

Numéro d'ordre :

50376
1995
45

Année : 1995

50376
1995
45

THESE

présentée en vue de l'obtention du

**Doctorat de l'Université
des Sciences et technologies
de Lille**

Spécialité : Informatique

par

CROISY Pascal

**Collecticiel temps réel et apprentissage coopératif :
des aspects sociaux et pédagogiques
jusqu'au
modèle multi-agent de l'interface de groupe**

Soutenue le 6 JANVIER 1995

Jury :

Président	M. Jean Marc GEIB
Directeur	M. Alain DERYCKE
Rapporteurs	Mme Joëlle COUTAZ M. Guy GOUARDERES
Examineurs	M. Dominique DECOUCHANT M. Anthony KAYE



SCD LILLE 1



D 030 305875 6

50376

1995

45

50376

REMERCIEMENTS

Les personnes que je souhaite remercier en tout premier sont mon épouse Dominique et mes deux filles Marion et Margot. Elles ont su me rappeler qu'elles existaient quand ma thèse me prenait trop de temps, et elles ont su aussi m'aider et m'encourager lorsque je baissais les bras.

Je veux remercier Madame Joëlle Coutaz, Professeur de l'Université Joseph Fourier de Grenoble et Monsieur Guy Gouardères, Professeur de l'Université de Pau d'avoir accepté d'être rapporteurs de ma thèse.

Je tiens à remercier Alain Derycke sans qui cette thèse n'aurait pas été possible. Il a su me diriger et m'encourager tout au long de ces trois années. Il m'a donné les moyens de réaliser ma recherche, il m'a fait confiance et a toujours respecté mes choix.

Je tiens aussi à remercier Claude Viéville, Frédéric Hoogstoel et Danièle Clément pour leur aide au niveau professionnel et amical. Claude et Frédéric ont toujours su choisir les bonnes options qui ont permis de terminer le projet dans lequel nous étions tous impliqués. Et enfin, Danièle a toujours été là pour m'écouter et me conseiller dans des domaines dans lesquels j'avais peu de compétence.

Le laboratoire Trigone ne serait pas comme il est actuellement, c'est à dire un endroit chaleureux et agréable, si Isabelle Logez n'en était pas la secrétaire.

Je veux aussi remercier Christian Brebion qui a su être présent quand j'avais besoin de lui, et qui a passé beaucoup de temps à relire ma thèse.

Et pour terminer, je tiens à remercier toutes les personnes que j'ai pu croiser, avec qui j'ai pu discuter et travailler : Laurent Barne, Patrice Carlier, Hervé Devos, Chantal D'Halluin, Jean-Noël Gers, Jacqueline Billon, Denis Lecoq et bien d'autres encore.

TABLE DES MATIERES

Introduction	1
---------------------------	----------

Chapitre 1 : Définitions, description et taxonomie	3
---	----------

I - Définitions du collecticiel.....	4
1 - Définitions	4
1.1 - La définition de Ellis & al.....	4
1.2 - La place du Collecticiel dans le CSCW	5
2 - Exemples de taxonomie	5
2.1 - Taxonomie par classes d'applications	5
2.2 - Classification espace temps	7
2.3 - Différentes stratégies (approches) pour la conception de collecticiels	8
II - Eléments pour une taxonomie du collecticiel temps réel	10
1 - WYSIWIS : What You See Is What I See	10
2 - La rétroaction de groupe (feed-back).....	11
3 - Gestion des conflits ou médiation?	12
4 - Espace d'information.....	12
5 - La notion de rôle et de droit.....	13
6 - Architecture logicielle d'un collecticiel.....	13
III - Analyse de quelques prototypes existants.....	17
1 - Les éditeurs	17
1.1 - Grove	17
1.2 - Alliance.....	19
1.3 - GroupDraw	21
1.4 - GroupDesign.....	22
2 - Les architectures d'accueil et les boîtes à outils	23
2.1 - GroupKit.....	23
2.2 - Rendezvous et le paradigme ALV	25
2.3 - Suite	27
3 - Extension des systèmes de fenêtrage	29
SharedX (ou XShared)	29
IV - Etat récapitulatif de l'analyse	32
1 - Le Wysiwis et sa relaxation	32
2 - L'espace d'information	32
3 - Gestion des conflits ou médiation.....	32
4 - La rétroaction	34
5 - Parallélisme d'interaction	34
6 - Intégration du multimédia.....	34
7 - L'interface	34
8 - Le support du groupe	34
9 - Etat récapitulatif.....	35
V - Bilan et perspectives	39

Chapitre II : La complexité d'une activité coopérative.....	43
--	-----------

I - Mieux comprendre le travail coopératif pour mieux l'assister !	44
1 - Le travail coopératif.....	44
1.1 - Conditions d'apparition du travail coopératif.....	44

1.2 - Les caractéristiques du travail coopératif.....	44
Le travail articulaire	44
Perception des autres.....	46
Dynamique.....	46
Conclusion	46
2 - Différents points de vue sur le travail coopératif.....	46
1.1 - Le point de vue social et organisationnel sur le travail coopératif.....	47
1.2 - Le point de vue communicationnel sur le travail coopératif.....	47
Le rôle du langage.....	47
La communication de groupe.....	48
La communication non verbale.....	49
3 - Un exemple de travail coopératif : l'apprentissage coopératif	50
Qu'est ce que l'apprentissage coopératif ?.....	50
Fondements théoriques de l'apprentissage coopératif	51
Conclusion à partir des aspects théoriques.....	52
II - Retour au Collecticiel.....	54
1 - Les leçons du passé.....	54
2 - Description simpliste d'une situation de travail coopératif mettant en jeu un système d'information et de communication.....	55
3 - L'espace d'information	56
4 - L'espace de conversation.....	57
Le problème de la communication non verbale	57
Le rôle du média dans la communication.....	57
5 - Articulation de l'espace d'information et de l'espace de conversation	58
6 - Le problème de la gestion des conflits.....	60

Chapitre III : Analyse et spécification63

I - Description générale.....	64
1 - Les choix qui structurent l'environnement Co-Learn.....	64
1.1 - Le choix de métaphores	64
1.2 - L'environnement doit être "façonnable"	65
1.3 - Le choix d'une démarche	66
2 - Le centre de ressources pédagogiques virtuel : Co-Learn.....	66
II - Le prototype de conférence temps réel de Co-Learn : RTTC (Real Time TeleConferencing)	70
1 - L'expérience du laboratoire Trigone	70
2 - Le contenu d'une pièce de type RTTC.....	72
2.1 - Les situations pédagogiques envisagées	73
Présentation d'information	73
Démonstration.....	73
Séminaires et discussions de groupe	73
Les jeux, les simulations, les projets de conception, et les études de cas.....	74
2.2 - La description des outils	74
L'outil de visualisation collectif et le télépointeur.....	75
L'outil de vote ou collecteur d'opinion	77
Le questionnaire de temps.....	78
L'outil de bavardage.....	79
L'outil de gestion du canal sonore.....	80
Le questionnaire de conférence.....	80
3 - L'architecture logicielle d'un outil.....	82
3.1 - Présentation de l'architecture logicielle.....	82
3.2 - La dimension multi-utilisateur.....	83
Prise en compte de la rétroaction de groupe	83
Un outil est utilisé dans un contexte social	83
3.3 - Exemple de spécification d'un outil	85
L'outil de télépointage.....	85
L'outil de vote	86
3.4 - La structuration et le contrôle au sein d'un outil	86

Chapitre IV : Conception et réalisation91

I - Architecture physique mise en oeuvre pour Co-Learn	92
II - Architecture logicielle	95
Le composant Organisation.....	95
Le système de communication (SC).....	96
Le Launcher	97
Le Hall Central (HC) et le Hall Local (HL)	97
Le Gestionnaire de Pièce Local (GPL).....	99
Le gestionnaire de Pièce Central	100
L'agent de médiation local.....	100
L'agent de médiation central	103
La présentation	103
III - La réalisation du composant présentation dans l'environnement Windows 3.1	105
3.1 - L'environnement de développement et ses caractéristiques	105
3.2 - La programmation d'une présentation.....	107

Chapitre V : Mise en oeuvre et évaluation112

I - La mise en oeuvre de RTTC au sein de l'institut CUEEP.....	113
1 - L'installation matérielle et logicielle	113
2 - La gestion des utilisateurs et les problèmes humains	113
II - L'expérimentation de RTTC.....	115
1 - Préparation de l'expérimentation.....	115
1.1 - La configuration matérielle.....	115
1.2 - Les ressources humaines.....	115
1.3 - Le Protocole de connexion des stations Co-Learn.....	115
1.4 - La formation	116
1.5 - La séance pédagogique	116
2 - Le déroulement de la séance	117
III - Conclusions sur l'utilisation du prototype au CUEEP.....	120
1 - Les aspects techniques et fonctionnels.....	120
2 - Les aspects pédagogiques	123
3 - Les aspects socio-cognitifs.....	123
IV - Définition de l'utilité du prototype.....	126
1 - La configuration matérielle et humaine	126
2 - Un scénario type pour la téléconférence temps réel.....	126
3 - Un exemple d'utilisation possible du système de conférence	128
3.1 Phase de présentation du problème.....	128
3.2 Répartition des sous-tâches	128
3.3 Travail personnel (même tâche, tâches différentes).....	128
V - Etude d'utilisabilité du système	129
La métaphore de Pièce	129
Les outils de la pièce de conférence temps réel	131
Conclusion.....	133

Chapitre VI : Proposition et recommandations134

I- Pourquoi une approche multi-agent ?	135
1 - Les caractéristiques d'un modèle multi-agent	135
2 - Quelques exemples de modèles multi-agent dans les interfaces homme machine	137
2.1 - Le modèle PAC.....	137
2.2 - Le modèle MVC	138
2.3 - Le modèle MVC étendu.....	139
2.5 - Rendezvous et le paradigme ALV	140
3 Conclusion sur le modèle multi-agent pour la conception de collecticiels	141
II - Un modèle d'architecture multi-agent pour système interactif de groupe	143
1 - Quels sont les objectifs du modèle en fonction des enjeux du collecticiel	143

2 - Les composants du modèle	143
Le composant Abstraction.....	144
Le composant Présentation.....	144
L'agent organisationnel	144
Ses connaissances :	144
Lien de communication :	145
Processus de traitement	145
Agent de Rétroaction.....	145
Ses Connaissances.....	145
Lien de communication.....	146
Le processus de traitement	146
Agent de rôle	146
Ses connaissances	146
Lien de communication.....	147
Processus de traitement	147
Agent d'interface	147
Ses connaissances.....	147
Lien de communication.....	148
Le processus de traitement	148
3 - La relation de délégation.....	148
4 - Les flux de messages et le filtrage	149
Le partage d'une ressource	150
La prise en compte de la rétroaction de groupe	151
5 - Apport du modèle	152
Avantages du modèle	152
Inconvénients	153
6 Conclusion.....	153
III - Quelques recommandations	155
1 - Les architectures de communication	155
2 - Les systèmes de fenêtrage.....	156
3 - Comment prendre en compte la dimension sociale ?.....	157
Conclusion.....	157

Conclusion	158
-------------------------	------------

Bibliographie	160
----------------------------	------------

Annexe 1	174
-----------------------	------------

Annexe 2	181
-----------------------	------------

Annexe 3	187
-----------------------	------------

INTRODUCTION

Nous sommes dans une société où la communication prend de plus en plus d'importance : développement des autoroutes de l'information, présentation au grand public du réseau Internet et de logiciels d'accès à l'information type Mosaïc (Emission "La marche du siècle" de Jean-Marie Cavada du 25 Octobre 1994). L'émergence de cette société de communication est accompagnée par l'émergence de nouvelles technologies : les stations multimédia avec des périphériques et cartes spécialisées pour la compression audio et vidéo. C'est à dire que l'on a intégré au niveau technologique dans la même station de travail plusieurs canaux de communication.

Ce besoin de communication multimédia devient une réalité dans des situations d'enseignement difficiles pour s'affranchir des distances. Il apparaît aussi dans des situations de formation continue pour s'affranchir du temps et de la distance, afin de fournir de nouveaux moyens de communication mieux adaptés aux activités professionnelles et aux besoins des personnes. En résumé, les situations qui nécessitent des moyens de communication sont généralement des activités dans lesquelles les coûts de transport (humain) et le coût du temps passé dans ceux-ci sont importants. C'est dans ce contexte que les environnements coopératifs médiatisés par les technologies de l'informatique peuvent être d'un grand apport.

Pourtant les collecticiels et plus généralement les CSCW¹ (Computer Supported Collaborative work) ne sont pas encore des produits fortement utilisés dans les entreprises ou les organisations. Ce malaise peut s'expliquer par la complexité de compréhension du travail collectif, qui met en jeu des acteurs de différentes natures (humaines et technologiques) qui sont encore mal connus. Chaque personne et chaque groupe sont différents alors que les collecticiels souvent réalisés par des informaticiens, ne prennent pas en compte ces différences interpersonnelles. Pourtant, lorsque l'on fait un bilan sur l'origine des publications dans les colloques internationaux (CSCW) et européens (ECSCW), le nombre de publications est réparti entre les disciplines de l'Informatique et celles des Sciences Sociales et Cognitives. Les collecticiels nécessitent de mettre en commun des compétences multiples appartenant à des disciplines différentes. L'échec actuel de certains collecticiels n'est-il pas lié à la composition en général mono-disciplinaire des équipes de recherche. La recherche sur les collecticiels nécessite sûrement la constitution d'équipes de chercheurs pluridisciplinaires, qui auront besoin d'apprendre à travailler ensemble. Alors essayons d'appliquer ce que l'on connaît du travail coopératif d'abord à l'univers de la recherche !

¹Le terme officiel en français correspondant à CSCW est Synergiciel

Introduction

Cette thèse s'intéresse aux collecticiels temps réel sous différents aspects : fondements théoriques, analyse, conception et réalisation de collecticiels et d'environnements collecticiels, mise en oeuvre et évaluation dans le domaine de l'enseignement.

Le chapitre I est un état de l'art des collecticiels temps réel qui a pour objectif de définir un état comparatif entre les différents prototypes, après avoir précisé les éléments de taxonomie.

Le chapitre II commence par une définition du travail coopératif : ses origines, ses fondements théoriques. Cette partie se termine par une présentation des fondements théoriques de l'apprentissage coopératif, afin de définir le champs applicatif associé à nos travaux. Ensuite, nous essayons de donner une interprétation des aspects théoriques dans le monde du CSCW, afin d'énoncer certains enjeux, en partant d'une situation simple de coopération médiatisée par des dispositifs informatiques.

Le chapitre III est consacré à l'analyse et les spécifications d'un collecticiel, ou plus précisément d'une collection de petits collecticiels qui composent un élément important d'un environnement CSCW, adapté à l'apprentissage coopératif à distance. Cet environnement a été réalisé au cours d'un projet nommé Co-Learn, soutenu par la Communauté Européenne dans le cadre du programme DELTA.

Le chapitre IV correspond à la description de la réalisation non pas de l'environnement total, mais des travaux réalisés par l'auteur de cette thèse.

Le chapitre V est consacré à la mise en oeuvre et à l'évaluation de l'environnement Co-Learn au sein de l'institut CUEEP (Centre Universitaire d'Enseignement et d'Economie Permanente). Nous voulons insister sur la difficulté de mise en oeuvre d'un environnement CSCW dans une organisation. La deuxième partie de ce chapitre est la présentation d'expérimentations de l'environnement en situation réelle, et d'une première étude de l'utilité et de l'utilisabilité du système, suite aux expérimentations réalisées au CUEEP et dans d'autres sites pilotes du projet (notamment ICBL Institute for Computer Based Learning de l'Université Herriot Watt qui est responsable de l'étude d'utilisabilité dans le projet).

Le chapitre VI a pour objectif de proposer tout d'abord une architecture multi-agent qui identifie des agents pour la gestion de la médiation, du contrôle et de la rétroaction de groupe. Ensuite nous proposons une liste de recommandations pour les futures générations de systèmes de fenêtrage et de boîtes à outils.

La conclusion donne les perspectives liées à ce travail de recherche en terme de projet sur les interfaces (apport du 3D et dimension métaphorique), et en terme d'architecture d'accueil. Nous décrivons aussi l'avenir de l'environnement Co-Learn.

CHAPITRE 1

DEFINITIONS, DESCRIPTION ET TAXONOMIE

Ce chapitre est un état de l'art des collecticiels temps réel qui a pour objectif de définir un état comparatif entre les différents prototypes après avoir précisé les éléments de taxonomie.

I - Définitions du collecticiel

Nous nous efforçons dans cette partie de donner un éclairage sur la définition de collecticiel ou "groupware". Ce qui est important pour nous, c'est de donner notre définition de collecticiel et de la replacer dans le contexte plus général du "Computer Supported Cooperative Work", CSCW. L'AFCEC propose la traduction suivante pour CSCW : "Activités Multiparticipantes Informatisées".

1 - Définitions

1.1 - La définition de Ellis & al

Ellis donne la définition suivante [Ellis 91] : "Computer based system that support group of people engaged in a common task (or goal) and that provide an interface to shared environment".

Il faut souligner dans cette définition trois termes importants :

- "group of people": c'est une condition nécessaire pour parler de travail collectif. Il faut d'abord parler d'un groupe de personnes, dont chaque membre a connaissance de l'existence de chacun. Chaque utilisateur doit avoir la possibilité de se faire connaître du groupe et inversement¹ ;
- "common task" : les utilisateurs ont un objectif commun² qui est la réalisation d'une tâche. Chaque utilisateur va orienter son activité pour atteindre ce but. La tâche commune va être définie par le but à atteindre, sa durée, le groupe concerné et son organisation sociale, les moyens mis à la disposition des participants ;
- "shared environment" : il regroupe l'ensemble des moyens mis à la disposition du groupe pour communiquer, pour créer, pour négocier, pour prendre des décisions... C'est cette partie qui fait l'objet de beaucoup de recherches actuellement, car c'est là que la puissance informatique peut apporter une grande richesse afin de supporter les activités de groupe. Nous verrons par la suite, un ensemble d'exemples qui montreront l'ampleur et les thèmes de recherche sur le collecticiel.

Lorsque nous parlerons de collecticiel, il faudra l'interpréter suivant la définition précédente.

¹Il existe également des groupes où l'anonymat est préservé

²L'objectif commun est une notion vague qui génère plusieurs interprétations de cette définition

1.2 - La place du Collecticiel dans le CSCW

Nous empruntons notre définition du CSCW à L. Bannon et K. Schmidt [Bannon 91].

"Le CSCW est le domaine de recherche répondant aux questions suivantes : quelles sont les caractéristiques spécifiques du travail coopératif comme opposé au travail effectué par des individus isolés ? Comment l'informatique peut être appliquée pour soutenir les problèmes logistiques du travail coopératif ? Comment les concepteurs abordent les problèmes délicats et complexes de conception des systèmes qui façonnent les relations sociales ?"

Ce qui définit le collecticiel, c'est la logistique technique à mettre en oeuvre pour assister le travail coopératif. Il est aussi évident que la conception des collecticiels nécessite de prendre en compte le fait qu'ils aient des répercussions sur les relations humaines, donc que le processus de conception doit tenir compte de la complexité de la gestion d'un groupe et du contexte organisationnel dans lequel le groupe pourra évoluer et dans lequel le collecticiel sera utilisé.

En résumé, le collecticiel est un système informatique créé pour assister une tâche clairement définie à l'avance (ce qui est en adéquation avec la définition de Ellis). On peut encore illustrer l'importance de l'existence d'un but, d'une tâche dans un collecticiel, en revenant à la création du terme groupware [Ganymède 93]. Il a été inventé par Johnson-Lentz pour décrire des processus et des procédures intentionnelles de GROUPE pour accomplir un but spécifique auquel il faut ajouter des outils logiciels (softWARE) conçus spécifiquement pour supporter et faciliter le travail de groupe (cité dans [Hiltz 90]).

En conclusion, le collecticiel s'intéresse à la conception des outils pour assister le travail en groupe, et le CSCW qui englobe le collecticiel s'intéresse aussi aux aspects sociaux, à la définition du travail coopératif, à l'impact de l'utilisation de collecticiels dans les organisations...

2 - Exemples de taxonomie

2.1 - Taxonomie par classes d'applications

La classification présentée ci-dessous est celle proposée par Ellis & al [Ellis 91]. Elle a pour but de donner un aperçu de l'étendu du "Groupware", et par la suite de situer les différentes activités les unes par rapport aux autres. Cette classification met en évidence les caractéristiques intrinsèques de ces systèmes.

Messagerie : on regroupe dans cette catégorie les échanges asynchrones de messages entre les utilisateurs d'un groupe. L'exemple le plus significatif est le courrier électronique que l'on peut trouver sur station de travail. Mais il faut aussi y inclure l'ensemble des CMC (Computer Mediated Conference) qui supportent la communication écrite entre les participants. Ces échanges se font de manière asynchrone : c'est-à-dire que les participants n'ont pas à se soucier de la présence de leur interlocuteur, le système se chargeant du stockage et de l'acheminement des messages. Ces systèmes offrent à l'utilisateur d'autres fonctionnalités telles que la structuration de la conversation (filtrage et tri des messages) [Malone 87], des outils qui vont permettre d'accroître la productivité du groupe (outil de vote...). De tels systèmes ont déjà fait leur preuve d'"utilisabilité"³, par exemple dans l'enseignement à distance (système Cosy utilisé à l'Open University).

Editeur multi-utilisateur : il s'agit ici de proposer à un groupe d'utilisateurs d'éditer et de composer un document ensemble, celui-ci étant naturellement multimédia. Dans ces systèmes, les utilisateurs pourront ensemble visualiser, modifier, créer le même document simultanément. Ceci implique la mise en place de stratégies afin de régler les conflits entre les participants. Ceux-ci vont pouvoir être gérés soit par le système informatique, soit par le groupe lui-même (par exemple un participant ayant le rôle d'animateur du groupe). On voit apparaître des éditeurs de dessin tel que GroupDraw [Greenberg 92] ou GroupDesign [Karsenty 93], de texte tel que Grove [Ellis 90] ou Alliance [Decouchant 92,93]. Le principal souci de ces éditeurs est d'implémenter une technique permettant la résolution des conflits, donc de permettre l'exploitation collective et simultanée d'un même document.

"Group Decision Support Systems" GDSS et salle de réunion informatisée : son but est de faciliter la prise de décision au sein d'un groupe, en proposant des outils de structuration, de présentation d'information, et de vote. La plupart de ces GDSS sont implémentés sous forme d'une salle de réunion informatisée : par exemple Colab de Xerox Parc [Stefik 87a] qui permet de gérer une réunion de cinq à six personnes dans une salle spécialement équipée, en proposant un système de dessin à main levée, d'annotation de document, et un outil d'aide à l'argumentation.

Conférence : dans ce type d'activité, l'ordinateur est placé comme un intégrateur du système de communication. Nous distinguons trois types de conférences :

- Conférence temps réel assistée par ordinateur : elle permet à un petit groupe de personnes d'interagir en temps réel comme dans une réunion de type face à face. Les membres du groupe peuvent se trouver dans une salle de réunion

³L "utilisabilité" a pour objectif d'appréhender la facilité d'apprentissage et d'utilisation d'un système par un utilisateur

informatisée ou être physiquement distants. Dans le deuxième cas, il faudra bien sûr prévoir un lien audio et/ou vidéo permettant la communication au sein du groupe. Ce qui est important, c'est l'application fournie par le système qui permet l'activité du groupe.

- Téléconférence assistée par ordinateur : les interactions du groupe sont supportées par les moyens de télécommunications actuels (lien audio et vidéo). L'ordinateur est un intégrateur des différents médias utilisés. C'est-à-dire que l'organe de restitution est la station de travail, sur laquelle on voit nos interlocuteurs, des documents audio ou vidéo. Ceci correspond à l'arrivée de la station multimédia.
- Téléconférence multimédia temps réel assistée par ordinateur : elle est l'association des deux précédentes. Elle permet de partager des documents et des applications multimédia tout en bénéficiant de liens de communication multimédia (image vidéo des interlocuteurs).

Agents intelligents : on introduit dans le système un agent intelligent informatique qui prend place comme participant du groupe. Par exemple, dans un jeu, un agent peut tenir le rôle d'un des joueurs.

La plupart des systèmes actuels regroupent des fonctionnalités des différents types de collecticiels décrits précédemment.

D'autres classifications sur les collecticiels sont proposées. Par exemple, Johansen [Johansen 88] définit 17 catégories de travail coopératif assisté par ordinateur. Cette taxonomie s'attache à définir le contexte d'utilisation. De plus Johansen propose une classification des systèmes coopératifs en prenant en compte les dimensions temporelle et spatiale.

2.2 - Classification espace temps

Johansen [Johansen 88], puis Rodden [Rodden 91] proposent une taxonomie par grandes classes d'applications représentées dans une matrice à deux dimensions (voir figure1). La dimension temporelle caractérise la nature des interactions utilisateurs. Elles peuvent être synchrones, c'est à dire que les utilisateurs utilisent simultanément l'application collective et en ont pleinement conscience, ou asynchrones, c'est à dire que les utilisateurs utilisent le collecticiel quand ils le veulent (indépendamment des autres). La dimension spatiale met en jeu l'organisation géographique des utilisateurs : soit les utilisateurs sont proches les uns des autres (dans la même pièce), soit ils sont très éloignés.

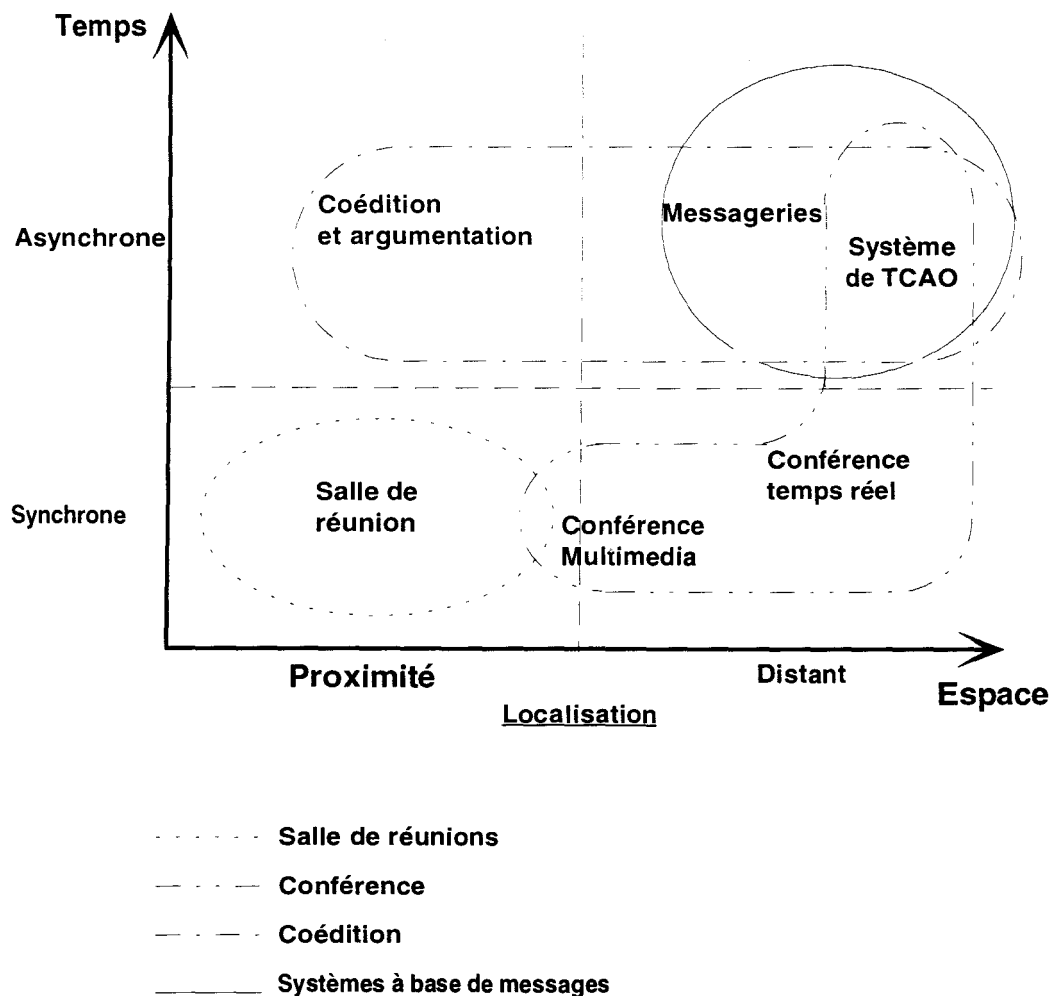


Figure 1 : Classification espace temps des collecticiels d'après Rodden

2.3 - Différentes stratégies (approches) pour la conception de collecticiels

Cette classification [Viéville 94a] est basée sur l'élaboration de différents points de vue ou approches sur la conception de collecticiel. Son objectif est de montrer la diversité des approches dans le domaine, et donc d'illustrer la complexité inhérente à l'étude des collecticiels et plus généralement du CSCW. Elle définit quatre classes :

- Interfaces multi-utilisateur distribuées : il faut voir ici principalement les systèmes basés sur la répartition de XWindow, en vue de son adaptation aux activités coopératives [Hill 92][Gust 88][Dewan 91]. Une définition plus générale serait l'ensemble des systèmes qui utilisent la structure d'un SGIU (Système de Gestion d'interface Utilisateur) ou UIMS [Ménadier 93] (User Interface Management System) afin de la répartir, soit au niveau du noyau fonctionnel (sémantique de l'application) soit au niveau de l'interface utilisateur et en utilisant les différents liens entre ces deux composants. Cette catégorie va bien souvent mettre en oeuvre des systèmes temps réel ;

- Espace d'information partagé : plusieurs approches du collecticiel se basent sur l'existence d'un espace d'information géré bien souvent par une base de données ou un serveur d'hypermédia [Haake 92]. Partant des systèmes de base de données multi-utilisateur ou des systèmes de gestion de fichiers, ils essaient de supporter les accès coopératifs et de définir une notion de rétroaction multi-utilisateur ou de groupe (perception des manipulations des autres)[Decouchant 94] ;
- Orienté médias et réseaux : dans cette catégorie, on peut définir deux sous approches. La première relative à la communication multi-partie et à ses extensions pour supporter le multimédia. Le concepteur fournit une architecture de base qui supporte la communication multi-partie en respectant la synchronisation temporelle et causale [Ravindran 92]. La deuxième rassemble l'ensemble des approches basées sur la messagerie électronique ou CMC (Conference Mediated Computer) [Kreifelts 93][Prinz93];
- Génie logiciel : cette approche est basée sur la conception logicielle d'un collecticiel. Il s'agit ici de définir quelles sont les méthodes d'analyse et de conception qui vont permettre d'identifier des composants génériques associés à la conception de collecticiel [Ben-Shaul 1993].

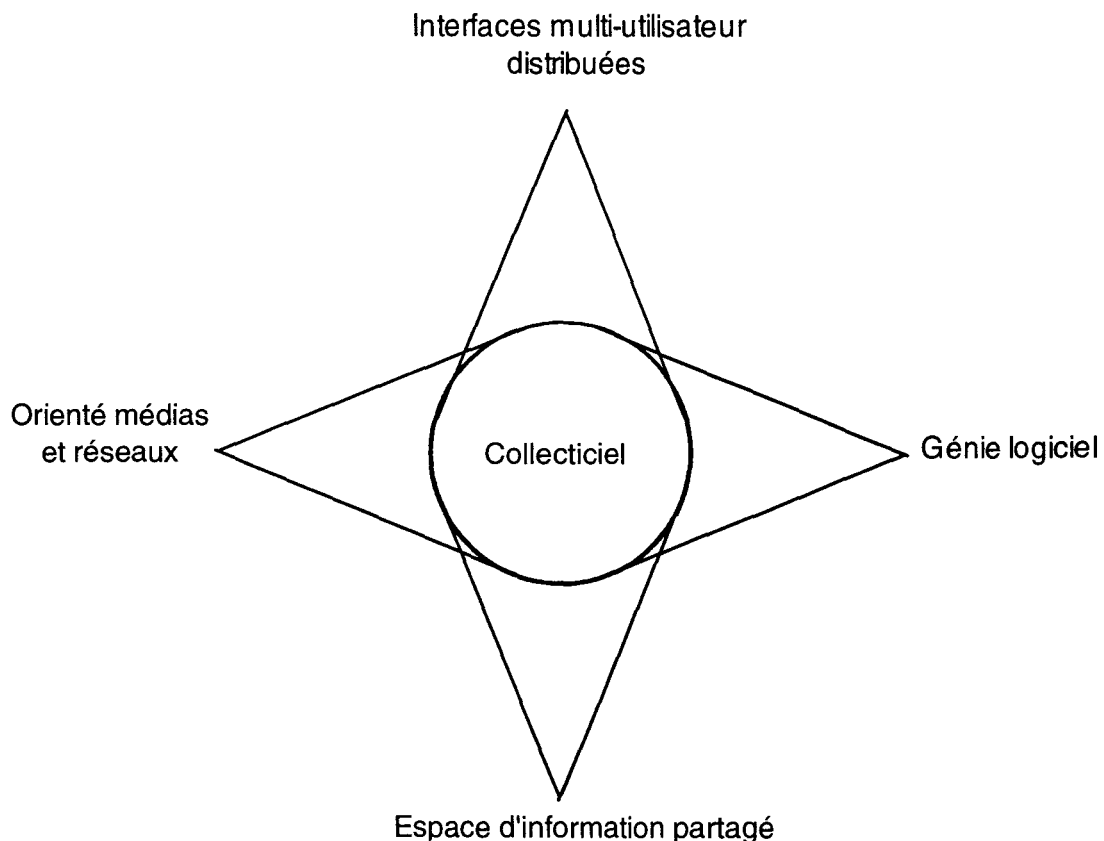


Figure 2 : Quatre approches pour la conception de collecticiel

II - Éléments pour une taxonomie du collecticiel temps réel

L'objectif de cette section est de définir une terminologie et un ensemble de concepts du collecticiel temps réel qui vont nous permettre de nous repérer dans l'analyse des systèmes existants. Cette terminologie trouve son origine dans l'ensemble de la bibliographie sur le CSCW, mais n'est en aucun cas exhaustive. Notre choix s'est porté sur les termes qui nous semblent génériques. C'est à dire ceux qui vont apparaître dans beaucoup de prototypes sous des formes différentes. En définissant cette terminologie, nous voulons ébaucher une grille de lecture qui va nous permettre d'établir un état comparatif des systèmes les plus pertinents.

1 - WYSIWIS : What You See Is What I See

Cette abstraction a été introduite par Stefik [Stefik 87a] et apparaît fondamentale dans la conception des systèmes multi-utilisateur temps réel où la visualisation d'information est centrale. Elle est inspirée des systèmes réalisés pour supporter les activités de réunion ayant un tableau noir pour support.

Respecter cette abstraction, c'est permettre à tous les utilisateurs de voir les mêmes informations sous la même forme et en même temps. Ceci va permettre aux utilisateurs de faire des références spatiales sur l'information visualisée. Décrire la position d'un objet visualisé dans l'espace WYSIWIS, va permettre une identification facile par l'ensemble des utilisateurs.

Le reproche que l'on peut faire aux systèmes qui respectent cette abstraction, est leur manque de flexibilité. L'activité humaine dans un contexte coopératif est pleine d'imprévus, de spontanéité, et les expériences faites avec certains systèmes tels que Colab [Stefik 87a, 87b] [Tatar 91], Grove [Ellis 91] et sur une plate-forme de prototypage du laboratoire Trigone [Vilers 92], montrent que le respect du WYSIWIS strict peut entraîner un dysfonctionnement au niveau de l'activité humaine.

C'est pour cela que Stefik introduit différents niveaux de WYSIWIS, en relâchant certaines contraintes. Ces relaxations vont être de quatre types : spatiale, temporelle, de population et de la congruence.

- la relaxation *spatiale*, permet de définir un espace de vision de l'ensemble des objets visibles (ensemble des objets visibles simultanément par tous les utilisateurs). Par exemple, elle permet à l'utilisateur de définir la taille de sa fenêtre, de poser son attention sur un sous ensemble des objets visibles par une sélection de la zone d'affichage, de placer cette fenêtre à l'endroit où il veut ;

- définir la relaxation *temporelle*, c'est permettre un délai dans la mise à jour de l'affichage entre les utilisateurs, celui-ci étant perceptible et consciemment défini à la conception du système. Le système introduit une désynchronisation entre les affichages d'informations associées à chaque utilisateur ;
- relaxation de la contrainte de *population* : c'est la possibilité de pouvoir créer des sous groupes. C'est à dire de permettre à un sous ensemble d'utilisateurs de voir la même information à un moment donné, l'information visible étant différente entre les sous groupes ;
- relaxation de la *congruence* : les utilisateurs voient la même information à une constante de représentation prêt. Par exemple, on voit le même objet à la couleur prêt, un tableau de valeurs apparaît sous forme de tableau ou de camembert (visions multiples d'un même objet).

2 - La rétroaction de groupe (feed-back)

Définir la rétroaction de groupe, c'est définir ce que voit un utilisateur de l'activité des autres. Ce concept pose un grand nombre de problèmes. Ils sont de différentes natures et appartiennent à des champs disciplinaires relativement différents tels que l'informatique, la psychologie et la sociologie.

Comment moi utilisateur du système, ayant une position hiérarchique ou "sociale" (par exemple rôle dans l'entreprise) contrôle cette divulgation d'information. Il peut exister un contrôle *temporel*. **Le quand ?** Mais qui décide quand la divulgation d'information se fait. **Par Qui ?** Ou bien, il n'existe aucun contrôle : je sais, ou je ne sais pas que toute mon activité dans le collecticiel est vue par les autres. Et que voient les autres utilisateurs de mon activité. **Le Quoi?** Le quoi va définir ce que l'on peut appeler la *granularité* de la rétroaction.

En conclusion, il apparaît qu'une plate-forme de support d'activités coopératives, se doit de gérer ces questions. Une solution possible est de répondre arbitrairement à l'ensemble de ces questions (par exemple en respectant le WYSIWIS strict). Mais dans ce cas, cette architecture ne répondra pas aux besoins de toutes les situations mais deviendra spécifique à un domaine et plus particulièrement à une organisation. La conclusion qui s'impose est de dire qu'un système support d'activités coopératives doit être flexible (malléable, reconfigurable) afin de tenir compte de la définition de l'organisation sociale dans laquelle est utilisée la plate-forme.

Dans le domaine de l'informatique, les questions qui vont se poser vont être aussi pluridisciplinaires : au niveau interface homme machine, quelles sont les métaphores

adaptées, comment définir l'accès à l'espace d'information (voir ci après), comment définir la perception visuelle du rôle que joue l'utilisateur, que deviennent les modèles utilisés en interface homme machine dans le contexte communication homme homme médiatisée par la machine ; au niveau système distribué et réseau : quels sont les réseaux les plus adaptés, quelles sont les fonctionnalités de communication nécessaires, quelles sont les performances à obtenir, le collectif a-t-il des besoins que n'offrent pas les systèmes à objets distribués.

Nous ne répondons pas ici aux problèmes, mais nous essayons d'illustrer comment le problème de la rétroaction de groupe est important dans le contexte synchrone, et comment il peut être transversal par rapport aux autres concepts du temps réel.

3 - Gestion des conflits ou médiation?

Le contexte du collectif amène un groupe d'utilisateurs à partager des applications et des informations. Dans une exploitation collective d'entité unique (processus ou information), il va se poser le problème d'accès concurrent à ces mêmes entités. Il faut donc que ces conflits soient gérés afin de permettre une exploitation effective du collectif, et que ceux-ci n'apparaissent pas comme un perturbateur de l'activité de groupe, ou qu'ils modifient fortement l'organisation de celui-ci. Dans un cadre présentiel, la gestion des conflits se fait sur le statut social [Anzieu 68], la prestance de l'individu et plus généralement par le canal de métacommunication ou communication non verbal [Peri 92]. La gestion du conflit pourra donc se faire par la définition de règles qui pourront exprimer une organisation sociale et hiérarchique basée sur la définition de rôle et de droits (problème de mécanisme d'analyse pour déterminer ces rôles) mais aussi par la prise en compte dans les environnements coopératifs de la métacommunication. Il faudra aussi tenir compte dans la stratégie de gestion des conflits (Algorithme de contrôle traduction de floor algorithm) [Rodden 91], de la nature même de l'activité supportée (services offerts par l'application), des contraintes et apports des systèmes informatiques, mais aussi des études faites dans d'autres domaines tels que la sociologie, la psychologie cognitive et l'ethnologie.

4 - Espace d'information

Les activités assistées par les collectifs temps réel vont souvent mettre en jeu le partage d'information (exemple : la coédition).

Cette information va être partagée par l'intermédiaire d'un système de visualisation qui va lui même prendre en compte la dimension de groupe. Par exemple, il va respecter le concept de WYSIWIS strict. Cet espace est souvent associé à un canal de

communication qui permet un échange d'information sonore entre les différents participants (voir chapitre II).

L'espace d'information va fournir des mécanismes qui vont être utilisés par les systèmes de visualisation. Ceux sont des mécanismes d'accès collectif à l'information basés sur la *structuration* même de l'information. Cette structuration peut être liée à la nature de l'information manipulée, ou à la représentation informatique de cette même information (approche objet).

La structuration de l'information va être utilisée pour définir la dimension de partage de celle-ci. Par exemple, l'éditeur coopératif de documents structurés Alliance [Decouchant 92,94] utilise les normes de structuration de document pour réaliser un éditeur coopératif.

5 - La notion de rôle et de droit

Nous avons vu dans le paragraphe sur la gestion des conflits, la possibilité de traiter ce problème en définissant des rôles et des droits attribués aux différents participants. Bien souvent, ces rôles et ces droits vont être associés à une fraction de l'information partagée, et ne permettront que des modifications à un seul participant à la fois sur une même entité d'information.

Le problème va se trouver dans l'analyse des situations afin de déterminer les rôles et les droits existants dans la tâche à gérer. Bien souvent dans les collecticiels, la définition de ces notions est statique et n'est pas le résultat d'une analyse de l'activité concernée, mais une définition à priori qui impose un modèle social.

Dans un collecticiel, il faudra s'intéresser aussi à la technique utilisée pour définir et affecter les rôles aux différents participants.

6 - Architecture logicielle d'un collecticiel

Il existe trois types d'architecture logicielle identifiés dans la littérature [Lantz 86] [Lauwers 90].

