-line options
    } elsif ($meth eq 'GET' || $meth eq 'HEAD') {
        $in = $ENV{'QUERY_STRING'};
    } elsif ($meth eq 'POST') {
        if (($got = read(STDIN, $in, $len) != $len))
            {$errflag="Short Read: wanted $len, got $got \n"};
    } else {
        &CgiDie("Lireparam.pm: Unknown request method: $meth\n");
    }

    in = split(/[&:]/, $in);
    push(in, ARGV) if $cmdflag; # add command-line parameters

    foreach $i (0 .. $#in) {
        # Convert plus to space
        $in[$i] =~ s/\+ / /g;

        # Split into key and value.
        ($key, $val) = split(=/, $in[$i], 2); # splits on the first =

        # Convert %XX from hex numbers to alphanumeric
        $key =~ s/%{[A-Fa-f0-9]{2}}/pack("c", hex($1))/ge;
        $val =~ s/%{[A-Fa-f0-9]{2}}/pack("c", hex($1))/ge;

        # Associate key and value
        $in{$key} .= "\0" if (defined($in{$key})); # \0 is the multiple separator
        $in{$key} .= $val;
    }
}

```

FIG. 7.4 – Partie de traitement des formulaires du type «GET»

```

} elsif ($ENV{'CONTENT_TYPE'} = m/^(multipart/form-data-#) {
    # for efficiency, compile multipart code only if needed.
    $errflag = !(eval <<'END_MULTIPART');

    local ($buf, $boundary, $head, $heads, $cd, $ct, $fname, $ctype, $blen);
    local ($bpos, $lpos, $left, $amt, $fn, $ser);
    local ($bufsize, $maxbound, $writefiles) = ($lireparam::bufsize, $lireparam::maxbound,
        $lireparam::writefiles);

    # The following lines exist solely to eliminate spurious warning messages
    $buf = "";

    ($boundary) = $type =~ /boundary="([^\"]+)"/; #"; # find boundary
    ($boundary) = $type =~ /boundary=(\S+)/ unless $boundary;
    &CgiDie ("Boundary not provided: probably a bug in your server") unless $boundary;
}

```

FIG. 7.5 – Partie de traitement des formulaires du type «multipart/form-data» ou «POST»

```

$boundary = "-" . $boundary;
$blen = length($boundary);

if ($ENV'REQUEST_METHOD' ne 'POST') {
    &CgiDie("Invalid request method for multipart/form-data: $meth\n");
}

if ($writefiles) {
    local($me);
    stat($writefiles);
    $writefiles = "/tmp" unless -d _ && -w _;
    # ($me) = $0 =~ m#(.*?)$#;
    $writefiles .= "/$lireparam::filepre";
}
$left = $len;

PART: # find each part of the multi-part while reading data
while (1) {
    die $@ if $errflag;
    $amt = ($left > $bufsize + $maxbound - length($buf) ? $bufsize + $maxbound - length($buf) : $left);
    $errflag = (($got = read(STDIN, $buf, $amt, length($buf))) != $amt);
    die "Short Read: wanted $amt, got $got\n" if $errflag;
    $left -= $amt;

    $in{$name} .= "\0" if defined $in{$name};
    $in{$name} .= $fn if $fn;

    $name =~ s/[^\w+]/ /; # This allows $insfn{$name} to be untainted
    if (defined $1) {
        $insfn{$1} .= "\0" if defined $insfn{$1};
        $insfn{$1} .= $fn if $fn;
    }
}
BODY:
while (($bpos = index($buf, $boundary)) == -1) {
    if ($left == 0 && $buf eq "") {
        foreach $value (values %insfn) {
            unlink(split("\0", $value));
        }
        &CgiDie("Lireparam.pm: reached end of input while seeking boundary of multipart. Format of CGI
input is wrong\n");
    }
    die $! if $errflag;
    if ($name) { # if no $name, then it's the prologue - discard
        if ($fn) { print FILE substr($buf, 0, $bufsize); }
    } else { $in{$name} .= substr($buf, 0, $bufsize); }
}
$buf = substr($buf, $bufsize);
$amt = ($left > $bufsize ? $bufsize : $left); # $maxbound == length($buf);
$errflag = (($got = read(STDIN, $buf, $amt, length($buf))) != $amt);
die "Short Read: wanted $amt, got $got\n" if $errflag;
$left -= $amt;
}
if (defined $name) { # if no $name, then it's the prologue - discard
    if ($fn) { print FILE substr($buf, 0, $bpos-2); }
} else { $in{$name} .= substr($buf, 0, $bpos-2); } # kill last \r\n
}
close(FILE);
last PART if substr($buf, $bpos + $blen, 2) eq "-";
substr($buf, 0, $bpos + $blen + 2) = "";

```

FIG. 7.5 – Partie de traitement des formulaires du type «multipart/form-data» (suite)

```

$amt = ($left > $bufsize + $maxbound - length($buf) ? $bufsize + $maxbound - length($buf) : $left);
$errflag = (($got = read(STDIN, $buf, $amt, length($buf))) != $amt);
die "Short Read: wanted $amt, got $got\n" if $errflag;
$left -= $amt;

undef $head; undef $fni;
HEAD:
while (($lpos = index($buf, "\r\n\r\n")) == -1) {
    if ($left == 0 && $buf eq "") {
        foreach $value (values %insfn) {
            unlink(split("\0", $value));
        }
        &CgiDie("Lireparam : reached end of input while seeking end of headers. Format of CGI input is
wrong.\n$buf");
    }
    die "$@" if $errflag;
    $head .= substr($buf, 0, $bufsize);
    $buf = substr($buf, $bufsize);
    $amt = ($left > $bufsize ? $bufsize : $left); # $maxbound == length($buf);
    $errflag = (($got = read(STDIN, $buf, $amt, length($buf))) != $amt);
    die "Short Read: wanted $amt, got $got\n" if $errflag;
    $left -= $amt;
}
$head .= substr($buf, 0, $lpos+2);
push (@in, $head);
@heads = split("\r\n", $head);
($cd) = grep (/^s*Content-Disposition:/i, @heads);
($ct) = grep (/^s*Content-Type:/i, @heads);

($name) = $cd =~ /\bname="([~"]+)"/i; #";
($name) = $cd =~ /\bname=([~\s;]+)/i unless defined $name;

($fname) = $cd =~ /\bfilename="([~"]*)"/i; #"; # filename can be null-str
($fname) = $cd =~ /\bfilename=([~\s;]+)/i unless defined $fname;
$incfn{$name} .= (defined $in{$name} ? "\0" : "");
(defined $name ? $name : "");

($ctype) = $ct =~ /^s*Content-type:\s*"([~"]+)"/i; #";
($ctype) = $ct =~ /^s*Content-Type:\s*"([~\s;]+)/i unless defined $ctype;
$inct{$name} .= (defined $in{$name} ? "\0" : ""); $ctype;

if ($writefiles && defined $name) {
    $ser++;
    $fn = $writefiles . ".$ser";
    open (FILE, ">$fn") || &CgiDie("Couldn't open $fn\n");
    binmode (FILE); # write files accurately
}
substr($buf, 0, $lpos+4) = "";
undef $name;
undef $ctype;
}

```

FIG. 7.5 – Partie de traitement des formulaires du type «multipart/form-data» (suite)

7.6.2 Module DOMAIN

Après la définition des modèles du domaine (cf. chapitre 4), nous expliquons maintenant la mise en œuvre du prototype. Cette partie contient 3 parties : les opérations sur l'item d'évaluation, les opérations sur l'unité logique d'apprentissage et le générateur du parcours d'apprentissage.

- **Item d'évaluation** : Les opérations que nous avons définies dans le chapitre 4 sont transcrites en programme Perl dans la figure 7.6.

```

sub ValueItem{                               ← Opération définie dans la section 4.2.3.a
    local($Item)=@_;
    return 0 if !$Item;
    my(%Mg, $a, $b, $c, $Value, $ob, $mg, $sa, $sf);
    $Mg{'q'}=4; $Mg{'v'}=3; $Mg{'a'}=2; $Mg{'i'}=1;
    $a=1; $b=1; $c=1;
    ($ob, $mg, $sa, $sf)=split(/\{(\;,\;)\}/, $Item);
    $Value= $a*$Mg{$mg} + $b*$sa + $c*$sf;
    return $Value;
}

sub EcartItem{                               ← Opération définie dans la section 4.2.3.b
    local($Item1, $Item2)=@_;
    my ($Item1, $Item2, $mg1, $mg2, $dx1, $dx2, $dy1, $dy2, $Item, $Min, $dx, $dy);
    return $Item1 if (($Item2 eq "aucun")||(!$Item2));
    ($Item1, $mg1, $dx1, $dy1)=split(/\{(\;,\;)\}/, $Item1);
    ($Item2, $mg2, $dx2, $dy2)=split(/\{(\;,\;)\}/, $Item2);
    return $Item2."(-.$mg2."-.$dx2."-.$dy2.)" if (($Item1 eq "aucun")||(!$Item1));
    if ($Item1 ne $Item2){
        return $Item1." ".$Item2;
    }
    $Min=&Min($mg1, $mg2); $dx=$dx1-$dx2; $dy=$dy1-$dy2;
    return 0 if (($Min eq "0") && ($dx eq "0") && ($dy eq "0"));
    $Item=$Item1."($Min.".".$dx.".".$dy.)";
    return $Item;
}

sub SommeItem{                               ← Opération définie dans la section 4.2.3.c
    local($Item1, $Item2)=@_;
    my ($Item1, $Item2, $mg1, $mg2, $dx1, $dx2, $dy1, $dy2, $Item);
    ($Item1, $mg1, $dx1, $dy1)=split(/\{(\;,\;)\}/, $Item1);
    ($Item2, $mg2, $dx2, $dy2)=split(/\{(\;,\;)\}/, $Item2);
    return 0 if ($Item1 ne $Item2);
    $Item=$Item1."(&Max($mg1, $mg2).".".$(dx1+$dx2).".".$(dy1+$dy2).)";
    return $Item;
}

```

FIG. 7.6 – Script d'une partie de module ECSAIWeb::Domain, opérations des items

- **Unité logique d'apprentissage** : Les opérations sur l'unité ULA sont de calculer les liens entre les unités. Chaque opérateur est transmis en programme Perl représenté dans la figure 7.7.
- **Générateur des liens** : Cette fonction est employée pour créer les liens entre les unités logiques d'apprentissage. Les enseignants-auteurs l'utilisent pendant la phase de construction des matériaux pédagogiques pour vérifier l'enchaînement des unités logiques dans l'univers d'apprentissage. Le système s'en sert pour générer automatiquement le parcours d'apprentissage pour chaque apprenant. L'une des opérations importantes est présentée dans la figure 7.8.