Architecture centralisée : il existe une seule instance de l'application, avec laquelle les utilisateurs interagissent. Cette architecture semble simple à mettre en oeuvre : elle repose sur la capture d'événements de bas niveau (souris) associé à un mécanisme de multiplexage/démultiplexage. Ce mécanisme va souvent être synonyme d'un modèle de gestion de la coopération (algorithme de contrôle) relativement pauvre : un seul utilisateur manipule le système à la fois. On peut aussi facilement imaginer des problèmes de performances de l'interface Homme Machine comparée à celles

obtenues en utilisant la même application en mono-utilisateur. SharedX (extension de X) [Gust 88] est un archétype qui respecte cette architecture logicielle.

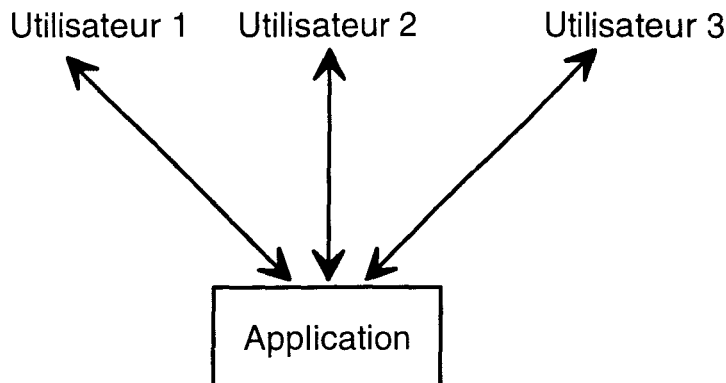


Figure 3 : Architecture centralisée

Architecture répliquée : une copie de l'application est associée à chaque utilisateur. Les applications vont communiquer afin de définir un état cohérent sur les données dupliquées dans chaque copie. La sémantique de l'information échangée peut être plus ou moins riche : par exemple, elle peut aller d'une diffusion d'événements souris (pour respecter le Wysiwis strict) jusqu'à des événements tenant compte de la sémantique de l'application (liés à la manipulation d'objet graphique). Ceci demande l'écriture d'algorithmes non triviaux (par exemple l'algorithme ORESTE [Karsenty 93]), qui vont devoir aussi prendre en charge la gestion de la concurrence [Ellis 89] dû à la modification simultanée de données partagées. La réplication de l'application permet plus facilement de gérer un parallélisme d'interaction entre les utilisateurs, par exemple permettre de modifier simultanément des objets graphiques différents. La prise en compte des nouveaux arrivants, ou retardataires n'est pas triviale comparée à l'architecture centralisée. On peut citer quelques exemples tels que GroupDraw [Greenberg 92], GroupDesign [Karsenty 94a], Groupkit [Roseman 92].

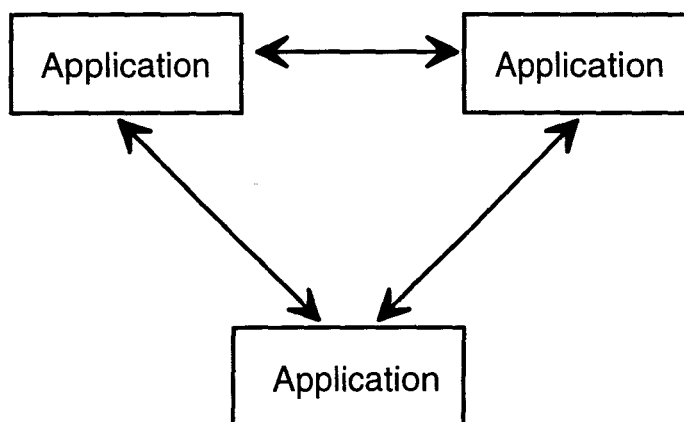


Figure 4 : Architecture répliquée

Architecture hybride ou client serveur : ce type d'architecture repose sur une distribution du Noyau fonctionnel et/ou de l'interface en sous éléments qui vont être répliqués ou non. Elles reposent sur la définition d'un modèle de distribution et de réplication des deux composants identifiés précédemment. Dewan [Dewan 92] distingue deux types d'architecture hybride : la première dans laquelle le système interactif est divisé en un objet d'application et un SGIU (Système de Gestion de l'Interface Utilisateur, traduction de UIMS User Interface Management System). L'objet d'application est centralisé alors que le SGIU est répliqué. La deuxième décompose le système interactif en un composant maître qui calcule la représentation à l'écran et un composant esclave qui gère les événements d'entrée. L'architecture hybride va permettre de réaliser un compromis entre les différentes caractéristiques de l'architecture centralisée et répliquée.

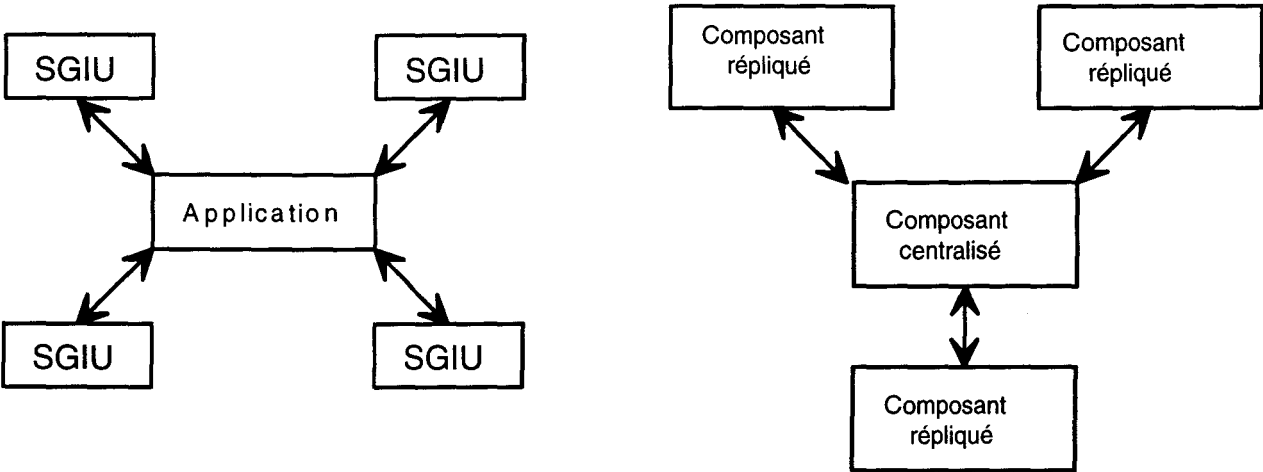


Figure 5 : Architecture hybride

Le tableau ci-dessous décrit un état comparatif des différentes architectures.

Architecture	Avantages	Inconvénients
Centralisée	- simplicité d'implémentation - développement minimum de logiciel - système de synchronisation simple (multiplexage)	- Performance de l'interface (prise en compte du multi-utilisateur) - modèle de coopération pauvre (tout ou rien)
Dupliquée	- performance de l'interface - permet un certain niveau de parallélisme d'interaction	- maintient de la cohérence des données - système de synchronisation non trivial - prise en compte des retardataires ou nouveaux arrivants
Hybride	- permet un compromis entre les différentes caractéristiques des deux autres architectures	

En conclusion, on peut dire que classer l'architecture d'un prototype dans un type d'architecture n'est parfois pas évident. La qualification d'architecture hybride se fait par négation sur les deux premières et donne donc peu d'information.

III - ANALYSE DE QUELQUES PROTOTYPES EXISTANTS

Cette partie présente quelques prototypes en essayant de décrire comment les éléments de taxonomie décrit précédemment, apparaissent ou sont traités. Ceci doit nous permettre d'aboutir à une description plus fine des éléments de taxonomie.

1 - Les éditeurs

Les éditeurs coopératifs sont l'ensemble des logiciels qui permettent l'édition collective d'information quelle soit textuelle, graphique ou sonore.

1.1 - Grove

Objectifs : Grove [Ellis 90] est un prototype d'éditeur de texte conçu pour être utilisé par un groupe de personnes travaillant simultanément sur le même texte.

Originalités :

- les fenêtres de groupe (traduction de "group window") sont une collection de fenêtres qui apparaissent sur plusieurs stations utilisateur à la fois et qui respectent le concept de Wysiwiw (voir figure 6). La notion de groupe définit les utilisateurs qui visualisent cette fenêtre. On affecte à chacun des membres des droits de lecture et d'écriture sur les items qui composent le texte. La figure 7 montre le formulaire qui permet d'affecter les droits.
- la vue (traduction de "view") est une représentation d'une portion de l'environnement partagé. Grove a trois types de vue : une vue privée qui contient les items de texte accessibles par un utilisateur particulier, une vue partagée qui contient les items de texte accessibles par un ensemble d'utilisateurs énumérés et enfin une vue publique qui contient les items accessibles par tous les utilisateurs.
- au niveau de l'interface, Grove utilise le concept de nuage pour indiquer que le texte est en cours de modification.

Caractéristiques :

L'interface de Grove donne une rétroaction immédiate à l'utilisateur qui modifie le texte. Les autres visualisent tout d'abord un nuage grisé sur la partie de texte en cours de modification. Ensuite le nouveau texte apparaît en couleur bleu vif, puis passe progressivement au noir puis à sa couleur finale.

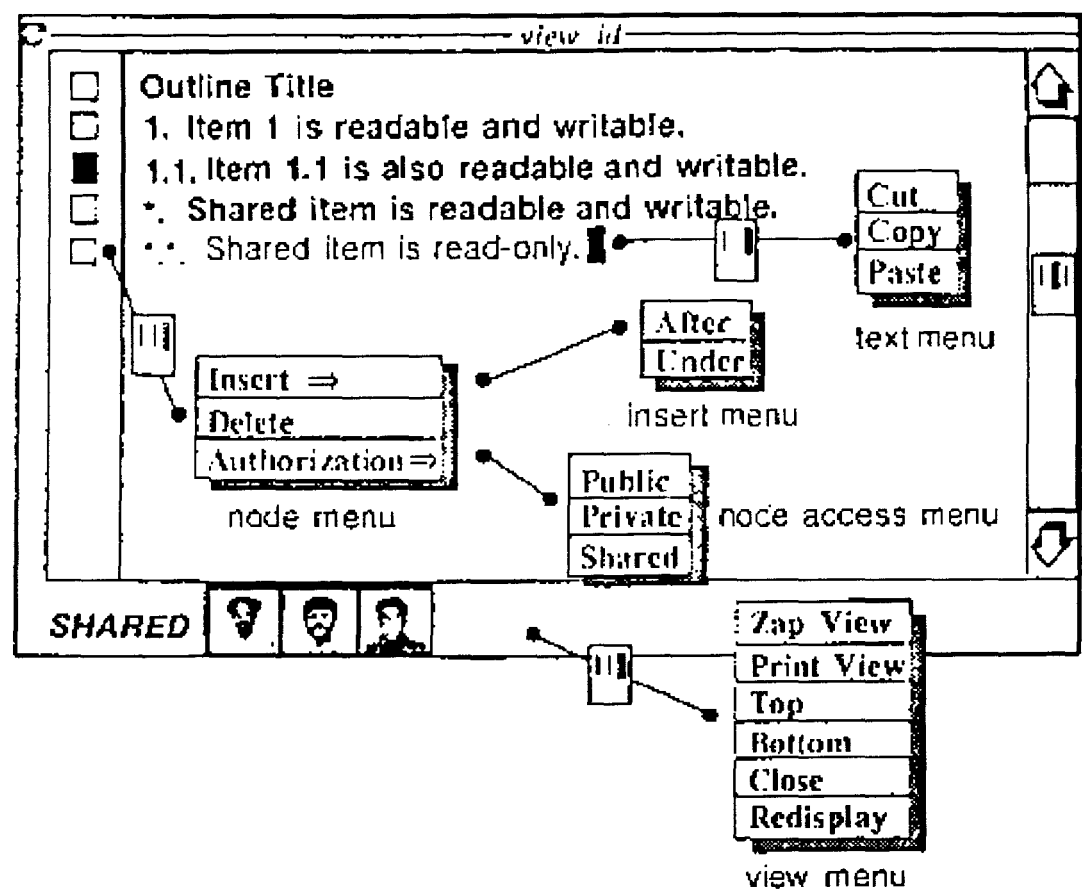


Figure 6 : Une fenêtre de groupe et ses menus associés

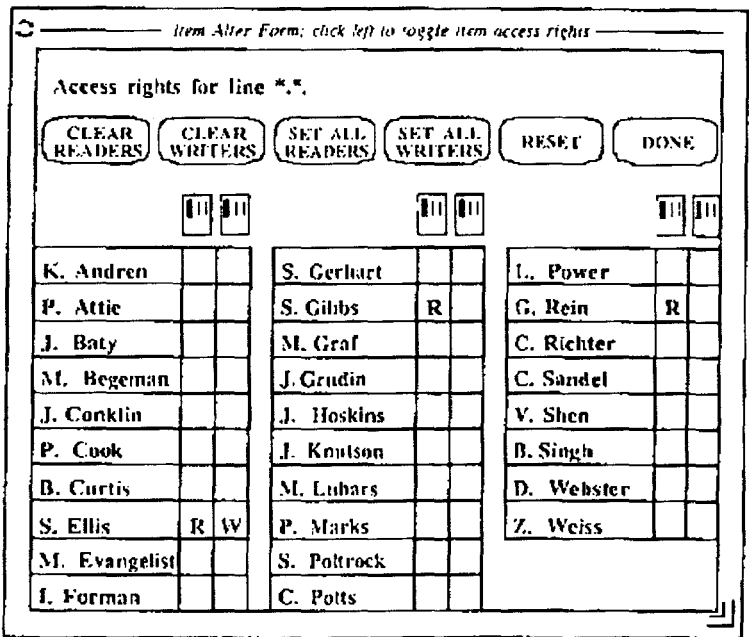


Figure 7 : La fenêtre d'affectation des droits sur un item

1.2 - Alliance

Objectifs : Alliance [Decouchant 92,94] est un atelier éditorial qui permet à un groupe d'utilisateurs, localisés sur des stations de travail, d'éditer simultanément les mêmes documents.

Originalités :

- Utilise la structuration du document pour exprimer et définir la coopération ;
- Définit un protocole de rétroaction qui permet l'exploitation collective du document.

Caractéristiques :

Les utilisateurs d'Alliance ont un statut de co-rédacteur par rapport au document. Ils peuvent agir sur les différentes parties (fragment) du document en fonction du rôle qui leur est attribué. Alliance définit quatre rôles qui correspondent à une liste d'actions possibles de la part des utilisateurs :

- le rôle de gestionnaire (symbolisé par une couronne) permet à l'utilisateur de modifier le contenu d'un fragment, d'attribuer des rôles aux autres utilisateurs et de créer ou détruire des parties de document ;
- le rôle de rédacteur (symbolisé par un crayon) permet à son détenteur de modifier le contenu d'un fragment ;
- le rôle de lecteur (symbolisé par un oeil) permet à son détenteur de consulter la partie considérée ;
- le rôle nul permet à son détenteur d'avoir connaissance de l'existence d'un fragment sans pouvoir en consulter le contenu.

La définition de ces rôles permet le travail simultanée de plusieurs rédacteurs sur des fragments distincts. Un seul rédacteur est actif à la fois sur un fragment spécifique. L'affectation des rôles est dynamique : un utilisateur peut changer son rôle (passer de rédacteur à lecteur) sans pouvoir pour autant obtenir plus de droits que lui en a attribué le gestionnaire du document.

On affecte au fragment un état parmi les suivant : fragment caché, fragment caché modifié et fragment caché et verrouillé. L'association des rôles des utilisateurs et de l'état d'un fragment permet de définir les manipulations d'un utilisateur a un instant donné. Ceci est symbolisé au niveau de l'interface par l'insertion d'icône symbolisant l'état du fragment.

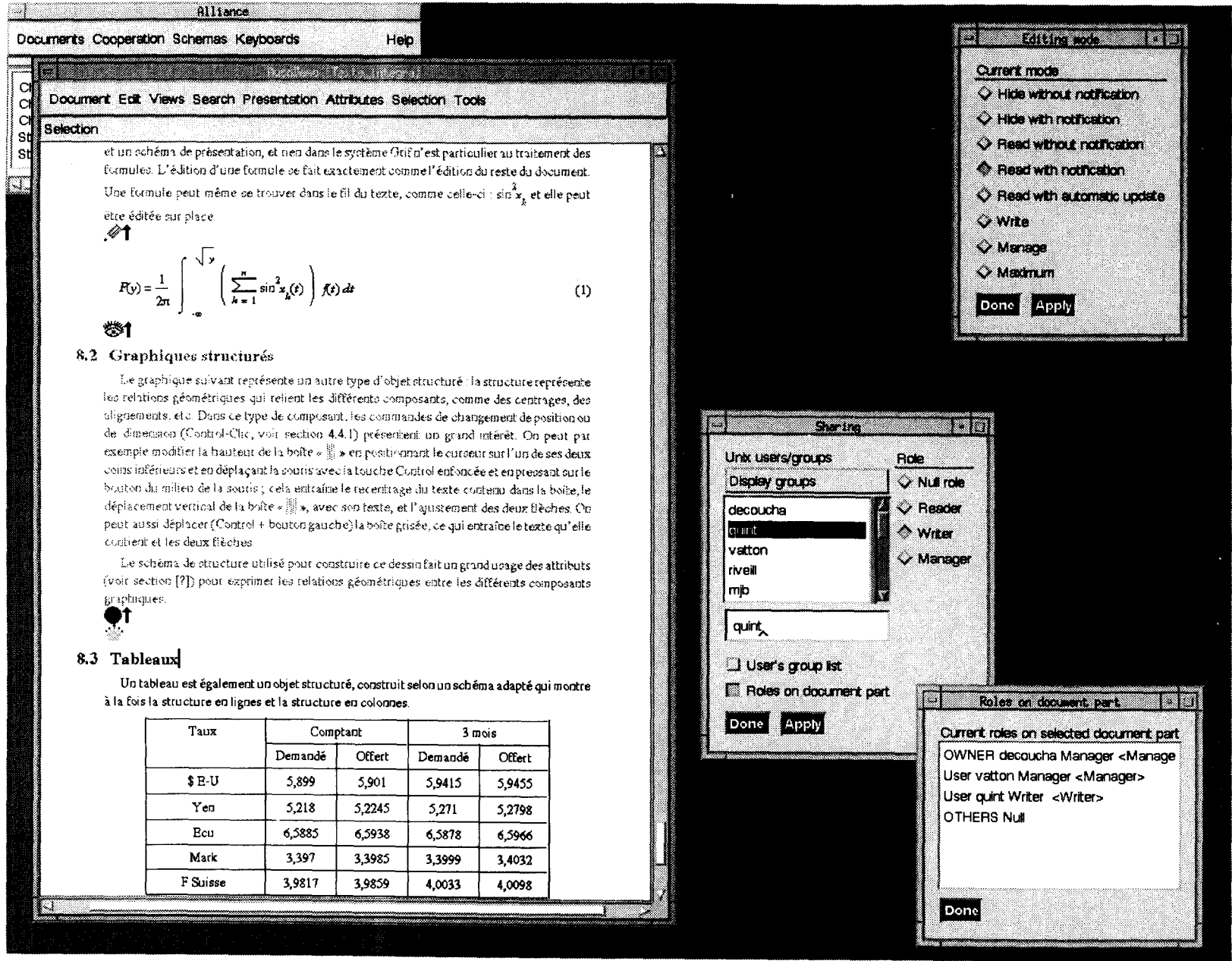


Figure 8 : L'interface d'Alliance

1.3 - GroupDraw

Objectifs : GroupDraw [Greenberg 92] est un éditeur de dessin orienté objets. Il est utilisé comme plate-forme d'expérimentation pour étudier différentes interfaces et architectures.

Originalités :

- Curseurs multiples et actifs simultanément permettant de traiter des interactions générées par plusieurs utilisateurs. Le curseur est adaptée à la sémantique de l'action réalisée et à l'utilisateur qui l'utilise (nommage).

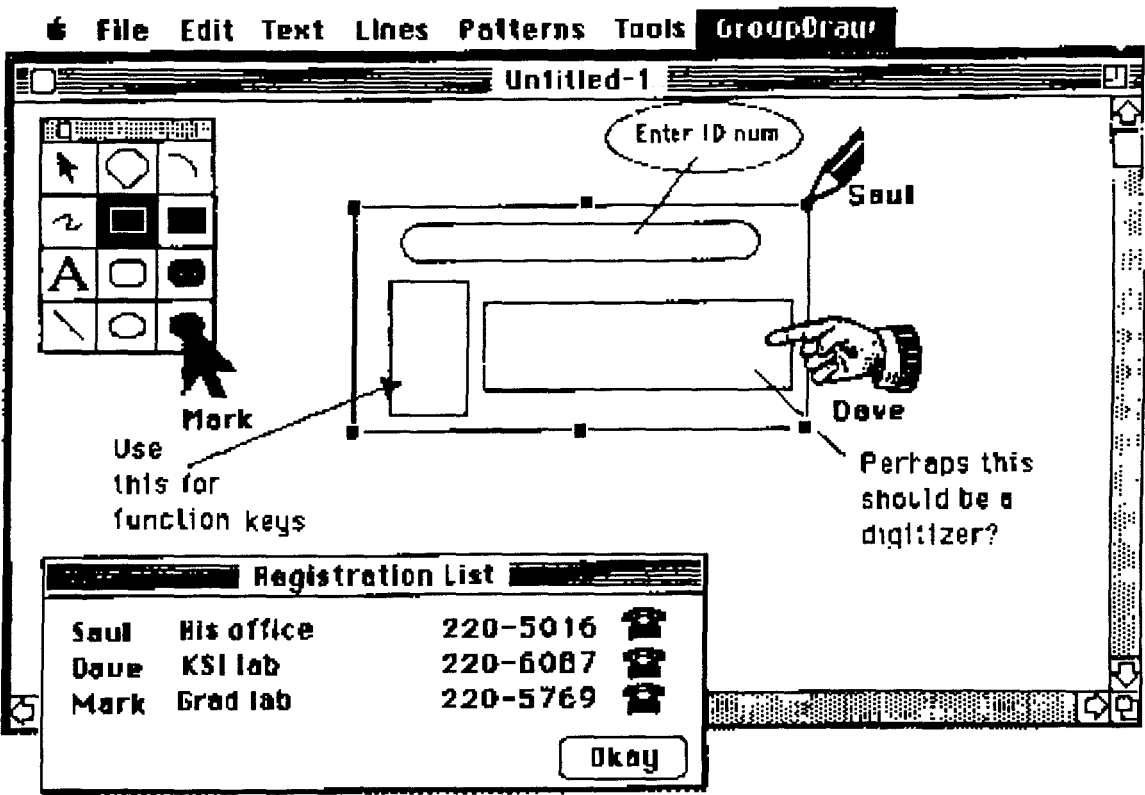


Figure 9 : Interface de GroupDraw

Caractéristiques :

Il permet de créer, de dimensionner, de déplacer des objets graphiques, à plusieurs utilisateurs simultanément : il gère plusieurs curseur simultanément et adapte leur aspect visuel à l'action en cours et à l'utilisateur (nommage). Il implémente le concept de Wysiwis, en relâchant la contrainte spatiale (fenêtre "scrollable") car la surface de dessin est trop petite. Une deuxième fenêtre donne des informations sur les participants (Registration list). Les conflits peuvent apparaître lorsque les utilisateurs veulent manipuler le même objet. La propriété de l'objet graphique est donnée à un

seul utilisateur. Dans le cas où un autre utilisateur a commencé à manipuler le même objet, il voit l'objet revenir à sa place originale. GroupDraw permet le dessin privé (je dessine, les autres ne voient pas), tout d'abord en proposant un espace scrollable, et en offrant la possibilité d'associer un statut à l'objet. Celui-ci décrit les droits des autres utilisateurs sur cet objet (visible, manipulable et privé). GroupDraw utilise une architecture répliquée. Un protocole de communication est défini entre les différents processus répliqués afin de réaliser la gestion de conflit présentée ci-dessus.

1.4 - GroupDesign

Objectifs : GroupDesign [Karsenty 94a] est une application de dessin structuré multi-utilisateur similaire à MacDraw. Elle est conçue dans le but d'être utilisée comme outil de dessin pour la réalisation de projet impliquant un grand nombre de personnes sur une longue période. Elle permet donc des interactions synchrones et asynchrones. Nous nous intéresserons ici seulement aux interactions synchrones.

Originalités :

- écho graphique et audio [Karsenty 94c] : c'est la définition de la représentation des actions d'un utilisateur sur les interfaces des autres utilisateurs.
- algorithme d'accès concurrent à des objets graphiques partagés permettant aux utilisateurs de manipuler simultanément les mêmes objets. Cet algorithme détecte les conflits suite aux manipulations des utilisateurs et propose une stratégie de résolution.

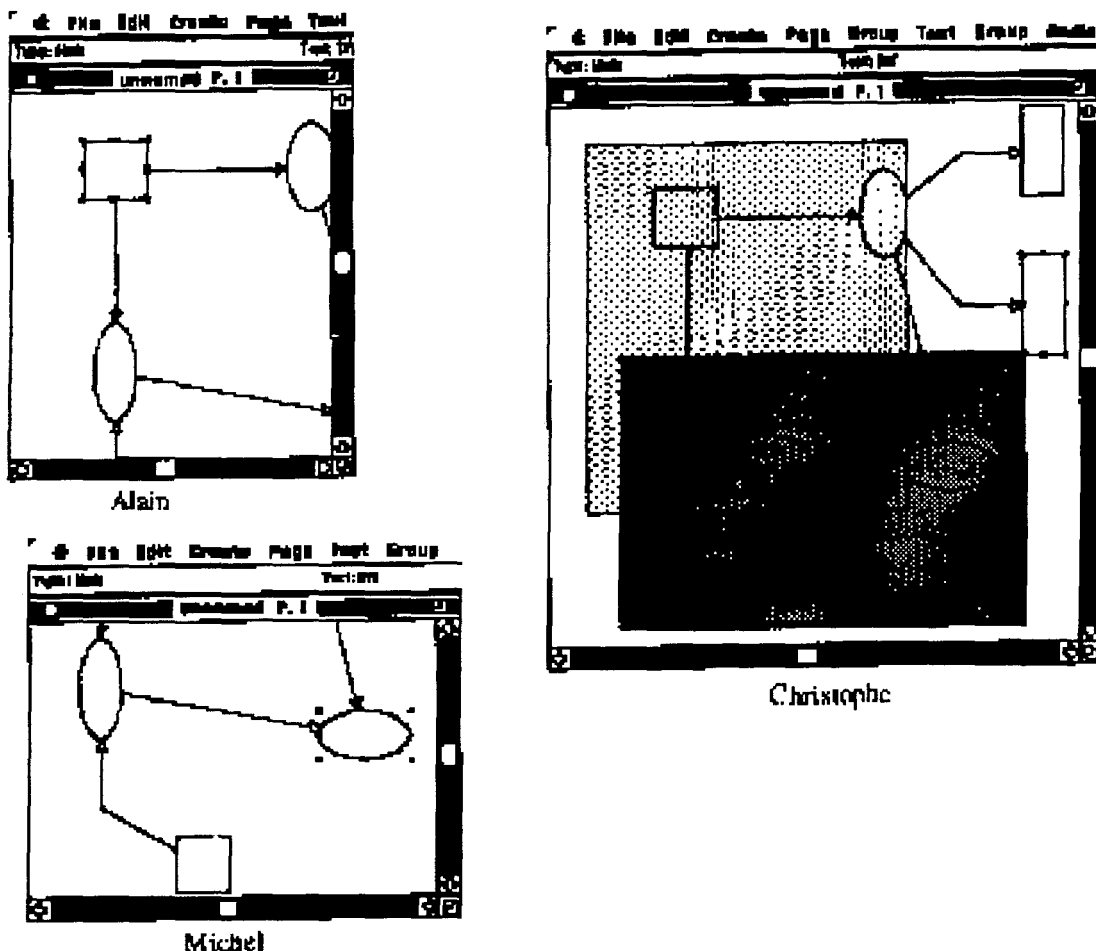


Figure 10 : Vision de l'interface de trois utilisateurs de GroupDesign

2 - Les architectures d'accueil et les boîtes à outils

2.1 - GroupKit

Objectifs : Groupkit [Roseman 92,94a,94b] est une boîte à outil pour la conception de collecticiel temps réel.

Originalités :

- une architecture de "*run-time*" (fonctionnelle) orientée objet qui gère des processus distribués et la communication entre ceux-ci. La figure 11 donne une représentation de cette architecture, qui met en oeuvre différents processus (registrar) qui permettent de gérer les conférences ;
- la notion de *transparence* "transparent overlay" permet de prendre en compte certaines fonctionnalités nécessaires dans les collecticiels, telles que la création d'un télépointeur pour supporter la gestuelle ;

- la notion d'*ouverture* "open protocole" permet au concepteur de créer différents types d'interface, et de techniques d'interaction pour s'adapter au contexte dans lequel est utilisé le collecticiel : par exemple l'algorithme de contrôle pour le partage des ressources, ou bien la définition de nouveaux protocoles de début de conférence.

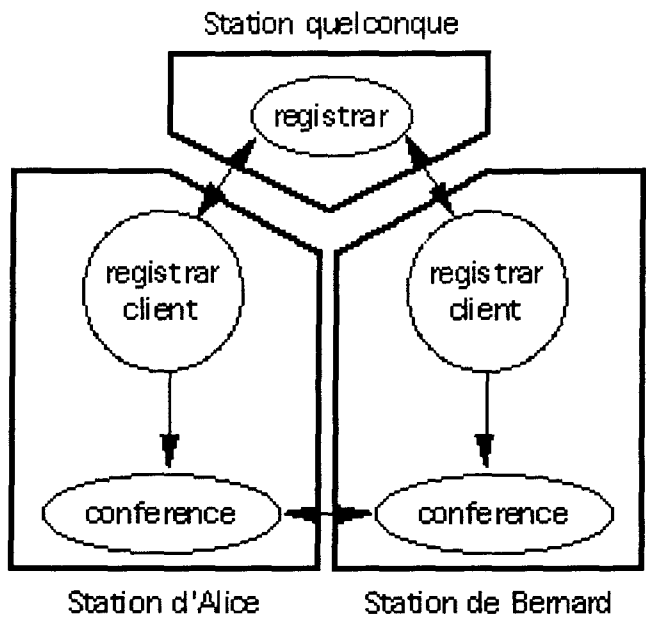


Figure 11 : Architecture de GroupKit

Caractéristiques :

Une application Groupkit est un ensemble de processus répliqués sur plusieurs machines. La figure 11 illustre cette distribution en prenant l'exemple de deux personnes communiquant à travers une conférence. Une conférence correspond à une application particulière comme un outil de dessin ou un éditeur de texte collectif. Groupkit fournit un mécanisme de gestion de conférence qui met en jeux un processus central nommé Registrar qui tient à jour une liste des conférences actives et de leurs participants. Il ne gère pas les politiques qui définissent comment est créée une conférence, ou comment un utilisateur entre dans une conférence.

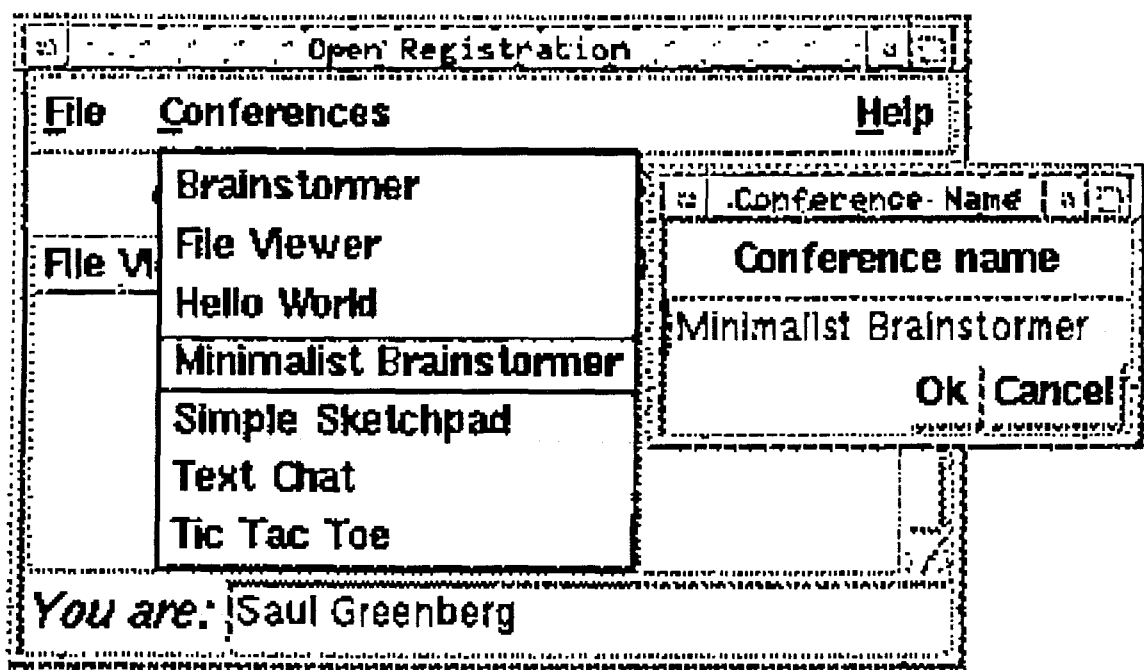


Figure 12 : Interface pour la gestion des conférences

Il existe un Registrar Client par utilisateur qui permet la création, la suppression, la sortie et l'entrée en conférence. GroupKit fournit trois Registrar Client qui définissent trois façons de gérer les conférences.

Les conférences sont des collecticiels développés à l'aide de la boîte à outil GroupKit. Elle propose un mécanisme d'envoi de messages qui permet la diffusion d'un message à tous les participants ou à un participant particulier. Le développement d'une application repose sur l'utilisation de la programmation événementielle : les messages reçus sont traités comme des événements qui déclenchent des traitements adaptés. Le dernier composant de la boîte à outil est une série de Widgets ou d'objets d'interface qui prennent en compte la dimension multi-utilisateur. On y trouve des barres de menu qui peuvent s'afficher simultanément sur toutes les stations, des boîtes de dialogue avec mise à jour automatique (lorsqu'on supprime un item appartenant à un exemplaire, tous les exemplaires se mettent à jour, des barres de défilement multi-utilisateur, etc.

2.2 - Rendezvous et le paradigme ALV

Objectifs :

Le projet Rendezvous [Patterson 90] [Hill 92] vise la construction d'un système interactif qui est utilisé simultanément par plusieurs utilisateurs possédant chacun une

station de travail. Rendezvous se situe dans une approche conception d'UIMS, basée sur la séparation entre l'interface utilisateur et l'application.

L'objectif global est de voir comment la communication homme homme peut être prise en charge, voir accrue, par des outils informatiques. Hill et al. s'intéressent à la création d'artefacts¹ qui facilitent la conversation. Ils peuvent se situer au niveau des objets de la conversation (qui sont les objets sur lesquels on parle), ou sur la conversation (actes de langage) elle-même (assister la conversation).

Originalités :

- l'utilisation d'un mécanisme de propagation de contraintes pour prendre en compte la dimension multi-utilisateur ;
- le paradigme ALV (Abstraction-Link-View). Les Liens (Link) sont des objets qui facilitent la communication entre l'Abstraction (application) et les objets Vues (interfaces utilisateurs).

Caractéristiques :

L'architecture de Rendezvous peut se décomposer en deux architectures :

- "Run-Time architecture" : cette architecture décrit l'organisation des ressources logicielles [Patterson 90 p.321]. Un terminal virtuel est dédié à chaque utilisateur, qui prend en charge les entrées sorties associées à l'utilisateur (niveau d'un terminal X). Ces terminaux virtuels communiquent avec un processus central appelé "Application process". Ce processus est lui même décomposé en processus légers qui vont se décomposer en deux types : le processus d'interaction (interaction process) et le "underlying process". A chaque utilisateur est associé un processus d'interaction, qui contient les objets d'interaction de l'utilisateur. Ceux-ci sont les objets qui sont responsables de l'affichage et de la gestion des interactions utilisateurs. Il existe un et un seul "underlying process" qui contient les objets de l'application, et prend en charge la gestion des sessions et de la coordination entre les utilisateurs.
- "Start-Up architecture" : ([Patterson 90 p.324]) par l'intermédiaire du terminal virtuel, les utilisateurs accèdent à un programme appelé RAP (Rendezvous Access Point) qui gère les sessions : commencer une session, entrer en session. Le RAP collabore avec le Rendezvous Server (RESERV) qui joue le rôle de serveur de nom pour la gestion des différentes sessions.

¹ Phénomène d'origine humain, artificiel dans l'étude des faits naturels.

La figure 13 montre les différents composants de ALV ainsi que leur distribution.

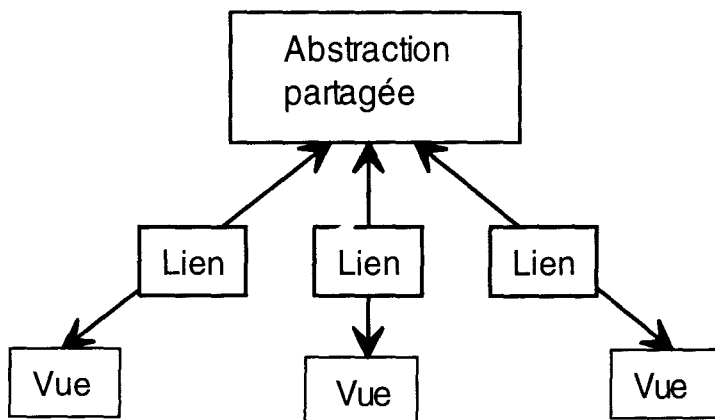


Figure 13 : L'architecture d'ALV

Le composant Vue ("View") présente l'information à l'utilisateur, permet à l'utilisateur de modifier l'information affichée. Il obtient ses informations à partir du composant Abstraction (objets de l'application). La présentation de l'information peut être différente entre les utilisateurs : on voit la même information sous une représentation visuelle différente. Chaque Vue possède l'information pour permettre un réaffichage plus rapide, mais cette information n'est pas considérée comme répliquée (copie des objets d'abstraction) mais comme redondante car elle est adaptée aux caractéristiques du composant Vue.

Le composant Lien ("Link") gère la communication entre l'Abstraction et la Vue qui lui est associée. Son rôle est d'adapter les données de l'Abstraction à celles nécessaires à la Vue, de maintenir la cohérence entre les données de l'Abstraction et les données correspondantes dans la Vue, et il assure la coordination des accès à l'Abstraction. Ce composant Lien est composé et implémenté comme un réseau de contraintes.

La boîte à outil contient une hiérarchie de classes d'objet Vue, Abstraction et Lien.

Déjà plusieurs applications utilisent cette boîte à outil : un prototype de conception d'interface basé sur Pictive (Telepictive [Miller 92]) et un ensemble d'outils de dessin [Brinck 92].

2.3 - Suite

Objectifs :

Suite [Dewan 90,91] propose une architecture flexible de haut niveau pour permettre la construction d'interface multi-utilisateur.

Originalités :

- offre des primitives de programmation des interfaces multi-utilisateur qui sont une extension des primitives mono-utilisateur ;
- niveau de collaboration incrémental.

Caractéristiques :

La démarche de Dewan est de vouloir étendre le modèle de Seeheim par ajout de composant pour qu'il prenne en charge les interactions multi-utilisateur. L'architecture de suite est la suivante :

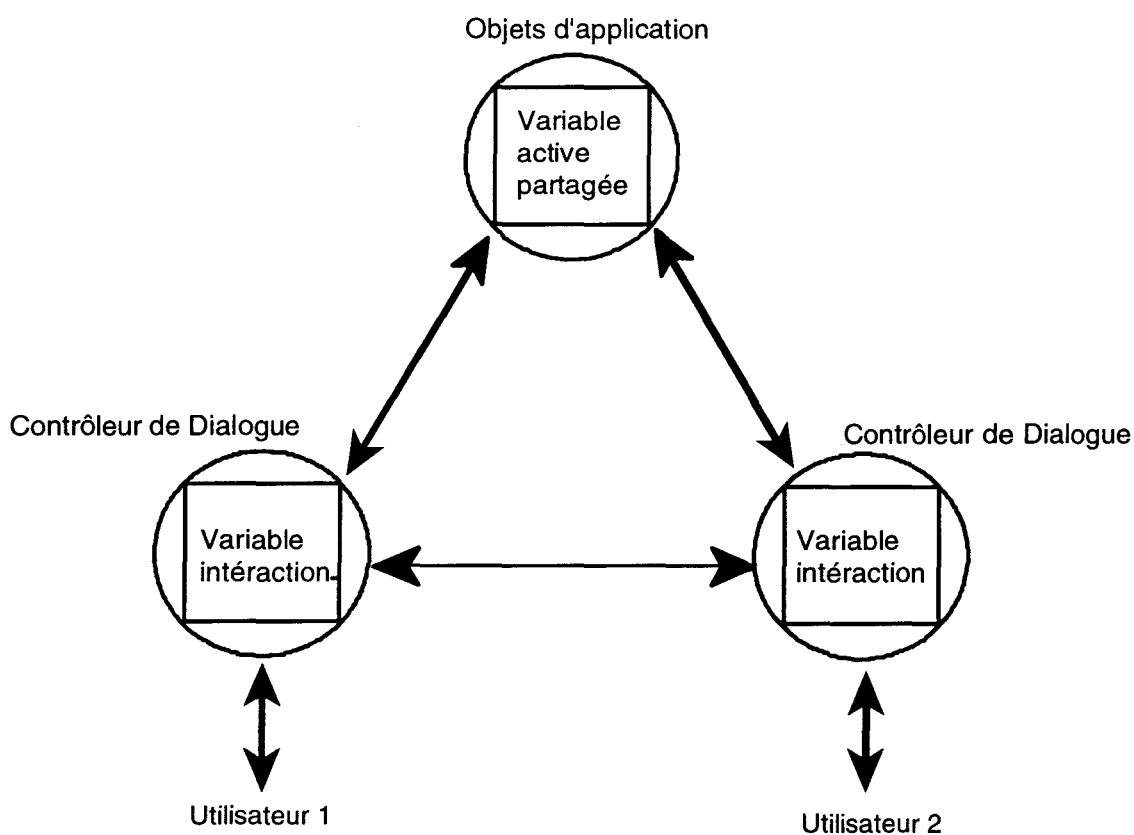


Figure 14 : Architecture de Suite Multi-utilisateur

Cette démarche permet à partir d'une application développée sur le modèle de Suite mono-utilisateur de construire facilement une version multi-utilisateur.