```

sub AccessibleApres{
    ← Opération définie dans la section 4.3.2.a
    local($UnitCourant, $UnitX) = @_ if @_;
    my(@WF, @WP, @WA, @N1, @A, @N)=();
    my ($Cnt);
    #Charger le niveau des connaissances d'une unite courante
    (@WP) = split(/ et /, &LireUnite($UnitCourant,"Antecedent"));
    $Cnt=0;
    while ($WP[$Cnt]){
        $WP[$Cnt]="1" if (($WP[$Cnt] eq "aucune")||($WP[$Cnt] eq ""));
        $Cnt++;
    }
    (@WF) = split(/ et /, &LireUnite($UnitCourant,"Prerequis"));
    $Cnt=0;
    while ($WF[$Cnt]){
        $WF[$Cnt]="1" if (($WF[$Cnt] eq "aucun")||($WF[$Cnt] eq ""));
        $Cnt++;
    }
    (@WA) = split(/ et /, &LireUnite($UnitCourant,"Apport"));
    @N1=&Union(@WF, 0, @WA);
    #Charger le niveau des connaissances d'une unite quelleconque
    (@A) = split(/ et /, &LireUnite($UnitX,"Antecedent"));
    $Cnt=0;
    while ($A[$Cnt]){
        $A[$Cnt]="1" if (($A[$Cnt] eq "aucune")||($A[$Cnt] eq ""));
        $Cnt++;
    }
    (@N) = split(/ et /, &LireUnite($UnitX,"Prerequis"));
    $Cnt=0;
    while ($N[$Cnt]){
        $N[$Cnt]="1" if (($N[$Cnt] eq "aucun")||($N[$Cnt] eq ""));
        $Cnt++;
    }
    my $tt1=&Inclusion(@A, 0, @WP); # A inclus WP (Inclusion sur l'ensemble)
    my $tt2=&InclusDans(@N, 0, @N1); # N inclus dans WA+WF
    return 1 if ($tt1 && $tt2);
    return 0;
}

sub AccessiblePour{
    ← Opération définie dans la section 4.3.2.a
    local ($Unit, $Apprenant) = @_ if @_;
    my (@A, @N, @P, @C, @tmp, @B)=();
    #Charger le niveau des connaissances d'une unite $Unit2
    # Antecedent de l'unité
    (@A) = split(/ /, &LireModule("AVANT"));
    my $Count=0;
    my ($stest1, $stest2)=(0);
    while ($A[$Count]){
        $A[$Count]="U".$A[$Count] if ($A[$Count]!~/U/);
        $Count++;
    }
    (@N) = split(/ et /, &LireUnite($Unit,"Prerequis"));
    $Count=0;
    while ($N[$Count]){
        $N[$Count]="1" if (($N[$Count] eq "aucun")||($N[$Count] eq ""));
        $N[$Count]=~s/i//;
        $N[$Count]=$main::query{'selectopic'}.$N[$Count];
        $Count++;
    }
    #Charger le niveau des connaissances de l'apprenant
    # l'ensemble des unités déjà activées
    (@P) = split(/ /, &LireModule("ACTIVES"));
    # et les connaissances pour l'apprenant
    (@C) = split(/ /, &LireModule("CONNAISSANCES"));
    @B=&Union(@A, 0, @P);
    $stest1=&Inclusion("U".$Unit, 0, @B); # Ux inclus (A union P) (Inclusion sur l'ensemble)
    $stest2=&InclusDans(@N, 0, @C); # N inclus dans C
    return 1 if ($stest1 && $stest2);
    return 0;
}

```

FIG. 7.7 – Opérations de l'unité logique d'apprentissage

```

sub CalculerDynamique{
  local ($username, $UniteChoisie, @EnsembleUnite)=@_ if @_ ;
  return 0 if !$username;
  my (@TmpD, $tmp, $Naturetmp, $Dynamique, $CountTmp);
  $tmp=0;@TmpD=();
  $CountTmp=0;
  while ($EnsembleUnite[$CountTmp]){
    $Dynamique=0;
    $Dynamique=&LienDynamique($UniteChoisie, $EnsembleUnite[$CountTmp], $username);
    if ($Dynamique){
      # Saugarder les infos.
      $TmpD[$tmp]=$Dynamique."("$UniteChoisie.", "$EnsembleUnite[$CountTmp].)";
      $tmp++;
    }
    $CountTmp++;
  }
  return "" if !$TmpD[0];
  return @TmpD;
}

sub CalculerStatique{
  local($UniteChoisie, @EnsembleUnite)=@_ if @_ ;
  my (@TmpS, $tmp, $Count, $Naturetmp, $Statique);
  $tmp=0;@TmpS=();
  #Calculer les liens statiques
  $Count=0;
  while ($EnsembleUnite[$Count]){
    $Count++,next if ($UniteChoisie eq $EnsembleUnite[$Count]);
    $Statique=0;
    $Statique=&LienStatique($UniteChoisie, $EnsembleUnite[$Count]);
    if ($Statique){
      # Saugarder les infos.
      $TmpS[$tmp]=$Statique."("$UniteChoisie.", "$EnsembleUnite[$Count].)";
      $tmp++;
    }
    $Count++;
  }
  return @TmpS;
}

sub LienStatique{
  ← Opération définie dans la section 4.3.2.b
  local($Unit1, $Unit2)=@_ if @_ ;
  # $Unit1 = UniteCourante
  # $Unit2 = UniteQuelconque
  return 0 if!@_ ;
  return 0 if ($Unit1 eq $Unit2);
  return "$main::msg{graphe.acces}" if ((&LireUnite($Unit1, "AccesDirect") eq ("U".$Unit2)));
  return "$main::msg{graphe.meme}" if ((&LireUnite($Unit1, "ItemRef") eq &LireUnite($Unit2, "ItemRef"))
    && (&LireUnite($Unit1, "Antecedent") eq &LireUnite($Unit2, "Antecedent"))
    && (&LireUnite($Unit1, "Prerequis") eq &LireUnite($Unit2, "Prerequis"))
    && (&LireUnite($Unit1, "Apport") eq &LireUnite($Unit2, "Apport")));
  return "$main::msg{graphe.suivi}" if &AccessibleApres($Unit1, $Unit2);
  return 0;
}

sub LienDynamique{
  ← Opération définie dans la section 4.3.2.c
  local ($Unit1, $Unit2, $Apprenant)=@_ if @_ ;
  return 0 if!@_ ;
  return 0 if ($Unit1 eq $Unit2);
  return 0 if !$Apprenant;
  my $stest1=&AccessiblePour($Unit1, $Apprenant);
  my $stest2=&AccessiblePour($Unit2, $Apprenant);
  return "$main::msg{graphe.equip}" if ((&LireUnite($Unit1, "ItemRef") eq &LireUnite($Unit2, "ItemRef"))
    && (&LireUnite($Unit1, "Apport") eq &LireUnite($Unit2, "Apport"))
    && ($stest1 && $stest2));
  return "$main::msg{graphe.implique}" if $stest2;
  return 0;
}

```

FIG. 7.8 - Script de module ECSAIWeb::Domain, calculer les liens

7.6.3 Module PRESENTATION

Les catégories de présentation des contenus d'une unité physique d'apprentissage à l'apprenant dont nous avons besoin sont déjà décrites dans la section 4.4. Nous décrivons maintenant comment nous traitons chaque catégorie de présentation. Remarquons que la présentation d'une unité physique est faite après le choix d'une unité logique d'apprentissage parce que le nom du fichier d'une unité physique est écrit dans l'unité logique. A partir des paramètres stockés dans l'ULA, ils nous offrent aussi la nature, le titre, et le nombre de page. Après ouverture du fichier d'UPA, les contenus de chaque ligne de donnée sont traités (1) pour chercher le type de présentation. Si c'est le cas alors nous obtenons le type de présentation et ses attributs (nom du module). Par exemple, le contenu d'unité U1, représenté dans la figure 7.10 est de présentation du type «normal». Alors la procédure pour traiter cette catégorie est le même type d'«exemple», c.-à.-d., le module «INFO».

```

sub displayAll{
  if (&isElement($main::Type,qattr, 'type')) { ← 1
    $type = $qattr{'type'};
    $type =~tr/A-Z/a-z/;
    $module = $present_map{$type}||$type;
    %attributes = ();
    %handlers = ();
    &handle( 'HINTS', '', '&HintDoSteps' ) if (($main::NatureUnité=~/^E/)||$main::query{'hints'});
    if (($module eq 'normal')||($module eq 'info')) { ← 2
      &handle( 'INFOS', '', '&ExampleDoSteps' ); ← 3
    }
    elsif ($module eq 'rappel') { ← 2
      &handle( 'BUTS', 'help', '&Rappel1DoButs' ); ← 4
      &handle( 'RAPPELS', '-', '&Rappel1DoResponses' ); ← 4
    }
    elsif ($module eq 'multi') { ← 2
      &handle( 'CHOICES', 'help', '&ChoiceDoChoices' ); ← 5
      &handle( 'ANSWER', 'ans help', '&ChoiceDoAnswer' ); ← 5
      &handle( 'RESPONSES', '-', '&ChoiceDoResponses' ); ← 5
    }
    elsif ($module eq 'sn') { ← 2
      &handle( 'ANSWER', 'ans tol eps help', '&NumericDoAnswer' ); ← 6
      &handle( 'RESPONSES', '-', '&NumericDoResponses' ); ← 6
    }
    elsif ($module eq 'se') { ← 2
      &handle( 'ANSWER', 'ans tol help', '&ExpressionDoAnswer' ); ← 7
      &handle( 'RESPONSES', '-', '&ExpressionDoResponses' ); ← 7
    }
    elsif ($module eq 'st') { ← 2
      &handle( 'ANSWER', 'ans help', '&TextDoAnswer' ); ← 8
      &handle( 'RESPONSES', '-', '&TextDoResponses' ); ← 8
    }
    elsif ($module eq 'tf') { ← 2
      &handle( 'ANSWER', 'ans help', '&TrueFalseDoAnswer' ); ← 9
      &handle( 'RESPONSES', '-', '&TrueFalseDoResponses' ); ← 9
    }
  }
  else{
    &CgiError( "$main::Type type not implemented - $module" );
  }
  last if&doModule(); # Executer les commandes offert par le sous-programme handle
  print "<hr size=1>";
} else {
  print $ _ ;
}
}

```

FIG. 7.9 – Script de module ECSAIWeb::Presentation


```

sub isElement{
  local($tag, *attrs, $allow) = @_;
  my ( $str, $str1, $str2, $str3 );
  %attrs = ();
  $str = $_;
  $str1 = ($str !~ s/^\s*<\s*([a-zA-Z]\w*)\s*//);
  $str2 = ($1 !~ /^$tag$/i);
  $str3 = ($str !~ s/^(("[\s<>"]+("[\s"]*)?\s*)*)>//);
  return 0 if $str1;
  return 0 if $str2;
  return 0 if $str3;
  %attrs = &getAttrs($1, $allow);
  $_ = $str;
  return 1;
}

sub getAttrs {
  local( $str, $allow ) = @_;
  my ( %allow, %attr, $name, $value );
  foreach $name (split(/\s+/, $allow)) {
    $name =~ tr/A-Z/a-z/;
    $allow{$name} = 1;
  }
  while( $str =~ s/[A-Za-z]\w\s(=\s(("["]*)|\s+))?// ) {
    $name = $1;
    $value = $4 ? $5 : ($3 || 1);
    $name =~ tr/A-Z/a-z/;
    if( $allow ) {
      if(!$allow{$name} ) {
        next;
      }
    }
    $attr{$name} = $value;
  }
  return %attr;
}

sub handle {
  local( $tag, $attrs, $subr ) = @_;
  $attributes{$tag} = $attrs;
  $handlers{$tag} = $subr;
}

sub doModule {
  $url = $main::ScriptName;
  $rc = 0;
  @tags = sort keys %handlers;
  $answer = $main::query{'ans'};
  # les variables suivantes doivent être mises à jour par le package de présentation
  $correctAnswer = "";
  $answeredCorrectly = 0;
  $responsePrinted = 0;
  print;
  LOOP: while( <main::IN> ) {
    $line=$_;
    $line =~ s/http:\//www-interne.loria.fr/\~sanrach/$main::ServerName/g;
    $line =~ s/http:\//www.loria.fr/\~sanrach/$main::ServerName/g;
    $_ = $line;
    ($rc = 1), last if s/\s*<\s*/QUESTION\s*>||i;
    foreach $tag (@tags) {
      if( &isElement( $tag, *attrs, $attributes{$tag} ) ) {
        print if !/\s*$/;
        $cmd = "$handlers{$tag}(*attrs)";
        eval $cmd; # exécute le command par ex. si $handlers{'ANSWER'} = "&Choice_doAnswer"
        next LOOP;
      }
    }
  }
}

```

FIG. 7.9 – Le script de module ECSAIWeb::Presentation (suite)

```

    }
  }
  print;
}
if ($answer && !$responsePrinted) {
  print "<hr size=1>\n<p>$main::msg{ 'test.mvrepond' } <font color='#FF0000'><b>$answer
</b></font><p>\n" if!((($module eq 'multi')||($module eq 'tf')));
  if ($main::TestPath) {
    &defaultResponse2();
  }else{
    &defaultResponse();
  }
}
return $rc;
}

```

FIG. 7.9 – Le script de module ECSAIWeb::Presentation (suite)

```

<PRESENTATION TYPE=Normal>
<h2>Expérimente:</h2>
<center>
<applet
name="SeriesEx"
code="VJad84469fd30"
codebase="http://www-interne.loria.fr/~sanrach/classes"
archive="SeriesEx.zip"
width="352"
height="380"
>
<param name="Trigger" value="1">
</applet>
</center>
<br><br>
<h4>Travail demandé<hr></h4>
<p>Observer les points suivants:</p>
<ul>
<li>Les courants dans tous les composants</li>
<li>La résistance équivalente par rapport aux résistances dans le circuit.</li>
<li>Des chutes des tensions par rapport à la tension appliquée.</li>
<li>Le rapport entre les tentions par rapport au rapport entre les résistances</li>
</ul>
</PRESENTATION>

```

FIG. 7.10 – Contenu d'une unité physique U1 : Présentation normale

7.7 Bases de données

Les bases de données principalement sont séparées en trois bases :

- la première base, «*la base de données pédagogiques*» que nous appelons «*l'univers d'apprentissage*», contient les bases de données des items d'évaluation, des unités logiques d'apprentissage et des unités physiques d'apprentissage (inclut son, graphique et texte). Cette base est fournie par les enseignants-auteurs au cours de la création de l'univers. Les outils pour manipuler cette base fait l'objet d'une discussion dans le chapitre précédent (cf. section 6.4),
- la deuxième base, «*la base de modèle d'élève*» est créée pendant un parcours d'apprentissage pour stocker les informations évolutives de l'apprenant. Selon le mode d'apprentissage choisi, nous la séparons en deux parties : l'un pour mode libre et l'autre pour mode guidé,

```
# Lire module dynamique d'un utilisateur :: mode libre #
sub LireModuleLibre{
    local($Objectif, $Lequel, $dir, $nom)=@_ ;          ← 1

# Ecrire module dynamique d'un utilisateur :: mode libre #
sub WriteModuleLibre{
    local($Objectif, $Score, $Unites, $Listes, $Titre, $ScoreTotal)=@_ ; ← 2
```

a) Lire et Ecrire les données sur le module d'élève en mode libre

Modèle :

```
<LIBRE_MATIÈRE_SUJET>
ITEM_ID |#| ITEM_DEF |#| LIST_UNITES |#| LIST_NATURE_UNITE |#| NOTE_OBTENU |#| NOTE_TOTAL
</LIBRE_MATIÈRE_SUJET>
```

Exemple :

```
<LIBRE_220102_i1>
i1 |#| Résistance en série |#| 1|11| |#| P_0|E_2| |#| 4 |#| 4
i2 |#| Polarité des chutes de tension dans un circuit série |#| 2|12| |#| P_0|E_1| |#| 1 |#| 1
i3 |#| Notation à deux indices |#| 3| |#| P_0| |#| 0 |#| 0
i4 |#| Loi des tensions de Kirchhoff |#| 4|13| |#| P_0|E_2| |#| 0 |#| 2
i5 |#| Résistance interne |#| 5|14| |#| P_0|E_1| |#| 1 |#| 3
</LIBRE_220102_i1>
```

b) La base de données d'élève X en mode libre

FIG. 7.11 – Liste de sous-programme pour gérer la base de donnée d'un profil d'élève en mode libre

En *mode libre* (figure 7.11), le nom de fichier est la suite du nom utilisateur et du langage choisi avec l'extension de fichier «*.lbr*». Par exemple, si le nom d'utilisateur est de «*sanrach*» et il a choisit la langue «*française*» comme interfaces, alors le nom de cette base devient «*sanrachfr.lbr*». Normalement, à l'entrée de menu principal du mode libre, l'élève

doit choisir la matière et le sujet à apprendre et nous les utilisons comme les variables globales. Les variables locales utilisées sont l'objectif et les champs que nous voulons lire (1) ou écrire (2). Par exemple, si nous voulons savoir quelle est la note obtenue pour l'objectif i5 alors il nous suffit de mettre «&LireModuleLibre("i5", "Note")».

En *mode guidé* (figure 7.12), nous réorganisons les contenus de base parce que ce mode est plus complexe que celui du mode libre. Nous devons stocker toutes les informations nécessaires pour calculer le parcours d'apprentissage: les connaissances actuelles (3), les objectifs (4), les unités restantes (5), les unités déjà activées (6), la situation précédente et actuelle (7), les liens proposés (8), et le parcours d'apprentissage choisi (9). Le nom du fichier de ce modèle est créé la même façon de celui du mode libre sauf nous changeons l'extension de «.lbr» à «.dyn».

- En fin la troisième base, «la base d'utilisateurs» a pour stocker des informations personnelles de chaque utilisateur.

Pour manipuler ces bases nous décidons de créer nos bases en format texte parce que cela ne nous gênera pas, à l'avenir, quand nous voulons transférer nos bases de notre système à l'autre système opérationnel et qu'il nous assure qu'il pourra toujours être exécuté sans besoin d'outils supplémentaires.

7.8 Générateur des menus

Pour générer les menus, nous avons utilisé plusieurs méthodes comme nous avons décrit dans le chapitre de description (chapitre 6): la barre de menus, le menu pop-up et le menu adaptatif. En appliquant le script du Java, présenté dans la figure 7.13, nous pouvons remplacer une image quand nous passons la souris au-dessus par l'autre image au niveau d'interface et nous employons le lien hypertexte pour lier à l'autre page ou pour envoyer le requête des commandes aux programmes situés dans le serveur. Nous avons exploité cette technique dans plusieurs parties, par exemple dans l'environnement partagé, messageries électroniques, et dans le module central, menu principal avant et après entrer dans le système, dont nous avons du mettre ce script en dehors du programme principal pour que nous puissions le réutiliser. Nous le réorganisons en mettant la partie paramètres dans le programme principal (figure 7.13.a) et puis tous les fonctions dans le fichier du script (figure 7.13.b). Nous employons la même technique pour créer le menu pop-up.

Le menu adaptatif est un menu pour aider l'apprenant à naviguer dans l'univers d'apprentissage et lui montrer où il en est. En utilisant les compétences de l'apprenant, nous pouvons exprimer les états pédagogiques sur les objectifs pédagogiques en six états différents (par défaut): très bien (objectif atteint, note obtenue >80%), bien, moyen, faible, très faible et l'objectif sans exercice. Chaque état est lié aux informations dans les couches différentes par exemple pour atteindre un objectif pédagogique, l'apprenant doit visiter les pages d'informations et s'il y a des exercices sur cet objectif, l'apprenant doit passer tous les exercices et avoir obtenu la note au moins 80%. La figure 7.14 présente la relation entre le menu adaptatif en mode libre et le contenu dans le profil de l'apprenant.

7.9 Générateur d'un parcours d'apprentissage

Le générateur du parcours d'apprentissage, selon le mode d'apprentissage, est séparé en deux générateurs: l'un pour le mode libre et l'autre pour le mode guidé. En mode libre, après le

```
# Lire module dynamique d'un utilisateur :: mode guide #
sub LireModuleGuide{
local($Lequel)=@_ if @_;

# Ecrire module dynamique d'un utilisateur :: mode guide #
sub WriteModuleGuide{
local ($Lequel, @Msg)=@_ if @_;
```

a) Lire et Ecrire les données sur le module d'élève en mode guidé

```
<CONNAISSANCES_220102_i1> ← 3
i11(a;1,2) i12(a;0,1) i13(a;0,0)
</CONNAISSANCES_220102_i1>
<OBJECTIF_220102_i1> ← 4
i11(a;2,2) i12(a;0,1) i13(a;0,0) i14(a;1,1) i15(a;2,1) i16(a;2,1) i17(a;2,1) i18(a;1,0) i19(a;1,0) i110(a;1,0)
</OBJECTIF_220102_i1>
<OBJECTIFS_220102_i1> ← 4
i11 i14 i15 i16 i17 i18 i19 i110
</OBJECTIFS_220102_i1>
<UNITÉS_220102_i1> ← 5
U4 U5 U6 U7 U8 U9 U10 U13 U14 U15 U16 U17 U18 U19
</UNITES_220102_i1>
<ACTIVES_220102_i1> ← 6
U1 U11 U2 U12 U3
</ACTIVES_220102_i1>
<STATUS_220102_i1> ← 7
U12
</STATUS_220102_i1>
<AVANT_220102_i1> ← 7
U4
</AVANT_220102_i1>
<LIENS_220102_i1> ← 8
implique(1,7) suivi-de(1,2) suivi-de(1,11)
implique(11,2) implique(11,7)
implique(2,7) suivi-de(2,3) suivi-de(2,12)
implique(12,3) implique(12,7)
implique(3,7) suivi-de(3,4)
</LIENS_220102_i1>
<LIEN_220102_i1> ← 9
suivi-de(1,11) implique(11,2) suivi-de(2,12) implique(12,3) implique(3,7)
</LIEN_220102_i1>
```

b) La base de données d'élève X en mode guidé

FIG. 7.12 – Liste de sous-programmes pour gérer la base de donnée d'un profil d'élève en mode guidé

```

print <<"      SCRIPT      ";
<SCRIPT LANGUAGE="JavaScript1.2">
<!--
var Images = new Array(10);
Images[0] = "login";
Images[1] = "inscrire";
Images[2] = "findkey";
Images[3] = "manual";
Images[4] = "close";
Images[5] = "copyright";
var Path = "$PathServerImg/";
//-->
</SCRIPT>
---- SCRIPT ----
:
:
:
# Se connecter
print "<TD class='bg-gray' align='center' ><a href=\"\$ScriptNameForm?mainmenu=yes&login=yes\"
onMouseOver=\"On(0);window.status='msg{'status.login'}';return true;\" onMouseOut=\"Off(0);
window.status='';return true;\" ><IMG SRC=\"\$PathServerImg/login_off.gif\" BORDER=0 ALT=\"
$msg{'menu.login'}\" name='login' width='40' height='40'></A></TD>\n";

# S'inscrire
print "<TD class='bg-gray' align='center' ><a href=\"\$ScriptNameForm?mainmenu=yes&askpass=yes\"
onMouseOver=\"On(1);window.status='msg{'status.registrer'}';return true;\" onMouseOut=\"Off(1);
window.status='';return true;\" ><IMG SRC=\"\$PathServerImg/inscrire_off.gif\"BORDER=0 ALT=\"
$msg{'menu.registrer'}\" name='inscrire' width='40' height='40'></A></TD>\n";
:
:
:

```

a) La description des paramètres pour le script du Java de la barre de menus

```

<SCRIPT LANGUAGE="JavaScript1.2">
<!--
function On(nb)
{
  var Img = new Image();
  Img.src= Path + Images[nb] + "_on.gif";
  document.images[Images[nb]].src = Img.src;
}
function Off(nb)
{
  var Img = new Image();
  Img.src= Path + Images[nb] + "_off.gif";
  document.images[Images[nb]].src = Img.src;
}
//-->
</SCRIPT>

```

b) Le contenu du script du Java de la barre de menus, *menu-image.js*

FIG. 7.13 – Script du Java pour manipuler la barre de menus

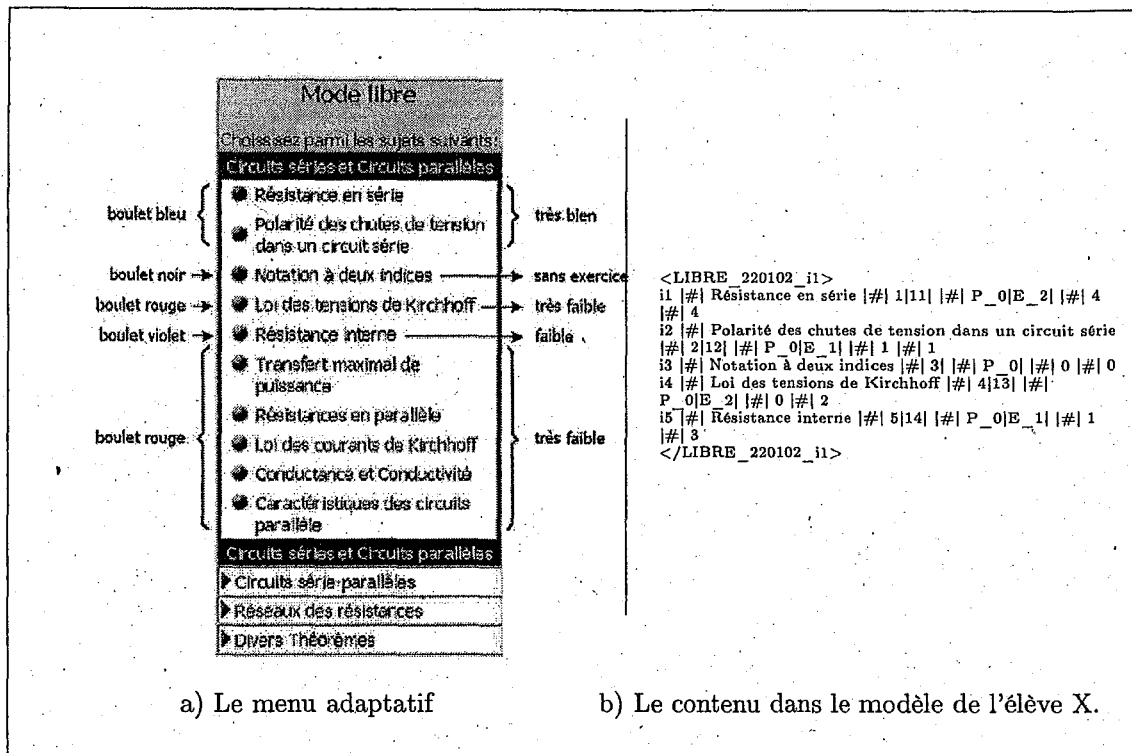


FIG. 7.14 – Relation entre le menu adaptatif et le contenu dans le profil de l'apprenant