3 - Extension des systèmes de fenêtrage

SharedX (ou XShared)

Objectifs :

SharedX [Gust 88] veut offrir la possibilité de partager une application XWindow, en permettant à plusieurs utilisateurs de visualiser et d'interagir simultanément sur la même application X (respect du Wysiwiw strict). C'est à dire que l'on cherche à rendre coopérative une application sans tenir compte de sa sémantique, mais en réalisant une extension du système de fenêtrage.

Originalités :

- fournit un mécanisme générique de collectivisation d'application : le Xmultiplexeur ;
- fonctionne sur tout type de terminaux XWindow.

Caractéristiques :

SharedX repose sur la modification du modèle client/serveur de XWindow. Dans la figure 15, on peut observer le modèle client/serveur X standard, composé d'un client X qui est l'application, et du serveur X qui peut être considéré comme un serveur d'affichage. Ces deux entités communiquent à l'aide d'un protocole appelé protocole X.

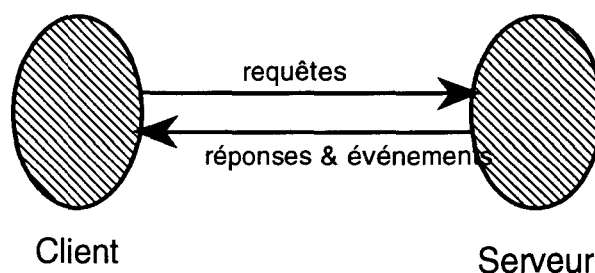


Figure 15 : Modèle client/serveur X standard

Dans l'architecture client/serveur de SharedX, un pseudo serveur appelé "shXBridge" vient s'intercaler entre le client X et les différents serveurs X (un par utilisateur au minimum). Il joue le rôle de démultiplexeur pour transmettre les événements d'affichage aux différents serveurs, et de filtrage pour permettre la prise en compte d'un algorithme de contrôle : il ne transmet au client que les événements émis par le serveur qui possède le contrôle.

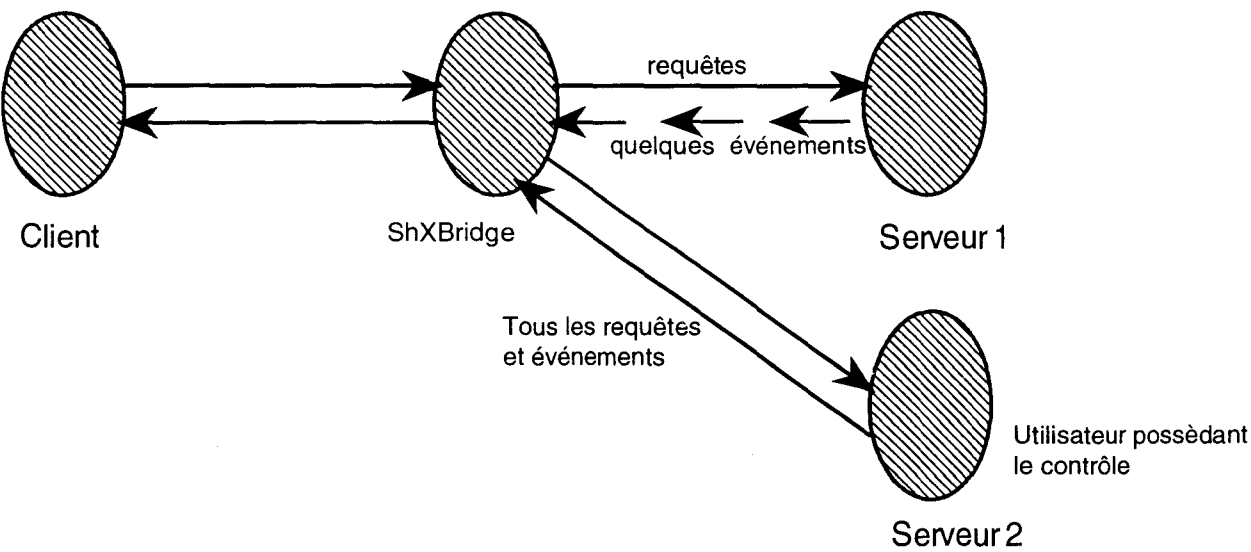


Figure 16 : Modèle client/serveur étendu

Vous trouverez dans [Baldeschwieler 93] et [Fitzgerald 91] d'autre moyens de distribution de l'environnement X.

La figure 17 montre l'interface correspondant au un panneau de contrôle qui permet de visualiser les utilisateurs, celui qui possède le contrôle (Ben), et qui permet de demander le contrôle.

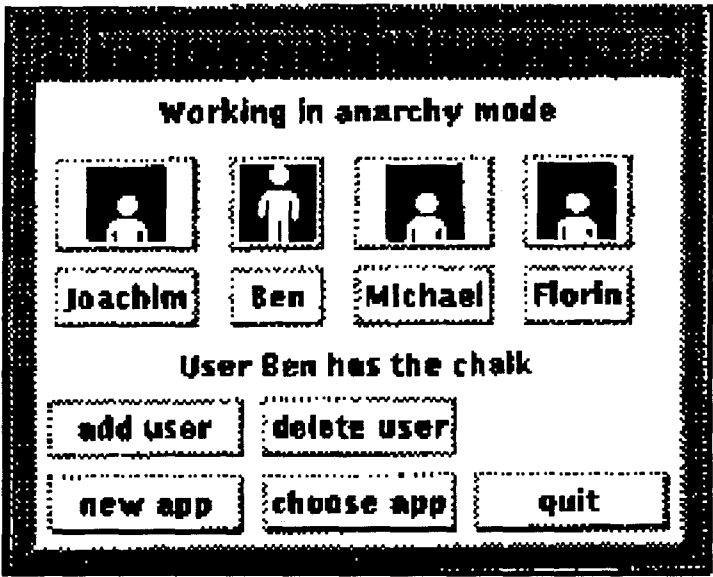


Figure 17 : L'interface de SharedX

Le nommage des utilisateurs se fait en désignant le serveur X (sous la forme HostName:0.0), ou par un nom si on définit des alias système (c'est le cas de la figure 17). Il existe deux modes de fonctionnement pour le partage de l'application : un mode

anarchique où chacun prend le contrôle quand il le veut et un mode courtois où l'utilisateur qui possède le contrôle peut le donner à tout participant (si un utilisateur demande le contrôle, il est alors représenté avec le bras levé). Le choix du mode se fait au lancement de SharedX en spécifiant un paramètre.

IV - ETAT RECAPITULATIF DE L'ANALYSE

L'objectif de cette dernière partie est tout d'abord de donner et synthétiser les caractéristiques de nos différents éléments de taxonomie afin de pouvoir construire un tableau comparatif récapitulatif des différents systèmes étudiés.

1 - Le Wysiwis et sa relaxation

Il est intéressant d'identifier comment les applications de visualisation d'information respectent ou pas le Wysiwis. Nous les classons en terme de Wysiwis strict et ensuite en fonction des différentes dimensions de relaxations décrites précédemment. Ce qui est encore plus intéressant c'est de comprendre pourquoi on relâche les dimensions du Wysiwis et comment ceci est réalisé.

2 - L'espace d'information

En ce qui concerne l'espace d'information, on s'attache à préciser d'abord la structuration de l'information utilisée par le collecticiel, et ensuite à identifier en quoi cette structuration aide à résoudre les problèmes posés par le collecticiel.

3 - Gestion des conflits ou médiation

L'objectif de cet élément de taxonomie est d'appréhender tout d'abord comment les collecticiels définissent la notion de conflit, quelles solutions ils utilisent pour le résoudre, et enfin comment ceci se traduit en terme d'interface utilisateur. Nous définissons des critères qui vont nous permettre de typer et classer les éléments de réponses aux trois questions précédentes.

Les conflits qui sont traités par les collecticiels sont souvent liés au partage de ressources. Ces ressources partagées sont soit la conséquence de contraintes techniques ou de contraintes humaines.

Un exemple de contrainte technique est la réalisation d'un télépointeur manipulable simultanément par plusieurs utilisateurs. Un exemple de contrainte humaine peut être la signification de manipuler ensemble un même objet graphique, ou la surcharge cognitive liée à la visualisation de plusieurs curseurs sur un espace Wysiwis.

Nous identifions trois techniques pour la gestion des conflits :

- Orienté fonction : on gère l'accès aux fonctions offertes par le système, de telle façon que deux fonctions conflictuelles ne puissent être utilisées simultanément. C'est à dire que le collecticiel agit sur les actes instrumentaux² possibles ;
- Orienté donnée : on va utiliser la structuration de l'espace d'information afin de définir des verrous sur une partie de l'information (mécanisme d'exclusion mutuelle) ;
- Orienté médiation humaine : soit le système admet qu'un autre canal de communication (sonore) va permettre de prendre en charge le conflit, ou le système adopte une politique optimiste basée sur le fait que le groupe ne va pas générer de conflit en utilisant le système.

Dans les deux premiers cas, c'est le système qui gère la politique de gestion de conflit. Nous définissons deux caractéristiques de l'interface qui définissent la façon dont l'utilisateur va percevoir cette politique. Ceci se traduit en terme de requête associée aux manipulations de l'utilisateur :

- requête implicite : générée par le système à la suite d'une action utilisateur. Par exemple, l'utilisateur d'un éditeur de dessin collectif charge un nouveau fichier par la commande "ouvrir" du menu "Fichier". Sans que l'utilisateur fasse une démarche supplémentaire, le système va vérifier s'il peut réaliser l'action. Il est nécessaire d'avertir l'utilisateur en cas d'échec ;
- requête explicite : l'utilisateur demande l'autorisation avant de réaliser une action. Dans ce cas, il est nécessaire de compléter l'interface utilisateur afin d'offrir l'accès à ces nouvelles fonctionnalités.

La politique de gestion ou "floor algorithm" peut être :

- statique : définit une fois pour toute au cours d'une session³ ;
- dynamique : possibilité de changer de politique au cours d'une session.

La prise en compte au niveau interface utilisateur peut être :

- a priori : seules les actions autorisées sont accessibles pour l'utilisateur. Il sait que sa requête va aboutir ;

² Les actes instrumentaux sont des actes que l'on réalise à l'aide d'un outil pour modifier ou créer de l'information.

³ Nous appelons session le temps d'utilisation d'un collecticiel.

- a posteriori : l'utilisateur génère des requêtes sans savoir si elles vont aboutir.

4 - La rétroaction

Notre état récapitulatif va essayer de répondre aux questions suivantes :

- Sous quelle forme apparaît elle ? Le comment ? Quelle perception à l'utilisateur de la rétroaction ?
- Quelle sont les causes de la définition de la rétroaction ? Le pourquoi ?

5 - Parallélisme d'interaction

Cet élément sert pour répondre à la question "Peut on travailler simultanément".

La réponse peut être un non strict. Dans le cas contraire, nous essayons de répondre aux questions suivantes :

- sur quels éléments (même objet, partie de texte) ? afin de définir la granularité ;
- avec quelles contraintes ? afin d'identifier les cas particuliers.

6 - Intégration du multimédia

Il s'agit de faire un bilan sur l'utilisation des nouvelles technologies multimédia dans les collecticiels. Ces aspects sont généralement présentés comme causes d'apparition, ou de fort développement du champ du CSCW.

7 - L'interface

Pour chaque prototype, nous avons essayé de donner une vision de l'interface. Notre état comparatif essaie de mettre en évidence l'existence de métaphores, et plus généralement de décrire les caractéristiques originales de l'interface en terme de visualisation et d'interaction.

8 - Le support du groupe

Le collecticiel temps réel nécessite l'existence et la présence d'un groupe. Cet élément nous permet d'apprécier comment le groupe apparaît au niveau de l'interface du collecticiel (information fournie sur le groupe telle que la liste des membres et une description de chacun), mais aussi comment il est géré (création du groupe, mise à jour du groupe).

9 - Etat récapitulatif

L'information contenue dans les tableaux suivant est la conséquence de notre interprétation des publications des auteurs concernés et pour certains prototypes à l'utilisation de ceux-ci (SharedX, Groupkit, Alliance). La présence de points d'interrogation peut signifier le fait que nous n'ayons pas de réponse, ou que nous émettons un doute sur notre interprétation

LES EDITEURS		
	GroupDesign	GroupDraw
WYSIWIS et relaxation	Strict + relaxation - spatiale - temporelle	Strict + relaxation - spatiale (taille de l'espace de travail, travail privé)
Espace d'information	Structuré en objet graphique	Structuré en objet graphique
Gestion de conflit ou médiation	- orienté données - implicite - a posteriori	- orienté données - implicite - explicite
Rétroaction	- écho graphique et audio associé aux manipulations des autres utilisateurs	- Wysiwis strict - associé à la médiation (rejet d'une manipulation, statut d'un objet)
Parallélisme dans l'interaction	- sur des objets différents - sur le même objet si actions non conflictuelles	- oui mais sur des objets différents si leur statut le permet
Intégration du multimédia	- écho audio	- néant
Interface	- écho graphique - utilisation de la couleur dans le mode identification - icône + zone colorée pour identifier les verrouillages d'objets	- multi-curseur
Support du groupe	- identification des membres de groupe soit par leur nom ou une couleur associée - administration du groupe ?	- fenêtre fournissant l'information sur chaque membre du groupe - administration du groupe ?
Domaine	Editeur de dessin collectif orienté objet	Editeur de dessin collectif orienté objet
Observation	- extension de MacDraw	

LES EDETEURS		
	Alliance	Grove
WYSIWIS et relaxation	Relaxation - spatiale (travail simultané, espace de travail) - congruence (définition de point de vue) - population - temporelle (travail privé)	Strict + Relaxation - spatiale (taille de l'espace de travail) - population (adapter un point de vue à un groupe particulier)
Espace d'information	Basé sur la structuration du document + définition de droit	Fenêtre de visualisation associée à un groupe
Gestion de conflit ou médiation	- orienté données - statique avec changement de droit dynamique - a priori	- orienté données - statique
Rétroaction	- associée à la médiation	- associée à la médiation (nuage qui définit l'étendu de la possession de l'information)
Parallélisme dans l'interaction	- oui mais sur des morceaux de texte différents	- oui mais sur des morceaux de textes différents (structuration réalisée par une sélection souris)
Intégration du multimédia	- néant	- néant
Interface	- utilisation des icônes pour la rétroaction de groupe	- concept de nuage et utilisation de la couleur pour la rétroaction
Support du groupe	- fenêtre de gestion de droits contenant la liste des membres du groupe - administration du groupe ?	- fenêtre de gestion de droits contenant la liste des membres du groupe - administration du groupe ?
Domaine	Edition collective de texte	Edition collective de texte
Observation	- Nouvelle version de Griffon (Basée sur un système à objet distribué), Alliance est développé sur UNIX - Utilisé la structuration du document	

LES ARCHITECTURES D'ACCUEIL			
	Groupkit	Suite	Rendez-vous
WYSIWIS et sa relaxation	Strict + relaxation ?	Strict + relaxation - congruence (adaptation visuelle) - spatiale	Strict + relaxation - congruence (adaptation visuelle) - spatiale
Espace d'information	?	- lié à la définition des variable active partagée	- lié à la structuration de l'abstraction
Gestion de conflit et médiation	- à la discrétion du concepteur d'application	- associé à la définition de Variable active partagé - plusieurs politiques par défaut (Dynamique ?)	- orienté données - orienté fonctions - implicite - explicite - statique
Rétroaction	- mécanisme support de la gestuelle	- sémantique (associé à la modification des variables actives partagées)	- sémantique (modification des objets de l'application) - associée à la médiation
Parallélisme dans l'interaction	- possible ?	- possible ?	- oui mais sur des objets différents
Intégration du multimédia	- non mais envisagée	- néant	-néant
Interface	- multi-cursseur nominatif - widgets Groupkit	?	?
Support du groupe	-gestion de conférence et non une gestion d'utilisateurs	?	?
Domaine	Orienté conception d'application	Orienté conception d'application	Orienté conception d'application
Observation	- notion de transparence - mécanisme d'envoi de messages (diffusion) - mécanisme de gestion des conférence paramétrable (open protocole)	- primitives de programmation d'interface multi-utilisateur - niveau de collaboration incrémental	- Paradigme ALV - Mécanisme de contrainte flexible et générique - Développement rapide (toolkit de développement, classe abstraite)

EXTENSION DES SYSTEMES DE FENETRAGE	
	SharedX
WYSIWIS et sa relaxation	- wysiwis strict
Espace d'information	?
Gestion de conflit et médiation	- deux modes : anarchique et courtoi - a posteriori pour l'application partagé
Rétroaction	- wysiwis strict
Parallélisme dans l'interaction	- non
Intégration du multimédia	- non
Interface	- ?
Support du groupe	- identification des serveurs X (alias possible) - ajout et suppression d'un utilisateur pendant une session
Domaine	Système de fenêtrage
Observation	- bibliothèque d'applications X - apprentissage simple

V - Bilan et perspectives

Le premier bilan est que plusieurs collecticiels existent de façon séparée pour répondre à une tâche bien précise : édition collective de texte ou de dessin, messagerie, etc. D'un autre côté les architectures d'accueil, ou les infrastructures systèmes pour le CSCW sont associées à l'utilisation de plate-formes matérielles spécifiques, et ne proposent pas d'applications vraiment significatives mais quelques outils pour les réaliser, et n'intègrent pas ou peu l'environnement déjà existant des utilisateurs..

Nous proposons de prendre comme perspectives la création d'un environnement CSCW.

Le constat est que les collecticiels s'ignorent l'un et l'autre, qu'ils sont indépendants l'un de l'autre, qu'ils implémentent leur propre modèle de coopération, qu'ils proposent des solutions souvent techniques pour assister une activité coopérative particulière. La figure 18 adaptée de [Navarro 93 page 290] résume la situation actuelle.

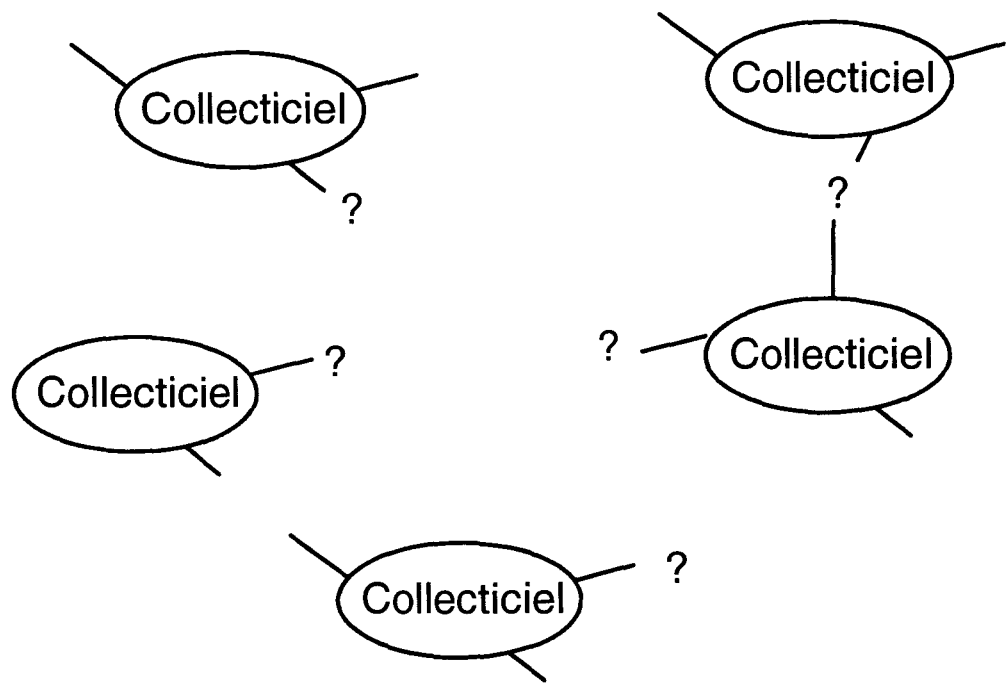


Figure 18 : Des collecticiels indépendants

L'objectif d'un environnement CSCW est de rassembler tous ces collecticiels afin d'offrir aux utilisateurs un environnement de travail riche et cohérent. Les logiciels mono-utilisateur doivent pouvoir trouver leur place dans cet environnement, afin d'intégrer plus facilement les phases de travail individuelles et collectives.

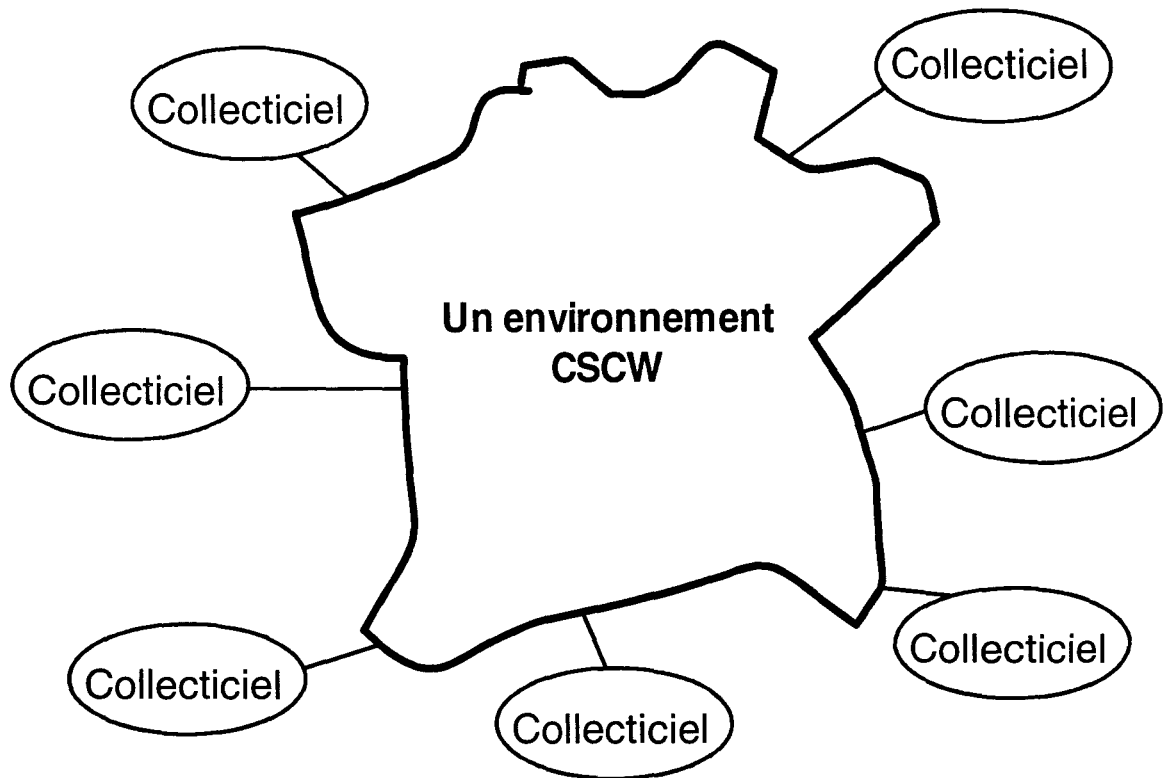


Figure 19 : Un environnement CSCW

L'environnement peut être vu comme un intégrateur. Il doit proposer aux utilisateurs :

- un cadre harmonique en terme de métaphore visuelle (voir les travaux sur la métaphore de "room" [Derycke 93b]) ;
- permettre une intégration en terme d'espace d'information : c'est à dire pouvoir manipuler la même information d'un collecticiel à l'autre, ou provenant d'un logiciel mono-utilisateur ;
- prendre en charge différents modes de communication (visuelle, sonore), et permettre une articulation avec l'espace d'information ;
- permettre aux utilisateurs de générer différentes activités avec différentes personnes : pour ceci l'environnement doit aider l'utilisateur à planifier ses activités dans le temps et à les coordonner avec les activités des autres ;
- prendre en charge le guidage, c'est à dire aider l'utilisateur à "se mouvoir" dans cet environnement.

Quelles sont les conclusions pour la conception des systèmes interactifs de groupes ?

Il nous paraît d'abord intéressant de faire un petit d'historique sur la relation Homme Système interactif. Les premières générations de systèmes interactifs ont répondu au type de relation 1 à 1, c'est à dire le dialogue entre un Homme et une application interactive. Ensuite, les systèmes ont répondu à un type de relation 1 à n, c'est à dire 1 Homme plusieurs applications, avec l'arrivée des systèmes à multi-fenêtrage permettant à un individu d'interagir sur plusieurs applications à la fois. Cette génération permet de prendre en compte les tâches plus complexes que doit mener un individu, en lui permettant d'utiliser plusieurs applications partageant la même information : c'est à dire que l'information produite par une application peut être manipulée par une autre. Ceci a été possible grâce à une standardisation du stockage de l'information, à l'apparition de mécanismes de communication entre applications tels que le presse papier ou des mécanismes plus évolués tels que les technologies OLE. Cette génération a vu naître des métaphores d'interaction telle que la métaphore du bureau.

La génération suivante des systèmes interactifs peut être associée à un type de relation n à 1. C'est à dire qu'un groupe de personnes interagit simultanément ou pas sur la même application. Ceci a donné naissance à un ensemble de collecticiels dédiés à certaines tâches telles que la coédition, l'aide à l'argumentation ...

Puisqu'il semble que nous nous dirigeons vers des types de relations de plus en plus complexes, la dernière génération doit être de type n à m, ou plusieurs utilisateurs veulent pouvoir travailler ensemble en utilisant plusieurs applications. Cette situation se justifie par la complexité croissante des tâches à réaliser, qui nécessitent la compétence de plusieurs personnes et l'association de différents outils pour assurer sa réalisation (voir chapitre II sur "la complexité d'une application coopérative").

C'est ce dernier type de relation qui doit être prise en compte dans un environnement CSCW. Pour permettre ce nouveau type d'interaction, il apparaît intéressant de réfléchir sur des nouvelles métaphores d'interaction, de proposer des modèles d'architecture qui permette de prendre en compte la communication interpersonnelles. Les nouvelles technologies du multimédia semblent être bien adaptées pour prendre en compte ce dernier aspect.

Il semble primordial d'intégrer dans les systèmes le son et la vidéo, en réfléchissant à leur intégration au niveau de l'interface en terme de présentation, de métaphore et de manipulation.

Le deuxième Bilan est que tous ces collecticiels ont besoin de valider leurs concepts en se confrontant à la réalité. C'est à dire qu'ils doivent être mis en situation réelle afin de permettre leur évaluation en terme de résultats techniques, et surtout en terme "d'utilisabilité". La conception de ces collecticiels s'étant souvent faite sans tenir compte

de la connaissance d'un existant (puisque'il n'existe pas ou peu), il est important de pouvoir se construire cet existant en évaluant systématiquement ces collecticiels. Ceci nous amène au problème de l'évaluation des collecticiels qui est complexe puisque nécessitant le déploiement d'une technologie (plusieurs stations utilisateurs dans des lieux différents), l'installation dans une organisation existante avec ses règles et ses modes de travail, la réalisation de systèmes relativement fiables pour être utilisés par des utilisateurs finaux, et surtout la création de méthodes d'évaluation.

Le troisième bilan est que les différents collecticiels implémentent chacun implicitement ou explicitement un modèle de coopération. Ce modèle de coopération est souvent basé sur la notion de partage de ressources et de la définition d'un algorithme de contrôle. Le point qui nous semble intéressant de souligner est que ces algorithmes ne doivent pas être uniques, de façon à ne pas contraindre l'activité du groupe. Nous pensons même que la conception de systèmes interactifs de groupe ne doit pas définir un modèle de coopération ou plusieurs, mais fournir plutôt un mécanisme générique qui puisse permettre au groupe de définir son propre modèle de coopération (mais des mécanismes de fonctionnement par défaut restent nécessaires). L'évaluation des collecticiels permettra bien sûr de recueillir des informations pour concevoir de nouveaux modèles de coopération.

Le dernier bilan est que les collecticiels laissent peu de place à l'utilisateur. Il est souvent référencé par le système sous un nom de machine (c'est le cas dans SharedX a moins de définir des alias système, ou de Groupkit). Il n'existe pas en tant qu'être social (objet social), et n'est donc pas présentable sous cet aspect aux autres utilisateurs

Résumé :

Penser la conception d'architecture d'accueil en terme environnement CSCW

Evaluation systématique des collecticiels en situation réelle

Le collecticiel doit pouvoir s'adapter en terme de modèle de coopération

L'environnement doit permettre une perception sociale de l'utilisateur

CHAPITRE II

LA COMPLEXITE D'UNE ACTIVITE COOPERATIVE (FONDEMENTS THEORIQUES)

L'objectif de ce chapitre est de décrire les fondements théoriques du travail coopératif afin de mettre en évidence certaines caractéristiques importantes de celui-ci qui peuvent influencer la conception de collecticiel. La deuxième partie de ce chapitre est consacrée à la description de ces caractéristiques en terme de collecticiel en partant d'une situation simpliste mais représentative des activités coopératives temps réel assistées par ordinateur.

I - MIEUX COMPRENDRE LE TRAVAIL COOPÉRATIF POUR MIEUX L'ASSISTER !

1 - Le travail coopératif

1.1 - Conditions d'apparition du travail coopératif

Schmidt [Schmidt 91] décrit ces conditions de la façon suivante :

"...le travail coopératif apparaît lorsque plusieurs acteurs sont concernés pour réaliser un travail et sont mutuellement dépendants dans leur travail et doivent coordonner et assembler leurs activités individuelles pour effectivement réaliser le travail qu'il leur est demandé".

La relation d'interdépendance dans le travail apparaît comme un facteur essentiel parmi les conditions d'apparition. Elle est la source du travail articulatoire (voir paragraphe 1.2), et un facteur stimulant de la communication entre les individus concernés. On peut en déduire qu'il faut s'intéresser à l'analyse du travail en terme d'identification de relations entre les tâches (analyse de la tâche, analyse de l'activité), et ensuite analyser la communication en terme de contexte, de contenu et en terme de "véhicule". Analyser la tâche permettra d'identifier un nombre de sous-tâches qui pourront soit être réalisées par un individu, donc entraîner un travail en parallèle entre sous-tâches indépendantes, ou d'identifier les tâches qui nécessitent une coordination entre les différents acteurs ou /et un assemblage entre les différents résultats des sous-tâches. Il sera aussi important d'essayer d'identifier des niveaux de coordination et d'assemblage afin de concevoir des outils (peut-être informatique) permettant le bon déroulement de cette coordination ou de cet assemblage.

1.2 - Les caractéristiques du travail coopératif

Le travail articulatoire

Une tâche dans laquelle des individus doivent coopérer génère des tâches secondaires de médiation¹ et de contrôle de l'association des individus. Ce surcoût de travail est appelé travail "articulatoire" [Schmidt 93]. On peut immédiatement en conclure que le passage d'un individu à plusieurs individus dans la réalisation d'une tâche n'entraîne pas forcément un gain d'un facteur N.

On peut ici faire une analogie avec les systèmes parallèles ou multiprocesseurs, où l'adjonction de processeurs n'entraîne pas forcément un gain significatif. Pour certains processus, il peut même y avoir détérioration des performances : la vitesse de prise de décision peut être modifiée puisque dépendante de plusieurs acteurs ; ou accrue de

¹ Ce sont des tâches d'arbitrage, de conciliation, de négociation ou de prise de décision.

façon infinie : ceci va correspondre aux processus qui ne peuvent pas être gérés par une seule personne.

Le travail articulatoire apparaît comme un élément à part entière dans le travail coopératif, en générant un ensemble d'activités qui définissent la dimension de *distribution* dans le travail coopératif.

Dans [Schmidt 93 page 18], on trouve une énumération de différentes dimensions dans le travail coopératif. Nous ne citerons ici que celles qui nous permettront de mieux comprendre comment le travail articulatoire peut être pris en compte dans les systèmes CSCW.

- (1) L'articulation en termes d'acteurs permet d'identifier dans le travail coopératif les différents acteurs concernés en identifiant leur rôle, les différentes tâches collectives et individuelles qui relèvent de leurs compétences ;
- (2) L'articulation en termes de tâches, qui va permettre de définir la tâche du groupe (but à atteindre et moyens à disposition) et ses relations avec les autres tâches incluses dans le travail à réaliser ;
- (3) L'articulation en termes d'activités ², c'est à dire avoir une connaissance du niveau d'avancement des autres dans leur plan d'action ainsi que sur le plan d'action lui-même : le quoi et le pourquoi au niveau du passé, du futur et du présent ;
- (4) L'articulation en terme de ressources partagées : partage d'information (définition de rôle, de droit et de visibilité sur cette information), partage d'outil (lequel, comment l'obtenir, comment l'utiliser, pourquoi faire, etc).

Ces différentes dimensions vont souvent être définies par l'environnement de travail, l'état de l'activité qui est conduite pour mener à bien ce travail, et certaines contraintes organisationnelles.

Malgré ce surcoût de travail, le travail coopératif est en pleine émergence. Ceci s'explique par la complexité des tâches à réaliser, qui nécessitent soit une augmentation de la capacité par le nombre ou bien un accroissement de cette capacité par la réunion de personnes possédant des connaissances complémentaires [Schmidt 93 page 5]. En résumé, le travail coopératif est incontournable lorsque la complexité de la tâche est telle qu'un individu ne puisse la réaliser seul.

² Dans [Derycke 94b] on trouve cette définition : suivant la tradition scandinave, le terme Activité est utilisé pour traduire un processus de travail comme un déroulement en cours d'Action. Le "cours d'Action" étant l'activité d'un (ou plusieurs) acteur(s) engagé(s) dans une situation qui est significative pour ce (ou ces) dernier(s), c'est-à-dire racontable et commutable par lui(ou eux) à tout instant [Pinky 92]. Ceci est proche de la notion d'activité de Vigotsky et autres [Vigotsky 85]. Le concept de tâche d'un autre côté est utilisé pour traduire une intention opérationnelle indépendamment du comment la réaliser. Une tâche est exprimée en terme de quoi, une activité en terme de comment. Une tâche peut être accomplie, une activité peut cesser [Schmidt 93].

Perception des autres

Percevoir les autres, c'est avoir conscience que leur activité intervient ou a des incidences sur notre propre travail. Le fait de communiquer avec les autres induit un niveau de perception, le fait de partager le même environnement de travail favorise cette perception, le fait de voir les autres interagir sur l'environnement ou de voir le résultat de leur activité contribue à la mise en place de la perception des autres. Cette perception va permettre à chaque individu de connaître l'état d'avancement dans lequel le groupe se trouve par rapport à l'objectif à atteindre, et permettra donc d'ajuster, d'affiner la stratégie définie par le groupe.

Dynamique

Le travail coopératif est constitué d'un certain nombre de tâches individuelles qui doivent être ordonnancées, coordonnées. Si ces tâches sont prescrites, le processus de coordination et d'ordonnancement est souvent figé d'avance et imposé aux différents acteurs impliqués dans le travail coopératif. Ceci est le cas des systèmes orientés tâche tel que les Workflow³. Si la décomposition en sous-tâches est **dynamique**, ceci entraîne que les individus engagés dans un travail coopératif doivent être capables d'alterner travail collectif (mettant en jeu plusieurs personnes) et travail individuel. Ils doivent donc gérer en parallèle différents flots d'information et plus généralement plusieurs processus. Les individus ont donc une part active dans la décomposition, et c'est eux qui contrôlent l'ordonnancement et la coordination entre leurs différentes activités.

Conclusion

Selon Schmidt, ce qui est important, c'est de comprendre les modes et les mécanismes d'interaction existants dans l'articulation du travail coopératif, afin d'identifier des dispositifs, des techniques, des outils (sûrement informatique) qui permettront de l'assister, d'obtenir un gain de compétences de la part des utilisateurs afin de mieux travailler en groupe.

2 - Différents points de vue sur le travail coopératif

L'objectif de cette étude est d'essayer de mieux cerner le travail coopératif en essayant de le caractériser en adoptant des points de vue différents. Le chapitre I est consacré à l'analyse des systèmes informatiques existants, ce chapitre est consacré à une étude plus théorique du travail coopératif afin d'en extraire certains enjeux pour la conception des futures générations de collecticiels, et plus précisément de susciter une

³ Le Workflow peut être défini comme étant une séquence de tâches réalisées par un individu ou un groupe d'individus qui respectent certaines procédures et certaines règles normalisées et définies auparavant.

discussion sur l'incidence de ces apports théoriques au niveau de l'interaction Homme Machine dans le contexte du travail coopératif [Derycke 94b].

1.1 - Le point de vue social et organisationnel sur le travail coopératif

Ce point de vue est à la base des caractéristiques qui ont été citées précédemment. Selon Alain Derycke [Derycke 94b], ce point de vue traite du rapport des technologies avec la société, de l'importance des relations humaines dans le travail. De façon assez simple, on peut montrer que l'analyse du travail coopératif prend sa source dans les écrits de Marx ou les travaux de Taylor. Ce qui est important dans cette approche, c'est le travail lui-même, les relations des divers agents concernés avec leur environnement, le rôle des artefacts⁴ dans les interactions de médiation, la compétence des différents agents, les relations existantes entre les agents aussi bien organisationnelles qu'humaines, la motivation et la nécessité pour chacun de rendre visibles ses activités.

Le point de vue organisationnel va imposer des règles au travers de l'environnement, en définissant des rôles et des responsabilités qui vont être attribués aux différents individus.

Au coeur de cette conception du travail coopératif est la notion d'interdépendance, qu'implique la nature sociale du travail. Comprendre cette interdépendance est indispensable pour mieux concevoir des environnements CSCW.

Le second point important de ce point de vue est le travail articulatoire, qui est lui même une conséquence de la notion d'interdépendance (Voir § 1.2).

1.2 - Le point de vue communicationnel sur le travail coopératif

Pour mener à bien cette production commune (résultat du travail coopératif), les personnes communiquent entre elles afin de pouvoir coordonner leurs activités individuelles, ajuster leur stratégie. Définir la communication de groupe n'est pas chose facile. L'objectif de ce paragraphe est de comprendre comment les personnes communiquent afin de voir comment les systèmes de communication se basant sur les systèmes informatiques peuvent la prendre en charge.

Notre objectif n'est pas de donner une définition, ni une modélisation de la communication, mais d'énoncer certaines caractéristiques qui peuvent influencer la conception d'un collectif.

Le rôle du langage

C'est un assemblage de symboles qui prennent la place de quelque chose ou le représente. Ce qui est important dans le langage c'est la signification associée à cet

⁴ Phénomène d'origine humaine, artificiel.

ensemble de symboles. Ceci correspond au niveau formel du langage qui associe à un mot une signification. Un deuxième niveau plus culturel qui contextualise le langage : l'interprétation d'un dialogue peut être influencé par le contexte, l'environnement dans lequel il est prononcé [Meyers 90 Chapitre V].

La communication de groupe

Encore une fois, il est très difficile de donner une définition d'un groupe. Pour cela nous utiliserons une définition de Hare [Meyers 90 page 251] : "Il y a en somme cinq caractéristiques qui différencient le groupe d'une collection d'individus. Les membres du groupe sont en interaction les uns avec les autres. Ils partagent un but commun et un ensemble de normes, ce qui oriente et limite leurs activités. Ils élaborent aussi un ensemble de rôles et un réseau d'attractions interpersonnelles, qui permettent de les différencier des autres groupes". L'interaction va mettre en jeu communication verbale et non verbale, et plus généralement un processus qui permettra à chacun d'avoir conscience de l'existence des autres. Les membres du groupe ne vont pas interagir et plus généralement communiquer par hasard, mais pour coordonner leur activité afin d'atteindre un but. De plus, le but à atteindre va influencer les types d'interaction. Le système de normes établit plus ou moins la manière dont les relations et les activités se dérouleront ou doivent se dérouler. Elles définissent explicitement ou implicitement les règles de fonctionnement du groupe. Lorsque le groupe persiste dans le temps, certains rôles apparaissent au sein de celui-ci (leader, personne ressource dans un domaine de compétence, personne habile dans la gestion des conflits...). Ce n'est qu'au cours du temps que ces rôles apparaissent, lorsque les membres du groupe apprennent à travailler ensemble, malgré l'existence de rôles imposés par l'environnement dans lequel le groupe fonctionne (par exemple le rôle d'enseignant).

Le contenu de la communication sera directement lié au but déclaré par le groupe. Par contre le processus de communication ne va pas seulement dépendre du contenu verbal (relatif au but), mais va aussi tenir compte de la manière dont les membres du groupe interagissent. Le processus va donner des informations sur : la prise de parole, l'existence de leader, la prise de décision etc.

La connaissance et l'analyse de ces processus va permettre d'élaborer des modèles qui peuvent nous aider dans la conception de systèmes informatiques pour assister les activités de groupe. Par exemple, John Dewey [Meyers 90 page 279] décrit un processus de décision en six étapes : 1. clarifier et définir le problème, 2. générer les solutions, 3. évaluer les solutions, 4. choisir la meilleure solution, 5. appliquer la solution, 6. évaluer les résultats. Le leader du groupe intervient positivement dans ces différentes phases comme catalyseur, facilitateur, directeur ou gestionnaire. Il va aider le groupe à franchir les différentes étapes. On trouve aussi d'autres analyses de

processus comme la gestion de conflit dans un groupe. Par exemple Pondy [Pondy 93 page 31] analyse le conflit en une série d'épisodes tels que la période de latence d'un conflit, l'apparition du conflit, la perception du conflit...

L'analyse de ces différents processus permet de définir la dynamique des groupes [Anzieu 68]. Il apparaît intéressant de décrire ce phénomène pour mieux comprendre le fonctionnement d'un groupe. Ceci permet d'aborder (comme on vient de le citer) la gestion des conflits dans un groupe, l'organisation du groupe en terme de rôle.

Pour conclure sur ce point de vue communicationnel, il apparaît que la conversation est régulée par un ensemble de règles, de normes que l'on peut caractériser de sociales et culturelles. Par exemple, McGrath [McGrath 90] identifie quelques normes qui régissent le processus de communication de groupe en face à face :

- une seule personne possède le contrôle à la fois ;
- il doit y avoir un locuteur par défaut (l'animateur du groupe, l'enseignant) ;
- au cours de la discussion, le locuteur courant peut exercer un contrôle sur le choix du locuteur suivant ;
- le temps de parole est partagé entre tous les interlocuteurs, pas forcément de façon équitable ;
- le temps de latence entre les différentes contributions des interlocuteurs est important : il ne doit pas être trop court pour éviter les interruptions, ou trop long car ceci entraîne la création de silence avec toutes ses conséquences (voir [Meyers 90] page 134) ;
- le passage d'un interlocuteur à un autre est signalé par des phénomènes appartenant à la communication verbale et non verbale, qu'il est important de respecter.