```
sub ChoixUnite{
  local($Objectif, $Lien, $Display)=@_;
  my $UnitChoisi="";
  my $Dir=&ReadUserDir($main::query{'username'});
  my $PathName = "$main::auteurpath/userdata/$Dir/$main::RealName.pro";
  open(PROFIL, "$PathName")||&CgiError("ChoixUnite : $PathName : $!");
  my (@Data) = <PROFIL>;
  close(PROFIL);
  my $Mode = $Data[0];
  chop ($Mode) if ($Mode = /\n$/);
  # Si l'apprenant choisit à apprendre toutes les informations avant de faire les exercices
  if ($Mode =~~/1/) {
    $UnitChoisi=&PresentationAbord($Objectif, $Lien, $Display);
    $UnitChoisi=&ItemAbord($Objectif, $Lien, $Display) if !$UnitChoisi;
  }else{
    $UnitChoisi=&ItemAbord($Objectif, $Lien, $Display);
  }
  $UnitChoisi=&ItemAttient($Objectif, $Lien, $Display) if !$UnitChoisi;
  $UnitChoisi=&ApportGlobal($Objectif, $Lien, $Display) if !$UnitChoisi;
  $UnitChoisi=&NiveauCritique($Objectif, $Lien, $Display) if !$UnitChoisi;
  return $UnitChoisi;
}
```

FIG. 7.15 – Sous-programme de raffinement du parcours d'apprentissage

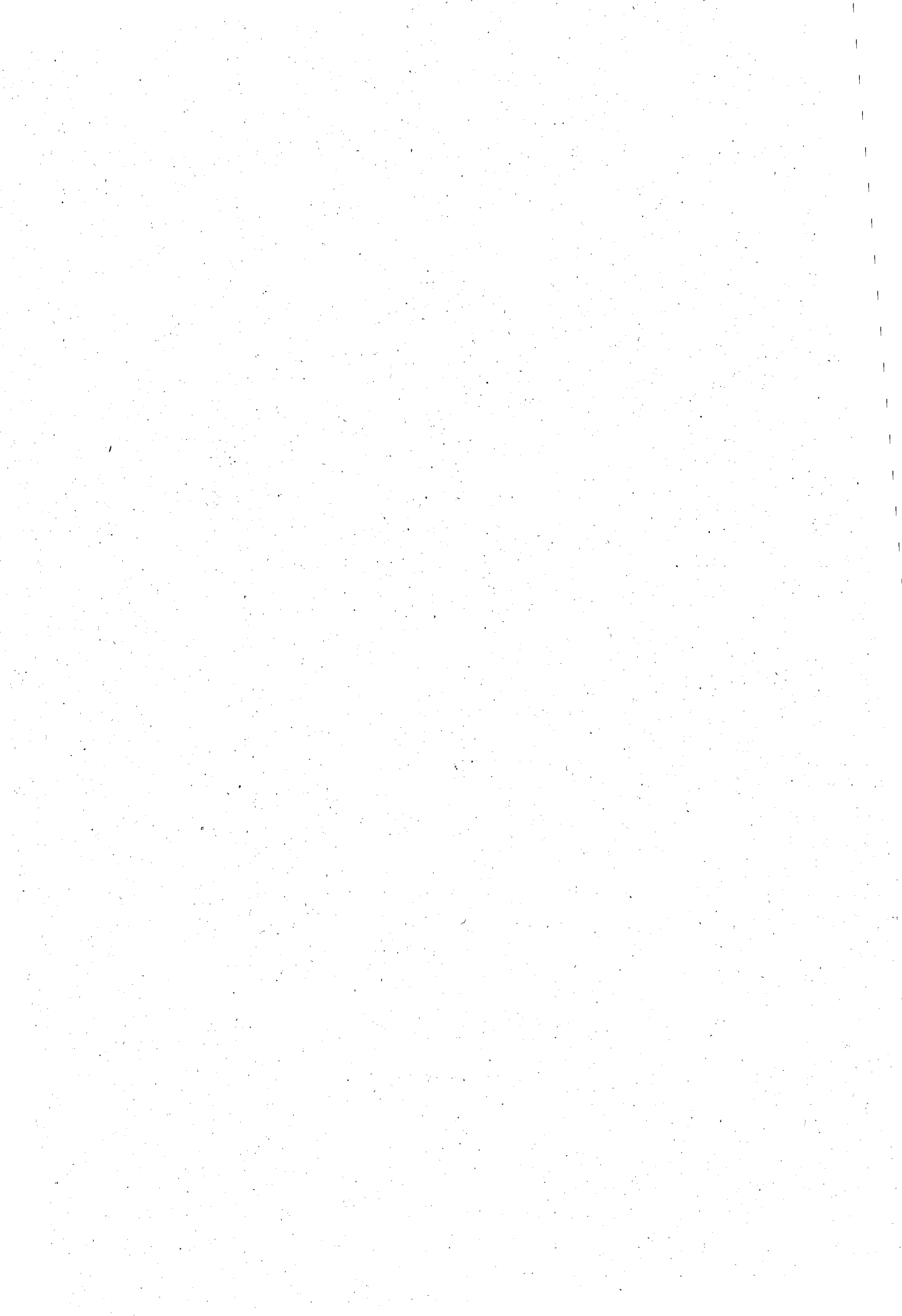
choix de l'objectif à apprendre, le système génère le plan de progression pour l'apprenant en offrant le parcours d'apprentissage au choix : la présentation seule, l'exercice seul et les deux. L'enchaînement des unités associées à l'objectif choisi est calculé automatiquement selon les informations stockées dans les unités logiques d'apprentissage (cf. section 6.5.2).

En mode guidé, après le choix d'un sujet à apprendre, le système prend la main et calcule le parcours d'apprentissage pour chaque apprenant. Pour compléter le réseau de l'univers d'apprentissage pour chaque sujet, le système utilise les informations stockées dans les unités ULA et dans le profil de l'apprenant pour calculer, à la fin de l'exécution d'unité courante, les liens entre l'unité courante et les unités restantes. En appliquant les méthodes raffinées, le système obtient l'unité suivante. A côté de l'apprenant, il doit suivre ce propos jusqu'à tous les objectifs associés au sujet choisi soient tous atteints. La figure 7.15 présente une partie de sous programme pour raffiner les liens.

7.10 Conclusion

Comme nous l'avons indiqué au début de ce chapitre, notre système est divisé en plusieurs parties. Cela nous pose des problèmes au niveau de l'organisation de tous les programmes, c.-à.-d., pendant la définition des fonctionnalités de chaque environnement et des liens entre les variables dans plusieurs environnements et modules. Avec cette organisation, elle nous permet à l'avenir de maintenir et d'ajouter facilement les nouvelles fonctionnalités dans chaque environnement. Imaginons que nous voulions ajouter, par exemple, une nouvelle catégorie d'exercice, il suffit d'ajouter le morceau de programme dans le module Présentation et Domaine sans toucher les autres programmes ni recompiler tous les programmes.

Troisième partie
Bilan et perspectives



Chapitre 8

Conclusion

L'objectif de notre travail était l'adaptation d'un environnement ouvert ECSAI pour produire un environnement de création des cours sur le Web. En conservant les idées initiales du système ECSAI, nous avons étudié le processus d'adaptation de l'environnement d'une part et des matériaux pédagogiques d'autre part, pour tirer partie des possibilités introduites par l'utilisation du Web. Nous avons souhaité que le nouveau système soit plus flexible et plus adaptatif.

Etant donné les nouvelles fonctionnalités des technologies gérées en réseau, il y a maintenant le potentiel pour que les étudiants partagent leurs expériences d'apprentissage les uns avec les autres et également interagissent plus dynamiquement avec leurs enseignants. La collaboration soutenue par ordinateur asynchrone a été un sujet fréquent de recherches, puis le travail synchrone en ligne a attiré l'attention dans la domaine CSCL⁴⁷. Puisque les interactions synchrones sont une partie essentielle de formation scolaire, nous avons choisi ici de nous concentrer sur les outils synchrones, utilisant l'interaction et la rétroaction actives immédiates entre les utilisateurs.

Enfin, nous espérons avoir produit des recommandations utilisables pour tous les auteurs qui ont à adapter des cours ou des environnements existants.

Bilan du travail

Réalisation de l'ECSAIWEB

Nous avons repris les principes d'ECSAI pour construire ECSAIWEB. Nous avons conservé les idées de l'ensemble d'unités indépendantes, des modèles du domaine et du générateur dynamique des liens entre les unités logiques d'apprentissage. Nous avons modifié quelques parties dans les modèles du domaine et les unités logiques d'apprentissage pour qu'ils supportent l'hypermédia et soient utilisables sur le Web. Par exemple dans ECSAIWEB, nous avons mis la valeur « mesure globale », la situation d'apprentissage dans le temps, à la valeur constante « a » pour conserver les algorithmes de calcul. Puis nous avons utilisé les techniques adaptatives de présentation et de navigation pour exprimer la situation d'apprentissage de chaque apprenant. Nous avons séparé les unités d'apprentissage en deux parties : les unités logiques d'apprentissage utilisées pour calculer les liens et le plan d'apprentissage et les unités physiques d'apprentissage utilisées pour présenter le contenu que nous voulons enseigner. De cette façon, les unités physiques peuvent être réutilisables par les unités logiques dans la même matière ou dans une autre matière.

47. sigle anglais de *Computer Supported Collaborative Learning*

Nous avons amélioré la façon de présenter le contenu et les exercices : plus flexible, plus interactif et plus adaptatif. Les unités physiques sont les nouveaux modèles que nous avons intégrés dans ECSAIWEB. Elles sont représentées dans des fichiers avec du code HTML en général et avec des balises spéciales qui sont utilisées pour identifier la nature de l'unité physique et qui permettent de présenter dynamiquement les contenus aux élèves. Ces modèles permettent également aux enseignants d'avoir plusieurs modalités de présentation des contenus aux apprenants : la présentation normale, la présentation d'exemple étape par étape, la présentation résumée et plusieurs types des exercices : la question à choix multiple, numérique simple, l'expression simple, texte et vrai ou faux. Nous avons aussi ajouté dans chaque question les explications (aides) et les rétroactions que les enseignants peuvent définir eux-mêmes ou bien utiliser les valeurs par défaut de notre système. Nous permettons aussi d'utiliser les médias de format particulier à condition que le plug-in associé soit installé.

L'idée de création d'un environnement ouvert de conception du système d'apprentissage est présentée depuis le début de ce travail. Nous avons voulu créer un environnement ouvert qui permet aux enseignants-auteurs de créer eux-mêmes leurs cours et d'être capable de modifier les cours conformément aux besoins de leurs élèves, parce que plusieurs produits existants n'offrent aucun outil de le faire. Les enseignants doivent obligatoirement les utiliser sans avoir les contraintes sur leurs propres connaissances et leurs élèves. Avec les systèmes autonomes, il est difficile de mettre à jour le contenu parce qu'il faut attendre de remplacer l'ancienne version par la nouvelle version, s'il le fait. Il est difficile aussi de la part des enseignants de suivre ou observer les activités de leurs élèves. Les enseignants doivent attendre jusqu'à la fin des travaux pour récupérer les tâches de chaque élève. Alors avec ces problèmes, nous avons décidé d'adapter ECSAI au Web. Selon les flexibilités et les études approfondies des modèles et l'environnement lui-même, ils nous permettent facilement d'adapter sur le Web.

La venue de HTML dynamique, les feuilles de styles et après expérimentation de notre système⁴⁸ nous ont mené à améliorer toutes les interfaces : plus interactives et amicaux. Les interfaces sont plus complexes au niveau de l'implantation mais elles sont faciles à utiliser⁴⁹.

Nous avons appliqué des techniques adaptatives dans l'environnement élève. Ces techniques font relative à l'utilisation des navigateurs et la présentation des contenus. L'annotation adaptative signifie que le système utilise des choix visuels comme les icônes, la police, et les couleurs, qui sont dynamiquement calculées en employant le modèle individuel de l'apprenant pour montrer le type et le rôle de l'enseignement de chaque lien (cf. section 3.2.2). La technique supplémentaire que nous avons introduit en mode guidé est le générateur de parcours d'apprentissage. Après calculer les liens entre l'unité courante et les unités restantes, nous avons modifié et ajouté les fonctions heuristiques (cf. section 6.5.2) qui aident à raffiner le parcours d'apprentissage. Donc nous obtenons toujours la meilleure unité pour la situation actuelle de l'apprenant. Ces fonctions ont pour but d'assurer que l'apprenant avec la bonne connaissance ait toujours le parcours optimal.

Nous avons décidé d'ajouter un environnement partagé. C'est un support de communication asynchrone et synchrone destiné à favoriser l'interaction, la coopération et le travail de groupe entre les enseignants, les étudiants et les enseignants et les étudiants. Chaque groupe dispose de trois outils de communication : un forum de discussion, une sorte de «*newsgroup*» et un courrier électronique destiné à faciliter le travail coopératif, et des salles de discussion (chatroom) destinées à discuter en temps réel entre utilisateurs.

48. La première maquette de notre système a été testée par vingtaines étudiants de maître, option informatique et formation, niveau master, KMITNB en Thaïlande, janvier 2000.

49. La deuxième maquette est disponible depuis octobre 2000.

Premières expérimentations