La communication non verbale

Lorsque nous communiquons avec les autres, il y a ce que nous disons (le quoi) et la manière dont nous le disons (le comment). Le quoi est associé à l'information et les données échangées alors que le comment fait référence à la façon dont votre message doit être interprété (blague, sérieux) et au genre de relation qui lie les individus (relation amicale, de travail). La communication non verbale peut prendre des formes relativement différentes [Meyers 90 page 141 à 150] :

- le paralangage : le timbre et le volume de la voix, les inflexions et l'accent mis sur certains mots, les coupures d'une phrase ;
- les gestes : ceux-ci accompagnent la langue orale et peuvent même la remplacer pour véhiculer certains messages. Le geste possède une dimension culturelle, il est automatique, et surtout il aide à la compréhension du discours parlé ;
- l'expression du visage et les mouvements corporels (froncer les sourcils, contact visuel...).

La communication non verbale aide à la compréhension du langage parlé. Elle inclut habituellement de l'information sur le contenu du message verbal. La question qui nous semble intéressante est de savoir comment la modification, ou la non communication de ce langage non verbal influe sur l'interprétation du langage parlé.

3 - Un exemple de travail coopératif : l'apprentissage coopératif

Nous voulons dans cette section nous focaliser sur un cas particulier du travail coopératif, qui est l'apprentissage coopératif. Ceci a deux objectifs : le premier est de donner une illustration concrète du travail coopératif dont chacun a plus ou moins connaissance et le deuxième est d'introduire l'apprentissage coopératif comme champ applicatif de nos travaux.

Qu'est ce que l'apprentissage coopératif ?

C'est beaucoup plus facile de définir ce que n'est pas l'apprentissage coopératif ou collaboratif que de donner une définition formelle de celui-ci.

On peut définir le domaine de l'apprentissage coopératif par les six éléments suivant [Derycke 93c] :

1. l'apprentissage est un processus individuel et pas collectif. Il est dépendant de facteurs extérieurs tels que le groupe et les interactions interpersonnelles ;
2. l'apprentissage n'est pas seulement un processus individuel, mais aussi un processus social. L'acquisition des connaissances ne peut se faire que par confrontation, interactions avec les autres ou avec un groupe ;
3. l'apprentissage coopératif nécessite des échanges entre pairs, une interaction entre individus possédant des rôles différents qui sont interchangeables (apprenant, enseignant, personne ressource, animateur) ;

4. le processus d'apprentissage coopératif doit être un processus synergique. C'est à dire que l'association des individus pour former un groupe d'apprentissage doit mener à un gain personnel au moins équivalent, voir supérieur, au gain obtenu par un apprentissage individuel ;
5. le bénéfice potentiel n'est pas toujours atteint à cause de phénomènes induits par l'apprentissage coopératif lui-même : manque d'initiative, les incompréhensions, les conflits, les compromis ;
6. l'apprentissage coopératif ne nécessite pas forcément un apprentissage en groupe, mais parfois la simple possibilité d'exprimer, de verbaliser ses problèmes aux autres afin d'obtenir un avis ou une opinion peut être suffisant pour franchir un nouveau cap du processus d'apprentissage.

En résumé, nous utilisons le terme apprentissage coopératif pour signifier le moyen d'acquisition individuel de connaissance, de pratique, de processus qui sont le résultat d'une interaction en groupe [Kaye 92].

Fondements théoriques de l'apprentissage coopératif

Notre étude se borne à décrire rapidement ces fondements théoriques dans l'objectif de confirmer ce que nous avons dit précédemment sur le travail coopératif, de présenter des notions qui seront utiles dans le chapitre sur la conception d'un environnement d'apprentissage et sur le chapitre incluant l'évaluation de ce même environnement.

La première approche que l'on peut prendre est associée à Vigotsky et Léontiev [Vigotsky 85]. Elle est définie comme une approche médiationnelle, qui donne un rôle prépondérant à un médiateur (qui peut être un autre apprenant, un formateur, mais aussi un document papier) qui va permettre à un individu de gravir un échelon dans son processus d'acquisition des connaissances. Pour citer Vigotsky : "Ce que vous pouvez faire aujourd'hui avec l'aide d'autrui, demain vous pouvez le faire seul". En conclusion, nous pouvons dire que le processus d'apprentissage doit permettre à un individu d'acquérir des connaissances par l'intermédiaire d'une relation dite de "médiation" avec un pair. Ce n'est pas par l'action que l'apprenant construit ses savoirs mais par l'interaction [Clément 94].

Ces travaux issus de Vigotsky, ont été repris et développés par W.Doise, G.Mugny, et A.N.Perret-Clermont [Mugny 84]. Ils insistent sur deux mécanismes importants de l'apprentissage, tous deux liés à la nécessité d'apprendre en groupe : le conflit socio-cognitif apparaissant entre deux partenaires d'une même tâche; le marquage social qui définit la nature sociale et l'enjeu social des tâches et de la situation. Le conflit socio-

cognitif n'est pas vraiment un conflit, mais plutôt une forme d'interaction sociale. C'est un mécanisme qui agit à chaque fois que deux personnes oeuvrent en commun. Dans le cas d'un travail de groupe, il se produit des échanges, une confrontation des points de vue propres à chacun et l'on s'est aperçu que c'est cette confrontation qui facilite le progrès cognitif. Ce qui est aussi fort intéressant, c'est que Doise et Mugny ont défini les conditions d'émergence, à travers lesquelles on trouve certaines caractéristiques énoncées précédemment sur le travail coopératif :

- les divers membres du groupe doivent être de niveaux différents, ou de points de vue différents. Chaque membre a une perception mutuelle de ses points de vue ;
- l'un des observateurs (ou plusieurs) adopte un point de vue critique ;
- la nécessité d'une certaine qualité de l'interaction sociale, c'est à dire qu'il y est un désir de rechercher la solution, d'un accord, d'un cadre de référence commun ;
- ne pas esquiver le conflit : ne pas chercher à répartir la tâche en sous-tâches où chacun travaille isolément ;
- l'existence de certains pré-requis qui peuvent être sociaux : savoir communiquer avec les autres de façon efficace ou avoir les moyens de le faire ; ou au niveau des connaissances ou des compétences nécessaires pour résoudre la tâche ;
- il faut absolument éviter les situations de domination car aucun participant ne progresse.

Il apparaît claire que la construction des savoirs possède une dimension sociale relativement importante, au même titre que la communication humaine. La notion de conflit (socio-cognitif) peut être vu non pas comme une compétition mais plutôt comme un chemin vers la coopération, la collaboration, un état nécessaire pour que des points de vue différents puissent s'ajuster progressivement.

Conclusion à partir des aspects théoriques

En conclusion, les données théoriques sur le travail coopératif vu sous les différents points de vue doivent être prises en compte dans la conception des collecticiels. Actuellement, il est difficile de faire une liste de recommandations pour la conception d'un collecticiel qui serait la conséquence des différents apports théoriques précédents. Il apparaît cependant que des caractéristiques du travail coopératif tel que le travail articulatoire, la prise en compte des caractéristiques de la communication de groupe (dynamique des groupes) soient des points importants à prendre en compte dans la conception de collecticiel.

Il apparaît maintenant important d'identifier comment ces théories ont été utilisées dans la conception de collecticiel, et comment la création de collecticiels sans tenir compte des apports des sciences sociales amène à un échec. Il est aussi important d'identifier en quoi l'utilisation de la technologie modifie, ou nécessite une interprétation différente des fondements du travail coopératif.

Premier Bilan :

Comment percevoir et organiser le travail articulatoire

Comment prendre en compte les multiples facettes de la communication

Comment intégrer les mécanismes de la dynamique des groupes

II - RETOUR AU COLLECTICIEL

Le travail coopératif via le média informatique va s'articuler autour d'un système d'information collectif (mémoire commune) et un système de communication. Dans ce paragraphe, nous essayons de définir comment les aspects plus théoriques vus précédemment peuvent s'interpréter dans un contexte de conception de collectifiel.

Pour nous aider dans cette démarche, nous proposons tout d'abord de définir une situation simpliste de travail coopératif, qui nous permettra de percevoir l'utilité de la description des points de vue précédents.

1 - Les leçons du passé

Nous voulons ici décrire quelques principes qui ont été énoncés pour la conception des collectifiels, à partir de l'utilisation de quelques prototypes, et de l'analyse des échecs suite à leur utilisation. Il s'agit ici d'illustrer comment l'échec de l'utilisation des collectifiels trouve une justification dans les aspects théoriques du travail coopératif.

Les travaux de Tatar & al [Tatar 91] énoncent quelques principes et défaillance d'un collectifiel nommé Cognoter. Celui-ci est un outil de dessin qui permet l'organisation d'idée pour un petit groupe de personnes [Stefik 87a], se trouvant dans la même pièce. Les principaux défauts constatés par les auteurs sont :

- le système impose implicitement un modèle de communication qui n'est pas adapté ;
- les utilisateurs rencontrent des difficultés de coordination ;
- les utilisateurs rencontrent des difficultés pour identifier les objets sur lesquels l'interlocuteur parle ;
- problèmes d'identification de l'utilisateur qui modifie l'environnement partagé.

D'autres travaux sur l'analyse de collectifiels (asynchrone) réalisés par Grudin [Grudin 88], montrent certaines défaillances dans la mise en oeuvre réelle d'un collectifiel, que nous pensons génériques :

- la disparité entre les personnes qui bénéficient du collectifiel et celles qui doivent réaliser le travail supplémentaire (dont le travail articulatoire) engendré par l'utilisation du collectifiel ;
- la complexité d'utilisation du système, qui amène les utilisateurs à le rejeter ;

- la difficulté d'analyser et d'évaluer les collecticiels, car ils nécessitent de nouvelles méthodes d'évaluation. Les évaluations doivent être réalisées sur des périodes de temps longues, mais elles sont complexes à mettre en oeuvre car on doit observer plusieurs personnes réparties sur plusieurs sites, et on doit prendre en compte la pluralité dans la composition d'un groupe, et les facteurs environnementaux.

La description rapide des expériences sur la mise en oeuvre de quelques collecticiels suscite des questions que nous avons résumées dans l'encadré suivant.

Deuxième Bilan :

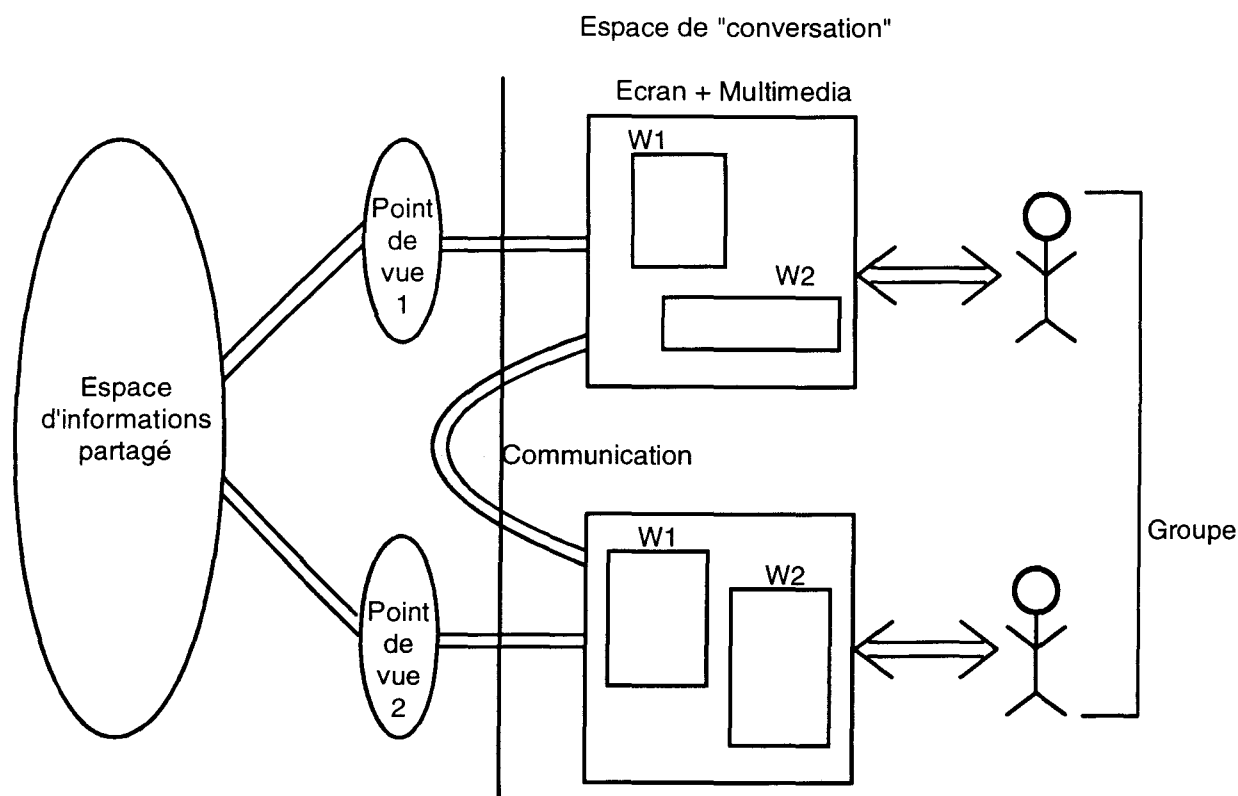
Le système informatique doit permettre la communication interpersonnelle : comment et jusqu'à quel point ?

Que doit on percevoir de l'activité des autres ? Comment savoir par qui est réalisé ce que je vois ou ce que j'entends ?

Comment réaliser l'évaluation d'un collecticiel ?

2 - Description simpliste d'une situation de travail coopératif mettant en jeu un système d'information et de communication

La figure 1 adaptée de [Derycke 94a] permet de représenter une situation typique d'un travail coopératif assisté par ordinateur où l'activité se réalise en temps réel. Plusieurs personnes sont engagées dans un travail coopératif. En fait les utilisateurs se trouvent devant une station de travail multimédia qui met à leur disposition des d'outils qui vont permettre de réaliser une tâche. Pour ceci, ils disposent de deux espaces différents : un associé aux informations (espace d'information) partagées par l'ensemble des utilisateurs qui contient des documents et constitue une sorte de mémoire de groupe ; le second associé à la communication entre les différents membres du groupe permettant des échanges verbaux et textuels, que l'on peut nommer espace de conversation. Pour l'instant, on peut considérer que les espaces sont disjoints, que l'**espace d'information** contient les objets de la conversation (sur quoi on parle) et que l'**espace de conversation** contient la conversation (ce qui se dit).



Figur 1 : Description simpliste d'une situation de travail coopératif

3 - L'espace d'information

Les différents utilisateurs peuvent partager exactement le même *point de vue* sur l'information collective afin de constituer un contexte de travail commun. Il constitue comme une mémoire de groupe, capable de stocker les informations entre deux séances de travail. L'activité des participants va se traduire en une série d'actes instrumentaux coordonnés et ordonnés, qui vont modifier l'espace d'information. Ces actes instrumentaux sont réalisés à l'aide d'outils tels que les éditeurs, des tableurs, des systèmes experts... Le travail coopératif étant une articulation de tâches individuelles et collectives, les points de vue contiendront l'information associée aux tâches collectives et aux tâches générées par le travail articulatoire, et l'information associée à la tâche individuelle de l'utilisateur concerné. La conception de ces points de vue peut être complexe et devra être dynamique car la décomposition d'une tâche coopérative est elle même dynamique, et tenir compte de la notion de rôle des individus dans la tâche à accomplir (voir travaux de B. Carré sur la représentation multiples d'objet avec point de vue [Carré 89]). Cette nécessité de développer la notion de point de vue pose des problèmes complexes d'articulation de la cohérence de ces mêmes points de vue. Les apports de l'Intelligence Artificielle peuvent être intéressants avec la conception d'agents filtrants et les travaux sur les univers multi-agent et la gestion coopérative des connaissances [Bourdon 93].

4 - L'espace de conversation

Pour nous, dans le contexte du collectif synchrone, l'espace de conversation contient obligatoirement un canal sonore.

Le problème de la communication non verbale

Le canal sonore ne permet pas de transmettre toute la communication non verbale. La création de nouveaux artefacts pour supporter la communication non verbale semble indispensable. L'utilisation de la vidéoconférence permet une transmission d'information plus importante liée à la conversation (expression faciale), mais ne semble pas répondre totalement aux problèmes [Edigo 88]. Elle ne permet pas d'avoir forcément la perception de l'activité de l'autre, ni forcément percevoir ce qu'il regarde. L'autre aspect est la prise en charge de la gestuelle, sur lequel nous reviendrons dans le paragraphe suivant, car elle met en jeu aussi l'espace d'information.

Le rôle du média dans la communication

Le fait d'utiliser des médias tels que la visioconférence et l'audioconférence modifie la communication, surtout la métacommunication comme nous l'avons vu précédemment. Par exemple, Clark et Brennan [Clark 91] ont cherché à caractériser les traits de la communication qui aide une personne à facilement comprendre une autre (le "grounding") à savoir :

- la co-présence : les utilisateurs sont dans le même environnement physique ;
- la visibilité : les utilisateurs se voient mutuellement ;
- l'audibilité : les utilisateurs communiquent en utilisant la parole. Un utilisateur peut entendre ce que les autres disent ;
- la co-temporalité : reçu comme il a été produit sans délai temporel ;
- la simultanéité : la possibilité de recevoir et d'envoyer simultanément ;
- le retour arrière : la possibilité de revoir ("rewiewability") ;
- la "revisability" : la possibilité de modifier les messages.

Les contraintes exprimées ci-dessus permettent de caractériser la communication en fonction du média utilisé (voir tableau ci-dessous).

Médium	Contraintes
Face à face ou présentiel	Co-présence, visibilité, audibilité, co-temporalité, simultanéité
Le téléphone	Audibilité, co-temporalité, simultanéité
La visioconférence	Visibilité, audibilité, co-temporalité, simultanéité
Courrier électronique	Retour arrière, "Revisability"
L'audioconférence	Co-temporalité, audibilité,

McGrath [McGrath 90] souligne aussi l'influence de l'utilisation des nouvelles technologies sur la communication humaine. Il insiste surtout sur la modification de la dimension temporelle de la communication. Pour lui, ce qui doit être synchronisé dans la communication de groupe, c'est la communication verbale et non verbale. Si les médias utilisés pour véhiculer la communication peuvent avoir des contraintes de co-temporalité différente, alors l'affirmation précédente ne sera pas respectée. Ceci entraîne qu'il faudra synchroniser les médias de co-temporalité différentes, ou utiliser des artefacts pour palier à ce problème.

5 - Articulation de l'espace d'information et de l'espace de conversation

La dimension communication dans les collecticiels nécessite l'articulation entre ces deux espaces. L'espace d'information qui permet la visualisation des objets sur lesquels on parle (objet de la conversation) doit être tout d'abord "synchronisé" avec l'espace de conversation qui contient les énoncés composant la conversation. Il est plus simple de donner un exemple pour illustrer l'importance de ce phénomène : un utilisateur manipule un collecticiel qui lui permet d'afficher de l'information dans un espace de visualisation collectif respectant le Wysiwis ; s'il existe un décalage dans le temps entre le moment où lui voit l'information et où les autres la voient (par exemple suite à des temps de transmission), il se peut qu'il commence à parler sur l'information alors que les autres ne partagent pas son point de vue visuel. Ce décalage temporel entre la conversation et la perception visuelle des objets de la conversation amène les différents participants à se trouver dans des espaces temporels différents, donc susceptibles d'engendrer des incompréhensions qui peuvent altérer le fonctionnement du groupe.

En conclusion, le respect du Wysiwis, ou surtout la relaxation de ces contraintes doit être réalisé en prenant conscience que d'autres média peuvent être utilisés, possédant une échelle temporelle différente en terme de perception pour l'utilisateur. Que se passe t-il lorsque l'on regarde une vidéo dont le son est décalé dans le temps par rapport à l'image ?

Au moment de la conception du collecticiel, il est important de prendre ce problème en considération, afin de proposer des solutions qui permettront aux utilisateurs de synchroniser leur discours sur le média le plus "lent". Concrètement pour l'exemple cité précédemment, il apparaît que l'utilisateur qui initie la commande doit avoir une rétroaction adaptée (curseur d'attente intelligent ou multi-utilisateur) qui lui permettent de connaître le moment à partir duquel tous les utilisateurs voient l'information.

Synchronisation temporelle entre les deux espaces

Le cas que nous avons présenté ci-dessus qui associe objets de la conversation et la conversation elle-même en terme d'échelle de temps, nous permet de soulever un autre degré d'articulation : la prise en compte de la gestuelle.

La communication au sein d'un groupe associée à un partage d'information, ne met pas seulement en jeu la parole, mais aussi la communication non verbale. Nous avons besoin de désigner, de montrer les objets sur lesquels nous sommes en train de parler. L'utilisation de repères spatiaux dans le langage est parfois difficile à utiliser, et peut apparaître comme ennuyeux. Ceci se traduit par des échanges verbaux du type "en haut, en bas, à gauche ...". Il est beaucoup plus naturel d'accompagner le discours de gestes de désignation, qui vont permettre d'identifier des objets sur lesquels on parle sans encombrer le discours d'utérances associés à des repères spatiaux.

Les collecticiels prennent en charge ces repères en proposant la notion de télépointeur ou de télécurseur [Greenberg 92] [Nakajima 93]. Comme le curseur est l'extension de la main pour manipuler des objets lorsque l'on utilise la métaphore de la manipulation directe, le télécurseur repose sur le même principe et est en plus visible par tous les utilisateurs. Il devient un artefact pour prendre en charge la gestuelle à travers un dispositif informatique. L'ambiguïté du télépointeur est contenue dans les deux phrases précédentes : l'outil qui nous permet de modifier l'espace d'information (réalisation des actes instrumentaux) est aussi utilisé pour prendre en charge la gestuelle qui appartient à l'espace de conversation. Plus généralement, on utilise le même concept pour réaliser des actes appartenant à des espaces différents. Dans la communication face à face, la main par l'intermédiaire d'un outil (stylo) sert à modifier l'espace d'information, et sert aussi pour faire des gestes de désignation. Mais il existe d'autres niveaux de communication qui permettent de dissocier le geste de la modification d'information (le paralangage). On peut conclure cette discussion par la question suivante : "le télépointeur est il toujours la bonne technique, le bon paradigme pour prendre en charge la gestuelle dans la communication médiatisée ?

Certains travaux [Benford 93] [Falhen 93] placent cette articulation en terme d'espace à un autre niveau, qui est l'immersion de l'utilisateur dans un environnement

virtuel. L'utilisateur est situé au milieu d'objets appartenant aussi bien à l'espace d'information qu'à l'espace de conversation. Cette situation est plus complexe que celle que nous avons décrit au début de cette section, car chaque utilisateur est représenté dans l'espace virtuel muni de ses canaux de communication.

Dans [Derycke 94a], on trouve une description des différents niveaux d'immersion de l'utilisateur dans un monde virtuel. La réelle question qui se pose est de savoir en quoi l'immersion de l'utilisateur dans cet environnement va lui permettre de mieux communiquer, et plus généralement en quoi la communication de groupe est elle mieux assistée ?

Il est important de se poser dès aujourd'hui la question de l'utilisabilité de l'interaction de l'utilisateur dans un monde virtuel ?

Articuler espace d'information et espace de conversation pour mieux prendre en charge la communication non verbale.

6 - Le problème de la gestion des conflits

Comment peut-on définir le conflit par rapport à notre situation simpliste ?

Les conflits qui apparaissent dans l'espace d'information vont être liés à la notion de partage de ressources. Dans le chapitre I, nous illustrons comment les collecticiels répondent aux problèmes en définissant un protocole d'accès à des ressources partagées. Dans ces approches, la notion de rôle et de droit est primordial.

En ce qui concerne l'espace de conversation, si on considère uniquement l'architecture physique qui la supporte, on pourra aussi le considérer comme une ressource partagée et définir un protocole qui gère le droit de parole. La technologie nous permet de bénéficier d'audioconférence qui ont la propriété d'offrir aux utilisateurs la possibilité de pouvoir parler quand on le veut tout en entendant la somme des voix de nos différents interlocuteurs. Comme nous l'avons précisé dans le paragraphe 2.2, la gestion des droits de parole dépend de certains phénomènes sociaux tels que le rôle, le statut, et certains faits appartenant au paralangage. La totalité de la communication n'étant pas prise en compte dans les collecticiels (même en utilisant la visioconférence), il apparaît que la régulation du flux verbal va être modifiée. Certains travaux comme ceux de Laurent Barme [Barme 94] proposent d'utiliser les capacités technologiques et les caractéristiques du son afin d'essayer de fournir des palliatifs qui permettent à chaque utilisateur de pouvoir mieux percevoir son interlocuteur "sonore". Nous pensons que ce type de services offerts par une audioconférence permettra plus facilement à chaque utilisateur de comprendre le discours des autres (en terme

d'interprétation), et permettra une meilleure gestion des échanges verbaux au sein du groupe.

A notre avis, la présence du canal sonore dans les collecticiels va permettre un niveau de détection des conflits, d'explicitation de ceux-ci et souvent ce même canal va être utilisé pour résoudre ces mêmes conflits. La plupart des collecticiels sont très orientés vers une régulation des conflits sur les données partagées. Ils laissent peu de place et prennent peu en compte le canal sonore. Le système impose un modèle de résolution de conflit et le canal sonore est souvent totalement oublié.

Dans les évaluations que nous avons réalisées, nous avons vraiment perçu l'importance du canal sonore pour la régulation des conflits (chapitre V). Il apparaît que le canal sonore est fortement sollicité pour gérer les conflits sur l'espace d'information. Ces conflits sont donc gérés au niveau humain (à mettre en opposition avec le niveau système qui impose un modèle de résolution). L'articulation entre les deux espaces pourrait apparaître sous la forme suivante : l'espace d'information ou plus exactement les outils de visualisation fournissent aux utilisateurs l'information sur l'existence d'un conflit, les moyens de le prendre en charge et permettent aux utilisateurs de le résoudre en utilisant le canal verbal. On peut étayer cette perspective en prenant en compte la nécessité du conflit cognitif, ou l'articulation de point de vue dans certains processus d'apprentissage. Une façon de stimuler le mécanisme de conflit socio-cognitif, est de permettre à plusieurs utilisateurs de manipuler la même information en même temps, et que le système puisse les prévenir qu'ils sont à deux à vouloir manipuler ces objets. *Le conflit n'est pas forcément une chose à résoudre a priori par un dispositif physique, mais il est un élément à part entière pour le bon fonctionnement de l'activité du groupe.* Le dispositif informatique est là simplement pour le relever, et offrir des moyens de communication et de résolution efficaces contrôlés par les utilisateurs. Un tel système pourrait par exemple être réalisé à partir de l'algorithme ORESTE d'Alain Karsenty [Karsenty 93] en offrant une notification aux utilisateurs lorsqu'il existe un conflit sur des actions non commutables, accompagné d'un mécanisme qui permette de reprendre l'activité dans un état cohérent, au lieu de prendre une décision arbitraire même si une rétroaction est fournie à l'utilisateur qui subit le refus.

Cette position sur l'algorithme de contrôle est aussi le résultat de l'étude de SharedX [Carlier 94] avec trois groupes tests⁵, qui met clairement en évidence que les

⁵ La notion de groupe test s'est traduite par la capacité des participants d'un groupe à utiliser l'outil informatique et leur niveau de connaissance sur les collecticiels. Le premier groupe était constitué de cinq personnes du laboratoire Trigone donc sachant manipuler des stations de travail et connaissant les collecticiels. Ils n'avaient pas connaissance des tenants et des aboutissants de l'expérimentation. Le deuxième groupe était constitué de cinq personnes inscrites en DEA Informatique connaissant la manipulation des stations de travail, mais ayant peu de connaissance sur les collecticiels. Le dernier groupe était constitué de cinq secrétaires de l'institut CUEEP sachant manipuler un PC en traitement de texte et ne possédant aucune connaissance sur les collecticiels.

utilisateurs verbalisent fréquemment les demandes de prise de contrôle en même temps ou avant qu'il utilise le dispositif informatique.

Il ne faut pas pour autant bannir l'algorithme de contrôle, car il reste une bonne solution face à des problèmes techniques liés à des dispositifs physiques (partage de ressources physiques), ou par des activités coopératives fortement contrôlées ou structurées (télécours, procédure de vote).

La solution que nous préconisons est d'extraire la politique de gestion des conflits de l'application elle-même. Une application coopérative sera décrite par la définition du couple (application, politique). Le composant politique est un objet instanciable plusieurs fois afin de fournir un système permettant d'utiliser la même application avec des comportements en terme de politique différents : ceci définit la malléabilité fonctionnelle de l'environnement. En se plaçant dans le cadre de l'interface homme machine, nous n'avons pas seulement un composant qui gère le dialogue entre l'application et l'utilisateur, mais aussi un composant à part entière qui prend en charge la communication Homme Homme et sa régulation.

Un collecticiel est un système dans lequel les politiques sont séparées de l'application. Le système est façonnable en fonction des besoins de communication et des habitudes de fonctionnement du groupe.

CHAPITRE III

ANALYSE ET SPECIFICATION

DE L'ENVIRONNEMENT D'APPRENTISSAGE

COOPERATIF COLEARN

I - DESCRIPTION GENERALE

Le projet Co-Learn est un projet de recherche et de développement du programme DELTA de la Communauté Economique Européenne (voir annexe I).

L'objectif de l'environnement d'apprentissage coopératif nommé **Co-Learn**, est de fournir une collection d'outils pédagogiques intégrés dans un environnement afin de supporter différents styles d'apprentissage coopératifs à distance. Les utilisateurs cible peuvent être aussi bien des utilisateurs naïfs qui connaissent peu de chose dans la manipulation d'outils informatiques, ou des utilisateurs chevronnés. Ce qui est important pour comprendre certains choix de l'environnement Co-Learn, c'est que les stations utilisateurs peuvent se trouver au foyer de l'utilisateur, au sein de son entreprise ou dans un centre de ressources d'un institut de formation. Cette mobilité des utilisateurs entraîne qu'une station n'est pas dédiée à un utilisateur, mais à n'importe quel utilisateur connu par le système Co-Learn.

Co-Learn veut promouvoir de nouveaux paradigmes pour l'enseignement à distance, que sont l'apprentissage coopératif ou collaboratif et un modèle de système centré sur la conversation et le dialogue.

On parle d'apprentissage collaboratif pour toute situation d'apprentissage dans laquelle l'acquisition d'un savoir par un membre d'un groupe est le résultat d'une interaction au sein du groupe.

Les systèmes centrés sur la conversation sont à opposer aux systèmes centrés sur l'activité qui sont conçus pour atteindre un objectif en réalisant une tâche au sein de laquelle le processus de communication peut être décrit à l'avance. Les systèmes centrés sur la conversation sont conçus pour supporter des activités moins formelles, basés sur l'analyse de la conversation et souvent dérivés de la théorie des actes de langage. Le Coordinator [Winograd 86] est un archétype de ces systèmes. L'environnement Co-Learn essaie d'associer les avantages de ces deux types de modèle de système, afin de pouvoir supporter les phases directives et non-directives qui constituent le processus d'apprentissage collaboratif.

1 - Les choix qui structurent l'environnement Co-Learn

1.1 - Le choix de métaphores

L'environnement est imprégné par deux métaphores : l'**outil** et la **pièce**. Elles sont utilisées comme métaphore de conception et comme métaphore pour l'interface utilisateur.

Comme le précise Alain Derycke [Derycke 95], l'idée de fournir une métaphore spatiale pour structurer les interactions de l'utilisateur n'est pas nouvelle. En effet, elle est présente dans la plupart des systèmes qui utilisent le concept de la salle de classe virtuelle et les hypermédia (pour exemple voir [Davies 93] et [Pozzi 94]). La métaphore de pièce apparaît aussi dans les systèmes qui veulent organiser l'interface utilisateur de telle façon qu'elle puisse contenir beaucoup d'informations, et permettre à l'utilisateur de gérer en parallèle plusieurs activités complexes [Card 87][Benford 93][Bly 93].

L'environnement de l'utilisateur est composé d'une collection d'outils conçus pour supporter une tâche de communication bien précise. Une pièce est le lieu où l'utilisateur trouve cette collection d'outils. Les outils présents dans une pièce définissent le caractère de celle-ci à deux niveaux différents :

- au niveau communication : les échanges sont synchrones ou asynchrones
- au niveau activité : l'activité que l'on peut mener dans cette pièce : cours magistral à distance, télé-tutorat, débat scientifique...

1.2 - L'environnement doit être "façonnable"

L'environnement est conçu pour supporter différents styles d'enseignements à distance, qui sont variés et nombreux. Pour ceci, Co-Learn doit être "façonnable", configurable, maléable, pouvoir être adapté à l'organisation dans laquelle il va être utilisé, et aussi aux utilisateurs par lesquels il va être manipulé.

Les métaphores citées précédemment vont nous aider fortement à atteindre cet objectif. On va pouvoir instancier plusieurs pièces de différents types afin de répondre aux besoins de l'organisme formateur. Dans chaque pièce, on pourra définir les outils qui vont s'y trouver et les utilisateurs potentiels. Et enfin, les outils vont pouvoir s'adapter aux besoins des utilisateurs en terme de comportement. Un outil est structuré en deux parties : l'interface qui prend en charge la communication avec l'utilisateur et une partie comportement qui définit le comportement de l'outil par rapport à la tâche à réaliser et à la structuration sociale de groupe (affectation de droits et de rôles aux participants).

Nous définissons pour l'environnement deux niveaux de malléabilité¹ :

- la **malléabilité structurelle** : pour répondre aux besoins d'une organisation, l'environnement Co-Learn permet de définir autant de pièces que l'on souhaite, et de placer dans ces différentes pièces les outils que l'on désire. Il n'est pas un

¹La malléabilité est la possibilité de s'adapter

environnement statique avec un nombre fixe de pièces avec des outils déjà affectés, mais la construction du centre de ressources est dynamique pour répondre aux besoins spécifiques d'une organisation [Hoogstoel 94] ;

- le **malléabilité fonctionnelle** : ceci est la capacité d'un outil à adapter son comportement aux besoins spécifiques d'un groupe. Ceci se traduit par exemple pour les outils partagés, par la définition de plusieurs politiques qui définissent chacune le partage entre les différents utilisateurs. Le nombre des politiques n'est pas figé. D'autres politiques peuvent être définies pour répondre à des besoins spécifiques, et venir enrichir l'environnement. Mais c'est aussi la possibilité pour un utilisateur de définir l'interface associée à cette outil. Dans le cas d'outil collectif, ceci se traduit par la possibilité offerte à l'utilisateur de définir comment il veut percevoir l'activité de l'autre, de définir l'aspect visuel de la rétroaction de groupe.

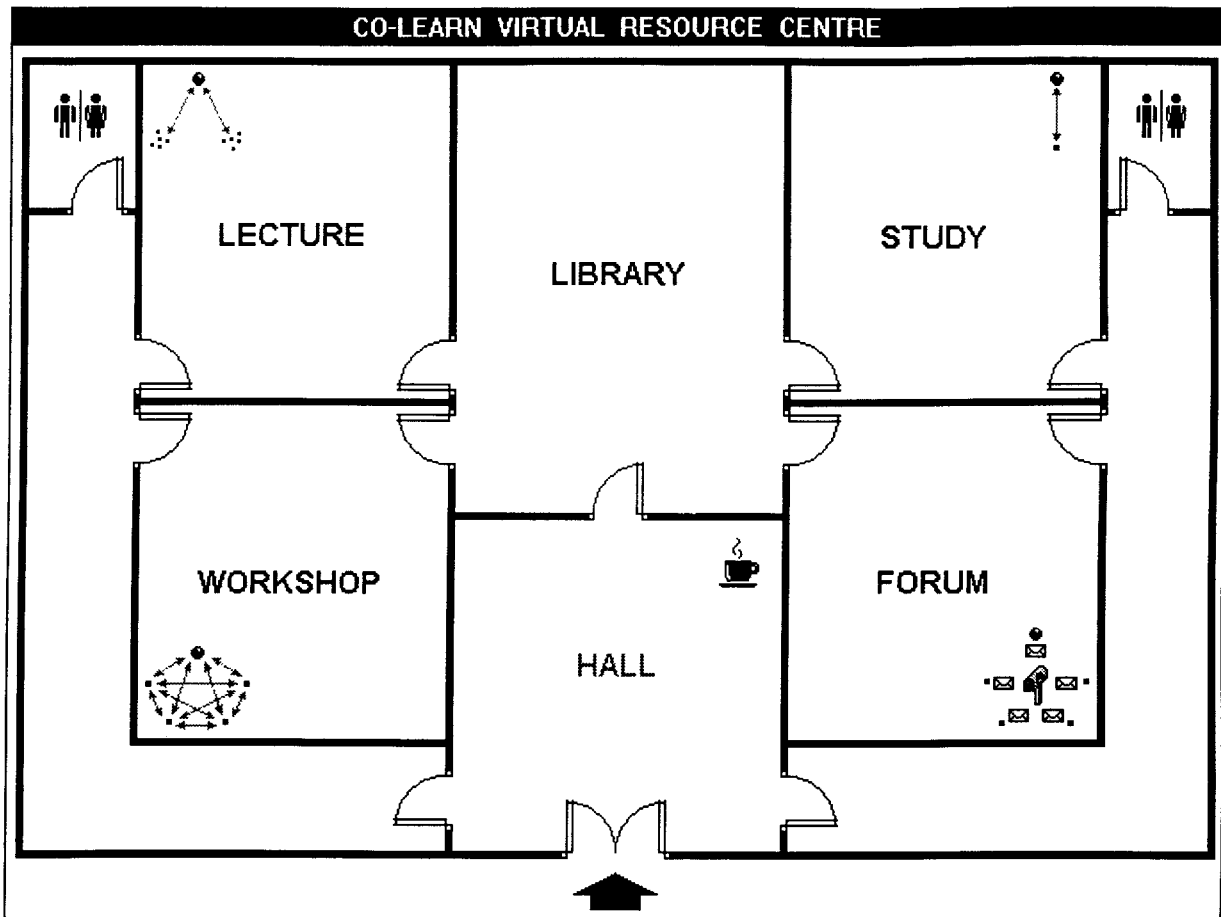
1.3 - Le choix d'une démarche

La démarche que nous avons adoptée est incrémentale. C'est à dire que l'on doit pouvoir facilement réutiliser des composants logiciels, créer de nouveaux outils, intégrer de nouveaux logiciels déjà existants. Ceci repose sur le choix d'une architecture Client-Serveur et d'une démarche d'analyse et de conception orienté objet.

2 - Le centre de ressources pédagogiques virtuel : Co-Learn

L'environnement Co-Learn peut être considéré comme un "centre de ressources pédagogiques virtuel" composé de différentes pièces dans lesquelles les apprenants et les enseignants vont pouvoir entrer pour y mener une activité d'enseignement : préparation de cours, suivre un travail dirigé, un débat scientifique, cours magistral...

L'accès à ce centre de ressources se fait après un contrôle d'identification et d'authentification, qui est réalisé par une connexion entre la station de l'utilisateur et le serveur Co-Learn (ceci doit être transparent pour les utilisateurs). Ensuite l'utilisateur peut accéder à l'ensemble des pièces (qui correspondent au cours) qui lui sont autorisées (choix sur une carte visualisant les types de pièce).



**Figure 1 : Interface de l'utilisateur prêt à choisir une pièce
(développement réalisé par Software de Base, partenaire du consortium Co-Learn)**

L'enseignant peut créer l'ensemble des pièces qui lui sont nécessaires pour assurer son enseignement. Il choisira l'un des quatre types de pièces² qui lui sont proposées (voir description ci-dessous), les personnes qu'il autorise à rentrer dans ces pièces, et enfin il décrira l'ensemble des ressources qui seront disponibles dans celles-ci (en terme d'outils). Le composant logiciel qui réalise ceci au niveau de la station utilisateur se nomme le Launcher [CoLearn 93b] en utilisant l'accès à une base de données à travers les services spécifiés par le composant GoldWing [CoLearn 93c], et le serveur d'activité. Le serveur d'activité définit le Modèle Organisationnel et prend en charge la malléabilité structurelle de l'environnement Co-Learn. Les travaux de Frédéric Hoogstoel portent sur la description et la réalisation de ce serveur d'activité et font l'objet d'une thèse en cours de rédaction [Hoogstoel 94].

Les types de pièce disponibles sont associés à des modes de communication (synchrone, asynchrone) et à la nature des interactions réalisables par les utilisateurs (les outils qui peuvent être mis à disposition de l'utilisateur).

²Nous reviendons dans le chapitre V sur cette spécialisation un peu arbitraire des pièces

La salle de télé-cours ou RTTI (Real Time TeleTeaching):

Ce prototype est une extension d'un produit commercial nommé télé-amphi [Seguin 92], et permet de réaliser un cours à distance (télé-cours) entre plusieurs sites différents. Le système repose sur la projection à distance de transparents qui ont été télé chargés avant le début du cours. L'enseignant peut contrôler la présentation des transparents sur les différents sites et réaliser un pointage à distance sur le document visualisé (désignation d'une partie du document). Les apprenants ont la possibilité de poser des questions en utilisant le canal sonore (les sites se trouvant connectés à une audioconférence), le tour de parole étant contrôlé par l'ordinateur sous le contrôle de l'enseignant. Pour plus de renseignements sur les fonctionnalités, vous pouvez consulter les documents de Delta [CoLearn 94b].

La salle de télé-assistance ou RTTA (Real Time Tele Assistance)

Ceci permet la connexion d'une station particulière dite tuteur à quatre autres stations dites élèves [CoLearn 93c]. La station tuteur est celle de l'enseignant et lui permet de visualiser l'état (aspect visuel) de l'une des quatre stations élèves afin d'aider un apprenant en difficulté. Un lien audio est immédiatement réalisé entre l'enseignant et l'apprenant qui est pris en charge. L'enseignant peut aussi manipuler à distance la station de l'apprenant, et inversement toutes manipulations de l'apprenant sont visualisées simultanément chez l'enseignant. Les apprenants ne forment un groupe que pour le tuteur. Il n'existe pas de possibilité de communication entre les apprenants. Les fonctionnalités de ce prototype sont très proches de celles fournies par des logiciels tels que CarbonCopy, NetRemote, Timbuktu. Son gros apport est d'être intégré dans l'environnement Co-Learn, et de fonctionner à travers le réseau RNIS.

Le forum ou ATC (Asynchronous TeleConferencing)

Ce prototype permet la conversation au sein d'un groupe de large échelle sous forme de messages textuels et sonores, avec possibilité de lier les messages avec des documents stockés dans une base de documents. L'intérêt de ce système est de fournir une structuration de la conversation en se reposant sur le typage des messages. Les concepts sous-jacents sont : la séparation entre l'espace où se réalisent les actions et l'espace de discussion, la décomposition de la conversation en échanges structurés. Cette structuration repose sur l'instantiation d'une classe d'échange qui définit la structuration de celui-ci (discussion libre, vote), et l'attribution d'un rôle à chaque participant impliqué dans l'échange afin de définir les actions possibles de l'utilisateur au sein de celui-ci [Viéville 95] [CoLearn 93f].

L'atelier ou RTTC (Real Time TeleConferencing)

Cette pièce est conçue pour permettre l'activité d'un faible nombre de participants probablement inférieur à 10, utilisant la métaphore de salle de réunion virtuelle. Chaque participant possède un espace privé où il peut concevoir des documents (avec un logiciel Windows) et ensuite les afficher dans un espace public en vue d'étayer une discussion au niveau du groupe présent dans la pièce. Les participants se trouvent *continuellement en audioconférence*, ce qui permet des discussions collectives. Une collection d'outils est à la disposition des utilisateurs afin de supporter l'activité de conversation : outils de vote, de pointage à distance, de gestion de temps. Les travaux relatés dans cette thèse sont directement liés à la réalisation de ce prototype.

La suite de ce chapitre est consacré à l'analyse et la conception du système de conférence temps réel RTTC.

II - LE PROTOTYPE DE CONFERENCE TEMPS REEL DE CO-LEARN : RTTC (REAL TIME TELECONFERENCING)

1 - L'expérience du laboratoire Trigone

Cette partie a pour objectif de décrire dans quel contexte a commencé la recherche relatée dans cette thèse.

Les travaux de Pierre Vilers, Alain Derycke et Claude Viéville avaient pour objet de créer un système de téléconférence adapté à l'enseignement [Derycke 90] [Derycke 93a] [Derycke 93d] [Viéville 90]. Le système qui a été réalisé permet l'enseignement des méthodes de planification en gestion de production, en utilisant la simulation de ces méthodes dans un atelier de production de pièces [Vilers 92].

Le prototype est utilisé pour un petit nombre de personnes se trouvant dans la même pièce, et disposant d'une station de travail. Cette configuration permet donc des conversations naturelles entre les participants.

Ce système relève du domaine du CSCW, dans lequel les méthodes d'analyse et de conception sont rares et peu éprouvées. C'est pourquoi, il a été choisi une démarche de conception reposant sur le maquettage et le prototypage incrémental, qui permettra d'évaluer le prototype au fur et à mesure de son avancement avec des utilisateurs potentiels (enseignement de ces méthodes). L'analyse du jeu de rôle, tel qu'il apparaît sous forme de jeu de société (utilisé en enseignement dans les IUT) fait apparaître plusieurs phases, qui ont été analysées en terme de tâches coopératives, en vue de construire des outils pour les supporter.

Il apparaît intéressant de décrire rapidement ces différentes phases afin de voir la nature des applications qui ont été réalisées.

Première phase : Présentation générale de la simulation

Cette phase consiste à expliquer les règles du jeu aux différents participants et à affecter un rôle (responsabilité d'atelier) à chaque utilisateur. Pour supporter cette phase, il fallait réaliser un espace de visualisation collectif respectant le concept de Wysiwis, qui permet d'enchaîner des transparents visualisés en même temps sur chaque station de travail. Cet espace est associé à un télépointeur qui permet de désigner des objets sur le transparent (curseur visible sur les stations de travail de chaque utilisateur). Un gestionnaire de conférence permet l'affectation de l'accès à l'espace Wysiwis (contrôle du passage de transparent) et le contrôle du télépointeur en mode FIFO (c'est le premier qui le demande qui peut l'utiliser).

Deuxième phase : Définition du plan de production

Elle permet au groupe de décider de la production hebdomadaire des différentes pièces à réaliser pour chaque poste, en fonction de données techniques (cadence des machines) et de données commerciales (prévisions sur les ventes).

Pour supporter cette phase, le prototype offre aux utilisateurs un tableur multi-utilisateur, qui permet de construire visuellement le plan de production, en prenant en compte les propositions du groupe. De plus les utilisateurs ont accès à une information commune (statistiques visualisées sous forme de graphe) lorsqu'ils le désirent. Lorsque le plan de production fait l'unanimité au sein du groupe, chaque utilisateur réalise le plan d'activité du poste dont il est responsable.

Troisième phase : Méthode de planification M.R.P.

Cette phase est une phase de simulation de la production suivant la méthode M.R.P. Elle permet de vérifier la validité du plan de production, en tenant compte d'événements extérieurs appelés incidents tels que panne d'une machine, grève du personnel ...

L'analyse de cette phase à permis de montrer la nécessité de partager de l'information (Représentation en temps réel de l'état des stocks). A partir de cette information chaque poste peut évoluer en fonction de son plan d'activité, les entrées et les sorties de stock ne nécessitant aucun accord des autres postes.

Quatrième phase : Méthode de planification Kanban

C'est une autre méthode de planification, qui n'a pas apporté de conclusion nouvelle. Seule l'interface utilisateur a changé, car l'activité d'un poste se traduit par des manipulations utilisateur directement sur l'espace de visualisation de l'information commune (manipulation de Kanban).

Conclusion :

Ce jeu de rôle est décomposé en différentes phases qui donnent lieu à la création d'applications coopératives Wysiwis (les deux premières phases), et d'applications coopératives dans lesquelles on partage de l'information et où l'activité du groupe demande un très faible niveau de coordination.

Ce système orienté tâche a permis d'appréhender la notion d'espace public permettant la visualisation collective d'une même information, permettant à un groupe de personnes de parler des mêmes objets. De plus, associer un télépointeur à cet espace permet plus facilement de référencer un objet. L'espace privé est affecté à

l'activité personnelle d'un membre du groupe, ayant peu de rapport avec l'activité collective et le système permet peu d'échange d'information entre les deux espaces.

L'utilisation de ce prototype a permis d'arriver aux hypothèses et aux questions suivantes :

- quel serait l'intérêt d'avoir une communication entre l'espace public et l'espace privé ?
- certains documents pourraient être réactifs et permettre aux utilisateurs une certaine liberté dans la découverte de l'information partagée, tout en gardant une visualisation collective du document. En d'autre terme, certaines parties du document ne pourraient elles pas évoluer sous l'action d'un utilisateur ?
- les applications telles que le tableur, et l'espace de visualisation public sont complètement liées à la tâche de gestion de production? Ne pourraient elles pas devenir des outils génériques indépendant de la tâche et être vus comme des outils supportant une activité de communication ?
- de même le contrôle de l'espace public et du télépointeur est unique. Le système doit offrir la possibilité à l'utilisateur de changer dynamiquement d'algorithme de contrôle. Oui, mais pour quel utilisateur? Cette réflexion nous amène à parler de rôle au sein du groupe, qui peut correspondre à un rôle dans l'organisation (par exemple l'enseignant), ou à un rôle qui structure l'activité du groupe : un rôle qui me donne des droits pour intervenir sur les outils partagés par le groupe.

Toutes ces réflexions et interrogations nous ont permis de spécifier certaines fonctionnalités du prototype RTTC. D'autres sont apparues au cours de l'utilisation de la plate-forme, grâce à notre démarche incrémentale et participative des utilisateurs potentiels.

2 - Le contenu d'une pièce de type RTTC

Identifier clairement les situations pédagogiques à associer aux différentes pièces présentes dans l'environnement Co-Learn est relativement difficile. Ceci est dû à la jeunesse des systèmes d'enseignement à distance médiatisés par les ordinateurs (CSCL : Computer Supported Collaborative Learning). Cette première liste de situations pédagogiques présentée ci-dessous, est décrite en terme de situation non médiatisée. L'association d'un outil à une situation pédagogique fera l'objet du second paragraphe. Cette association est à prendre pour le moment comme hypothèse. Un des objectifs (traité dans le chapitre V) de l'évaluation de l'environnement Co-Learn sera d'éprouver

l'utilisabilité des outils en mettant en place des scénarios réels avec de véritables apprenants.

2.1 - Les situations pédagogiques envisagées

Cette partie va nous permettre de décrire des situations d'enseignement qui sont susceptibles d'être supportées par l'environnement Co-Learn et plus précisément dans la pièce de conférence temps réel. Ces situations sont décrites en terme de situation non médiatisée, avec une indication sur le niveau d'interactivité et une description de la collaboration du groupe.

Présentation d'information

C'est une situation d'enseignement très traditionnelle, où un enseignant est chargé de transmettre un savoir, une pratique à un groupe de personnes, en utilisant toutes sortes de documents visuels. Le niveau d'interactivité est faible, et limité à des échanges sous forme de questions de la part des apprenants et des explications de la part de l'enseignant.

Démonstration

La notion de démonstration implique la maîtrise d'un modèle psychomoteur et à pour objet l'acquisition d'un processus par un apprenant ou un groupe, suivant l'exemple donné par l'enseignant réalisant le processus lui-même. Un exemple, tiré des situations réelles d'expérimentation de Co-Learn, est l'apprentissage de la calligraphie nécessaire pour l'écriture du Japonais.

Séminaires et discussions de groupe

Ces scénarios impliquent 6 à 12 personnes, dont l'objectif est de partager des informations, des idées, et d'en générer de nouvelles qui seront le résultat d'une discussion ou d'une négociation. Ils peuvent être précieux dans un processus d'apprentissage, car non seulement ils permettent de diagnostiquer et d'éclaircir des erreurs de compréhension et d'interprétation du contenu pédagogique, mais donne au participant une place active dans son processus d'acquisition des connaissances.

Les séminaires sont souvent des présentations plus ou moins formelles qui permettent à un participant de présenter son travail, suivi d'une discussion sous le contrôle d'un animateur (tuteur dans le monde pédagogique). Le niveau d'interactivité peut être très important au sein du groupe. Il donne lieu à utilisation simultanée de plusieurs canaux de communication (canal sonore, visuel), et nécessite souvent une gestion de l'activité du groupe en terme de coordination, soit par le groupe lui même ou

par un participant jouant le rôle d'animateur (pour plus d'information voir [CoLearn 92c]).

Les jeux, les simulations, les projets de conception, et les études de cas

La collaboration entre acteurs est souvent à la base du fonctionnement des petits groupes, ou des groupes de travail, dans lesquels le leader du groupe agit principalement en tant que régulateur et comme personne ressource. Dans les jeux et les simulations, chaque participant joue un rôle spécifique qui lui donne des droits et des devoirs (voir le jeu du kanban [Vilers 92]). Dans les études de cas, les participants disposent d'une information sur un sujet particulier. On leur demande de réaliser collectivement une évaluation (ou une explication) du problème, d'écrire un rapport, en suscitant une ou des démarches à suivre. Les projets de conception impliquent un groupe dans une tâche de spécification relative à la création d'un produit, d'un système...

Toutes ces situations peuvent être structurées en un certain nombre de phases, par exemple :

- une phase de remue-méninge ou de conceptualisation de la tâche à réaliser ;
- distribution ou choix de rôle et de sous tâches entre les membres du groupe ;
- travail individuel et discussion de groupe ;
- phase de coordination et de synthèse des résultats.

En conclusion, la description de ces scénarioss précise dans quel contexte et pour quel objectif l'environnement Co-Learn a été conçu. L'objectif du paragraphe suivant est de décrire les outils qui ont été conçus pour supporter ces différentes situations. Ceci permet aussi de décrire une partie des objectifs de l'évaluation de l'environnement : les outils tels qu'ils ont été conçus, assistent-ils bien les activités d'enseignement médiatisées.

2.2 - La description des outils

Nous allons décrire maintenant les différentes fonctionnalités d'un outil, en s'attachant à identifier dans quelle situation il peut être utilisé.

La figure 2 montre l'interface utilisateur lorsque celui-ci vient de rentrer dans la pièce. Nous pouvons y distinguer quelques outils (les représenter tous sur un même écran est difficile), et surtout la palette représentant la boîte à outils qui permet de sortir de la pièce, d'accéder aux outils privés, et surtout de naviguer entre les outils Cette

navigation est utile car les outils sont nombreux, et se réalise comme suit : lorsqu'un outil n'est pas visible, l'utilisateur clique sur l'icone qui lui est associé sur la palette et celui-ci apparaît en avant plan.

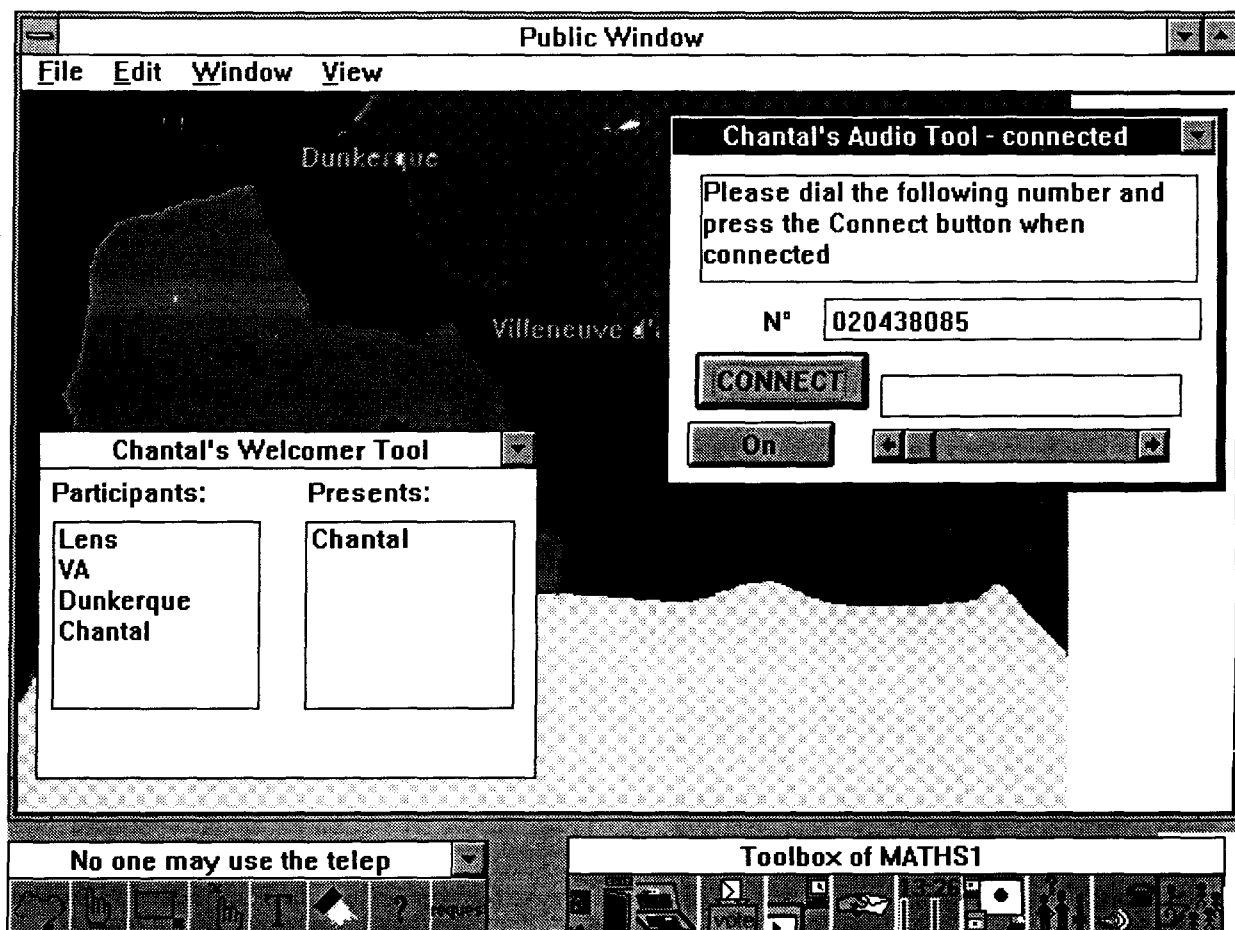


Figure 2 : L'utilisateur vient d'entrer dans la pièce

L'outil de visualisation collectif et le télépointeur

L'outil de visualisation collectif ou fenêtre publique gère l'affichage simultané du même document sur toutes les stations utilisateur. Il respecte le concept de Wysiwiw, pour que tous les utilisateurs voient exactement le même document. Dans une activité de communication, cet espace de visualisation va contenir des objets sur lesquels on va discuter : ce sont les objets de la conversation à distinguer de la conversation elle-même qui est l'ensemble des actes de langage (échangés par le canal sonore). Cet outil pourra être utilisé comme outil d'aide à la conversation nécessitant des supports visuels. Il permettra un certain niveau de couplage entre l'espace d'information et l'espace de conversation.

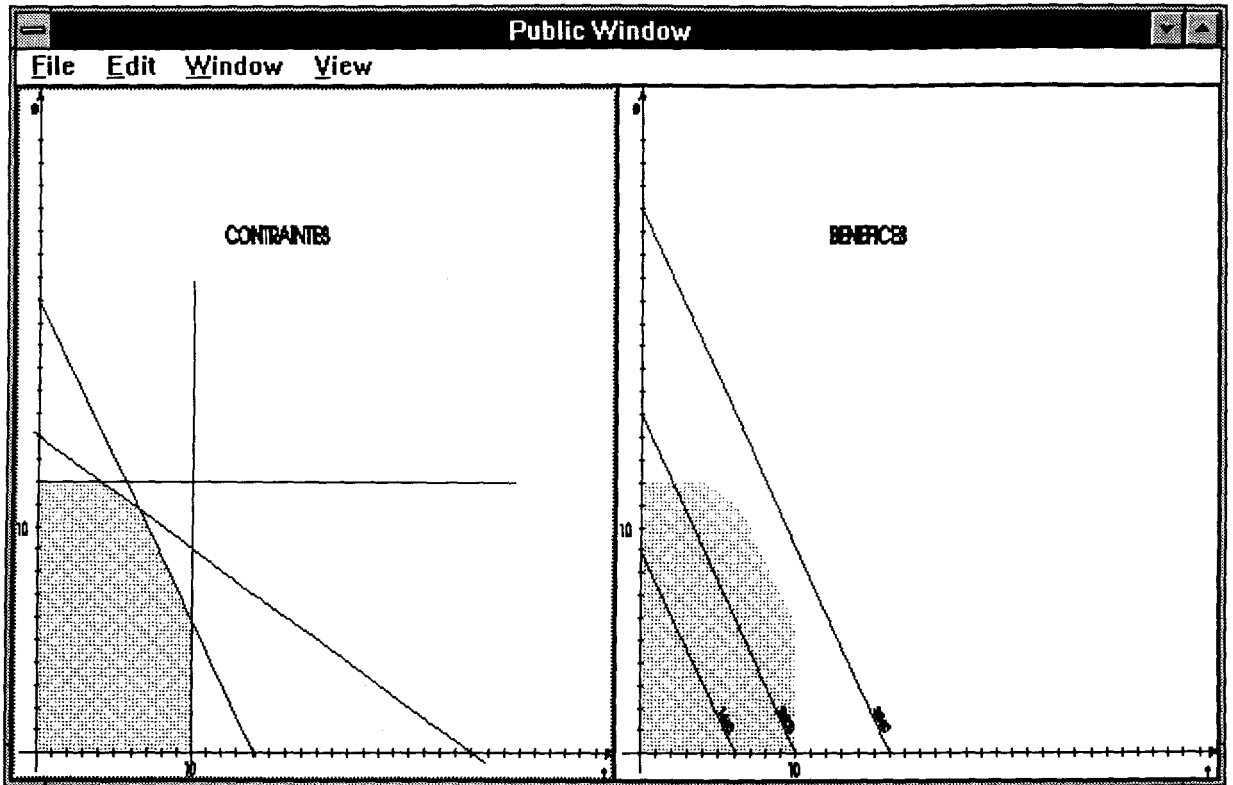


Figure 3 : L'outil de visualisation collectif ou fenêtre publique

Les documents sont de deux types et dépendent du moment de leur création :

- documents partagés préparés à l'avance : ils sont préparés avant la conférence par différents participants pour illustrer leur point de vue ou exprimer une idée. C'est exactement le type de document que l'enseignant prépare avant sa séance. Ils sont placés dans l'espace public donc accessibles pendant la conférence et peuvent être manipulés par chaque participant (sous réserve de posséder les droits suffisants).
- documents privés : ce sont les documents créés pendant la conférence par l'utilisateur en utilisant un outil privé (comme tableur, outils de dessin ...). L'outil privé appartient à l'environnement personnel de l'utilisateur : dans l'environnement Co-Learn, il correspond à tout logiciel fonctionnant dans l'environnement Windows respectant les formats du presse papier suivant : texte, bitmap, windows métafile et le format spécifique généré par l'éditeur Write³.

L'outil de visualisation collectif peut être vu comme un **presse papier collectif** [Pattel 93]. Au niveau de l'interface utilisateur chaque utilisateur pourra copier un document privé dans l'outil de visualisation. Celui-ci deviendra visible pour tous les utilisateurs, qui auront la possibilité de copier ce nouveau document dans leur monde

³Write est un éditeur de texte fourni en standard avec l'environnement Windows

privé. Ce presse papier collectif peut être vu comme un mécanisme d'échange d'information entre un utilisateur et le groupe.

L'outil de visualisation collectif est **multi-document**. C'est à dire que le groupe peut simultanément visualiser plusieurs documents à la fois (voir figure 3). Ceci afin de pouvoir supporter des tâches de comparaison de documents. Dans le milieu de l'enseignement ceci peut être adapté pour supporter une activité d'étude de cas, pour permettre la présentation de deux solutions d'un même problème, ou deux démarches différentes pour arriver à la solution d'un problème. Ceci est pour nous l'originalité de cet outil.

Si on considère l'outil de visualisation collectif comme un outil de conversation, il semble tout naturel de lui associer un outil de télépointage. Celui-ci va permettre de réaliser des références spatiales dans le monde des objets de la conversation (objets visibles dans l'outil de visualisation) en adéquation avec le contenu du discours. Cet outil va permettre d'articuler l'espace de conversation et l'espace d'information. Il permet de prendre en charge une partie de la méta-communication ou communication non verbale : la gestuelle. Il permet de déposer une marque, de déplacer un curseur visible par le groupe, de mettre du texte sur la fenêtre associée à l'outil de visualisation collectif.



Figure 4 : le télépointeur

L'association de l'outil de visualisation et du télépointeur est être interprété suivant un modèle multi-couche. Une couche pour visualiser les documents et une autre couche transparente pour gérer la gestuelle. L'outil de visualisation a une fonction de sauvegarde qui stocke une partie de la gestuelle dans le document. Cette fonction peu être interprétée comme une fusion entre les deux couches afin de rendre persistant une partie de la gestuelle.

L'outil de vote ou collecteur d'opinion

Cet outil peut être utilisé par le groupe pour prendre rapidement une décision. Dès la conception de cet outil on peut identifier trois phases dans le processus de vote. Une première qui correspond à définir la motion du vote. Celle-ci peut être un texte que l'initiateur du vote pourra préciser, ou le vote pourra faire référence à un sujet traité par un autre outil de la pièce, ou tout simplement par le canal sonore. Cette phase d'initiation se termine par le choix du mode de vote, qui définit le contenu d'un vote

(vote possible) et à la période au cours de laquelle le résultat est visible. La seconde phase correspond au processus de vote et dépend du mode choisi. Nous avons identifié quatre modes qui seraient intéressants pour nos situations d'enseignement :

- vote à main levée : correspond au tour de table, au cours duquel chaque participant répond par "oui" ou "non", où les résultats intermédiaires sont visibles en temps réel : liste des personnes qui ont voté "oui", liste des personnes qui ont voté "non". C'est l'initiateur du vote qui décide de la fin de celui-ci ;
- vote anonyme : il ressemble beaucoup au précédent mais la vision des résultats intermédiaires ne contient que le nombre de vote et non une liste de personnes ;
- vote à bulletin secret : ceci correspond à un tour de table, mais au cours duquel les résultats intermédiaires ne sont pas visibles de telle façon que le vote d'un participant ne puisse influencer les autres ;
- vote numérique : les utilisateurs choisissent une valeur entre 1 et 10.

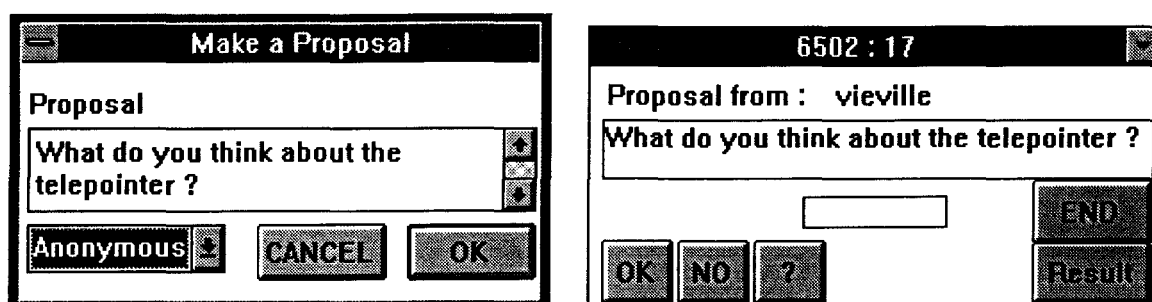


Figure 5 : deux phases de l'outil de vote

Le gestionnaire de temps

Cet outil est conçu pour rythmer temporellement une session d'utilisation de RTTC. Il fournit trois niveaux de service :

- il donne l'heure courante ;
- il permet de spécifier en début de session la durée de celle-ci. Chaque participant pourra alors visualiser le temps écoulé, ou le temps qu'il reste. Lorsque le temps est écoulé, l'outil prévient les utilisateurs en utilisant un son généré par la station et paramétrable ;
- il offre une fonction de chronomètre qui permettra de borner une tâche précise.

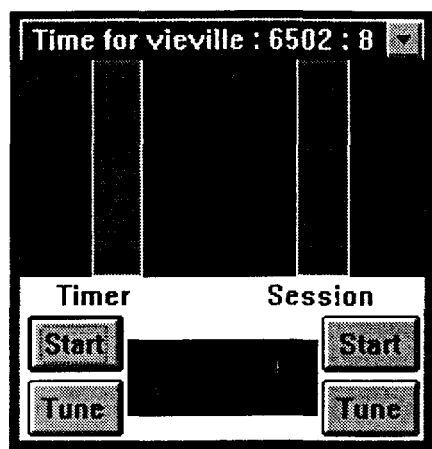


Figure 6 : Le gestionnaire de temps

Cet outil reste toujours sous le contrôle de l'utilisateur. Il ne contraint pas l'activité du groupe, mais donne des informations dont le traitement est laissé à l'initiative des utilisateurs.

L'outil de bavardage

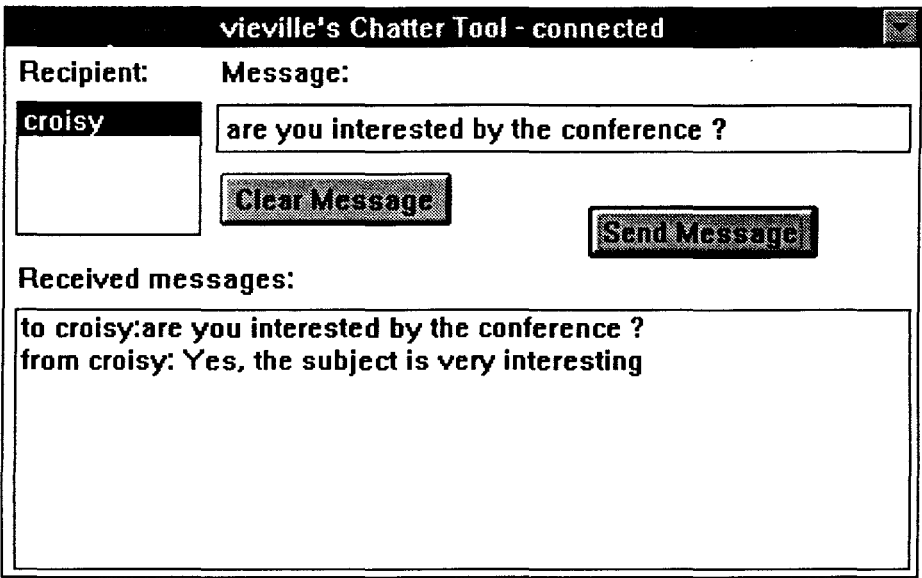


Figure 7 : L'outil de bavardage

Cet outil permet aux différents participants d'envoyer des petits messages textuels aux membres du groupe sans perturber l'activité de celui-ci. L'utilisateur tape son message, et choisit son ou ses destinataires. Il peut l'envoyer à un utilisateur particulier, l'outil fournissant la liste des personnes présentes dans la pièce, ou l'envoyer à toutes les personnes présentes dans la pièce. Lorsqu'un utilisateur reçoit un message, il est averti par un son généré par la station, ou l'affichage de la fenêtre associé à l'outil. Celle-ci permet aussi la visualisation des derniers messages envoyés. Dans le domaine

de l'enseignement, cet outil sera très utile pour que le médiateur ou le tuteur puisse s'adresser à un apprenant sans que les autres le sachent.

L'outil de gestion du canal sonore

Le canal sonore est primordial dans les conférences temps réel. Le service de base que l'on peut offrir est tout simplement une audioconférence, qui permet à chaque participant de parler quand il le veut et d'entendre la somme des voix de ses différents interlocuteurs (voir chapitre IV paragraphe Architecture physique)

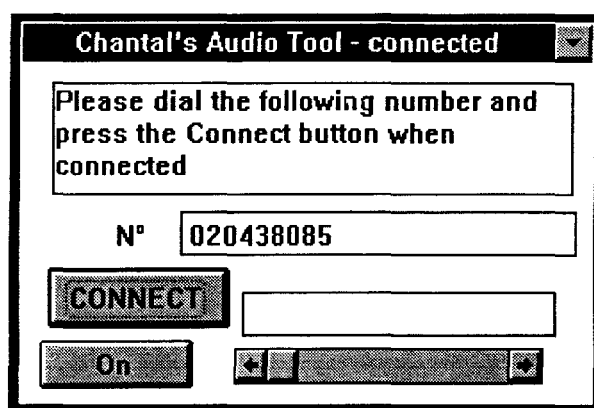


Figure 8 : l'outil de gestion du canal sonore

L'outil de gestion du canal sonore offre la possibilité à chaque utilisateur de contrôler son système de gestion de son local. Il peut régler le confort d'écoute, et couper ou mettre son micro en fonction. Ces fonctionnalités sont utiles lorsque l'utilisateur réalise une tâche à l'extérieur du système Co-Learn : il ne veut pas perturber l'audioconférence à cause de son environnement local (par exemple plusieurs utilisateurs présents devant la station, ou utilisation d'appareils bruyants dans son activité personnelle), être lui même dérangé pendant une activité individuelle au risque de s'exclure de l'activité du groupe.

L'objectif futur (non réalisé au moment de l'écriture de cette thèse) sur cet outil, est d'intégrer des fonctionnalités de détection de locuteur et de contrôle de droits de parole.

Le gestionnaire de conférence

Cet outil est utile afin de donner un point de vue général sur la pièce de conférence temps réel utilisé. Il donne de l'information sur la liste des personnes attendues, la liste des personnes présentes actuellement, la liste des outils présents dans la pièce, et finalement un état récapitulatif de l'utilisation, et du partage des outils parmi les participants.

Il offre aussi la possibilité de s'absenter temporairement de la conférence, en prévenant les autres utilisateurs afin que l'absence soit prise en compte pour ne pas perturber l'activité du groupe. Il permet aussi de connaître l'état du canal sonore d'un participant (un utilisateur a coupé son micro, ou mis son volume d'écoute à 0) et de le modifier. Certaines fonctionnalités de cet outil nécessite d'être contrôlées et sont à offrir à une personne qui aura un rôle particulier dans le groupe comme le tuteur dans un groupe d'apprentissage, car ces fonctionnalités utilisées consciemment permettent de mieux coordonner et structurer l'activité du groupe. Par exemple, pouvoir modifier le volume d'écoute des outils de gestion du son de tous les utilisateurs peut être un moyen efficace pour resynchroniser l'attention des différents participants.

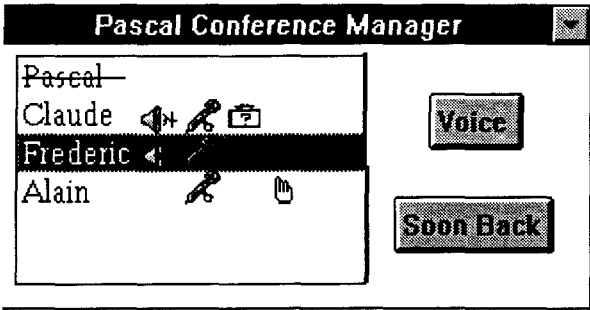


Figure 9 : le gestionnaire de conference

La figure 10 montre le gestionnaire de conférence simplifié, qui ne montre que la liste des personnes attendues et la liste des personnes présentes. Actuellement, seule cette version est disponible dans la pièce de conférence temps réel de Co-Learn.

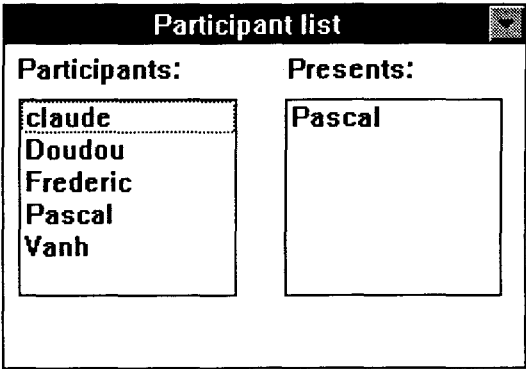


Figure 10 : Le gestionnaire de conférence simplifié

3 - L'architecture logicielle d'un outil

3.1 - Présentation de l'architecture logicielle

Un outil est composé de deux parties. La première est indépendante du matériel et même du système d'exploitation et du système de fenêtrage. Comme dans Liza [Gibbs 89], nous l'appelons **Noyau** (traduction de "Kernel"). Cette partie représente l'état d'un outil, gère les données qui lui sont associées. Par exemple dans un outil de vote, le noyau stocke la motion, le nombre de voix positives et négatives, les utilisateurs qui ont voté et l'algorithme de calcul du résultat, mais aucune information en relation avec l'interaction utilisateur.

Les interactions utilisateur sont gérées par la seconde partie de l'outil qui l'on peut appeler **Présentation**. Elle affiche les objets qui lui sont fournis par le noyau, et reçoit tous les messages générés par les organes d'entrées/sorties (nommés événements d'entrée).

Cette décomposition n'est pas sans rappeler les modèles qui décomposent un système interactif en différents composants. Le modèle de Seeheim [Coutaz 90 page 139] décompose un système interactif en trois composants : le noyau fonctionnel, le contrôleur et la présentation. Pour la spécification d'un outil, nous n'avons pas besoin d'un composant Contrôleur, nous avons préféré décrire les relations entre les composants Noyau et Présentation.

Dans la suite de ce chapitre, pour être consistant avec une approche modèle de système interactif, nous appellerons Noyau Fonctionnel (NF) la partie noyau de l'outil, et conserverons Présentation (P).

L'architecture d'un outil respecte le concept de client/serveur. Concevoir un outil sera tout d'abord d'identifier les composants NF, P et de décrire les services existant entre ceux-ci :

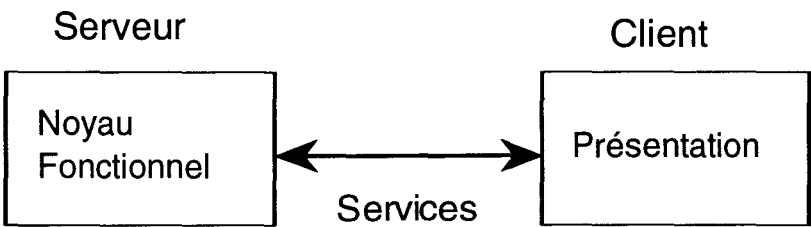


Figure 11 : Architecture Client/Serveur d'un outil

Spécifier un outil, va consister à identifier les données significatives de l'outil ainsi que les manipulations associées, la représentation visuelle et les différentes actions utilisateurs, et enfin, les services existants entre les deux composants.

3.2 - La dimension multi-utilisateur

Quand plusieurs utilisateurs utilisent un outil, le NF (serveur) est créé une seule fois, mais la Présentation (le client) est répliquée sur chaque site utilisateur. Le NF et la Présentation dialogue en échangeant un certain nombre de messages. Définir les services, c'est lister l'ensemble de ces messages. Ils peuvent être classés en deux catégories en fonction de la partie qui les génère. Une présentation envoie des messages d'événements d'entrée (traduction de "input events") alors que le NF diffuse des messages de requêtes d'affichage à toutes les présentations.

Prise en compte de la rétroaction de groupe

Les systèmes qui supportent la communication interpersonnelle doivent proposer les possibilités de visualiser pour tout le groupe certains événements qui vont correspondre aux interactions d'un utilisateur (voir chapitre I).

Cette rétroaction ne concerne pas le NF, mais les Présentations. Ceci nous amène à créer un nouveau type de message : les messages de rétroaction visuelle (Visual feedback). Par exemple, ce type de message va contribuer à réaliser des outils qui respectent l'abstraction du Wysiwis strict.

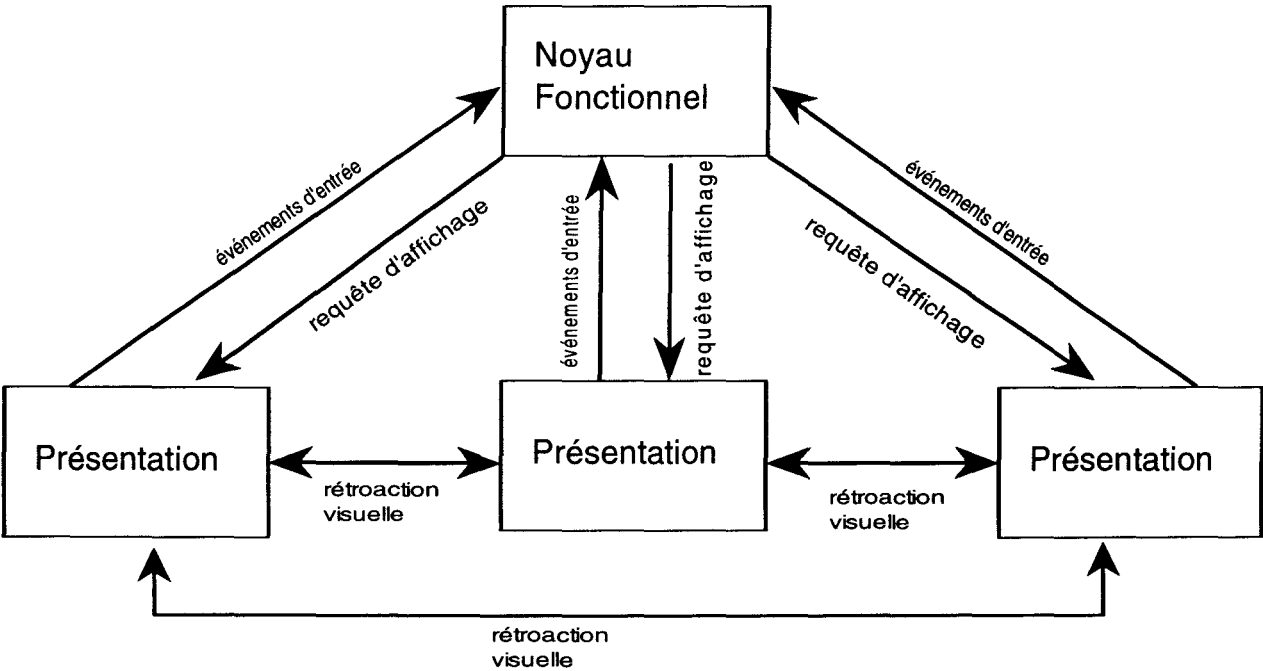


Figure 12 : Architecture logicielle multi-utilisateur d'un outil

Un outil est utilisé dans un contexte social

L'usage d'un outil ne peut pas être défini une fois pour toute. Son comportement peut être différent en fonction du contexte social dans lequel il est utilisé. Comment cet

outil peut supporter ses différents comportements qui vont dépendre du rôle des utilisateurs et des règles de fonctionnement du groupe. De plus ces mêmes règles peuvent évoluer au cours de l'utilisation de l'outil pour permettre à ce même groupe de mieux organiser son activité. Par exemple, à un outil de désignation, il faudra associer un algorithme de contrôle qui correspondra au protocole par lequel le groupe utilise et se partage l'outil. Prendre en compte le contexte social, défini au niveau des outils la malléabilité fonctionnelle de l'environnement.

Le contrôle du respect de ces règles de fonctionnement et des rôles au sein du groupe ne va pas être pris en compte au niveau de l'outil lui même, mais par un agent associé à l'outil (voir chapitre III). Cet agent contient les règles de fonctionnement, une définition des rôles et des règles de visualisation. Ces règles de visualisation permettent de décrire la rétroaction visuelle fournie à l'utilisateur, associé non pas à la sémantique de l'outil, mais à un état visuel qui traduit la phase de l'activité en fonctionnalité accessible à l'utilisateur.

Pour concevoir un outil, il est important dans un premier temps de ne pas prendre en compte ce contexte social. Toutes les fonctionnalités offertes à l'utilisateur sont accessibles.