Les premiers réactions d'utilisations ont eu lieu dans le département Informatique et Formation, Faculté d'Education Technique, KMITNB avec une vingtaine étudiants de maître en «*master*». Deux tiers de ces étudiants ont déjà eu des expériences personnelles d'enseignement au moins un an dans lycées/collèges public ou privés. Les autres viennent de terminer leurs études en «*Bachelor of Science in Technical Education*», mais ont aussi eu des expériences d'enseignement pendant leurs études. Cette expérimentation porte sur les interfaces utilisées et les fonctionnalités proposées par ECSAIWEB. Ces étudiants ont joué le rôle d'une part de l'enseignant-auteur et d'autre part de l'apprenant. Les fonctionnalités que nous avons proposées sont satisfaisantes parce qu'il y a eu des outils nécessaires pour aider à créer les cours sur le Web. Mais les interfaces sont trop compliquées à utiliser. De plus le système lui-même est complexe. Après cette expérimentation et après avoir présenté et discuté notre travail dans plusieurs conférences, nous avons décidé de regrouper les outils en plusieurs catégories : outils pour créer l'univers d'apprentissage, outils pour visualiser le parcours d'apprentissage et etc., (voire figure 6.12 et 6.13). En utilisant Javascript, nous avons pu créer le menu POP-UP et grouper les sous-menus en un seul menu au lieu d'afficher tous les menus d'un seul coup (voire figure 6.9). Nous avons modifié également les interfaces pour simplifier la définition de l'univers d'apprentissage. Par exemple au lieu de saisir directement la valeur $i1(a; 2; 0)$ aux formulaires, nous avons proposé une nouvelle fenêtre incluant les informations pour aider à remplir les paramètres. En utilisant la technique du menu POP-UP, nous avons pu choisir un élément dans la liste par exemple, la liste des items ← $i1$: Résistance en série, la liste des mesures globales ← Abordé, la liste de savoir-appliquer ← 2 et la liste de savoir faire ← 0 (voire figure 6.15, définir l'apport de connaissances). Après cette amélioration, la deuxième maquette est disponible depuis octobre 2000.

Tendances

Un certain nombre de facteurs peuvent être considérés comme jouant un rôle positif dans le déploiement de WEDS :

- la prise de conscience du besoin en formation continue et du potentiel des WEDS, en particulier pour la prise en charge de la logistique d'un dispositif de formation en entreprise,
- l'augmentation de l'offre en formation, en particulier dans les domaines techniques,
- l'augmentation du nombre d'accès à l'Internet au sein des organisations, même les plus petites (parfois grâce au rôle stimulateur des pouvoirs publics) ainsi que la mise en place de nombreux intranets et extranets : ceci implique que les postes «*clients*» existent de plus en plus fréquemment et que la culture du Web s'impose graduellement
- la pression exercée par la communauté des utilisateurs de l'Internet pour augmenter la bande passante des communications.

Obstacles

Malgré le potentiel de développement important représenté par notre système, il faut souligner que l'adoption généralisée se heurte à un certain nombre d'obstacles :

- les limitations actuelles de la bande passante et, plus encore, les irrégularités de débit empêchent l'utilisation du fichier de grande taille par exemple la vidéo, le son, les images et des autres formats multimédias,
- le coût initial important du développement des cours (parfois on prend plusieurs heures de travail pour une heure de «*cours*»), il est donc difficile de trouver les volontaires pour

expérimenter notre système),

- la résistance institutionnelle aux approches «*technologiques*» pour la formation suscite la crainte d'une dépersonnalisation accrue dans les rapports humains,
- les formateurs en place craignent que le passage aux WEDS supprime leurs emplois (ce que le point précédent contredit bien évidemment),
- la qualité moyenne, voire médiocre, d'un certain nombre de produits actuels risque de rebuter les nouveaux utilisateurs

Perspectives

Evolution de l'environnement

Après expérimentation de notre système, la comparaison avec les systèmes WEDS existants et la discussion avec les enseignants, les enseignants-auteurs et les experts du domaine WEDS, nous ont permis d'améliorer plusieurs fonctionnalités dans ECSAIWEB. Le premier point, dans l'environnement auteur, actuellement toutes les fonctions de cet environnement permettent aux enseignants-auteurs de créer et modifier librement les matériaux pédagogiques. La demande sur cet environnement est que les enseignants-auteurs doivent avoir la possibilité de travailler en dehors du système pour préparer les nouveaux cours ou améliorer les anciens cours. Cela devient possible parce que des modèles du domaine de notre système sont indépendants les uns les autres. Les enseignants-auteurs peuvent créer directement les nouveaux contenus dans chaque modèle en dehors du système, puis mettre à jour ce qui sont déjà faits séparément. Pour améliorer les matériaux existants, les enseignants-auteurs doivent télécharger tous les contenus, puis les modifier et enfin les mettre à jour. Cette possibilité peut être mise en œuvre par l'utilisation du Java.

Au moment que nous avons commencé notre travail, il n'existait pas encore des outils pour créer ou manipuler les DTD comme XML⁵⁰ [W3Cd]. Nous avons donc décidé de modéliser nous-même par exemple les unités physiques d'apprentissage en ajoutant les balises spéciales pour identifier la nature de l'unité physique et ses catégories. Mais en 1998, un groupe de travail XML (initialement connu sous le nom de comité d'examen éditorial SGML⁵¹) constitué sous les auspices du Consortium du World Wide Web (W3C) a publié la spécification de base du XML version 1.0. XML est un langage de balisage extensible qui décrit une classe d'objets de données appelés documents XML et décrit partiellement le comportement des programmes qui les traitent. XML est un profil d'application ou une forme restreinte de SGML, le langage normalisé de balisage généralisé (ISO 8897). Par construction, les documents XML sont des documents conformes à SGML. En fait, notre travail sur les unités physiques est proche de cette norme. Nous envisagerons donc de transcrire nos unités physiques vers XML dans la prochaine version.

Le deuxième point, dans l'environnement élève, la demande sur cet environnement est d'ajouter le choix du parcours d'apprentissage en mode guidé parce qu'en ce moment le parcours d'apprentissage est généré et sélectionné automatiquement par le système en utilisant les compétences de chaque apprenant et les critères pour raffiner les choix. Ce qu'il faut changer dans ce mode est de permettre à l'apprenant de choisir lui-même le chemin d'apprentissage. Il est possible d'implanter ceci appliquant la technique adaptative d'annotation pour annoter les concepts ou les unités logiques qu'ils doivent être suivis. Nous pouvons nommer cette méthode, selon la nature de choix, «*semi-guidé*». Cette méthode prépare les unités suivantes en annotant l'unité

50. sigle anglais de *Extensible Markup Language*

51. sigle anglais de *Standard Generalized Markup Language*

«*doit-être-suivi*» (lien «accès-direct-à» et le concept déjà atteint) , «*suggère-à-faire*» (le même concept) et «*peut-être-suivi*» (les liens restants).

ECSAIWEB comme support de formation à distance

Il est possible que nous puissions adapter ECSAIWEB comme un support de la formation à distance. Selon la nature de l'environnement de type client-serveur, nous pouvons l'utiliser directement sans aucunes modifications de l'environnement. Mais ce qu'il faut faire est d'ajouter une certaine fonctionnalité dans le système , représenté dans la liste suivante:

- la table des matières (syllabus),
- la bibliographie,
- les répertoires des élèves avec la liste des adresses électroniques,
- la liste des salles de discussion,
- le calendrier des engagements de cours, des exercices, des notes et autres communications,
- le module de mis à jours des exercices sur site Web,
- etc.

Intégrer les compagnons dans ECSAIWEB

Une organisation moderne réussie se caractérise par les employés qui prennent la propriété et la fierté dans les choses qu'ils font. Elle a un climat où les employés à tous les niveaux sont aidés pour rencontrer la mission de l'organisation. Elle stimule et récompense la formation continue, encourage la formation et les liens des collègues, et distribue la connaissance de l'organisation vers la main d'œuvre entière. Mais une organisation moderne réussie ne se produit pas simplement spontanément. Elle se développe en raison du changement d'organisation où un climat créé encourage la croissance des communautés de bases des employés et soutient activement des efforts de collaboration d'apprentissage.

Un composant commun d'apprentissage de collaboration est un «*réseau informel de Peer-Help*». C'est un réseau social des apprenants et des connaissances qui sont consultés pour l'aide et le conseil aux sujets d'apprentissage. Les réseaux informels de «*Peer-Help*» existent dans chaque organisation. Ils constituent un composant crucial de la formation et de l'acclimatation pour le novice apprenant et représentent un élément essentiel de mémoire d'organisation partagée. Le but de notre projet, à l'avenir, est d'employer la technologie pour soutenir l'apprentissage collaborative et l'intégration de réseau informel de «*Peer-Help*» dans notre système.

Afin d'employer efficacement la technologie moderne de l'information un apprenant doit posséder moins de qualifications de procédurales et plus de résolution des problèmes et de jugement. Le travail performant soutenu par un bon système d'aide en ligne ou des listes de contrôle de tâche peut aider les apprenants pour surmonter beaucoup d'impasses simples. Néanmoins, des apprenants plus expériences peuvent apporter de l'aide sur des demandes moins fréquentes mais plus complexes, alors que les apprenants novices tendent à apporter de l'aide sur des demandes plus fréquentes et normalement moins complexes.

L'approche que nous voulons adopter est de développer un système de support orienté vers la tâche qui est le système d'aide par les compagnons. Il y a deux aspects à cette approche. Le premier comporte l'utilisation d'un système adapté à la tâche pour classer les demandes d'aide et la connaissance de l'utilisateur. La second implique le soutien basé sur la connaissance pour localiser un compagnon quelque part dans l'environnement d'apprentissage collaboratif qui est prêt, volontaire, et capable de fournir l'aide sur la demande. La combinaison de ces aspects

produit un environnement efficace pour livrer l'aide *«just-in-time»*. Elle crée également un environnement fertile pour étudier l'apprentissage collaboratif et le changement d'organisation. En conclusion, la nécessité pour maintenir des profils de connaissance des compagnons potentiels a produit beaucoup de défis intéressants de recherche dans le domaine de modèle d'utilisateur et d'étudiant.