En conclusion, nous proposons la spécification suivante pour décrire une représentation externe de l'outil. Dans celui-ci, on trouve le nom de l'outil, une description et les différents types de messages (événements d'entrée, requête d'affichage, et rétroaction visuelle)

Nom de l'outil	permet l'identification de l'outil
Description	permet d'expliquer les différentes fonctionnalités de l'outil
Événements d'entrée	liste des messages envoyés par la présentation au noyau fonctionnel
Requête d'affichage	liste des messages envoyés par le noyau fonctionnel permettant à la présentation de donner la représentation visuelle de l'outil
Rétroaction visuelle	liste des services qui permettent de transmettre les modifications visuelles d'une présentation aux autres présentations

3.3 - Exemple de spécification d'un outil

Une analyse complète des outils en suivant la méthode OOA⁴ de Shlaer & Meylor [Shlaer 88] permet d'arriver à une décomposition de l'outil en terme d'objet et d'identification de relation entre ces objets [CoLearn 92b]. Les deux tableaux suivants contiennent la spécification associée à l'outil de télépointage et à l'outil de vote que nous avons défini pour décrire une représentation externe de l'outil. Pour une bonne compréhension de ces tableaux, l'annexe II donne la spécification de l'outil de télépointage et de l'outil de vote suivant la méthode OOA.

L'outil de télépointage

Nom de l'outil	Télépointeur
Description	Il permet de désigner des objets visualisés avec l'outil de visualisation collectif
Événements d'entrée	- Créer une marque : l'utilisateur à l'aide de la souris désigne l'endroit où il veut placer une marque : marque statique, désignation rectangulaire, ou texte. - Supprimer un transparent : suppression de toutes les marques.
Requête d'affichage	- afficher une marque : affiche une marque à un endroit spécifique du transparent - effacer toutes les marques : efface toutes les marques présente sur le transparent
Rétroaction visuelle	- Déplacer une marque : l'utilisateur déplace une marque sur le transparent. Ce déplacement devient visible pour tous les utilisateurs.

⁴Méthode de spécification orientée objet choisie par le consortium Co-Learn

L'outil de vote

Nom de l'outil	Le collecteur d'opinion
Description	Il permet de prendre rapidement une décision et repose sur un mécanisme de vote
Événements d'entrée	- proposer un vote : ceci correspond à la définition d'une motion sur laquelle il faut voter - demander le résultat : correspond à une demande pour connaître l'état courant du vote - accord : l'utilisateur vote oui - désaccord : l'utilisateur vote non - indécis : l'utilisateur ne prend pas position - annulation : l'utilisateur annule sa réponse précédente - fin : la fin du vote est atteinte
Requête d'affichage	- résultat : affichage du résultat du vote
Rétroaction visuelle	

3.4 - La structuration et le contrôle au sein d'un outil

On parle de structuration lorsque l'activité associée à un outil peut être spécifiée à l'avance. Ceci est le cas pour un outil de vote, dans lequel le processus de communication est structuré et connu.

On parle de contrôle au sein d'un outil, lorsque l'accès à cet outil peut être conflictuel entre les différents utilisateurs. Il y a nécessité de définir une dimension de partage de l'outil, qui doit tenir compte de l'organisation du groupe et de son organisation sociale [McGrath 90]. Ceci se traduit en terme de collecticiel par la définition d'un algorithme de contrôle.

L'activité est un phénomène qui évolue dans le temps et qui permet de décrire les actions effectivement réalisées par l'utilisateur.

Pour spécifier la structuration et le contrôle au sein de nos outils collecticiels, nous avons adopté le modèle AMIGO [Pankoke 89]. La modèle Amigo a été conçu pour décrire les processus de communication au sein d'un groupe d'individu, le système de communication étant basé sur l'utilisation d'un réseau d'ordinateurs interconnectés (par exemple les messageries). Ce modèle est adapté à la modélisation des systèmes de communication qui nécessitent une régulation des échanges entre les différentes entités mises en oeuvre. Ceci signifie que le modèle identifie qui communique et avec qui, quand les échanges doivent ils avoir lieu et quels échanges. Cette structuration définit explicitement le terme activité pour le modèle Amigo. De plus, il donne les moyens de spécifier la coordination des échanges entre les entités communicantes.

Nous décrivons maintenant notre modèle d'analyse des activités de communication qui va nous permettre de spécifier la structuration de la communication et le contrôle au sein des outils de l'environnement Co-Learn. Nous illustrerons notre modèle en prenant un exemple significatif d'une activité de communication fortement structurée qui est le vote, et une activité qui nécessite le contrôle d'une ressource partagée comme le télépointeur.

Dans nos activités, chaque participant va jouer un **rôle** qui lui donne des droits et des devoirs. Par exemple, dans l'activité de vote on peut définir le rôle d'initiateur du vote, et le rôle de votant. Pour le télépointeur, nous identifions le rôle de gestionnaire et d'utilisateur. Le choix d'un rôle va déterminer les actions possibles pour l'utilisateur, et va déterminer les **messages** qui vont être échangés. L'activité gère la coordination entre tous les messages en provenance des différents rôles. Pour ceci elle possède des **règles** de coordination. Les règles sont déclenchées par la réception de messages spécifiques appelés déclencheurs. L'activation d'une règle va agir sur les rôles afin de modifier les messages qu'ils peuvent envoyer. Pour gérer la structuration de l'activité, nous définissons des **phases** qui décompose l'activité en entité qui détermine un état stable dans la communication, relativement indépendant des rôles : les phases sont connues de tous les rôles. Par exemple, dans l'outil de vote, nous identifions une phase d'initiation du vote qui permet à un utilisateur de définir la motion, et une phase de vote. Pour l'outil télépointeur, il y a une phase où il est utilisé et une autre où il est libre. L'activité est vue comme une succession de phases. Les changements de phase sont contrôlés par l'activité et se réalisent à la suite du traitement d'un message au travers de l'exécution d'une règle. Afin de prendre en compte, les changements au sein d'une phase qui ne concerne qu'un utilisateur d'un même rôle nous introduisons au sein même de la phase un **état**. L'état peut être vu comme un attribut de l'activité qui prend des valeurs qui sont utilisées par les règles afin de pouvoir identifier non par un rôle précis dans le système de communication, mais une instance unique de ce rôle, donc un utilisateur.

Pour résumer les différents composants du modèle sont :

- les **rôles** qui permettent d'affecter des droits et des devoirs aux utilisateurs ;
- la **phase** qui est une entité de décomposition pour laquelle on peut définir un état stable pour tous les rôles ;
- l'**état** qui apparaît comme un attribut de l'activité permet dans notre cas de considérer un exemplaire d'un rôle ;

- les **règles** qui structurent les échanges de **messages** et font évoluer l'activité en terme de phase et d'état.

Pour illustrer complètement ce modèle, nous développons maintenant complètement l'exemple du télépointeur, et précisons seulement les rôles et les phases pour l'outil de vote.

L'outil de vote :

Dans cette activité, nous identifions :

- deux rôles : le rôle de votant et le rôle d'initiateur ;
- trois phases : la phase d'initiation, la phase de vote, la phase de résultat.

Le télépointeur :

Nous identifions dans cette activité :

- deux rôles : celui du gestionnaire et celui d'utilisateur ;
- deux phases : la phase outil utilisé (1), et outil libre (2) ;
- deux états qui permettent de dissocier les actions possibles pour deux utilisateurs ayant le même rôle, l'un des deux possédant l'outil.

Les messages définis pour gérer la coordination au sein de l'activité sont les suivants :

- demander le contrôle de l'outil ;
- relâcher le contrôle ;
- prendre le contrôle ;
- donner le contrôle.

Les tables ci-dessous contiennent les messages qui peuvent être envoyés par le rôle en fonction de la phase courante et de l'activité courante pour le rôle du gestionnaire et le rôle de l'utilisateur.

		Etat	
		Je ne possède pas (1)	Je possède (2)
Phase	Outil libre (1)	- demander le contrôle - prendre le contrôle - donner le contrôle	Impossible
	Outil utilisé (2)	- prendre le contrôle - donner le contrôle	- donner le contrôle

Rôle du gestionnaire

		Etat	
		Je ne possède pas (1)	Je possède (2)
Phase	Outil libre (1)	- demanderContrôle	Impossible
	Outil utilisé (2)	- aucun message	- aucun message

Rôle de l'utilisateur

Les règles que l'on peut définir sont les suivantes :

Règle 1 : Le message demanderContrôle(Y) permet à l'utilisateur Y de demander le contrôle de l'outil. Si la phase courante est la phase 1 (outil libre) alors on passe en phase 2 et l'utilisateur Y devient l'utilisateur courant. L'utilisateur courant passe en état 2 et les autres en état 1

- Si (demanderContrôle (Y)) alors

si (phaseCourante == 1) alors

PhaseCourante = 2 ;

utilisateurCourant=Y;

pour utilisateurCourant état(2) sinon état(1);

finSi

finSi

Règle 2 : Le message prendreContrôle(Y) permet à un utilisateur Y de prendre le contrôle de l'outil s'il est connu comme gestionnaire de l'outil Y

```
- Si (prendreContrôle(Y)) alors      si gestionnaire(Y) alors
                                     PhaseCourante = 2 ;
                                     utilisateurCourant=Y;
                                     pour utilisateurCourant état(2) sinon état(1);
                                     finSi
finSi
```

Règle 3 : Le message donnerContrôle permet à l'utilisateur Y de donner le contrôle de l'outil à Z. Si l'outil est libre alors on donne le contrôle à Z. Si l'outil est utilisé par X qui n'est pas un gestionnaire on le contrôle à Z.

```
Si (donnerContrôle(Y,Z)) alors      si (phaseCourante == 1) alors
                                     PhaseCourante = 2 ;
                                     utilisateurCourant=Z;
                                     pour utilisateurCourant état(2) sinon état(1);
                                     sinon si (!gestionnaire(X)) alors
                                     utilisateurCourant=Z;
                                     pour utilisateurCourant état(2)
                                     sinon état(1);
                                     finSi
                                     finSi
finSi
```

Exprimer comme nous venons de le faire les tableaux représentant les rôles par les messages échangés en fonction de la phase et de l'état, associé à la définition des règles, définit le **mode** de fonctionnement de l'outil. Durant la phase d'analyse, plusieurs modes de fonctionnement pourront être définis, qui caractériseront la malléabilité fonctionnelle de l'outil.

Chapitre IV

Conception et Réalisation

du prototype de

conférence temps réel de Co-Learn : RTTC

Ce chapitre décrit la réalisation du prototype de conférence temps réel de Co-Learn. Une première partie décrit la plate-forme technique et logicielle utilisée, en insistant sur les problèmes significatifs résolus. Une deuxième partie relative à la mise en oeuvre de ce prototype au sein de l'institut CUEEP, explicite les exigences à respecter pour permettre l'évaluation du prototype en situation réelle. La dernière partie décrit le déroulement d'une séance d'évaluation, une description sommaire du protocole d'évaluation ainsi que quelques résultats sur l'évaluation technique et sur l'utilisabilité du système.

Ce qui est décrit dans ce chapitre n'engage en rien les différents partenaires impliqués dans le projet Co-Learn. Certains aspects du projet ont été adaptés, ou complétés pour répondre à une problématique de recherche cadrée dans les activités du laboratoire Trigone.

I - ARCHITECTURE PHYSIQUE MISE EN OEUVRE POUR CO-LEARN

La figure 1 montre l'ensemble des composants physiques qui définissent l'architecture de l'environnement Co-Learn, donc utilisée pour la réalisation du prototype de téléconférence temps réel (RTTC). Cette architecture est adaptée pour la présentation de RTTC dans le fait ou elle inclut des composants physiques spécifiques pour ce prototype : le micro-casque connecté directement sur la carte son et le câble d'interconnexion entre la carte RNIS et la carte son (décrit dans la suite du document).

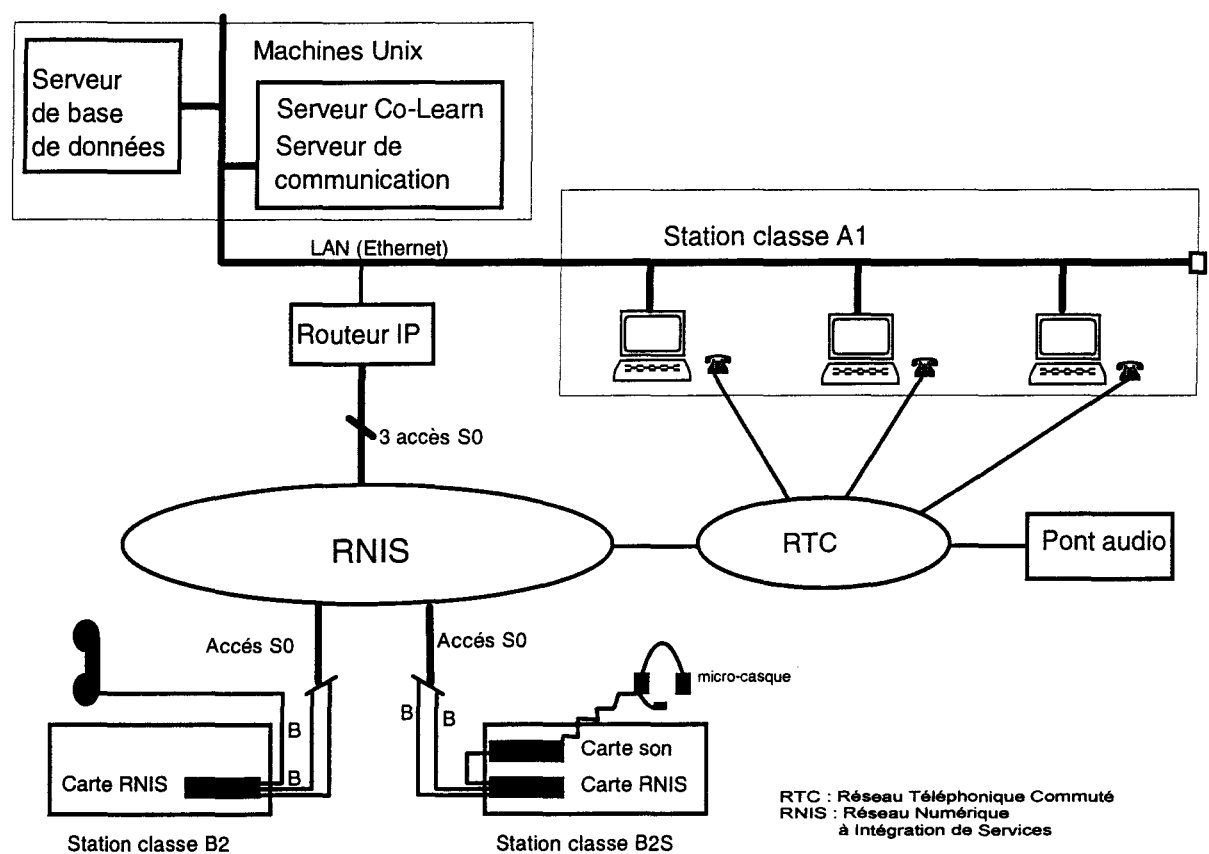


Figure 1 : Architecture physique pour RTTC

Il faut distinguer dans cette architecture trois types de composants [Viéville 94b] :

- un composant serveur¹ qui regroupe des composants standards du marché (Physiques et logiciels) et des composants spécifiques logiciels développés par le consortium² Co-Learn. Ce composant se constitue d'un ensemble de stations Unix connectées par Ethernet, d'un pont audio numérique et d'un routeur IP. Les stations Unix accueillent : le serveur d'activité qui permet la définition des rôles des utilisateurs et encore d'autres fonctions d'administration et de gestion d'une organisation (voir [Hoogstoel 94]) développées en utilisant la base de données orientée objet Gemstone

1 Le terme serveur est à interpréter comme un composant qui fournit des services.
2 L'ensemble des partenaires impliqués dans le projet Co-Learn.

(de Servio Logic Corporation) et son interface SmallTalk80 ; un gestionnaire de communication (composant spécifique) qui prend en charge la communication entre les sites client en offrant des mécanismes de diffusion et de sérialisation de messages. Le pont audio est actuellement un composant indépendant qui offre un service d'audioconférence. Le routeur IP permet la connexion entre le réseau RNIS³ et le réseau éthernet.

- un composant de service d'accès qui définit l'accès physique de la station cliente : le réseau de transmission cible est le réseau RNIS. Chaque station doit posséder un accès RNIS de type S0⁴ pour communiquer avec le serveur Co-Learn et les autres stations. Le service d'accès doit fournir deux types d'accès : un accès au serveur Co-Learn et un autre pour le pont audio. L'accès au serveur Co-Learn se fait au travers d'un lien IP⁵. Toute station possédant un lien IP avec le réseau éthernet sur lequel se trouve le serveur Co-Learn pourra accéder à celui-ci. L'accès au pont audio peut se faire soit par l'intermédiaire du RNIS ou bien par le réseau téléphonique classique (RTC : réseau téléphonique commuté). Actuellement, nous ne supportons pas l'accès au serveur Co-Learn et au pont audio sur le même canal : nous utilisons un canal B pour accéder au serveur, et le deuxième canal B pour l'accès au pont audio pour une station de classe B2S (voir ci-dessous).

- un composant client qui correspond au matériel disponible pour l'utilisateur : il s'agit d'un PC sous Windows 3.1, muni d'une carte RNIS, optionnellement d'une carte son aux normes MPC⁶ accompagnée d'un micro-casque ou d'un micro plus enceintes acoustiques. La station cliente peut accéder au serveur Co-Learn soit par l'intermédiaire du RNIS ou d'un réseau local. Pour faciliter la compréhension, nous avons créé une classification [Viéville 94b] des stations Co-Learn (stations qui accèdent au serveur) en fonction de leur équipement et du mode d'accès :

- A : accès au serveur Co-Learn par l'intermédiaire d'une carte éthernet
- B : accès au serveur Co-Learn par l'intermédiaire d'une carte RNIS
- 1 : accès au pont audio avec un téléphone extérieur
- 2 : accès au pont audio directement par le PC

3 RNIS Réseau Numérique à Intégration de Services.

4 Un accès S0 est composé de trois canaux, deux à 64 Kbits par seconde nommé canal B et un canal D à 16Kbits par seconde.

5 IP Internet Protocol : protocole de connection inter réseaux.

6 MPC Multimédia PC : norme qui standardise l'accès à des périphériques multimédia pour les machines de type PC.

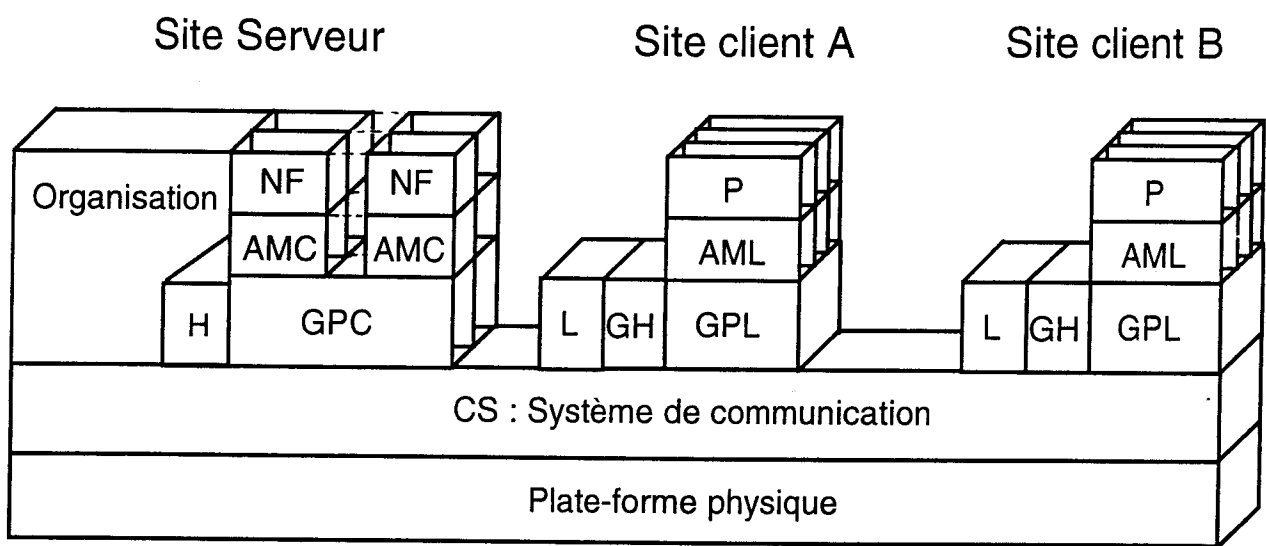
- S : le PC dispose d'une carte son aux normes MPC qui fournit des possibilités de connectique avec la carte RNIS. Ceci signifie que l'on pourra contrôler le son par l'intermédiaire d'outils informatiques.

Une classe est définie par la combinaison suivante "A ou B + 1 ou 2 [+ S]" (les [] signifie le caractère optionnel). Par exemple une station de la classe B2S est un PC fonctionnant sous Windows équipé d'une carte RNIS qui accède simultanément au serveur Co-Learn et à l'audio conférence, et qui peut manipuler le son provenant de l'audio conférence à partir des outils Windows. Ceci nécessite la connexion physique entre la carte RNIS et la carte son [Viéville 94b].

II - ARCHITECTURE LOGICIELLE

L'objectif de cette partie est de décrire l'ensemble des composants logiciels qui ont été réalisés pour concevoir le prototype RTTC. Certains de ces composants sont génériques à l'environnement Co-Learn, donc utilisés par les quatre prototypes. Nous décrivons plus précisément les composants associés à RTTC qui correspondent au travail réalisé pour cette thèse (même si nous avons participé à la spécification des autres composants). Nous nous contentons de décrire les autres composants en terme de services utilisés par les composants de RTTC, ou bien sous forme d'une description lorsqu'elle est nécessaire pour la compréhension de la réalisation du prototype.

La figure 2 montre l'ensemble des composants mis en oeuvre dans la réalisation. L'utilisation d'un schéma en trois dimensions nous permet d'exprimer l'existence de plusieurs exemplaires d'un même composant.



HC : Hall Central
GPC : Gestionnaire de Pièce Central
AMC : Agent de Médiation Central
NF : Noyau Fonctionnel

L : Launcher
HL : Hall Local
GPL : Gestionnaire de Pièce Local
AML : Agent de Médiation Local
P : Présentation

Figure 2 : Architecture logicielle de RTTC

Le composant Organisation

Le composant organisation va contenir de l'information sur la structuration de l'organisation dans laquelle l'environnement Co-Learn est utilisé. Il représente la partie administrative de l'organisation qui stocke des informations sur : les utilisateurs, les

pièces qui constituent l'organisation, les personnes impliquées dans l'activité d'une pièce avec leur rôle institutionnel (rôle dans l'organisation), et leur rôle par rapport aux différents outils qui vont être présent dans la pièce. La description des activités associées à un outil en terme de médiation (ce sont des classes génériques qui vont permettre de créer des instances d'outils [Hoogstoel 94]).

Le système de communication (SC)

Le service de communication [CoLearn 93] offre des mécanismes de transfert de données entre les différents composants de l'architecture Co-Learn. Il offre un niveau d'abstraction permettant une complète indépendance des applications par rapport au type de support physique utilisé.

Il offre les fonctionnalités suivantes :

- connexion d'un site client au site serveur ;
- permet l'envoi de données entre le site serveur et un site client spécifique, ou vers tous les sites client. La métaphore de Pièce a aussi été utilisée dans le SC pour organiser les liens de communication. C'est à dire que le site serveur au travers de son SC peut envoyer des données à tous les sites client dont l'utilisateur est considéré comme présent dans la pièce. Ceci permet au SC de gérer les connexions dans plusieurs pièces simultanément ;
- permet l'envoi de données entre deux sites clients; entre un site client et tous les autres sites clients. C'est à dire qu'il offre un mécanisme de diffusion de données (multicast) au sein d'une pièce ;
- il offre aussi des mécanismes plus évolués de téléchargement d'informations plus complexes, tel que des documents. Ceci est défini pour permettre un téléchargement de document sur les sites clients, qui peuvent être utilisés par les applications de l'environnement Co-Learn. Nous n'avons actuellement pas utilisé ces fonctionnalités dans RTTC pour cause de disponibilité de la version et des problèmes de fiabilité et de performances.

Le SC fournit un mécanisme de sérialisation dans la diffusion de données. Ce mécanisme est implicite car il utilise un processus centralisé sur une machine Unix qui réalise le mécanisme de diffusion, en diffusant à tous les sites les données avant de traiter les événements en réception suivant. Ceci est très important car le SC garantit que les données arrivent toutes dans le même ordre sur tous les sites client.

Le Launcher

Ce composant se trouve sur chaque site client. Il prend en charge l'aspect intégration des quatre prototypes. Il prend en charge l'authentification des utilisateurs lors de leur entrée en session, la navigation qui permet à l'utilisateur de choisir la pièce où il veut se rendre, la gestion administrative d'une organisation : enregistrement des utilisateurs, création des cours et des pièces [CoLearn93b]. Une session Co-Learn correspond au temps au cours duquel l'utilisateur est présent dans l'environnement Co-Learn.

Le Hall Central (HC) et le Hall Local (HL)

Ces deux composants sont sollicités lorsque l'utilisateur veut utiliser le système. Le Hall central peut être considéré comme un processus toujours en activité qui scrute les demandes d'accès au système qui lui sont envoyés par un HL à la suite de manipulation d'un utilisateur par l'intermédiaire du Launcher. Ce composant est exécuté sur demande de l'utilisateur sur la station cliente. Ceci se traduit en utilisant les concepts de l'environnement Co-Learn comme suit : une personne qui veut entrer dans l'organisation doit le demander par l'intermédiaire d'un HL, qui après avoir contacté le HC va lui spécifier quelles sont les salles auxquelles il a accès. Lorsqu'il aura choisi la salle où il veut aller, l'ensemble des composants logiciels présents sur la figure 2 seront créés. C'est à dire que le HL va créer le Gestionnaire de Pièce Local GPL, qui va communiquer avec le Gestionnaire de Pièce Central GPC afin de créer un exemplaire de chaque outil (Agent de Médiation + Présentation) sur le site client. Ceci n'est pas sans rappeler l'architecture du mécanisme d'entrée en conférence de Groupkit [Roseman 92], à la différence prêt que les utilisateurs sont référencés en tant qu'objet social dans Co-Learn (Nom, Prénom, adresse, photo ...). De plus dans l'environnement Co-Learn, on pourrait prévoir des règles de démarrage : par exemple sur un seuil de participant minimum.

Lorsque deux utilisateurs se trouvent dans la même pièce, la figure 2 montre l'ensemble des composants logiciels présents ainsi que leur distribution.

Ces deux composants sont fortement impliqués dans la gestion d'une session. C'est à dire qu'ils vont permettre l'entrée en session, la sortie, le changement de pièce. La figures 3 illustre l'aspect dynamique de la répartition et de l'instanciation des composants logiciels.

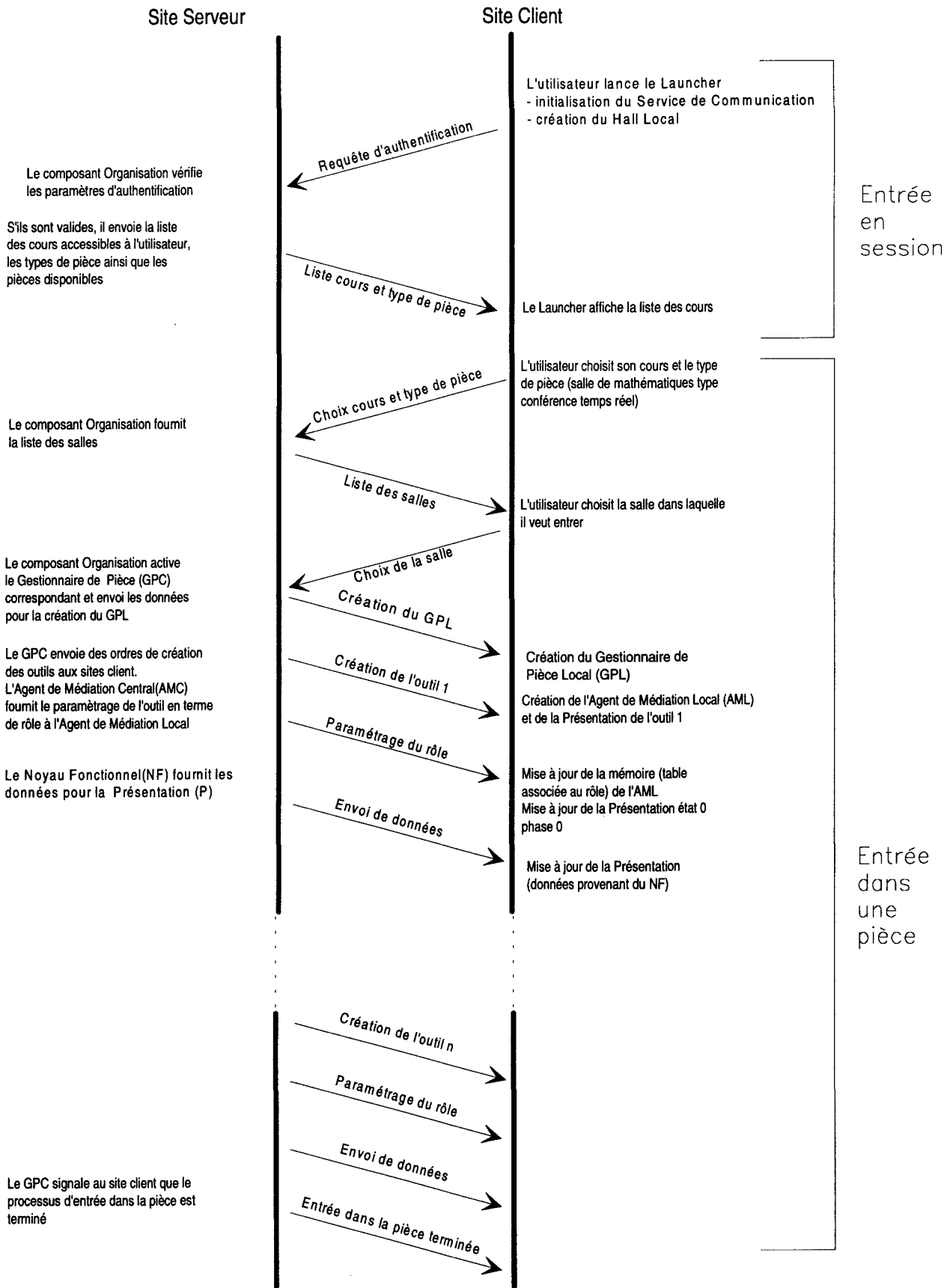


Figure 3 : Entrée en session et entrée dans une pièce

Pour quitter une session, il faut tout d'abord sortir de la pièce, ce qui a pour conséquence de détruire le GPL et tous ces composants supérieurs. Quitter définitivement la session se fait en enfonçant un bouton du Launcher, qui a pour conséquence de détruire le Hall et le Launcher lui même.

Le Gestionnaire de Pièce Local (GPL)

La pièce est un lieu où l'on trouve des outils. Dans la figure 2, l'outil sur le site client représente les deux composants Présentation P, et l'Agent de Médiation Local (AML) et constitue l'application collecticiel. Le GPL va offrir un niveau d'abstraction supplémentaire par rapport au SC, en proposant un niveau d'envoi de message. C'est à dire que le GPL va permettre aux entités qui l'utilisent d'envoyer un message à la partie centrale d'un outil appelé Noyau fonctionnel, ou de diffuser un message aux parties correspondantes de l'outil sur tous les sites (multicast). Pour ceci il gère des identifiants d'outil qui lui sont fournis par le Gestionnaire de Pièce Central GPC lors de l'entrée de l'utilisateur dans la pièce. Cet identifiant est ajouté aux messages lors de l'envoi de celui-ci par le GPL, et permet à ce même GPL dans sa fonction de réception de diffuser le message vers l'outil concerné (voir figure 4).

La relation de causalité entre les différents messages est assurée par la fonction de diffusion offerte par le SC. Par relation de causalité, nous parlons du fait que les messages sont ordonnés de la même façon sur toutes les stations. La relation d'ordre peut s'exprimer ainsi : si M1 avant M2, alors M1 arrivera avant M2.

La dernière fonctionnalité du GPL est de permettre la création des outils. Par l'intermédiaire d'appel de fonction système de l'environnement Windows, il lance les outils avec les paramètres nécessaires à leur exploitation dans l'environnement Co-Learn (principalement les paramètres nécessaires à l'envoi de message). Ceci suppose que les fichiers exécutables de chaque outil sont disponibles sur le site client.

En résumé, la fonction principale du GPL est la création des outils et le routage des messages (figure 4).

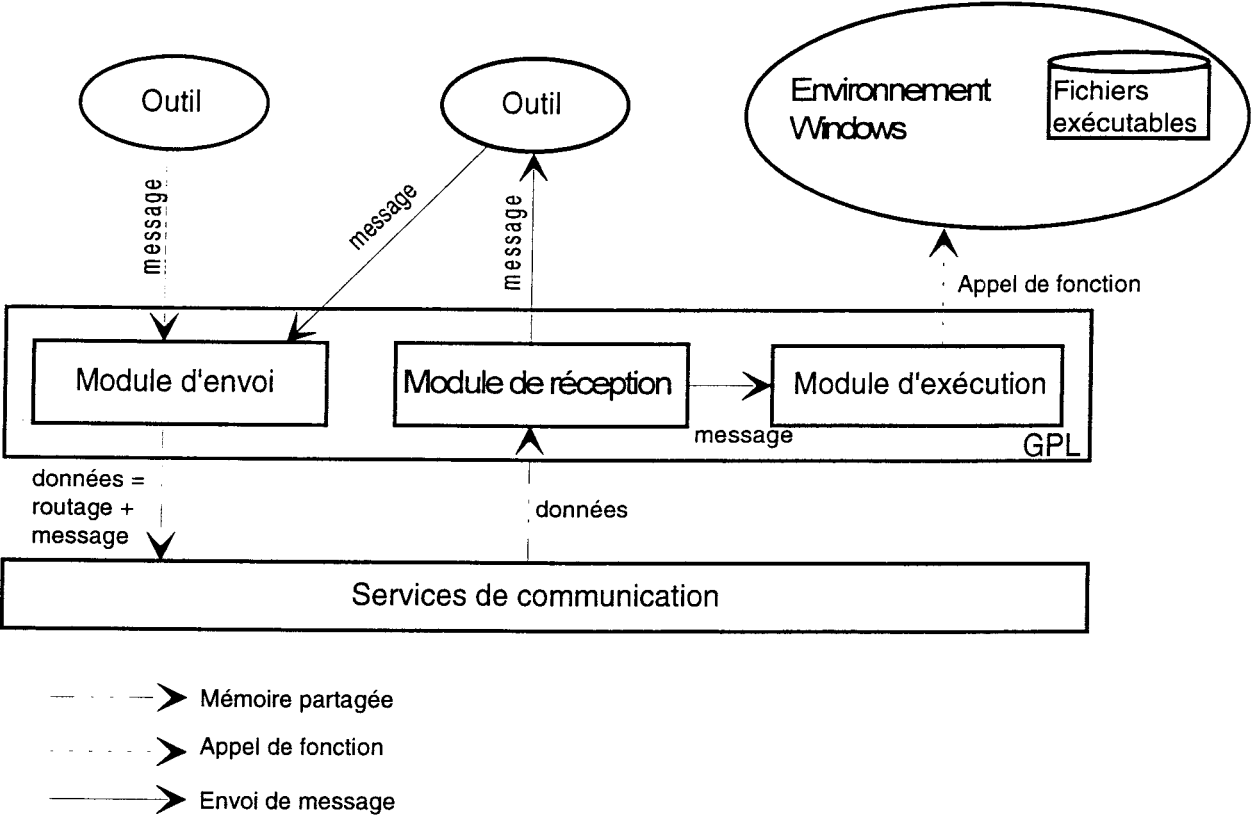


Figure 4 : Le gestionnaire de Pièce Local

Le gestionnaire de Pièce Central

Il est capable de changer les agents de médiation associés à une activité. Ceci est possible si la Présentation offre les moyens à l'utilisateur de le faire. Les agents de médiation sont en nombre fini, et créés à partir de classes d'activité génériques [Hoogstoel 94] qui peuvent être instanciées afin de pouvoir utiliser le même outil dans la même organisation dans des pièces différentes. L'utilisation d'un environnement orienté objet (SmallTalk80), nous permettra dans l'avenir de compléter ces classes génériques à partir des conclusions de l'évaluation du prototype. Comme son homologue sur les sites clients, il utilise les identifiants d'outil pour réaliser le démultiplexage des messages reçus.

L'agent de médiation local

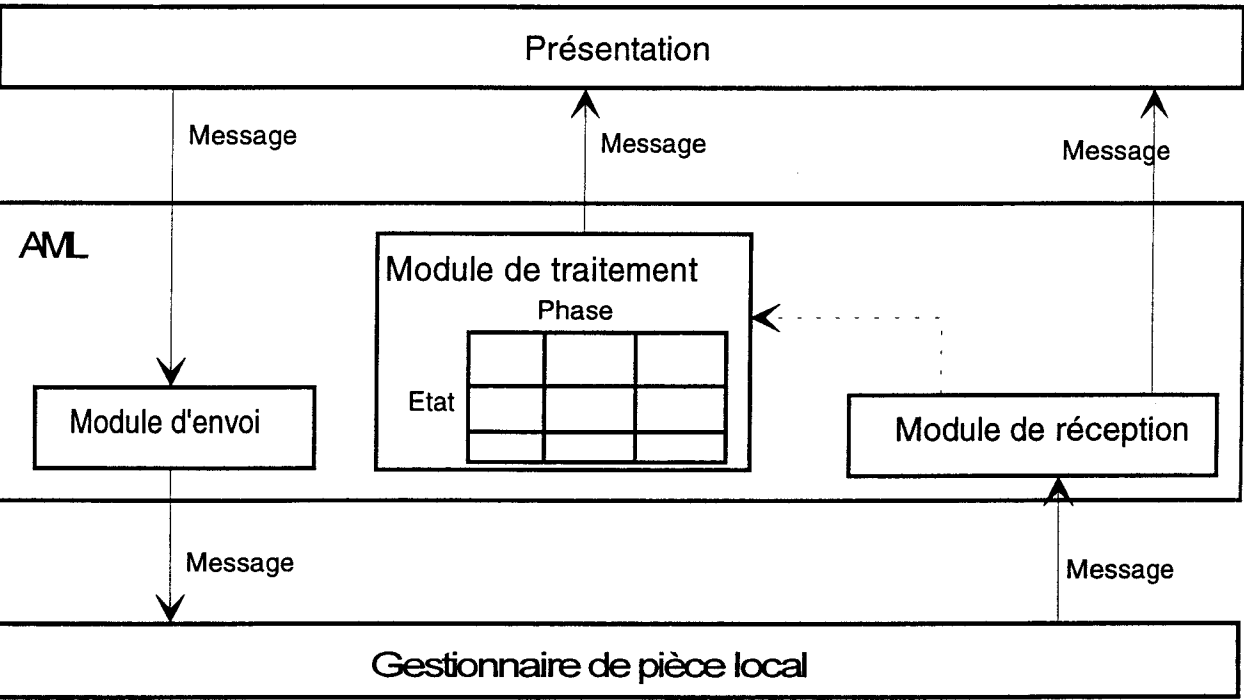
Tout d'abord, il peut être considéré comme un véhicule pour les messages . C'est à dire qu'il fournit sans interprétation les messages qui lui sont envoyés par la Présentation au GPL. Nous rappelons que les actions utilisateurs sont toujours réalisables, car nous avons adapté la politique suivante : la Présentation montre à l'utilisateur que les actions qui lui sont autorisées et qui aboutiront. C'est pour cette raison que les messages en provenance de la Présentation sont diffusés directement au GPL, sans traitement préalable. Ce choix est arbitraire. Un des objectifs du prochain

prototype sera de pouvoir modifier ce comportement, c'est à dire de pouvoir passé d'un mode à priori à un mode à posteriori (voir chapitre I). Ceci nous permettra de définir une nouvelle dimension, à ajouter à la caractéristique de système flexible et configurable.

Les messages en provenance du GPL sont de deux types :

- le premier regroupe les messages pour la Présentation : les messages liés à la rétroaction et ceux envoyés par la présentation ou le Noyau Fonctionnel. Ils ne nécessitent aucun traitement ;
- les deuxièmes sont des messages liés au changement d'état ou de phase dans l'activité, et sont envoyés par l'Agent Central de Médiation.

Ce deuxième type de message est traité par l'AML : il contient en paramètre la phase et l'état courant, ce qui permet en utilisant la table mémorisée par l'AML, de générer les messages nécessaires à la Présentation pour qu'elle s'adapte à l'état courant de l'activité. Ces messages vont être associés à des modifications des objets d'interface contenus dans la Présentation.



-> Appel de fonction
- > Envoi de message

Figure 5 : L'agent de médiation local

La figure 6 montre le contenu de la table contenue dans le module de traitement de l'agent de médiation local. Elle correspond au mode négociation pour un utilisateur ayant le rôle de gestionnaire du télépointeur. Ce gestionnaire à les mêmes droits que l'utilisateur simple, à savoir droit de demander et de relâcher le contrôle, auxquels on ajoute le droit de le donner et de le reprendre. Il y a deux états pour cet outil : je possède et je ne possède pas, et deux phases : outil utilisé, outil libre.

		Etat	
		Je ne possède pas	Je possède
Phase	Outil libre	Possesseur (nil) MontrerLeDemander CacherLeRelacher MontrerLePrendre MontrerLeDonner ListeDesModes	Impossible
	Outil utilisé	Possesseur(@2) CacherLeRelacher MontrerLeDonner MontrerLePrendre ListeDesModes	Possesseur(@2) MontrerLeRelacher CacherLeDemander MontrerLeDonner CacherLePrendre ListeDesModes

Figure 6 : Exemple de table contenu dans l'AML

Le tableau de la figure 6 représente la table contenue dans l'AML. Contrairement à la description faite dans le chapitre précédent sur l'analyse réalisée pour faire l'outil télépointeur, elle contient des messages qui sont directement interprétables par la Présentation : ils tiennent compte de l'aspect visuel. On faut les interpréter comme suit : Possesseur(nil) signifie qu'il n'y a pas de possesseur de l'outil, MontrerLe Demander signifie montrer l'icone utilisée pour demander la ressource, CacherLeRelâcher signifie faire disparaître ou griser l'icône qui correspond à l'action de relâcher le contrôle La présence des @ permet de définir des variables. Dans notre realisation, nous disposons de dix variables. Elles sont affectées par le GMC avant de donner l'ordre de changement de phase et d'état. Par exemple, @2 représente le nom du possesseur de l'outil et sera mise à jour à chaque changement de possesseur. Ces variables nous permette d'obtenir un mécanisme générique indépendant de l'outil. C'est à dire que tous les GML sont créés à partir d'une classe générique unique indépendamment de l'outil qu'il gère.

L'un des objectifs de l'architecture décrite dans le prochain chapitre sera d'offrir une relative indépendance entre la définition de l'activité et sa représentation, ainsi que les modes d'interaction de l'utilisateur.

Les changements de phase ou d'état générés par L'Agent de Médiation Central, vont entraîner l'envoi des messages contenus dans le tableau à la Présentation. La communication entre l'Agent de Médiation Central et l'AML, en ce qui concerne les changements de phase ou d'état est relativement simple. C'est un message unique qui contient deux paramètres : un pour exprimer la phase et un deuxième pour préciser l'état. Les changements de phase ou d'état sont donc performant dans le fait où ils ne consomment que très peu de bande passante. Ce système de changement de phase et d'état est indépendant de l'état et de la phase précédente. Ceci est important si on veut facilement rendre persistant les outils entre les sessions et que les retardataires récupèrent l'outil dans l'état courant. Le GMC n'a pas besoin de rejouer la séquence des changements d'état et de phase mais simplement à donner la phase et l'état courant après mis à jour les variables. Ceci a pour conséquence que la séquence de messages contenus dans chaque case du tableau (figure 6) doit permettre de donner une représentation visuelle identique quelque soit l'état et la phase précédente. C'est pour ceci que l'on précise toutes les fonctionnalités accessibles et non accessibles.

L'agent de médiation central

Il contient les règles de médiation qui gèrent les changements d'état et de phase d'une activité, ceci pour chaque rôle. Ces règles sont activées sur la réception de messages envoyés par les sites clients. Ces déclencheurs sont identifiés lors de l'analyse de l'activité (voir chapitre précédent), et sont utilisés par la Présentation. Ils sont donc utilisés par le concepteur de la Présentation.

L'AMC est créé à partir de classes génériques qui définissent les caractéristiques d'une activité : l'ensemble des rôles et les médiations possibles. Ces classes stockées dans le composant Organisation, servent à initialiser de nouvelles activités en associant à chaque participant un rôle, et un mode de médiation. Il existe une collection de médiations associées à une activité, qui reflète dans le cadre de RTTC la façon dont les utilisateurs se partagent une ressource (floor algorithm). Changer dynamiquement une politique va correspondre à changer les règles de l'AMC, et à changer la table associée contenue dans l'AML.

Les messages reçus par l'AMC, qui ne font pas partie des déclencheurs, sont transmis au noyau fonctionnel. Ils correspondent à des modifications de l'utilisateur au niveau de la Présentation qui entraînent une modification des données du NF.

La présentation

La présentation doit prendre en charge plusieurs dimensions :

- premièrement elle doit définir l'interface associée à la présentation des informations contenues dans le Noyau Fonctionnel ;
- deuxièmement elle prend en charge les interactions utilisateurs ;
- et enfin elle doit donner une représentation visuelle du rôle, donc des fonctionnalités accessibles pour l'utilisateur à chaque instant.

Elle est dépendante du système sur lequel elle est les implémenter. Il est donc difficile de décrire plus la réalisation de la Présentation sans décrire la plate-forme utilisée pour le développement.

Dans la partie suivante, nous décrivons l'implémentation dans l'environnement Windows 3.1 de composant Présentation.

III - LA REALISATION DU COMPOSANT PRESENTATION DANS L'ENVIRONNEMENT WINDOWS 3.1

3.1 - L'environnement de développement et ses caractéristiques

La plate-forme de développement est un PC sous Windows 3.1. Les développements ont été réalisés en C++ (Borland C++), en utilisant la bibliothèque d'objets d'interface nommée "Object Window Library" ou OWL. Nous ne voulons pas décrire ici l'environnement Windows mais simplement donner les caractéristiques que nous avons utilisées dans nos développements.

Les principales caractéristiques de l'environnement Windows sont :

- la notion de fenêtre ;
- la programmation événementielle ;
- les bibliothèques de lien dynamique ;
- les mécanismes de communication inter-applicatifs multiples et puissants : le presse papier, les échanges dynamiques de données (DDE Dynamic Data Exchange) et les mécanismes de lien d'objets (OLE Object Linking and Embedding).

La fenêtre est un concept clé de l'environnement. Son utilisation et sa définition rappellent les caractéristiques d'une approche objet. Une fenêtre appartient à une classe qui définit son comportement par défaut. Chaque fenêtre peut être comparée à un objet actif qui est capable de traiter et de générer des événements. Dans le monde Windows les événements se traduisent par des envois de messages Windows. A chaque fenêtre est associée une file de messages gérée par le système qui mémorisent l'ensemble des messages avec plusieurs stratégies d'insertion (priorité de certains messages). L'application correspondante pourra traiter les messages lorsque le système lui donnera les ressources nécessaires. L'environnement Windows définit plusieurs classes de fenêtres qui définissent des comportements de base telle que les fenêtres dialogue, les barres de défilement . Développer une application revient à créer une nouvelle classe de fenêtre qui héritera des comportements d'une classe de base pour laquelle on définira des traitements spécifiques correspondant à la réception de certains messages. Les environnements de développement vont eux aussi fournir une hiérarchie de classe de fenêtre en offrant une encapsulation des mécanismes de base de façon à simplifier les développements.

La programmation événementielle revient à définir un traitement à la réception d'un message. A tout message traité par une classe de fenêtre correspond une fonction qui est activée à la réception de celui-ci. La figure 7 montre un squelette d'une classe de fenêtre qui réagit à l'enfoncement du bouton droit de la souris écrit en Borlandc++ 3.1 en utilisant la bibliothèque de classe fournit par "Object Window Library".

```
class ExempleFenêtre : public TWindow
// cette classe hérite de TWindow et réagit à l'enfoncement du bouton droit de la souris
{
public :
    ExempleFenêtre (PTWindowsObject unParent, LPSTR unTitre); //constructeur
    virtual void WMRButtonDown (RTMessage Message)
        = [WM_FIRST + WM_RBUTTONDOWN_DOWN];
    ....}

void ExempleFenêtre :: WMRButtonDown (RTMessage Message)
// fonction membre associée au traitement du message bouton droit enfoncé
{
    ...
}
```

Figure 7 : Description d'une classe de fenêtre en Borlandc ++ 3.1

Les bibliothèques de liens dynamiques ou DLL regroupent des fonctions qui peuvent être appelées par d'autres programmes ou bibliothèques. Elles sont chargées en mémoire lors du premier appel par une application d'une de leurs fonctions ou explicitement par une fonction spécifique de l'API Windows (Application Protocol Interface). Le lien est calculé dynamiquement au moment de l'exécution. Une DLL est partageable par plusieurs applications. L'avantage de ces bibliothèques est le gain en place puisque le code correspondant n'est pas inclu dans chaque application et la facilité de maintenance : la modification d'une DLL ne nécessite pas d'édition de lien pour les programmes qui l'utilisent.

Le presse papier est une application Windows particulière. Elle permet la communication simple de données entre applications. Toutes les applications peuvent communiquer avec le presse papier pour lui fournir des données ou au contraire lui demander les données qu'il contient. Ceci est réalisé au niveau du développement par l'appel de fonctions spécifiques fournies par l'API Windows. L'interface utilisateur est relativement standardisée et correspond aux fonctions Couper, Coller, Copier du menu Edition.

Les échanges dynamiques de données (DDE) sont basés sur le mécanisme d'envoi de message inclu dans l'environnement Windows. Ce mécanisme permet l'échange de données entre un serveur DDE et un client DDE. On parle alors de conversation DDE. Dans une conversation, il existe plusieurs liens qui définissent la façon dont les informations transitent entre le serveur et le client : passive, automatique et active.

Le mécanisme de liaison et d'inclusion d'objet est une extension du mécanisme de DDE. Il permet le transfert d'information entre un serveur et un client OLE. L'information est structurée sous forme de document. Un document est un ensemble structuré de données de différents types (graphique, texte, image). Le document utilise les mécanismes OLE afin de manipuler les différentes données qui le constituent.

3.2 - La programmation d'une présentation.

Le composant Présentation est une application Windows manipulable par l'utilisateur par l'intermédiaire de la souris et du clavier, et communiquant avec son Agent de Médiation Local par l'intermédiaire d'une conversation DDE. La Présentation est dans cette conversation le serveur et l'Agent de Médiation le client. La Présentation supporte donc deux canaux d'interaction.

Cette communication va permettre la réception des messages et l'envoi de message en utilisant une syntaxe bien particulière. Cette syntaxe est la syntaxe standard utilisée dans les échanges DDE : [opcode(paramètre 1,....., paramètre n)]. En ce qui concerne l'envoi de messages, nous utilisons deux liens actifs dans la conversation : un pour l'envoi de messages au site serveur (pour l'AML ou le Noyau Fonctionnel), et un autre pour l'envoi de messages aux présentations correspondantes sur les autres sites (pour la rétroaction de groupe).

Pour réaliser ceci, nous définissons une classe de fenêtre qui offre les fonctionnalités de communication DDE adaptées à l'environnement nommé TWindowCoop. Elle hérite (héritage multiple) de la Classe TWindow qui définit le comportement d'une fenêtre standard, et d'une classe nommée DdeSvr qui prend en charge la communication DDE, fournit deux fonctions pour l'envoi de messages ainsi qu'une fonction de traitement des messages reçus par le canal DDE. La figure 8 montre une description des classes DdeSvr et TWindowCoop.

```

//Classe DdeSvr
class DdeSvr
{
public:
    int Topic;
    DdeSvr(char *,char*,HANDLE);
    ~DdeSvr();
    void SendData(char *);           // envoi au site serveur
    void BroadcastData(char *);      // envoi aux présentations correspondantes
    virtual BOOL Exec(char *)=0;     // fonction de traitement des messages DDE reçus
private:
    // fonctions associées à la gestion de la communication DDE
    static HDEDEDATA FAR PASCAL _export CallBack( WORD, WORD, HCONV, HSZ, HSZ,
HDEDEDATA, DWORD, DWORD );
    BOOL MatchTopicAndService( HSZ, HSZ );
    BOOL MatchTopicAndItem( HSZ, HSZ );
    HDEDEDATA Execute(HDEDEDATA);
    HDEDEDATA DataRequested(WORD, int);
    DWORD idInst;
    HCONV hConv;
    BOOL Advised;
    HSZ hszService, hszTopic, hszBltem, hszSltem;
    char data[MAXSIZE],Cmde[MAXSIZE];
    FARPROC lpfnCallBack;
    static DdeSvr *pStaticThis;
};

//classe TWindowCoop
class TWindowCoop:public TWindow,public DdeSvr
{
public:
    char *roomName;    // mémorise l'identificateur de pièce ( argv parameter)
    char *userName;    // mémorise le nom de l'utilisateur
    int myId;          // mémorise l'identificateur d'outil
    HWND toolBox;      // mémorise le handle de la fenêtre boîte à outil
    TWindowCoop (PTWindowsObject AParent, LPSTR ATitle, char *,char *);
    TWindowCoop::~TWindowCoop();
    virtual LPSTR GetClassName() { return ("TWindowCoop"); };
    virtual void GetWindowClass (WNDCLASS& WndClass);
    virtual BOOL Exec (char *);    // boucle de traitement des messages reçus

    // messages de communication avec la boîte à outil
    virtual void WMStop (RTMessage) = [WM_FIRST + WM_STOP];
    void BeHide(RTMessage) = [WM_FIRST + WM_NCLBUTTONDOWN];
    void BeVisible(RTMessage) = [WM_FIRST + WM_TOOL];
};

```

Figure 8 : Les classes DDESvr et TWindowCoop

La fonction de traitement des messages est appelée à la réception de chaque message. Le message est alors analysé (par un petit analyseur syntaxique de commande DDE) pour obtenir l'opcode et les paramètres afin d'appeler la fonction de la Présentation correspondante. La figure 9 montre la fonction de traitement des messages associés à l'outil télépointeur. On y distingue le traitement des messages envoyés par l'Agent de Médiation Local et les messages qui transitent entre les présentations.

```

BOOL TToolBar:: Exec(char *cmd)
{
...
// liste des opcodes possibles
static char *CmdList[]={"possesseur","texte",.....,"MontrerLeDemander",
.....,"rectangle","poseMarque"};
cmds.SetCmdString(cmd);
while (c=cmds.NextCmd())
{
    switch (c)
    { // traitement de chaque message reçu
    case 1: // Possesseur
        ...
        break;
    case 2: // texte (atext,X,Y,acolor) affiche atext à la position X, Y avec la couleur acolor
        char s[50];
        strncpy(s,cmds.NextParam(),49);
        X = atoi(cmds.NextParam());
        Y = atoi(cmds.NextParam());
        aColor = (COLORREF)(atoi(cmds.NextParam()));
        WIS_DC = GetDC(WysiwiHandle);
        SetBkMode(WIS_DC,TRANSPARENT);
        SetTextColor(WIS_DC,aColor);
        TextOut(WIS_DC,X,Y,s,strlen(s));
        ReleaseDC(WysiwiHandle , WIS_DC);
        ((PTTextTool)State->Tools[4])>textes->add(*(new TAnnot(s,X,Y,aColor)));
        break;

        ...
    case 15 : // CacherLeRelacher
        ReleaseOK=FALSE;
        index = concat(&toolTab[0],index,releaseIndex);
        MoveWindow(HWindow,Attr.X,Attr.Y,0,0,TRUE);
        break;

    case 16 : // MontrerLeDemander
        index = add(&toolTab[0],index,takeIndex);
        MoveWindow(HWindow,Attr.X,Attr.Y,0,0,TRUE);
        break;
    default : return FALSE;
    }
    return TRUE;
}

```

Figure 9 : Fonction de traitement des messages de l'outil télépointeur

La construction d'une nouvelle présentation revient à définir une sous-classe de TWindowCoop pour récupérer le comportement standard d'une fenêtre, les fonctions d'envoi et de réception de messages provenant l'Agent de Médiation, et des messages Windows de communication avec la boîte à outil pour gérer des problèmes de navigation (faire apparaître ou disparaître l'outil). Dans l'annexe III, nous décrivons la réalisation de l'outil de visualisation et du télépointeur.

Les relations entre le Noyau Fonctionnel et la Présentation

La réalisation des présentations dans l'environnement Windows, nous amène à répliquer les données contenues dans le noyau fonctionnel afin de permettre la gestion

du rafraîchissement des fenêtres. Nous sommes même allés plus loin dans la réplication de cette information car pour certain outil tel que le télépointeur, il n'existe pas de Noyau Fonctionnel : la création des marques est diffusée à toutes les présentations sans stockage au niveau du NF. Dans ce cas nous avons eu une approche complètement répliquée en ce qui concerne les données manipulées par les présentations, la gestion du contrôle de l'outil restant à la charge de l'agent de médiation central situé sur le serveur. Par contre, pour un outil tel que le vote, le Noyau Fonctionnel contient toutes les données relatives au vote ainsi que les méthodes de calcul du résultat. L'information répliquée dans chaque Présentation est mise à jour à la suite de messages générés par le Noyau Fonctionnel vers toutes les présentations.

On peut définir trois stratégies dans la réalisation d'une présentation :

- (1) le Noyau Fonctionnel contient toutes les données du domaine et les présentations ne contiennent qu'une réplication de l'information nécessaire à l'affichage. La mise à jour des présentations se fait sur l'initiative du Noyau Fonctionnel par diffusion de messages à toutes les présentations. Cette stratégie se rapproche de celle utilisée par Hill pour réaliser les composants ALV [Hill 92] ;

- (2) le Noyau Fonctionnel contient les informations du domaine qui doivent être persistante afin de retrouver les outils dans l'état où ils étaient lorsque l'utilisateur a quitté la pièce, et de permettre l'arrivée des retardataires. Le noyau fonctionnel est une copie de référence. Le noyau fonctionnel assure donc la cohérence entre l'information répliquée entre les Présentations. En ce qui concerne l'information nécessaire aux Présentations qui n'est pas reconnue comme persistante, la cohérence entre les répliques est assurée par les Présentations en utilisant le système de diffusion de messages entre ces dernières. La causalité des événements est assurée par le composant Gestionnaire de communication ;

- (3) Le noyau fonctionnel n'existe pas. C'est le cas du télépointeur dans lequel nous avons décidé de ne pas rendre persistant les marques qu'il dépose sur le document.

Il faut remarquer que ces trois stratégies sont possibles car le système de gestion des rôles et de contrôle est extérieur à l'outil. L'annexe trois donne une vue de la réalisation d'outil utilisant la stratégie 2 et 3.

Le choix entre les stratégies se fait par rapport à des critères que nous avons identifiés : le niveau de persistance de l'état visuel d'un outil, la performance de l'interface utilisateur, la prise en compte du retardataire. Par exemple pour le télépointeur, nous voulions obtenir une performance de l'interface importante (visualisation instantanée sur les sites distants) et pas de prise en compte des

retardataires, ni de persistance de la désignation : nous avons choisi la stratégie 3. Pour la fenêtre de visualisation collective, nous avons choisi la stratégie 2, pour prendre facilement en charge l'arrivée des retardataires (l'information étant demandée au NF) et la présentation contient une copie de l'information pour avoir une bonne performance de l'interface lors des réaffichages.

Conclusion :

La réalisation d'une présentation demande beaucoup de travail en terme de codage. Par exemple le source de l'outil téléporteur contient environ 2000 lignes de code, l'outil de visualisation collectif 700 lignes auxquelles il faut ajouter une bibliothèque permettant l'affichage des formats DIB (Device Independent Bitmap), WMF (Windows Meta File) et le format généré par l'éditeur Write, soit 2000 lignes. La complexité du système pose des problèmes dans la phase de développement car il est difficile de tester le comportement d'une présentation en cours de réalisation : il faut disposer de tout le système réaliser toute la phase de connexion, avoir à disposition plusieurs machines Il a donc fallu aussi développer des outils d'aide au développement : par exemple Claude Vièville (Ingénieur de Recherche du Laboratoire Trigone) a développé un simulateur de l'Agent de Médiation qui permet de valider le comportement en terme de rôle. Nous disposons maintenant d'une hiérarchie de classe permettant de développer plus facilement les Présentations. Par exemple, elle a servi à développer une présentation d'un outil de visualisation et d'annotation de vidéo par une personne extérieur au Projet Co-Learn (mémoire de DEA de Franck Houzé [Rouzet 93]).

CHAPITRE V

MISE EN OEUVRE ET EVALUATION

DU PROTOTYPE DE CONFERENCE

TEMPS REEL DE CO-LEARN

L'objectif de ce chapitre est de démontrer comment la mise en oeuvre et l'évaluation d'un prototype dans une véritable organisation pose des problèmes dont certains ont été mentionnés dans le chapitre II : l'existence du travail articulatoire, la difficulté d'utilisation d'un outil, la remise en cause de pratiques éducatives, la fiabilité du prototype, des problèmes de coordination, et les problèmes liés aux bénéfices pour les personnes qui utilisent le système. La seconde partie de ce chapitre sera consacrée plus particulièrement à l'évaluation. Nous donnerons une description d'un scénario afin de définir l'usage et le résultat de l'évaluation de celui-ci en situation réelle selon trois aspects : un aspect informatique, un aspect pédagogique et un aspect psycho-cognitif. Nous concluons sur l'évaluation par une description d'une étude d'utilisabilité du système.

I - La mise en oeuvre de RTTC au sein de l'institut CUEEP

Dans cette partie nous voulons montrer tout d'abord le problème de la mise en place physique et logicielle d'un dispositif, et ensuite l'ensemble des démarches à suivre avant de pouvoir passer réellement à la phase d'expérimentation et d'évaluation.

1 - L'installation matérielle et logicielle

L'utilisation du prototype a nécessité la mise en place de cinq stations Co-Learn¹ dans des lieux relativement différents. Nous avons choisi d'installer deux sites sur le campus universitaire de Villeneuve d'Ascq et trois autres situés à Lens (40 km de Villeneuve), Dunkerque (80 km) et Calais (70 km). L'implantation de ces sites facilitera le recrutement des stagiaires et ne leur demandera pas de démarches supplémentaires en terme de déplacement (ils suivent leur formation sur ces mêmes centres). La première phase a été la mise en place physique des différentes stations avec leur configuration logicielle. Afin de gérer plus facilement ces stations distantes, nous avons trouvé des correspondants techniques locaux (souvent des informaticiens ou des personnes sensibilisées à l'utilisation de l'informatique) afin qu'ils puissent nous aider dans la maintenance logicielle et matérielle. Durant cette phase d'installation, nous avons vérifié les connexions RNIS de chaque site, avec parfois des choix de locaux non judicieux pour les accès : prise installée dans un petit local tout noir donc peu convivial, ou installée dans une salle machine où se trouvent plusieurs personnes donc peu adaptée pour l'utilisation du canal sonore.

2 - La gestion des utilisateurs et les problèmes humains

Le système installé, nous sommes passés à la phase de formation des formateurs à l'utilisation du système. Les formateurs sont les enseignants qui animeront les séances d'enseignement à distance avec de véritables apprenants inscrits dans des formations au sein de l'institut. Le premier problème pour nous concepteurs et réalisateurs est de fournir un système relativement fiable. Il faut prouver aux futurs utilisateurs que le système est suffisamment fiable pour être utilisé en situation réelle. Il faut souligner que la plupart de ces formateurs sont intégrés dans l'équipe OPEN² qui s'intéressent aux Sciences de l'Education et plus précisément à l'enseignement ouvert à distance. Il est important pour la bonne réussite des expérimentations que celles-ci prennent en compte la problématique de recherche de cette équipe (voir [Grudin 88] pour l'aspect profit dans l'utilisation d'un CSCW). L'aspect fiabilité du système est aussi important pour ces formateurs, qui vont utiliser pour l'évaluation des apprenants inscrits dans les

¹Nous appelons station Co-Learn tout PC possédant les logiciels spécifiques de l'environnement et capable d'accéder au serveur Co-Learn

²L'équipe OPEN du laboratoire Trigone s'intéresse à l'enseignement ouvert (Open Learning). Une partie de son activité est d'utiliser des systèmes d'enseignement à distance (EAD) et de réfléchir sur les scénarios pédagogiques qui peuvent être mis en place

formations qu'ils dispensent : on ne peut pas faire perdre le temps des stagiaires qui sont en formation. Face à ceci, il est nécessaire de convaincre, d'expliquer les enjeux aux différents partenaires concernés, et de coordonner les actions de tout le monde. Si la conception des collecticiels nécessite la mise en commun de compétences appartenant à des domaines différents, la mise en oeuvre et l'évaluation d'un collecticiel sont aussi des tâches collectives. Pour finir, on peut dire que l'utilisation d'un nouvel outil pose toujours des problèmes et que son acceptation par les personnes qui vont l'utiliser est très délicate, même si ces personnes dominent le contenu pédagogique. Ce nouvel outil va obligatoirement modifier la pratique pédagogique des formateurs. Il est donc indispensable qu'ils maîtrisent l'outil, qu'ils comprennent les objectifs et les concepts de l'environnement, afin qu'ils puissent adapter leurs scénarios pédagogiques.

Avant de commencer l'expérimentation, il faut préparer des contenus pédagogiques et les adapter aux contraintes du système. Cette phase est relativement simple, dès que l'utilisateur accepte les contraintes du système en terme de présentation, de format et de fonctionnalités.

En conclusion, la mise en oeuvre du prototype a été pour nous une leçon pratique sur le travail de groupe, et nous a confronté à la vraie nature du travail coopératif, qui dans notre cas était rarement médiatisé.

II - L'expérimentation de RTTC

1 - Préparation de l'expérimentation

1.1 - La configuration matérielle

Le dispositif physique se compose de trois stations Co-Learn de la classe 2BS (stations connectées sur RNIS intégrant le contrôle du canal sonore) situées à Calais, Dunkerque, Villeneuve d'Ascq, et de deux stations Co-Learn sur réseau local (utilisation d'un téléphone pour le canal sonore).

1.2 - Les ressources humaines

Nous disposons de 20 stagiaires inscrits dans les modules MATHS9, MATHS10, MATHS11³ répartis sur 4 sites : ce qui signifie 3 à 4 personnes devant une station Co-Learn. La dernière station est utilisée par 2 formateurs EAD⁴. Cette situation n'est pas celle qui a été prévue pour RTTC. Le modèle idéal que nous avons prévu au début du projet, est un utilisateur par station. Nous reviendrons sur cet aspect dans le § III.3.

Pour cette séance nous avons un correspondant technique sur chaque site, et des observateurs de différentes disciplines : Sciences de l'Education, Psychologie et Informatique. Un observateur particulier était relié au système simplement par téléphone : son rôle était d'essayer de comprendre ce qui se passait sans intervenir dans l'audioconférence.

1.3 - Le Protocole de connexion des stations Co-Learn

La connexion d'une station étant une opération majeure sur le serveur, les sites vont se connecter les uns après les autres. Dans un premier temps, tous les responsables techniques se retrouvent en audioconférence. Les stations RNIS utilisent l'outil de téléphone disponible sous Windows (développé au laboratoire Trigone) et les autres le téléphone. Ensuite, chaque correspondant technique entrera dans le Centre de Ressources Virtuel de Co-Learn l'un après l'autre. Cette phase de connexion et de vérification est importante afin de commencer l'évaluation dans un état stable et opérationnel du système.

³Ceci correspond à des niveaux de formation en Mathématiques

⁴Les formateurs EAD sont impliqués dans les formations à distance dispensées au sein de l'institut CUEEP. Leur pratique est basée sur l'échange de documents avec les apprenants et sur des permanences téléphoniques

1.4 - La formation

La formation des utilisateurs novices se fera sous forme d'une séance de 10 minutes animée par une personne connaissant parfaitement le fonctionnement du système. Cette formation doit permettre aux participants d'apprendre à utiliser quelques outils (outil de visualisation collectif, Télépointeur et Outil de dialogue) par la pratique.

1.5 - La séance pédagogique

La séance démarre avec une vision dans la fenêtre publique de 6 courbes et 6 tableaux. La figure 1 montre une vision de l'interface lorsque l'utilisateur entre dans la pièce. Nous ne voyons que les courbes, les tableaux étant contenus dans un autre document.

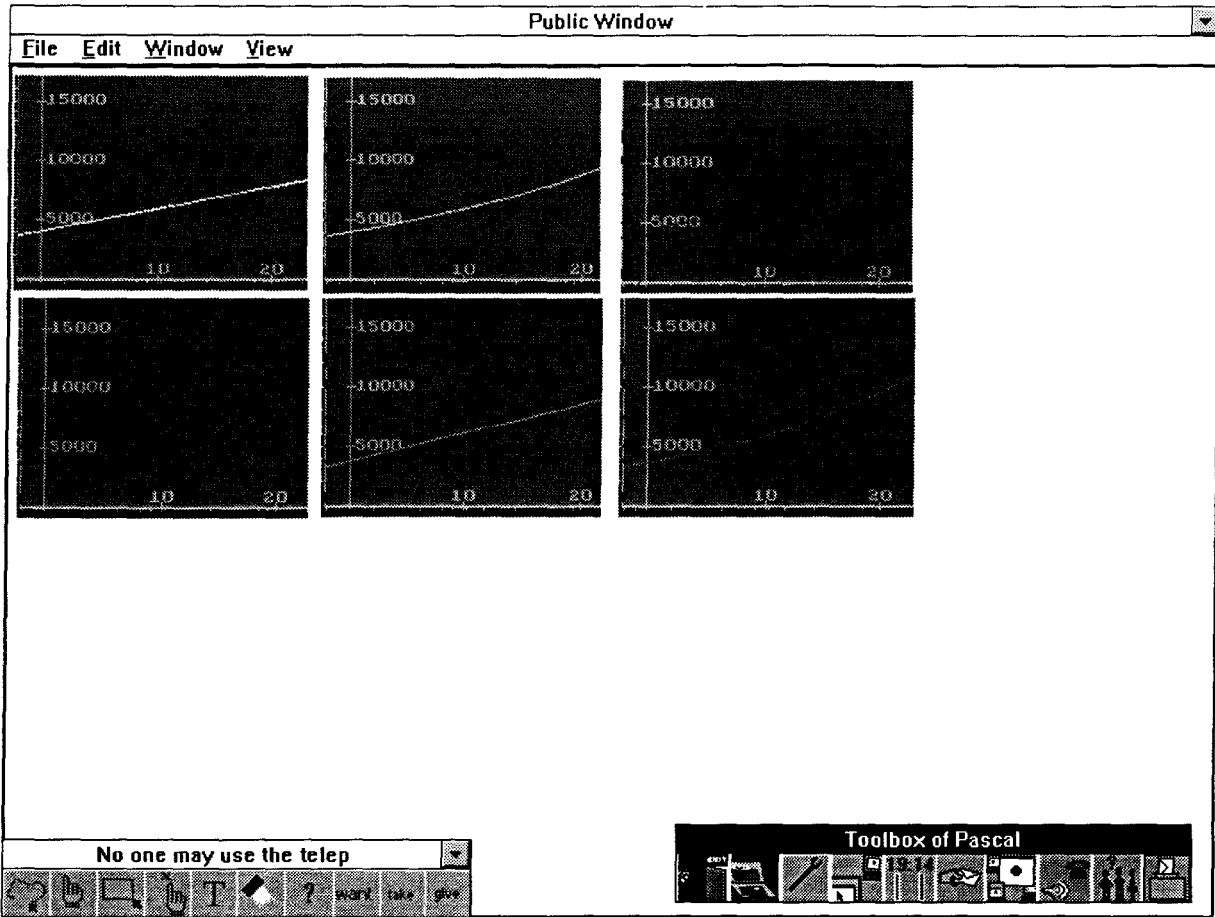


Figure 1 : L'utilisateur vient d'entrer dans la pièce

Le scénario pédagogique est le suivant :

- 1 : produire un énoncé qui puisse correspondre aux documents visualisés en utilisant un outil de l'espace privé : paintbrush ;
- 2 : discussion sur la validité des énoncés (un par participant). Visualisation des énoncés à l'aide de la fenêtre publique;

3 : rechercher les équations associées aux 6 graphiques et tableaux. Distribution de 2 à 3 équations par site;

4 : validation des résultats par le groupe;

5 : continuer le travail à l'extérieur du système pour calculer combien on paie dans chacun des cas : les courbes correspondent à des tarifs de location (extension du scénario pour utiliser l'ATC téléconférence asynchrone).

2 - Le déroulement de la séance

La description suivante de la séance est basée sur la perception d'un observateur se trouvant avec un groupe d'utilisateur.

La séance a commencé par un test de connexion de toutes les stations Co-Learn. Seules deux stations Co-Learn ont réussi à se connecter au serveur⁵. La station de Dunkerque n'a jamais réussi à réaliser entièrement la procédure d'entrée dans la salle de conférence temps réel. Les stations sur réseau local n'ont eu aucun problème à la connexion.

Seule l'audio conférence étant disponible pour tout le monde, le groupe à communiquer principalement en utilisant ce canal. La distribution de documents papiers sur les sites non connectés a permis le début du travail. La séance de formation n'a pas été faite. Des explications verbales ont été données sur chaque site, avec très peu de manipulations de la part des utilisateurs.

La description qui suit est un ensemble de réflexions non structurées qui ont été prises en note pendant la séance.

L'audioconférence ne peut être utilisée que pour la communication de groupe. Par exemple, pour régler les problèmes techniques il faut prévoir un autre canal de communication comme un téléphone classique point à point.

De même, la communication locale "prend le dessus" sur les deux autres espaces de conversation que sont les messages sonores de la machine et l'audioconférence. Les participants ne réagissent pas aux messages sonores de notification générés par les outils, lorsqu'ils communiquent localement ou que chacun est occupé à un travail personnel. Interpellé sur ce fait, les utilisateurs ont répondu qu'ils avaient entendu quelque chose mais qu'ils n'avaient pas perçu le contenu.

⁵Ce problème a été identifié et corrigé depuis que nous disposons d'une couche de communication fiabilisée (Travail de notre partenaire UPM Université Politechnique de Madrid)

Des problèmes d'identification du formateur sont apparus : on percevait au niveau de l'audioconférence deux formateurs alors que le système Co-Learn n'en faisait apparaître qu'un (les deux formateurs se trouvent derrière la même station). Ceci reflète un mauvais usage (ou une perversion de l'utilisation) du système par les expérimentateurs. En effet, le système prévoit un seul rôle par station. Nous reviendrons dans la suite du chapitre sur ce problème de la présence d'un groupe devant la station Co-Learn.

Les messages sonores générés par les outils Co-Learn perturbent l'audioconférence. Ceci est un problème d'ergonomie qui doit nous amener à réfléchir sur la combinaison entre les fonctionnalités offertes par une audioconférence et l'utilisation des possibilités multimédia de la station.

La communication mise en place entre deux sites par le formateur (pour mettre en contact deux sites ayant un problème commun) ne permet plus aux autres de communiquer car cette communication en sous groupe monopolise l'audioconférence.

A Calais, le correspondant technique est apparu comme l'intermédiaire entre le groupe de stagiaires et le système Co-Learn.

Une calculette dans l'espace privé se serait avérée très utile. Il faut préciser que l'espace privé est un groupe de programme Windows facilement accessible à partir de la boîte à outils. Ceci veut dire que les utilisateurs novices n'ont pas à chercher dans l'environnement Windows, l'application dont ils ont besoin. Bien préparer l'espace privé en fonction du scénario pédagogique est très important si on veut que l'espace privé tienne le rôle pour lequel il a été prévu.

Les stagiaires ont manipulé vraiment le système qu'à la fin (2 heures après le début de la séance). "On n'a commencé à s'amuser qu'à la fin". Ceci ne concerne que le groupe où se trouvait l'observateur.

L'activité de groupe nécessite parfois de capter l'attention de tous les participants : soit pour une explication orale, ou l'utilisation d'un outil public tel que l'outil de visualisation collectif. Cette synchronisation des participants en utilisant le canal sonore n'apparaît pas toujours efficace lorsque plusieurs personnes se trouvent devant la station et que l'espace de communication locale est utilisé.

L'incompréhension d'un problème par le canal vocal, peut être parfois facilement résolu par l'utilisation de l'outil de communication (écrire une équation peut être plus significatif que l'épeler). L'utilisation de l'outil de dialogue, ou de l'annotation textuelle du Télépointeur (si l'information est pour le groupe) est adaptée pour régler le problème. Il apparaît très important que les utilisateurs aient connaissance de tous les moyens de

communication mis à leur disposition. L'efficacité des utilisateurs en terme d'utilisation des outils de communication ne pourra se faire que par un apprentissage progressif. La possibilité de définir les outils présents dans une pièce (à sa création ou entre deux utilisations de celle-ci) va permettre aux formateurs de commencer sa formation avec peu d'outil, et ensuite d'en ajouter afin de proposer un apprentissage incrémental de l'environnement, ou d'adapter le choix des outils aux besoins du groupe. Ceci correspond exactement à ce que nous appelons la malléabilité structurelle et fonctionnelle.

La prise de parole dans l'audioconférence s'est traduite plusieurs fois par la phrase "Allô!". Y a t'il besoin d'un apprentissage de la communication dans l'audioconférence ? La définition d'un protocole simple d'intervention dans l'audioconférence s'avère utile. Il faudra mener une réflexion sur la définition de nouvelles fonctionnalités pour l'audioconférence [Barme 95][Derycke 94a][Barme 94].

La phrase "tout le monde est à l'écoute" a plusieurs fois été prononcée par le formateur.

Dans la phase de conclusion, le temps de parole des formateurs a été relativement important (peut être un paramètre à mesurer dans les futures expérimentations).

Les documents papiers ont permis de palier aux problèmes techniques, mais ont entraîné chez les sites connectés un travail papier important et une sous utilisation de l'espace privé informatique.

Conclusion

La description de cette séance d'utilisation correspond à la toute première avec des étudiants inscrits dans des modules réels de l'institut CUEEP. Cette présentation peut paraître relativement peu valorisante pour l'utilisation du système, mais nous pensons qu'elle illustre très bien les problèmes rencontrés lors de l'utilisation d'un tel système dans une organisation. L'utilisation de collecticiels dans les laboratoires qui les ont créés, n'amène pas ce type de description et ne révèle pas non plus tous les vrais problèmes.

D'autres expérimentations menées au sein de l'institut CUEEP et dans d'autres sites pilotes, ont permis d'élaborer les conclusions présentées dans le reste du chapitre.

III - Conclusions sur l'utilisation du prototype au CUEEP

Dans cette partie, nous voulons faire part des conclusions que nous émettons suite à l'évaluation du prototype au sein de l'institut CUEEP. Ces conclusions ne sont pas celles obtenues par une mise en place d'un processus d'évaluation strict et rigoureux, mais plutôt des constatations qui demandent à être validées. Elles résultent du travail réalisé par Pascal Croisy (auteur de cette thèse), Danièle Clément (Maître de Conférence en Psychologie), Chantal d'Halluin (Maître de Conférence, Docteur d'état en Sciences de l'Education) et de discussions avec les autres membres de l'Equipe OPEN et NOCE⁶ du laboratoire Trigone.

Ces premières constatations [Croisy 94c] vont nous permettre d'élaborer un programme d'évaluation et d'observation intégrant des dispositifs de capture d'écran et d'enregistrement audio et vidéo. Nous avons déjà utilisé ces techniques pour réaliser un enregistrement vidéo qui illustre les fonctionnalités du système. Les constatations sont de trois niveaux : un premier niveau relatif aux systèmes informatiques mis en place (complétées dans la partie V), un deuxième niveau s'intéressant à la pédagogie et plus particulièrement aux scénarios pédagogiques mis en place et à l'importance de ces dispositifs de communication dans l'enseignement ouvert [Derycke 90], et enfin un niveau socio-cognitif chargé d'appréhender la dimension sociale du processus d'apprentissage.

1 - Les aspects techniques et fonctionnels

Les diverses expérimentations ont permis de valider l'architecture physique en terme de performances et d'utilisation de différents réseaux (Ethernet, RNIS). Néanmoins, nous devons encore accroître la fiabilité de certains composants logiciels. Pour le télépointeur des efforts sont à faire sur certaines fonctionnalités tels que les déplacements avec trace. Cette fiabilité et les performances⁷ de cet outil dépendent fortement de la couche de communication fournie par notre partenaire de l'Université de Madrid (voir chapitre IV). Les performances dépendent donc du réseau RNIS et de la couche de communication. D'autres explorations sont faites afin de mesurer les performances des drivers de la carte RNIS, et de leur intégration dans l'environnement Windows. Nous espérons que ces investigations nous permettront d'augmenter les performances générales du système afin de fournir un plus grand confort d'utilisation aux utilisateurs.

⁶L'équipe NOCE Nouveaux Outils pour la Communication Educative du laboratoire Trigone (à laquelle l'auteur de cette thèse appartient) s'occupe de la création, la mise en oeuvre et l'exploitation d'outil de communication (par exemple des collecticiels) qui permettent l'enseignement à distance. Ces travaux sont fortement couplés avec l'équipe Open

Les métaphores de Pièce et d'Outil nous ont aidé dans l'explication du fonctionnement du système. En ce qui concerne la métaphore de Pièce, nos investigations ne peuvent être que superficielles car nous n'avons utilisé qu'une seule pièce. Nous ne pouvons donc pas conclure sur l'aspect structurant des pièces dans le centre de ressource. La seule constatation possible est que dans les échanges verbaux ou les petits entretiens réalisés de façon spontanée après les séances, la métaphore de pièce apparaît utile pour les utilisateurs en terme de repère par rapport aux différentes formations qu'ils pourraient suivre : une pièce correspond à une formation. La métaphore d'outil semble bien adaptée pour diriger et structurer l'activité des participants. Un certain niveau de compréhension et d'habitude est à acquérir pour que les participants identifient les activités permises par chaque outil, de telle façon à ce qu'ils puissent les utiliser simultanément et surtout savoir choisir le bon outil en fonction de ce qu'ils veulent faire. En conclusion, les dimensions métaphoriques ne pourront être vraiment validées que suite à une utilisation sur une longue durée des différentes pièces du système par des groupes d'apprenants. La notion de longue durée est importante, car la maîtrise de l'environnement demande du temps : il ne suffit pas seulement de savoir manipuler l'outil informatique, mais il faut aussi l'utiliser efficacement. Il faut que les groupes utilisent plusieurs pièces de l'environnement si on veut valider l'aspect structurant pour l'organisation de la métaphore de pièce.

Suite à une petite enquête de satisfaction, nous pouvons dire que l'utilisation des outils au travers de leur interface est jugée facile par les participants. Pourtant certains problèmes d'ergonomie liés à l'aspect visuel des icônes ont été constatés [Smith 94].

Il est difficile de conclure actuellement sur l'adéquation entre les fonctionnalités offertes par les outils et leur utilisation réelle. Ce qui semble difficile pour les utilisateurs, c'est de choisir le bon outil en fonction de ce qu'ils veulent faire. Nous avons souvent constaté que lorsque le formateur suscite l'utilisation de l'outil à l'apprenant (lorsqu'il arrive à percevoir le problème de l'apprenant), celui-ci l'utilise et atteint son objectif. Ceci pose le problème de la définition de l'engagement d'un outil dans une activité.

Au niveau fonctionnel, de nouvelles fonctionnalités sont apparues comme nécessaires. Elles sont soit demandées par les utilisateurs du système, ou identifiées par les concepteurs du système suite à l'observation de ces séances d'expérimentations. Pour le moment, elles ne sont pas encore bien spécifiées, mais on peut les énoncer comme ceci :

⁷Les performances de l'outil télépointeur se mesurent selon deux critères : un critère spatial qui évalue comment sont restitués les déplacements du curseur sur les postes distants et un critère temporel qui évalue le décalage entre les différentes mise à jour des interfaces

- nécessité de compléter les fonctionnalités du télépointeur (aide au discours) : par exemple en mathématique pouvoir déplacer une règle pour illustrer le parallélisme existant en deux droites d'un document. On aborde peut être ici la nécessité de compléter les fonctionnalités d'un outil pour l'adapter à un type d'application (la géométrie) ;

- le besoin de contrôler plus finement le canal sonore : offrir la possibilité de savoir comment les autres perçoivent le canal sonore : l'état du micro, le volume d'écoute. Ceci est nécessaire surtout pour le médiateur⁸ afin de mieux gérer le canal sonore : s'il peut contrôler le volume de restitution, il pourra interpellier l'ensemble du groupe ou une personne particulière. L'autre problème du canal sonore est la possibilité de savoir qui parle, d'avoir la possibilité de faire des apartés. Certaines de ces fonctionnalités ont été spécifiées et non implémentées (par exemple la version complète du gérant de conférence non réalisée). Les autres posent des problèmes complexes soit au niveau réalisation, soit au niveau de leur impact sur le travail coopératif [Barme 94] ;

- la nécessité d'avoir des outils de création multi-utilisateur qui permettent à ces derniers de ne pas montrer uniquement le résultat, mais aussi la démarche de conception de celui-ci. Ceci apparaît comme primordial dans certaines situations pédagogiques, mais ne remet en aucun cas en cause l'existence de l'outil de visualisation. Ce type d'outil devrait venir enrichir l'environnement de telle façon qu'il puisse assister d'autres situations pédagogiques ;

- les participants ont demandé de pouvoir visualiser plusieurs documents en même temps (plus de deux). Mis à part les problèmes de place sur l'écran, une fenêtre de visualisation collective avec la possibilité de pouvoir dynamiquement visualiser plusieurs documents en même temps permettrait de voir les travaux de chaque membre du groupe, ou de conserver un petit historique de solutions possibles.