Glossaire

CBL : Computer Based Learning

CDROM : Compact Disc Read Only Memory

CGI : Common Gateway Interface

Courseware : Logiciel spécialisé dans l'enseignement d'une discipline, d'une méthode ou de certaines connaissances. Le terme normalisé par l'Office de la langue française est *didacticiel* ou parfois on appelle des matériaux pédagogiques.

CSCL : Computer Supported Collaborative Learning

CSS : (Cascading Style Sheets) – Fichier texte qui contient une liste de marqueurs HTML ainsi que le formatage associé à chacun.

DHTML : Dynamique HyperText Markup Language

DOM : (Document Object Model) – Modèle d'interface permettant de diviser les sections d'un document HTML en éléments ou en objets pouvant être déplacés individuellement à l'intérieur d'une fenêtre de navigateur, et ce, indépendamment de la plate-forme et du langage utilisés. Le modèle DOM constitue le coeur du HTML dynamique. Grâce au modèle DOM, on peut utiliser des scripts pour avoir accès au contenu, à la structure et aux polices de caractères d'un document HTML, et ainsi le modifier de façon dynamique.

EAO : Enseignement Assisté par Ordinateur, cf. sigle anglais, *CAI – Computer Assisted Instruction*

ECSAI : Environnement de Conception de Systèmes d'Apprentissage Intelligent

ECSAIWEB : Environnement de Conception de Systèmes d'Apprentissage Intelligent sur le Web

EIAO : Environnement Interactif d'Apprentissage avec Ordinateur

ftp : file transfer protocol

Gopher : Système basé sur une structure de menus, qui permet la recherche d'information dans Internet, l'accès à cette information et sa visualisation.

HTML : (HyperText Markup Language) – Langage utilisé pour définir la structure et la présentation à l'écran d'une page Web, à l'aide de balises interprétées par le navigateur Web. Le langage HTML permet ainsi de spécifier la mise en forme du texte, l'insertion d'image et de séquences sonores ou vidéo, de liens hypertexte qui pointent vers d'autres pages, et de scripts JavaScript ou d'applets Java qui permettront de rendre la page plus interactive.

HTTPD : (HyperText Transfer Protocol Daemon) – Processus démon qui attend sur le port 80 (norme) de sa machine hôte (host en anglais) des requêtes HTTP. Ce processus interprète les requêtes et émet une réponse. C'est un serveur Web, il reçoit des requêtes provenant de clients navigateurs qui demandent des URL, et le démon répond en envoyant les pages HTML, les images demandées. Termes équivalents – serveur Web, serveur HTTP.

HyperCard : HyperCard est souvent décrit comme «paquet de cartes». Les cartes peuvent être liées les unes avec les autres. HyperCard est parfois décrit comme une sorte de base de données pour organiser des informations et les enchaîner ensemble.

Hypertexte : Système de présentation de l'information sous forme de documents textuels dont certains mots, mis en évidence par une couleur par exemple, renvoient vers une autre partie du même document ou vers d'autres documents. En cliquant sur ces liens hypertexte à l'aide de la souris, l'utilisateur peut ainsi «naviguer» d'un document à l'autre de manière très intuitive.

KMITNB : King Mongkut's Institute of Technology North Bangkok

PC : Personal Computer

Perl : (Practical Extraction and Report Language) Perl est une langue interprétée optimisée pour les dossiers arbitraires des textes, extrayant l'information à partir de ces dossiers des textes, et imprimant des rapports basés sur cette information.

POP3 : Protocole de messagerie électronique, permettant à un abonné qui fait appel à un fournisseur d'accès à Internet, de récupérer ses courriels en attente sur un serveur de courrier, sans que le micro-ordinateur utilisé soit continuellement sous tension.

RNIS : Réseaux Numérique à Intégration de Service, cf. sigle anglais de *ISDN - Integrated Service Digital Network*

SEW : Systèmes d'Education basés sur le Web, cf. sigle anglais de *WES - Web-based Education System*

SEW-AI : Systèmes d'Education Adaptatifs et Intelligents basés sur le Web, cf. sigle anglais, *AIWED - Adaptive and Intelligent Web-based Education Systems*

SGML : Un langage normalisé permettant de décrire les relations entre le contenu d'un document informatique et sa structure. Ce langage SGML spécifie un système de marquage du contenu d'un document en définissant les titres, auteurs, chapitres, sous-chapitres d'un document, et en identifiant les graphiques et autres types de données. Cette description standardisée permet donc de lire les documents créés selon cette norme sur n'importe quel type de machine, ce qui facilite leur réutilisation et leur circulation sous forme électronique. Le SGML assure en particulier que tout document pourra être relu, quelle que soit l'évolution des systèmes et des traitements de texte. Le langage SGML est aujourd'hui surtout utilisé dans les agences gouvernementales, les industries aéronautique et pharmaceutique ainsi que dans l'édition.

STI : Système Tutoriel Intelligent, sigle anglais, *ITS - Intelligent Tutoring System*

TCP/IP : Protocol de communication pour réseaux hétérogènes (locaux ou étendus) provenant de la fusion de deux protocoles. Le premier, TCP qui gère le découpage des données en paquets avant leur routage, et leur réassemblage sur le poste récepteur. Le deuxième, IP qui gère le routage des paquets au travers des nœuds du réseau. Il existe des implémentations de TCP/IP pour pratiquement tout type d'ordinateur et de système d'exploitation. Chaque machine ou nœud du réseau est identifié par son adresse IP, sous forme d'un nombre de 32 bits. Internet est le plus grand réseau TCP/IP du monde.

Telnet : Protocole d'émulation de terminal qui permet à l'internaute d'entrer en communication avec un hôte. Internet à partir de son propre ordinateur pour utiliser des programmes ou consulter des données qui y sont stockées.

UM : User Modeling

UPA : Unité Physique d'Apprentissage

URL : (Unified Resource Locator) – Protocol et mécanique associé, qui permet de nommer et d'identifier l'ensemble de ressources (site, documents, images, vidéo, programmes ...) disponibles sur le Web. Par extension, l'acronyme URL est souvent synonyme de nom d'une ressource. URL contient également une méthode d'accès au document recherché, le nom du serveur et le chemin d'accès au document.

VRML : Virtual Reality Markup Language

WAIS : Système basé sur le modèle client-serveur qui permet la recherche d'information à partir de mots clés dans le contenu de documents préalablement indexés dans ce but par ce même système, l'accès aux documents ainsi trouvés et leur visualisation.

Webcam : caméra vidéo numérique reliée directement à l'ordinateur.

WEDS : Web based Education Delivery System

WWW ou Web : (World Wide Web) – Système basé sur l'utilisation de l'hypertexte, qui permet la recherche d'information dans Internet, l'accès à cette information et sa visualisation.

XML : Une évolution du langage SGML permettant aux concepteurs de documents HTML de définir leurs propres marqueurs, dans le but de personnaliser la structure des données qu'ils comptent présenter. Il faut noter que c'est en utilisant un navigateur compatible que l'utilisateur peut exploiter les marqueurs personnalisés du langage XML. De ce fait, ce langage est mieux adapté à la gestion de documents longs et complexes, comme on en trouve dans les intranets, puisque l'utilisateur peut sélectionner le type d'information qu'il souhaite consulter.

Bibliographie

- [ASF99] S. R. Alpert, M. K. Singley, and P. G. Fairweather. Deploying intelligent tutors on the web: architecture and example. *International Journal of Artificial Intelligence in Education*, 10:183–197, 1999.
-
- [BC99a] Marie-José Barbot and Givovanni Camatarri. *Autonomie et Apprentissage: L'innovation dans la formation*. Education et Formation: «pédagogie théorique et critique». Presses Universitaires de France, 1999.
-
- [BC99b] P. Brusilovsky and D.W. Cooper. ADAPTS: Adaptive hypermedia for a web-based performance support system. In *Proceedings of the 2nd Workshop on Adaptive Systems and User Modeling on the WWW*, 1999.
-
- [BDG95] R. Byrnes, R. Debreceeny, and P. Gilmour. The development of a multiple-choice and true-false testing environment on the web. In *Proceedings of The First Australian World Wide Web Conference, AusWeb95*, Ballina, Australia, 1995.
URL – <http://www.scu.edu.au/sponsored/ausweb/ausweb95/papers/education3/byrnes/>.
-
- [BE94] C. Boyle and A. O. Encarnacion. Metadoc: an adaptive hypertext reading system. *User Modeling and User-Adapted Interaction*, 4(4):1–19, 1994.
-
- [Bea94] I. Beaumont. User modeling in the interactive anatomy tutoring system ANATOM-TUTOR. *User Modeling and User Adapted Interaction*, 4(1):21–45, 1994.
-
- [BES98] P. Brusilovsky, J. Eklund, and E. Schwarz. Web-based education for all: a tool for development adaptive courseware. In *Proceedings of 7th International World Wide Web Conference*, Brisbane, Australia, 14–18 April 1998.
URL – <http://www7.scu.edu.au/programme/fullpaper/1893/com1893.html>.
-
- [BP94] P. Brusilovsky and L. Pesin. An intelligent learning environment for CDS/ISIS users. In J. J. Levonen and M. T. Tukianinen, editors, *Proceedings of The interdisciplinary workshop on complex learning in computer environment (CLCE94)*, pages 29–33, Joensuu, Finland, 1994. EIC.
URL – http://cs.joensuu.fi/~mtuki/www_clce.270296/Brusilov.html.
-
- [Bru92] P. Brusilovsky. Intelligent tutor, environment and manual for introductory programming. *Educational and Training Technology International*, 29(1):26–34, 1992.

URL - <http://www.contrib.andrew.cmu.edu/~plb/papers.html> .

- [Bru96] P. Brusilovsky. Method and techniques of adaptive hypermedia. *User Modeling and User Adapted Interaction*, 6(2-3):87-129, 1996.
- [Bru97] P. Brusilovsky. Efficient technique for adaptive hypermedia. In C. Nicholas and J. Mayfield, editors, *Intelligent Hypertext: Advanced Techniques for the World Wide Web*, number 1326 in Lecture Notes in Computer Science, pages 12-30. Berlin: Springer Verlag, 1997.
- [BSW96a] P. Brusilovsky, E. Schwarz, and G. Weber. ELM-ART: An intelligent tutoring system on world wide web. In G. Guathier C.Frasson and A. Lesgold, editors, *Third International Conference on Intelligent Tutoring Systems, ITS'96*, number 1086 in Lecture Notes in Computer Science, pages 261-269. Berlin: Springer Verlag, 1996.
URL - <http://www.contrib.andrew.cmu.edu/~plb/ITS96.html> .
- [BSW96b] P. Brusilovsky, E. Schwarz, and G. Weber. A tool for developing adaptive electronic textbooks on WWW. In *Proceedings of WebNet'97, World Conference of the Web Society*, San Francisco, CA, 15-19 October 1996.
URL - <http://www.contrib.andrew.cmu.edu/~plb/WebNet96.html> .
- [C2T] C2T2 (Centre for Curriculum, Transfer and Technology, Canada). A web tool for comparative analysis of on-line educational delivery applications. <http://www.c2t2.ca/landonline/>.
- [CED] CEDITI (Centre de Diffusion des Technologies de l'Information, Belge). <http://cediti.be/weds>.
- [CG77] B. Carr and I. Goldstein. Overlays: a theory of modeling for computer-aided instruction. *Memo 406 of Massachusetts Institute of Technology*, February 1977.
URL - <http://www.ai.mit.edu/publications.old/pubDB/makehtml> .
- [CIT] CIT (Center for Instructional Technology, Marshall University, USA). Comparison of online course delivery software products. <http://www.marshall.edu/it/cit/webct/compare/>.
- [CLK93] T. Chan, C. Lin, and H. Kuo. OCTR: A model of learning stages. In P. Brna and S. Ohlsson and H. Pain, editors, *Proceedings of AI-ED93*, Scotland, 1993. ACCE.
- [dRdCP94] F. de Rosis, B. de Carolis, and S. Pizzutilo. User tailored hypermedia explanations. In *conjunction with Proceedings of Fourth International Conference on User Modeling*, Hyannis, MA, 15-19 August 1994.
URL - <http://wwwis.win.tue.nl/ah94/deRosis.html> .
- [EH96] F. Espinoza and K. Höök. An interactive www interface to an adaptive information system. In *Proceedings of the workshops on "User Modeling for Information Filter*

on the World Wide Web" at the fifties international Conference on User Modeling, 1996. http://www.sics.se/~kia/espinoza_hook.html.

-
- [Gav91] E. Gagnon. *Environnement de Conception de Système d'Apprentissage: une modélisation de la connaissance pédagogique*. PhD thesis, Université Henry Poincaré - Nancy I, 1991.
-
- [Gav94] E. Gagnon. Instructional Expertise in ECSAI. In *Proceedings International Conference on Computed Aided Learning and Instruction in Science and Engineering*, pages 249–257. Paris, France, 1994.
-
- [GG92] E. Gagnon and M. Grandbastien. Représentation des connaissances dans un générateur de système d'EIAO. In C. Fråsson, G. Gauthier, and G. I. McCalla, editors, *Second International Conference on Intelligent Tutoring System, ITS'92*, number 608 in *Lecture Notes in Computer Science*, pages 209–216. Berlin: Springer Verlag, 1992.
-
- [GH95] M. Gonschorek and C. Herzog. Using hypertext for an adaptive help system in an intelligent tutoring system. In J. Greer, editor, *Proceedings of AI-ED'95, 7th World Conference on Artificial Intelligence in Education*, pages 274–281, Washington, DC, 16-19 August 1995.
-
- [GMC+98] J. Greer, G. McCalla, J. Collins, V. Kumar, P. Meagher, and J. Vassileva. Supporting peer help and collaboration in distributed workplace environments. *International Journal of Artificial Intelligence in Education*, 9:159–177, 1998.
-
- [GrG94] M. Grandbastien and E. Gagnon. ECSA: An environment to design and instantiate learning material. In T. de Jong and L. Sarti, editor, *Design and Production of Multimedia and Simulation Based Training Material*, Kluwer Academic Publisher, Amsterdam, 1994.
-
- [Gra99] M. Grandbastien. Teaching expertise is at the core of its research. *International Journal of Artificial Intelligence in Education*, 10:335–349, 1999.
-
- [GSS96] M. W. Goldberg, S. Salari, and P. Swoboda. World wide web – course tool: An environment for building WWW-based courses. In *Proceedings of Fifth International World Wide Web Conference*, volume 28 (7–11) of *Journal of Computer Networks and ISDN Systems*, page 1219, Paris, France, 1996.
-
- [Har99] Linda Harasim. A frame for online learning: The virtual-u. *Computer*, pages 44–49, September 1999.
-
- [HBG96] H. Hohl, H.D. Böcker, and R. Gunzenhäuser. Hypadapter: An adaptive hypertext system for exploratory learning and programming. *User Modeling and User-Adapted Interaction*, 6(2–3):131–156, 1996.
URL – http://laser.minerva.at:81/deutsch/d/data/KO/305319_6_2_3_1996.html.
-

-
- [HDJG93] P. Holt, S. Dubs, M. Jones, and J. Greer. The state of student modeling. In *Student Modeling: The Key to Individualized Knowledge-Based Instruction*, 125:66-97, 1993.
-
- [HKW⁺96] K. Höök, J. Karlgren, A. Wærn, N. Dahlbäck, C.G. Jansson, K. Karlgren, and B. Lemairer. A glass box approach to adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 6(2-3):157-184, 1996.
URL - http://laser.minerva.at:81/deutsch/d/data/KO/305319_6_2_3_1996.html .
-
- [HN99] N. Henze, and W. Nejdl. Adaptivity in the KBS-Hyperbook System. In *Proceedings of the 2nd Workshop on User Modeling and Adaptive Systems on the WWW*, Toronto, Canada, 11 May 1999. Holds in conjunction with the World Wide Web Conference (WWW 8), and the Seventh International Conference on User Modeling (UM 99).
URL - <http://www.kbs.uni-hannover.de/henze/paperadaptivity/Henze.html> .
-
- [Kap93] V. Kaptelinin. Item recognition in menu selection: The effect of practice. In *Proceedings of INTERCHI'93 (Adjunct Proceedings)*, pages 183-184, Amsterdam, 24-29 April 1993.
-
- [KFC93] C. Kaplan, J. Fenwick, and J. Chen. Adaptive hypertext navigation based on user goals and context. *User Modeling and User-Adapted Interaction*, 3(3):193-220, 1993.
-
- [KK94] J. Kay and B. Kummerfeld. An individualised course for the C programming language. In *Proceedings of Second International WWW Conference*, 17-20 October 1994.
URL - <http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Educ/kummerfeld/kummerfeld.html> .
-
- [KMN94] A. Kobsa, D. Mümller, and A. Nill. KN-AHS: An adaptive hypertext client of the user modeling system BGP-MS. In *Proceedings of Fourth International Conference on User Modeling.*, pages 99-105, Hyannis, MA, 15-19 August 1994.
URL - <http://www.ics.uci.edu/~kobsa/papers/1994-UM94-kobsa.ps> .
-
- [KW94] A. Kushniruk and H. Wang. A hypermedia-based educational system with knowledge-based guidance. In T. Ottman and I. Tomek, editors, *Proceedings of ED-MEDIA'94 - World conference on educational multimedia and hypermedia*, pages 335-340, Vancouver, Canada, 1994.
-
- [LCY95] M. Lai, B. Chen, and S. Yuan. Toward a new educational environment. In *Proceedings of 4th International World Wide Web Conference*, Boston, USA, December 11-14, 1995.
URL - <http://www.w3.org/pub/Conferences/WWW4/Papers/238/> .
-
- [LZ95] M. Linard and R. Zeiliger. Designing navigational support for educational software. In Unger Blumental, Gornostaev, editor, *5th International Conference EWHCI'95*, number 1015 in Lecture Notes in Computer Science, pages 63-78. Berlin: Springer Verlag, 1995.
-

-
- [Mai] Gilles Maire. Un nouveau guide internet - éléments d'html dynamique. <http://www.imagnet.fr/ime/frame41.htm>.
-
- [Mal94] J.C. Mallery. A common lisp hypermedia server. In *Proceedings of the First International Conference on the World Wide Web, WWW94*, CERN, Geneva (Switzerland), 25-27 May 1994.
URL - <http://www94.web.cern.ch/WWW94/jcma.ps>.
-
- [MENRT00] Le Ministère de l'éducation nationale, de la recherche et de la technologie. Étude comparative technique et pédagogique des plates-formes pour la formation ouverte et à distance, 2000.
<http://www.educnet.education.fr/superieur/plateforme.htm>
-
- [Mit00] A. Mitrovic. Porting SQL tutor to the web. In *Proceedings of the workshop on "Adaptive and Intelligent Web-Based Education Systems" of ITS2000*, pages 37-44, Montreal, Canada, 2000.
-
- [MO98] M. Milosavljevic and J. Oberlander. Dynamic hypertext catalogues: Helping users to help themselves. In *Proceedings of the 9th ACM Conference on Hypertext and Hypermedia (HT'98)*, Pittsburgh, PA, USA,, 20-24 June 1998.
URL - <http://www.mri.mq.edu.au/~mariam/papers/ht98/>.
-
- [MS89] J. D. Moore and W. R. Swartout. Pointing: A way toward explanation dialogue. In *Proceedings of Eight National Conference on Artificial Intelligence*, volume 14(3), pages 457-464, 1989.
-
- [MV95] J. Martin and K. VanLehn. Student assessment using Bayesian nets. *International Journal of Human-Computer Studies*, 42:575-591, 1995.
URL - <http://www.pitt.edu/~vanlehn/distrib/journal/HCS95-abs.html>.
-
- [Par88] C. L. Paris. Tailoring object description to a user's level of expertise. *Computational Linguistics*, 14(3):26-34, 1988.
-
- [PGL95] T. A. Perez, J. Gutierrez, and P. Lopsteguy. An adaptive hypermedia system. In J. Greer, editor, *Proceedings of AI-ED'95, 7th World Conference on Artificial Intelligence in Education*, pages 351-358, Washington, DC, 1995.
-
- [PSH95] A. Paiva, J. Self, and R. Hartley. Externalising learner models. In J. Greer, editor, *Proceedings of "Intelligent Educational Systems on the World Wide Web" of AI-ED95*. AACE, 1995.
-
- [Ram92] P. Ramsden. *Learning to teach in higher education*. London: Rutledge, 1992.
-
- [Rek83] T. Rekkedal. The written assignment in correspondence education. effects of reducing turn-around time. an experimental study. *Distance Education - An International Journal*, 4(2):231-252, 1983.

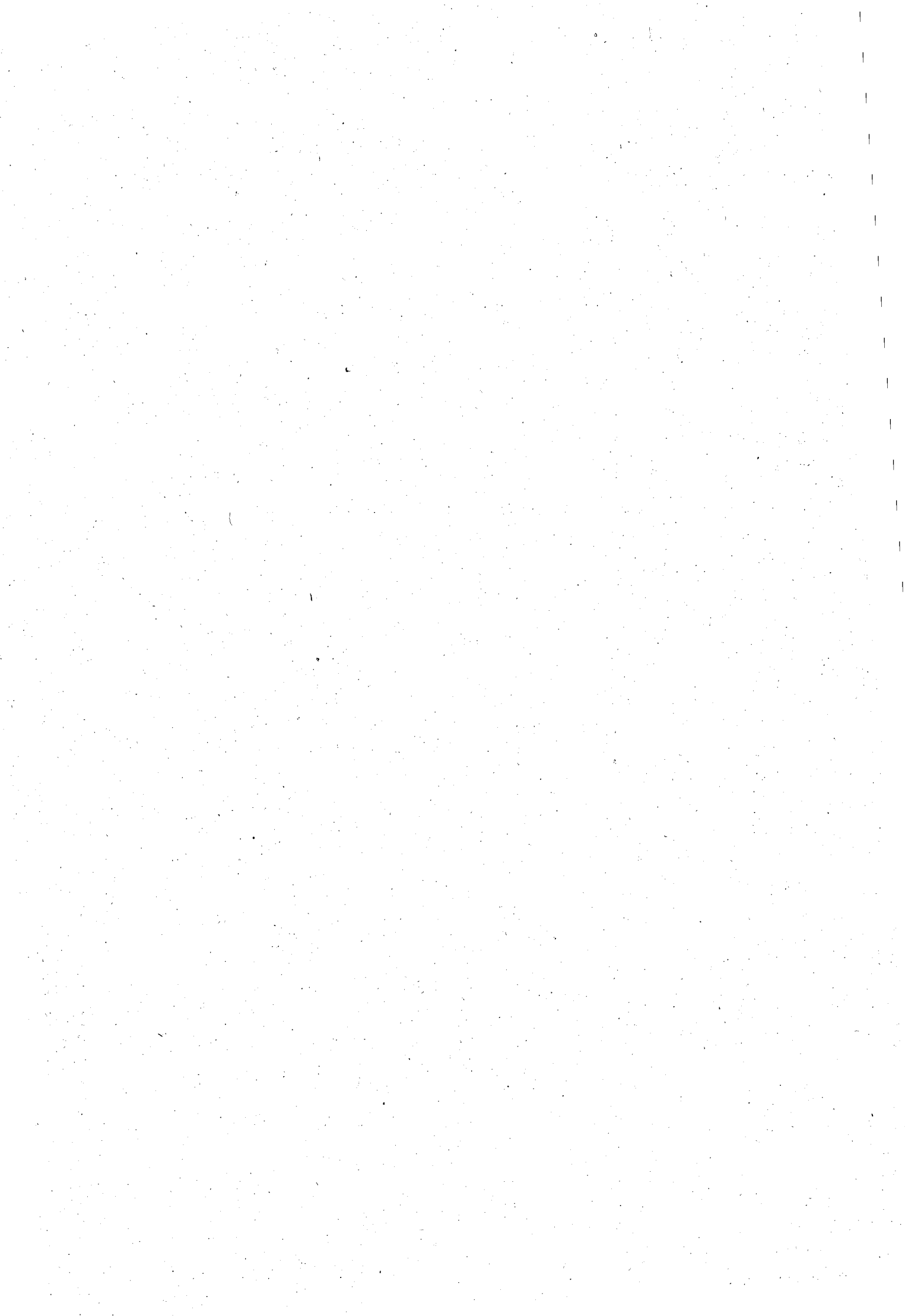
URL - <http://www.usq.edu.au/dec/decjournal/v4n283/issue.htm> .

- [RFJM95] R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell. WebWatcher: A learning apprentice for the world wide web. In *Proceeding of AAAI Spring Symposium on Information Gathering from Distributed, Heterogeneous Environments*, Stanford, CA, 1995.
URL - <http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-6/web-agent/www/webagent-plus.ps.Z> .
- [Rit97] S. Ritter. Pat online: A model-tracing tutor on the World Wide Web. In P. Brusilovsky, K. Nakabayashi, and S. Ritter, editors, *Proceeding of the workshop on "Intelligent Educational Systems on the World Wide Web"*, Kobe, Japan, 1997.
- [San99] C. Sanrach. *Mode d'Emploi d'ECSAIWEB*. LORIA No. 99-R-419, 1999.
- [San00] C. Sanrach. ECSAIWEB: An adaptive learning environment on the web. In *Proceeding of "The Third International Symposium on Wireless Personal Multimedia Communications"*, WPMC'00, volume 2 of 2, Bangkok, Thailand, 2000.
- [SB82] D. Sleeman and J. S. Brown, editors. *Intelligent Tutoring Systems*. Academic Press, 1982.
- [SG00a] C. Sanrach and M. Grandbastien. Adapter un environnement de conception de système d'apprentissage au web. In *Actes du colloque international "Technologie de l'Information et de la Communication dans les Enseignants d'ingénieurs et dans l'industrie"*, TICE2000, pages 43-52, Troyes, France, 2000.
- [SG00b] C. Sanrach and M. Grandbastien. An adaptive and intelligent web-based environment. In Vladik Kreinovich and Jiraphun Daengdej, editors, *Proceeding of "The International Conference on Intelligent Technologies"*, Intech2000, Bangkok, Thailand, 2000.
- [SG00c] C. Sanrach and M. Grandbastien. ECSAIWEB: A web-based authoring system to create adaptive learning systems. In P. Brusilovsky, O. Stock, and C. Strapparava, editors, *Adaptive Hypermedia and Adaptive Web-based System*, number 1892 in Lecture Notes in Computer Science, pages 214-226. Berlin: Springer Verlag, 2000.
- [Shu95] V.J. Shute. Smart: Student modeling approach for responsive tutoring. *User Modeling and User-Adapted Interaction*, 5(1):1-44, 1995.
URL - http://laser.minerva.at:81/deutsch/d/data/KO/305319_5_01_1995.html .
- [Site M206] Site M206. Computing: an object-oriented in British open university.
<http://www-cs.open.ac.uk/m206/m206home.html>.
- [SJ97] D. Suthers and D. Jones. An architecture for intelligent collaborative educational systems. In R. Mizoguchi B.de Boulay, editor, *Proceeding of "Intelligent Educational Systems on the World Wide Web" of AI-ED97*, pages 55-62, Kobe, Japan, 1997.

-
- [Spe] M. Specht. ACE – Adaptive Courseware Environment.
URL – http://129.26.167.17:8080/pub_html/html/NRHM/NRHM.html .
-
- [Sta] Active State. <http://www.activestate.com>.
-
- [Ste97] M. K. Stern. The difficulties in the web-based tutoring, and some possible solutions. In P. Brusilovsky, K. Nakabayashi, and S. Ritter, editors, *Proceeding of the workshop on "Intelligent Educational Systems on the World Wide Web" of AI-ED97*, Kobe, Japan, 1997.
-
- [Tea] NCSA HTTPd Development Team. CGI–common gateway interface.
<http://hoohoo.ncsa.uiuc.edu/cgi/overview.html>.
-
- [Top] TopClass. <http://www.avilar.com>.
-
- [Vas96] J. Vassileva. A task-centered approach for user modeling in a hypermedia office documentation system. *User Modeling and User-Adapted Interaction*, 6(2–3):185–223, 1996. URL – http://laser.minerva.at:81/deutsch/d/data/KO/305319_6_2_3_1996.html .
-
- [Vas97] J. Vassileva. Dynamic course generation on the www. In *Proceeding of the workshop on "Intelligent Educational Systems on the World Wide Web" of AI-ED97*, Kobe, Japan, 1997.
-
- [W3Ca] W3C organisation. CSS–Cascading Style Sheets. <http://www.w3.org/Style/CSS/>.
-
- [W3Cb] W3C organisation. DOM–Document Object Model. <http://www.w3.org/DOM/>.
-
- [W3Cc] W3C organisation. HTML–Hypertext Markup Language.
<http://www.w3.org/MarkUp/>.
-
- [W3Cd] W3C organisation. XML–Extensible Markup Language.
<http://www.w3.org/XML/>.
-
- [WCS91] L. Wall, T. Christiansen, and R. L. Schwartz. *Programming Perl*. O'Reilly & Associates, Inc, USA, 1991.
-
- [WCS96] L. Wall, T. Christiansen, and R. L. Schwartz. *Programmation en Perl, 2^e édition*. O'Reilly International Thomson, Paris, France, 1996. Traduction de Jean Zumdel.
-
- [Web] WebMentor. <http://www.wbtsystems.com>.
-
- [ZOF93] Z. Zhao, T. O'Shea, and P. Fung. Visualization of semantic relations in hypertext systems. In T. Ottman and I. Tomek, editors, *Proc. ED-MEDIA '93, World conference on educational multimedia and hypermedia*, pages 556–564, Orlando, FL, 1993. AACE.
-

Liste de Publications

- [San99] C. Sanrach. *Mode d'Emploi d'ECSAIWEB*. LORIA-INPL, LORIA No. 99-R-419, 1999.
-
- [San00] C. Sanrach. ECSAIWEB: An adaptive learning environment on the web. In *Proceedings of «The Third International Symposium on Wireless Personal Multimedia Communications», WPMC'00*, volume 2 of 2, Bangkok, Thailand, 2000.
-
- [SG00a] C. Sanrach and M. Grandbastien. Adapter un environnement de conception de système d'apprentissage au web. In *Actes du colloque international «Technologie de l'Information et de la Communication dans les Enseignants d'ingénieurs et dans l'industrie», TICE2000*, pages 43–52, Troyes, France, 2000.
-
- [SG00b] C. Sanrach and M. Grandbastien. An adaptive and intelligent web-based environment. In Vladik Kreinovich and Jiraphun Daengdej, editors, *Proceedings of «The International Conference on Intelligent Technologies», Intech2000*, Bangkok, Thailand, 2000.
-
- [SG00c] C. Sanrach and M. Grandbastien. ECSAIWEB: A web-based authoring system to create adaptive learning systems. In P. Brusilovsky, O. Stock, and C. Strapparava, editors, *Adaptive Hypermedia and Adaptive Web-based System*, number 1892 in Lecture Notes in Computer Science, pages 214–226. Berlin: Springer Verlag, 2000.
-



**AUTORISATION DE SOUTENANCE DE THESE
DU DOCTORAT DE L'INSTITUT NATIONAL
POLYTECHNIQUE DE LORRAINE**

o0o

VU LES RAPPORTS ETABLIS PAR :

Monsieur DERNIAME Jean-Claude, Professeur, UHP - NANCY 1 (rapporteur interne)

Monsieur PEGUCHET Jean-Pierre, Professeur, INSA Rouen, Mont Saint Aignan

Monsieur TCHOUNIKINE Pierre, Professeur, LIUM, Le Mans

Le Président de l'Institut National Polytechnique de Lorraine, autorise :

Monsieur SANRACH Charun

à soutenir devant un jury de l'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE,
une thèse intitulée :

**"Modèles pour la conception et la réalisation de formation au travers de réseaux :
L'environnement ECSAIWeb".**

NANCY BRABOIS
2, AVENUE DE LA
FORET-DE-HAYE
BOITE POSTALE 3
F - 54 5 0 1
VANDOEUVRE CEDEX

en vue de l'obtention du titre de :

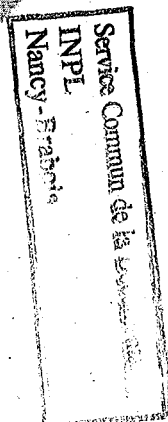
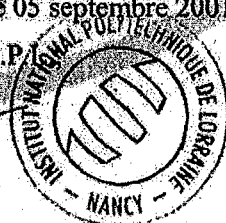
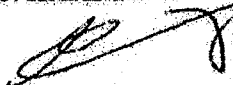
DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

Spécialité : « **Informatique** »

Fait à Vandoeuvre, le 05 septembre 2001

Le Président de l'IN.P.

J. HARDY



Résumé

Ce travail porte sur la conception et la réalisation de formations au travers de réseaux. L'une des difficultés rencontrées est que les systèmes proposés sont rarement acceptés par les enseignants car ils leur sont fermés : très peu de personnalisation des contenus et de prise en compte de leur pédagogie. L'objectif du travail est donc de produire des modèles généraux et de les implanter dans un environnement de création de cours et de déploiement de formation pour le Web, appelé ECSAIWEB. Cet environnement sera ouvert aux enseignants pour la création de matériaux et de stratégies pédagogiques et tirera au mieux partie des possibilités offertes sur le Web, tout en restant simple à déployer.

Le modèle du domaine, décrit à partir de concepts appelés items, est issu de l'environnement existant ECSAI. Ils sont organisés en réseaux et servent à modéliser en termes de contenus et de caractéristiques pédagogiques les modules d'apprentissage, appelés unités logiques. Ces unités logiques sont aussi organisées en réseau dynamique qui évolue au fur et à mesure des actions de l'apprenant et qui permet de calculer le parcours d'apprentissage en mode guidé. Les items servent également de base à un modèle d'apprenant utilisant le modèle classique de recouvrement, mais nécessitant une exploitation selon deux points de vue : construction dynamique du parcours apprenant ou adaptation des présentations. L'adaptation des présentations a nécessité également une modélisation de la présentation des unités à l'aide de balises de type XML, c'est le modèle des unités physiques.

Nous avons créé un environnement auteur ouvert. L'enseignant-auteur peut construire et organiser des matériaux pédagogiques dans sa matière, et observer leur utilisation afin de les améliorer. Nous avons créé un environnement élève qui permet l'exploration de ces matériaux pédagogiques, et un environnement partagé. Le système fonctionne selon un mode client-serveur sous Windows, utilise un navigateur Web standard.

Mots-clés : environnement d'apprentissage basé sur le Web, environnement d'auteur basé sur le Web, système tuteur intelligent basé sur le Web, formation sur le Web, système de distribution de cours basé sur le Web.

Abstract

This work concerns the design and the realization of formations through networks. One of the encountered difficulties is that the suggested systems are seldom accepted by the teachers for they are closed to them: very little personalization of the contents and taking into account of their pedagogy. The objective of work is thus to produce general models and to establish them in an environment for creating courses and deploying formations for the Web, called ECSAIWEB. This environment will be opened to the teachers for the creation of teaching materials and strategies and will draw the best part of the possibilities offered on Web, while remaining simple to deploy.

The domain model, described starting from concepts called items, results from existing environment ECSAI. They are organized in networks and are used to model in terms of contents and teaching characteristics of the learning modules, called logical units. These logical units are also organized in dynamic network which evolves progressively with the student's actions and that allow to calculate the learning path in guided mode. The items are used also as basic for a student model using the traditional overlay model, but requiring an exploitation according to two points of view: dynamic construction of the learning course or adaptive presentations. The adaptive presentations also required a modeling of the units presentation using the XML tags, it is the model of the physical units.

We created an open authoring environment. The teacher-author can build and organize teaching materials in his matter, and observe their use in order to improve them. We created a student environment that allows the exploration of these teaching materials, and a shared environment. The system functions according to a client-server mode under Windows, uses a standard Web navigator.

Keywords : Web-based learning environment, Web-based authoring environment, Web-based intelligent tutoring system, formation on Web, Web-based education delivery system.