- pendant la phase d'apprentissage de chaque outil, avoir une version strictement WYSIWIS permettrait une explication par démonstration. Par exemple voir l'ouverture des menus sur l'outil de visualisation collectif faciliterait la compréhension et la recherche des items dans un menu. Nous verrons dans le chapitre suivant comment nous envisageons la création d'une boîte à outils d'objets d'interaction permettant de passer d'un mode Wysiwis strict à un mode Wysiwis relâcher.

La dernière constatation est l'importance du canal verbal pour la communication, mais aussi pour coordonner les activités des différents utilisateurs et se partager les différentes outils. L'accès à l'outil télépointeur en cliquant sur l'icône correspondant est souvent accompagné d'un échange verbal du type "je prends le télépointeur". Cette

⁸Le médiateur est un participant possédant un rôle particulier.

conclusion ne remet pas en cause l'existence de l'algorithme de contrôle, mais relativise son importance. Nous sommes prudents et nous ne voulons pas conclure trop vite car les expérimentations n'ont pas été jusqu'alors suffisamment nombreuses. Par compte, nous envisageons de réaliser des expérimentations en modifiant cet algorithme de contrôle. C'est à dire pouvoir définir plusieurs modes, allant du mode anarchique jusqu'au mode fortement contrôlé afin de mesurer l'impact de l'algorithme de contrôle sur l'activité.

2 - Les aspects pédagogiques

Nous ne voulons pas réaliser ici une étude très précise des aspects pédagogiques, car ce n'est pas le but de cette thèse. Mais il nous semble intéressant d'aborder ces aspects pour illustrer comment le collecticiel, sa conception, sa mise en oeuvre et son évaluation sont des problèmes qui mettent en jeu plusieurs disciplines et plusieurs compétences (voir [Ellis 91] ou [Karsenty 94b] pour une illustration de la dimension multidisciplinaire du collecticiel). Il en sera de même pour la partie suivante sur les aspects socio-cognitifs.

Comme tout système d'enseignement qui utilise un système technique complexe, l'enseignant impliqué dans l'utilisation de l'environnement Co-Learn doit gérer un grand nombre d'information en même temps. Tout d'abord il doit maîtriser l'utilisation de l'environnement, préparer ses documents pédagogiques en respectant les contraintes imposées par le système, apprendre à gérer la communication sur une audioconférence, gérer l'utilisation des outils de l'environnement par les apprenants, percevoir les problèmes des apprenants ou du groupe, s'assurer que le travail coopératif réalisé est efficace. En conclusion, nous pouvons dire que l'utilisation du système Co-Learn nécessite une redéfinition des acteurs impliqués dans le processus pédagogique ainsi que de leur rôle respectif. Une évaluation à grande échelle doit permettre la définition de ces nouveaux acteurs, ou tout simplement une redéfinition de leur rôle, ainsi que des besoins exprimés par les tuteurs eux-mêmes qui nous⁹ permettrons de spécifier une nouvelle génération d'environnement CSCL (Computer Supported Collaborative Learning).

3 - Les aspects socio-cognitifs

La communication dans l'environnement Co-Learn

Les moyens de communication, surtout le canal sonore ont grandement facilité la communication entre le tuteur et les apprenants, mais aussi entre les apprenants eux-mêmes.

⁹il faut comprendre ici tout les personnes impliquées dans le processus d'évaluation : concepteur du systèmes, utilisateurs (apprenants et tuteurs) et évaluateurs

Le canal sonore a permis rapidement d'énoncer les objectifs du cours et de distribuer des tâches et des responsabilités aux différents participants. Il a aussi permis de synchroniser et de coordonner les activités de chacun.

La communication verbale est aussi utilisée pour vérifier la fiabilité du système. Par exemple, quand les participants émettent un doute par rapport au fonctionnement, le tuteur demande à chacun d'eux, station par station ce qu'ils voient.

L'utilisation du canal sonore par l'intermédiaire d'une audioconférence nécessite une autorégulation du droit de parole par les participants.

Au niveau de la tâche

La présence de plusieurs apprenants devant une station de Co-Learn est néfaste. Cette organisation induit une surcharge cognitive importante. Concrètement, nous pouvons dire que sur chaque site un apprenant utilisant le canal sonore doit gérer simultanément quatre espaces différents :

- l'espace de communication local : sur chaque site les apprenants présents communiquent afin de résoudre la tâche ;
- l'espace de communication verbale du groupe se trouvant dans la pièce virtuelle ;
- l'espace visuel fournit par les outils ;
- l'espace correspondant à la tâche où ils sont impliqués.

En plus, ils doivent gérer l'interaction avec les différents outils Co-Learn, les outils de l'espace privé appartenant à l'environnement Windows.

L'existence de sous-groupes (autour d'une station) dans le groupe nécessite d'abord que la communication et la structuration des sous-groupes se fassent, pour que la communication au niveau du groupe puisse se mettre en place.

Nous pouvons dire qu'il est important de réduire cette surcharge cognitive. La solution immédiate que nous préconisons est de prévoir une seule personne par station Co-Learn et de créer des pièces qui ont au départ peu d'outils.

Remarque : il faut préciser que la présence d'un groupe devant une station n'est pas le cahier des charges du début du projet. Ceci implique qu'il faut à notre avis repenser une partie de la conception afin de définir une station de groupe.

Cette station de groupe va modifier la téléprésence. En effet les autres utilisateurs doivent avoir conscience que derrière cette station se trouvent plusieurs utilisateurs. Il

est donc normal de percevoir plusieurs voix sur l'audioconférence. Il est important que le système fournisse une identification de chaque participant ou du sous-groupe se trouvant derrière la station. Cette identification est importante par rapport à des activités comme le vote.

Pour prendre en compte plusieurs personnes sur le même site, on pourrait mettre en place un ensemble de stations Co-Learn dans la même pièce. Dans ce cas, il faudrait prévoir une architecture réseau légèrement différente, en regroupant les stations locales derrière un routeur afin de n'utiliser qu'un accès S0. Ceci modifierait aussi le protocole de diffusion des informations sur le réseau pour des raisons d'optimisation et de coût. La figure 2 donne une vision de l'architecture physique possible.

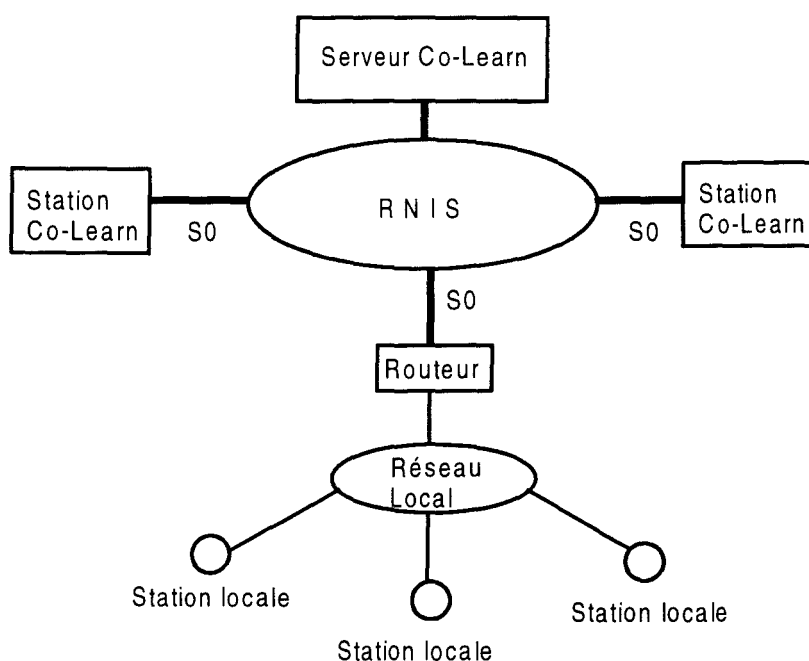


Figure 2 : Architecture pour prendre en compte un groupe d'utilisateurs sur un même site

Cette architecture physique peut aussi être intéressante dans des cas où les groupes sont éparpillés sur plusieurs pays. Elle permettrait d'optimiser les coûts de communication, en utilisant des communications locales et un seul lien longue distance pour chaque site. L'utilisation d'un réseau local ne prend pas en compte le canal sonore. Il faudrait donc un dispositif extérieur comme le téléphone, ou que ce réseau local prenne en charge le transport du son.

IV - Définition de l'utilité du prototype

Cette partie a pour but d'expliquer notre vision (celle de l'auteur de cette thèse) sur l'utilité du prototype dans le contexte de l'enseignement à distance. Son objectif est de définir la configuration matérielle et humaine idéale de telle façon à éliminer les problèmes qui ont été exprimés dans la partie précédente. La proposition suivante concernant le scénario pédagogique a simplement pour but de définir un scénario type, et de voir comment les outils Co-Learn peuvent être utilisés en fonction des différentes phases du scénario.

1 - La configuration matérielle et humaine

La configuration idéale semble être les stations Co-Learn qui intègrent le son et son contrôle (Classe B2S) équipées d'un microcasque, devant laquelle se trouve un seul participant. L'équipement physique permet d'assurer une meilleure gestion du canal sonore, une disponibilité totale de l'utilisateur pour celui-ci, et un isolement en terme de bruit pour l'audioconférence (bruits générés par l'environnement local). Il faut sûrement penser au confort de l'utilisateur : porter un microcasque pendant une heure n'est pas agréable. Nous pensons aussi qu'il sera nécessaire de permettre aux utilisateurs de quitter momentanément une pièce virtuelle, bien sûr en prévenant les autres de son absence temporaire.

2 - Un scénario type pour la téléconférence temps réel

Dans la figure 2, nous essayons de donner une représentation d'un scénario générique adapté à la pièce de conférence temps réel.

Nous avons découpé l'activité du groupe en phase dans lesquelles nous suggérons l'utilisation de certains outils, et nous identifions les types de documents qu'il est possible d'utiliser. Les traits en pointillés expriment l'utilisation de documents pendant la phase concernée. Les autres précisent l'enchaînement temporel.

Les documents utilisés dans une phase vont permettre d'expliquer le problème et d'exprimer les objectifs à atteindre dans la tâche à réaliser. Leur caractéristique est d'être indépendant du scénario pédagogique choisi. Les documents annexes vont faciliter le travail dans les autres phases du scénario pédagogique. Ils vont être adaptés au public choisi, à leur niveau de connaissance, au degré d'autonomie souhaité par le formateur pour les apprenants. Ils vont aussi être adaptés aux logiciels de l'espace privé. Ces documents vont aussi permettre d'obtenir un cadre de référence pour la présentation dans l'outil de visualisation collectif (papier millimétré, repère orthonormé).

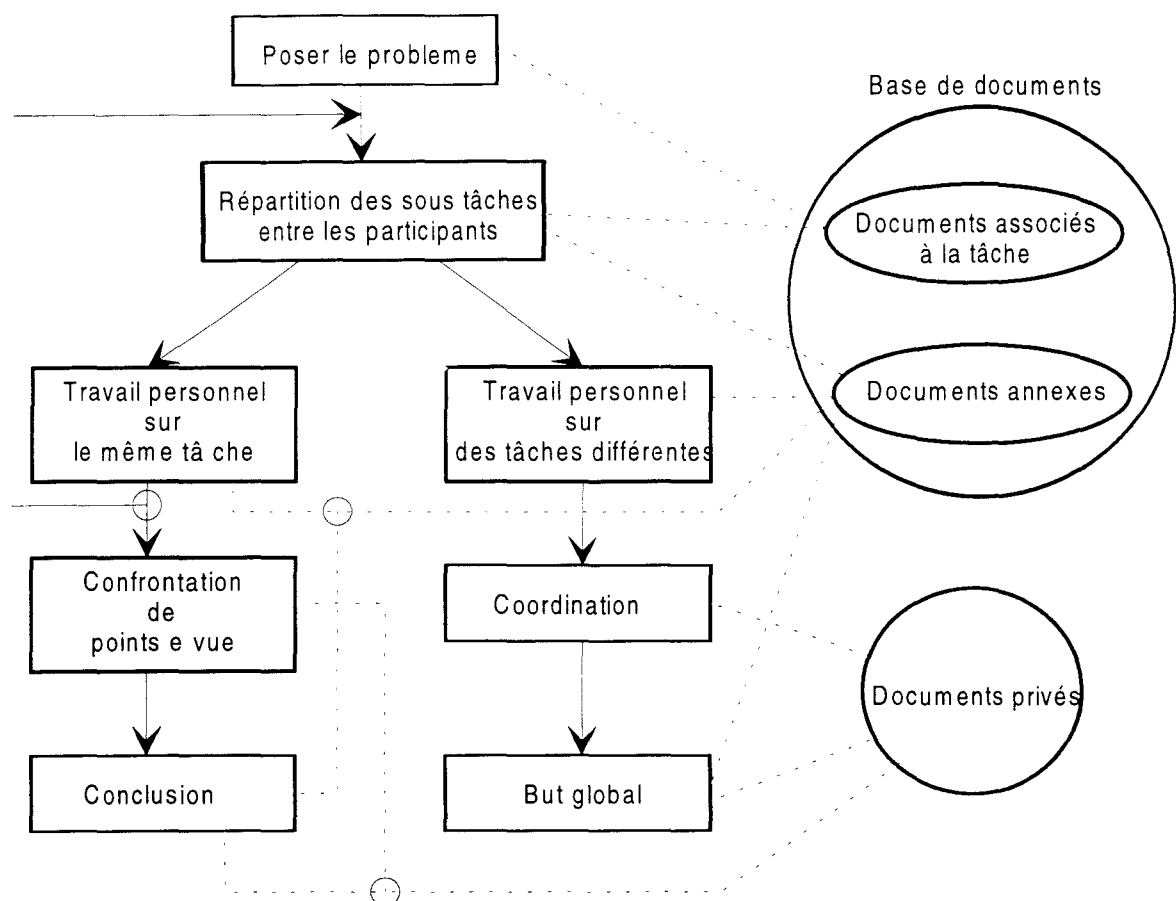


Figure 2 : Scénario type pour la pièce de conférence temps réel

La répartition des tâches entre les participants peut se faire de plusieurs façons. Ici nous en avons identifié deux. La première est le cas où les participants travaillent sur la même tâche afin d'aboutir à une phase de confrontation de points de vue ou de confrontation de démarches (stimuler et créer les conflits socio-cognitif). La deuxième dans laquelle la tâche globale est répartie en sous-tâches qui peuvent être réalisées de façon indépendante, mais une coordination entre tous les groupes est nécessaire afin d'atteindre le but global (construction sociale du savoir).

Ces deux modèles ne sont pas exclusifs. Au contraire le formateur devra souvent faire face à un mélange (entrecroisement) de ces deux scénarios type.

Pourtant, il nous apparaît relativement important d'avoir en tête ces deux types de scénario, car ils vont avoir une influence sur l'utilisation des outils, et également sur les documents à préparer par le formateur.

3 - Un exemple d'utilisation possible du système de conférence

3.1 Phase de présentation du problème

Cette phase sera faite par le tuteur (plus généralement le médiateur) pour expliquer le problème, en utilisant des documents visuels qu'il aura préparé à l'avance.

Dans cette phase, les outils à privilégier sont l'outil de visualisation collectif et le télépointeur.

Il est aussi important de préciser la démarche globale de la session en insistant sur l'utilisation du monde privé par les participants. Il faut aussi donner quelques consignes pour l'utilisation des outils privés, et préciser que des documents annexes (quadrillage, repère, etc) sont disponibles pour travailler dans cet espace.

3.2 Répartition des sous-tâches

Pour permettre des échanges entre participants dans les phases suivantes, il faudrait prévoir dans l'outil de visualisation collectif un document qui "mémorise" les tâches à faire et les participants concernés. Ceci permettrait d'aider les participants à communiquer (1 à 1) en utilisant par exemple l'outil de dialogue surtout si les participants ont des tâches identiques. La conception du document peut être faite à l'avance si le formateur affecte arbitrairement les tâches, ou en utilisant un outil de son espace privé et la communication entre espace privé et public (presse papier collectif).

3.3 Travail personnel (même tâche, tâches différentes)

Il est nécessaire pour la poursuite de la session que des documents visuels soient réalisés par les participants : pour assister la conversation et permettre des comparaisons ou des confrontations. Ceux-ci sont nécessaires pour démarrer la discussion (condition de création du conflit socio-cognitif).

Dans cette phase, l'utilisation de l'audioconférence risque d'être perturbante pour les utilisateurs non intéressés par les échanges. Le médiateur doit réguler l'utilisation du canal sonore, voire proposer d'autres moyens de communication (par exemple l'outil de dialogue). Au contraire si l'information est pertinente pour le groupe, il doit faire la démarche de capter l'attention de tout le monde.

Dans cette phase, l'outil de dialogue peut être d'une grande utilité pour une communication bipartite, et l'audioconférence pour la communication au sein du groupe.

Une réflexion préalable du formateur sur les problèmes que vont rencontrer les apprenants pendant la session est importante, car ils lui permettront de créer des documents qui pourront être utilisés pendant cette phase, afin de gagner du temps.

V - Etude d'utilisabilité du système

La seconde évaluation du système que nous voulons relater est une étude d'utilisabilité réalisée par Carmel Smith¹⁰. Les résultats de cette analyse sont publiés dans un rapport de Delta [Smith 94].

Le processus d'évaluation est le résultat de deux études :

- une première qui correspond à une expertise de l'interface réalisée par des spécialistes de l'interface homme machine à partir de l'utilisation du système et des guides utilisateurs ;
- une deuxième relative à l'observation d'utilisateurs (tuteurs et apprenants) mis en situation réelle.

Les conclusions suivantes ont été corroborées par d'autres expérimentations réalisées dans d'autres sites pilotes¹¹ (par exemple au CUEEP).

Notre objectif n'est pas de décrire le processus d'évaluation et toutes ses conclusions, mais seulement d'énoncer les conclusions en relation avec les métaphores de l'interface et les conclusions spécifiques sur le prototype RTTC. Ces conclusions seront complétées et commentées en donnant notre point de vue.

La métaphore de Pièce

La métaphore de pièce et la façon dont son utilisation a dirigé la conception, est peut être mal appropriée et a dirigé le projet dans une mauvaise direction [Smith 94]. Cette conclusion nous paraît excessive mais les propositions qui en découlent sont à considérer sérieusement.

Dans la conception d'un centre de ressources pédagogiques virtuel, cette métaphore représente plus le point de vue de l'administrateur : par exemple pour l'affectation des ressources [Derycke 95]. En effet, la métaphore de Pièce est une façon de structurer un centre de ressources virtuel complexe. A une pièce, on associe l'activité d'un groupe. Elle met en jeu des objets sociaux qui partagent la même activité d'apprentissage (par exemple un groupe de personnes suivant le même cours). On prépare la pièce pour une activité particulière en lui affectant des ressources, qui correspondent à une collection d'outils privés (pour un utilisateur) et publics (pour le groupe). L'utilisateur qui accède au système navigue dans le centre de ressources

¹⁰Carmel Smith travail à ICBL Institute for Computer Based Learning de l'Université Herriot Watt. Elle est intervenue dans le projet comme spécialiste de l'interface homme machine et plus particulièrement de l'évaluation de ces interfaces.

¹¹Les sites pilotes sont les instituts ou les organisations qui sont en charge de l'évaluation de l'environnement en situation réelle

virtuel en recherchant les ressources dont il a besoin. On peut dire que nous disposons d'un mode de navigation par les ressources.

Un défaut lié à l'utilisation de la métaphore de Pièce par le Launcher¹² est la notion de typage arbitraire des pièces (4 types) et leur représentation sous forme d'une carte statique ([Smith 94] page 45). Une analyse du modèle conceptuel montre que la mise en oeuvre de la métaphore de Pièce lui donne une caractéristique statique. Cette représentation devrait plutôt être dynamique afin de pouvoir s'adapter à l'organisation dans laquelle l'environnement est utilisé, et s'adapter à l'utilisateur pour lui montrer l'ensemble des ressources auxquelles il a accès.

La notion de typage des pièces est due à notre avis à la séparation du développement des pièces génériques entre deux partenaires : Télésystème pour les pièces de télécours et de téléassistance, et le Laboratoire Trigone pour les pièces de conférence temps réel (ou atelier) et du forum (conférence asynchrone). En ce qui nous concerne (Trigone), les outils présents actuellement dans le forum et l'atelier peuvent être mixés dans une pièce : c'est à dire que l'on peut trouver une fenêtre de visualisation publique dans une pièce dite de forum et inversement trouver un browser¹³ de tâches dans une pièce dite de type atelier. Avec cette approche générique en terme d'outil, la pièce devient un lieu où l'on trouve des moyens de communication qui peuvent être synchrone ou asynchrone. C'est la malléabilité structurelle de l'environnement qui permet cela.

L'utilisation de la métaphore de Pièce dans l'environnement Co-Learn, ne prend pas en compte la dimension temporelle, qui est peut être le point d'entrée le plus adapté pour l'utilisateur. En effet, l'utilisateur est plus concerné par l'ensemble des activités auxquelles il doit participer. Sa recherche dans le centre de ressources doit être dirigée par l'espace des activités et non par la répartition géographique de pièces. Cet espace est dynamique et possède une dimension temporelle qui n'est pas évidente à prendre en compte avec une métaphore spatiale. La métaphore de cahier de texte (emploi du temps) serait peut être une bonne candidate. Elle permettrait à l'utilisateur de connaître quel est le cours qu'il doit suivre, et plus généralement les pièces qui concernent sa propre activité. Le système pourra alors le "téléporter" directement dans la pièce où le cours a lieu. Du point de vue de l'environnement Co-Learn, ceci peut signifier que la pièce identifiée comme le Hall devrait fournir l'accès à un "cahier de texte" adapté à l'utilisateur. La métaphore de la pièce reste valide mais c'est le mode de navigation dans l'organisation qui change. Pour les administrateurs de cours, la métaphore de la

¹²Le launcher est le logiciel qui prend en charge la gestion du centre de ressource virtuel ainsi que la navigation des utilisateur dans ce centre. Il a été développé par nos collègues de Software de Base (Madrid) et a déjà été présenté dans le chapitre IV

¹³Le browser de tâche est un outil qui permet de créer des tâches et de visualiser le planning associé

Pièce est très structurante. A notre avis, elle permet facilement de comprendre la préparation d'une pièce virtuelle qui correspond à la définition d'un ensemble d'outils qui vont pouvoir être engagés dans l'activité.

La prise en compte de la dimension temporelle dans le centre de ressource virtuel pourrait aussi apparaître dans le mécanisme d'entrée dans une pièce. Actuellement, l'utilisateur se retrouve devant une porte correspondant à la pièce où il veut entrer. Il suffit qu'il pousse cette porte pour participer à l'activité menée dans la pièce. Ici encore la mise en oeuvre de la métaphore de pièce ne permet pas à l'utilisateur de connaître l'état de la pièce. Par exemple dans une pièce temps réel, l'utilisateur ne sait pas si la conférence vient de commencer ou va commencer. Ceci pourrait être pris en compte par le gérant de conférence associé à la pièce qui pourrait définir une vision externe de la conférence en cours. On pourrait définir un "sas d'accueil" qui permettrait à l'utilisateur qui entre, de connaître l'état de la pièce (personnes présentes, activité en cours, heure de début de la conférence ...).

En conclusion, on peut dire que la métaphore de Pièce peut avoir une mise en oeuvre qui permette de prendre en compte des aspects temporels. On pourrait définir des métaphores dites de second degré, comme le cahier de texte pour offrir un moyen de navigation efficace dans le centre de ressources pédagogiques virtuel, et définir un "sas d'entrée" qui permette de définir l'entrée dans une pièce en fournissant des informations sur l'état de la pièce.

Les outils de la pièce de conférence temps réel

Les évaluations menées par Carmel Smith [Smith 94] et nos propres évaluations [Croisy 94c] ne permettent pas actuellement de conclure sur l'utilisabilité des outils impliqués dans la pièce de conférence temps réel. Cependant Carmel Smith explicite certains indicateurs susceptibles d'être à l'origine de problèmes d'utilisabilité.

Les utilisateurs des outils peuvent avoir différents rôles, comme animateur, tuteur ou apprenant. Certains outils possèdent plusieurs modes de fonctionnement, comme le télépointeur et le collecteur d'opinion. En fonction du rôle de l'utilisateur, les outils auront un comportement différent dans chaque mode. [...], certains outils ont aussi des états, par exemple actif, initié, bloqué, demandé, etc. L'interface associée à l'outil donne peu d'information sur les rôles des utilisateurs, le mode et l'état des outils.

Ceci pose le problème de l'interprétation et de la compréhension du fonctionnement d'un outil par un utilisateur dont l'interface fournit peu d'information. Cette constatation est importante puisqu'elle remet en cause un choix que nous avons fait pour répondre à la question : **Que doit on montrer à l'utilisateur comme information sur le**

mécanisme définissant le partage d'un outil ? Nous avons fait un choix minimaliste, qui consiste à montrer le minimum d'information à l'utilisateur. C'est à dire qu'il est informé de l'état de l'outil par rapport à son rôle (à un instant donné les états peuvent être différents), et de l'ensemble des actions qu'il a le droit de faire. L'interface de l'outil ne donne pas d'information sur le rôle des autres participants. Nous avons choisi de faire évoluer l'interface en même temps que les phases et les états, en cachant les fonctions devenues inaccessibles et non en les montrant dans un état inactif (par exemple une icône grisée). Ce choix un peu arbitraire a été fait pour simplifier l'interface de l'outil qui parfois est déjà complexe.

En conclusion, il est important de réfléchir à la question suivante : **Faut-il montrer l'ensemble des fonctionnalités accessibles à l'utilisateur, en lui précisant celles qui sont actuellement disponibles et celles qui ne le sont pas ?** Nous pensons qu'il sera très difficile de répondre à cette question, mais nous avons certaines hypothèses pour y arriver. Le choix fait dans Co-Learn de cacher les fonctionnalités non actives pour un rôle, permet de simplifier l'interface. Le choix contraire qui serait de montrer les fonctions non actives pour le rôle, pourrait renforcer la notion de téléprésence [Buxton 92] et de groupe. En effet, les utilisateurs pourraient se poser la question de savoir pourquoi ils n'ont pas accès à ces fonctionnalités, et dans certains cas demander ces accès pour mieux gérer leur participation dans l'activité du groupe. La perception des fonctions inactives serait une information utile, si on veut que l'utilisateur participe à la construction de son interface. Une fonctionnalité présentée comme non accessible à un utilisateur peut être une information sur l'activité des autres. Par exemple, dans l'outil de visualisation collectif lorsqu'un utilisateur utilise la fonction de chargement d'un document, les autres voient l'item du menu en grisé. Ceci traduit que la fonction est inaccessible car quelqu'un l'utilise. Ceci renseigne les utilisateurs sur l'activité d'un participant, sans donner son identification au niveau de l'outil. **Quelle est l'information nécessaire et utile au niveau de l'interface des outils qui permette de renforcer la téléprésence ?**

Dans cette première phase d'identification de certains critères susceptibles d'apporter des problèmes d'utilisabilité, Carmel Smith met en évidence le problème de perception de l'intention des participants. Il est important de savoir ce que les autres sont en train de faire, mais aussi d'avoir une perception sur ce qu'ils veulent faire. Par exemple, l'outil audio n'intègre pas de fonctionnalités permettant de savoir qui parle, qui veut parler, pendant combien de temps. Co-Learn ne permettant pas la transmission de toute la métacommunication, la gestion du canal verbal est un véritable problème. Dans le cas de l'environnement Co-Learn, Carmel Smith souligne que la gestion du droit de parole est souvent liée à la gestion du télépointeur qui permet un certain niveau de support de la gestuelle (composant de la métacommunication). Dans ce cas, le fait de

demander le télépointeur signifie l'intention de la part d'un utilisateur de prendre la parole. Cette constatation est importante et doit être confirmée car elle peut avoir des répercussions sur les fonctionnalités offertes par une audioconférence qui peuvent être en inadéquation avec le comportement d'un outil de prise en charge de la gestuelle. Le concepteur de collecticiel ne doit pas seulement fournir plusieurs canaux de communication, mais aussi vérifier la cohérence et la cohabitation entre ceux-ci.

Conclusion

Nous pouvons dire que les expérimentations et les évaluations de la pièce temps réel ont permis de poser certains problèmes importants pour la réussite des collecticiels : prise en compte de la téléprésence, perception de l'activité des autres.

Co-Learn n'a pas permis actuellement de répondre à ces questions. Mais il a permis au moins de les poser et de les situer dans un contexte d'utilisation réelle. L'architecture et les caractéristiques de l'environnement (malléabilité structurelle et fonctionnelle) vont permettre de définir différentes configurations qui font varier certains paramètres, tels que la téléprésence et le niveau de rétroaction, etc. Cette étude nous a permis de trouver certaines caractéristiques à observer. Maintenant il faut étudier l'impact de la variation de ces paramètres sur l'activité du groupe. Il est important de construire des modèles de fonctionnement des groupes qui nous permettent de dialoguer avec les futurs utilisateurs ou plus généralement l'organisation concernée, afin de pouvoir configurer le système.

CHAPITRE VI

PROPOSITIONS ET RECOMMANDATIONS

POUR LA PROCHAINE GENERATION

DE SGIU¹

L'objectif de ce chapitre est tout d'abord de proposer un modèle d'architecture multi-agent qui permette de modéliser des systèmes interactifs de groupe. Cette contribution sur les architectures sera suivie par une liste de recommandations pour les futures générations de systèmes de fenêtrage et de boîtes à outils afin qu'ils permettent de réaliser plus facilement des collecticiels. On peut formuler la problématique sous forme d'une question : si nous considérons que toutes les applications sont des collecticiels, et que les applications mono-utilisateur sont des collecticiels utilisés dans un contexte social particulier, quelles sont alors les répercussions sur la conception des SGIU ?

¹SGIU Système de gestion d'Interface Utilisateur traduction de UIMS User Interface Management System

I- Pourquoi une approche multi-agent ?

1 - Les caractéristiques d'un modèle multi-agent

Avant d'essayer de caractériser les modèles multi-agent, il nous semble intéressant de définir ce qu'est un agent.

Un agent peut être défini comme un système réactif à certains événements qui sont générés par l'environnement dans lequel se trouve l'agent, ou par d'autres agents. L'agent qui réagit à ce stimuli (événement) subit des transformations et génère de nouveaux stimuli [Abowd 90]. Un agent peut être considéré comme un système de traitement d'information composé d'un récepteur, d'un émetteur, et d'une mémoire à deux niveaux qui enregistre les événements venant du récepteur et qui est capable de traiter plusieurs classes d'événements² [Coutaz 90]. Un événement peut être caractérisé par les éléments suivant : il appartient à une classe, il véhicule de l'information, il est produit par un émetteur et détecté par un récepteur.

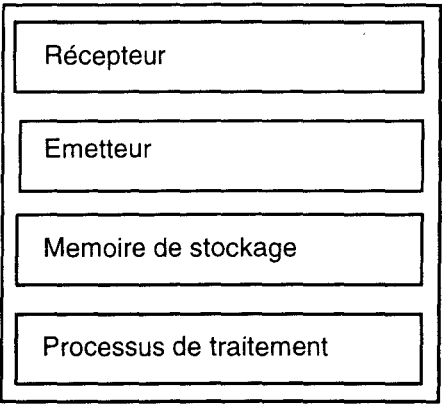


Figure 1 : Un modèle d'agent

La dynamique d'un agent peut être défini comme suit : le récepteur qui agit comme un filtre capte les classes d'événements que peut traiter l'agent, l'événement est traité par le processus de traitement, ce qui a pour conséquence de changer l'état de l'agent, et de générer d'autres événements.

Bhandaru et Craft [Bhandaru 90] donnent une description d'un modèle d'agent qui complète la précédente, en définissant les "connaissances" contenues dans un agent qui rendent possible la coopération entre agents. Ce modèle est conçu pour définir une architecture qui permet le travail coopératif orienté tâche (la coopération à un objectif définit). Les "connaissances" d'un agent dans ce contexte sont les suivantes :

- la connaissance de l'objectif à atteindre ;

²Le terme classe à la même signification que dans une approche objet

- la connaissance liée au domaine : l'agent possède les connaissances nécessaires pour atteindre l'objectif cité précédemment ;
- il possède un plan d'actions ;
- il possède un rôle : ceci définit la responsabilité de l'agent par rapport aux autres agents ;
- il possède une "capacité" : ceci définit les rôles potentiels de l'agent ;
- il gère des priorités : il existe une pondération entre les différentes activités menées par l'agent qui définit un niveau de priorité ;
- il possède un mécanisme d'anticipation sur son activité future.

Pour conclure sur la définition d'agent, nous énumérons certaines propriétés qui doivent être communes à tous les agents [Cardozo 93 page 71] :

- les agents sont autonomes dans le sens où ils peuvent définir leurs propres objectifs et leur propre démarche ;
- les agents sont capables de prendre part à des interactions complexes en utilisant des primitives de communication de haut niveau indépendantes du domaine dans lequel ils sont utilisés ;
- les agents sont indépendants de l'architecture ;
- les agents sont capables de percevoir les changements dans l'environnement auquel ils appartiennent, de changer leur comportement, et d'intégrer ces changements dans leur propre modèle interne ;
- les agents sont capables de prendre en compte les objectifs et les compétences (savoir faire, responsabilité) des autres agents, afin de pouvoir résoudre collectivement les problèmes.

Dans une approche multi-agent de l'interface, le système interactif est vu comme un ensemble d'agents spécialisés qui réagissent à des événements et qui produisent des événements [Coutaz 90]. La notion de spécialisation sera illustrée par la description de modèle multi-agent existant. Concevoir un modèle multi-agent c'est définir un modèle de coopération, de collaboration, et un modèle de communication entre plusieurs agents afin que ceux-ci puissent prendre en charge le dialogue entre une application et un utilisateur.

Apports des modèles multi-agent

- indépendance entre l'application et l'interface (principe de base des SGIU) ;
- répartition de la complexité du système interactif entre plusieurs agents spécialisés ;
- modularité dans la conception, réutilisabilité dans l'implémentation.

2 - Quelques exemples de modèles multi-agent dans les interfaces homme machine

Bien que nous nous intéressions au modèle multi-agent pour les interfaces de groupe³, nous pensons qu'il est intéressant de présenter des modèles conçus pour les systèmes interactifs mono-utilisateur (sans avoir de préjugés sur leur utilisation pour les systèmes interactifs de groupe) et des modèles prévus pour concevoir des systèmes interactifs de groupe. Ceci nous permettra d'illustrer la spécialisation d'agent, la communication entre agents, et de voir comment certains modèles ont été étendus pour essayer de définir les caractéristiques qui ont permis de les étendre.

2.1 - Le modèle PAC

PAC [Coutaz 90] est un modèle multi-agent donc chaque agent est composé de trois entités :

- la **Présentation** qui définit l'image que perçoit l'utilisateur de l'agent : représentation externe ;
- l'**Abstraction** qui est l'ensemble des concepts et des fonctions de l'agent : fonctionnalités sémantiques ;
- le **Contrôle** dont le rôle est de maintenir la cohérence entre l'Abstraction et la Présentation. Il joue un rôle d'arbitre entre les composants pour résoudre les conflits, les rafraîchissements, etc, ainsi qu'un rôle de traducteur entre les structures de données manipulées par la Présentation et celles de l'Abstraction qui sont différentes.

Une application basée sur le modèle PAC décompose l'application interactive en un ensemble d'agents PAC structurés hiérarchiquement de manière récursive (Figure 2).

³Nous préférons le terme interface de groupe au terme interface multi-utilisateur, ce dernier n'incluant pas forcément une dimension coopération entre les utilisateurs

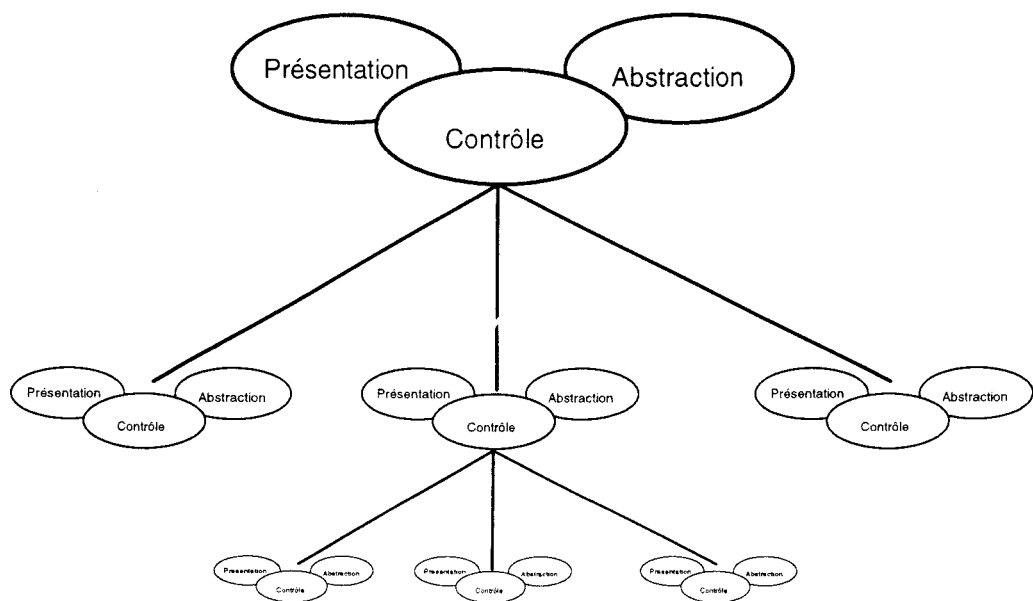


Figure 2 : Le modèle PAC d'un système interactif

Le modèle décompose le dialogue homme système interactif en un ensemble de petits dialogues : dialogue à plusieurs fils. Ceci permet un certain niveau de parallélisme dans les traitements des interactions, un agent PAC étant une unité d'exécution.

Ce qui peut être intéressant dans ce modèle pour les systèmes interactifs de groupe, c'est cette décomposition fine du système interactif qui peut permettre une granularité fine pour le partage des objets de l'application entre les utilisateurs, et la dimension parallélisme qui peut permettre les interactions simultanées d'un utilisateur et la prise en compte simultanément de la rétroaction associée à l'activité des autres utilisateurs.

2.2 - Le modèle MVC

MVC est un modèle multi-agent utilisé dans l'environnement Smalltalk80 [Golberg 84][Dugerdil 90]. Ce modèle est composé d'un triplet d'agents : le **Modèle**, la **Vue** et le **Contrôleur**.

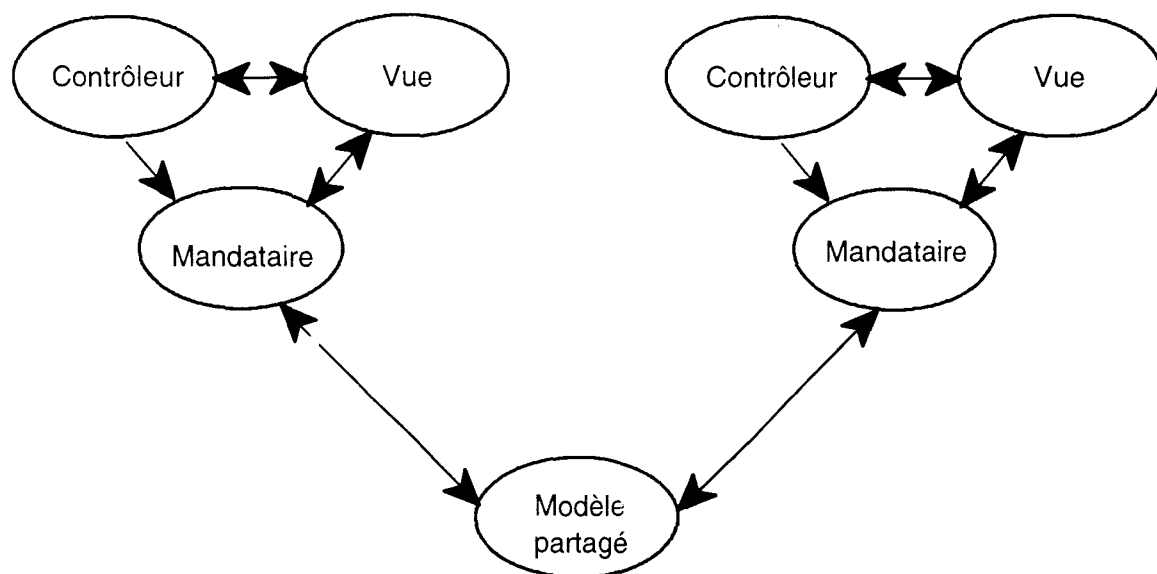


Figure 4 : Le modèle MVC étendu

La notion de rôle et de droit est prise en compte au niveau du contrôleur, en fournissant à l'utilisateur un menu en fonction de son rôle. Les changements de rôle sont pris en compte par un gérant de conférence distribué sur les différents sites.

Nous avons aussi travaillé [Croisy 91] sur une plus forte distribution du Modèle Partagé, en réfléchissant sur un mécanisme qui permettrait un certain niveau de délégation sémantique du Modèle vers les Mandataires. Cette étude théorique a montré la faisabilité du mécanisme, et aurait sûrement augmenté les performances de l'interface homme machine, mais complexifié fortement la mise en oeuvre.

2.5 - Rendezvous et le paradigme ALV

Nous avons déjà présenté ce modèle dans le chapitre I. Ce modèle est proche du modèle PAC. On peut faire une analogie entre la Présentation et la Vue, le Contrôleur et le Lien, l'Abstraction représentant le même composant. L'avantage de ce modèle ainsi que de MVC est d'avoir un coupage faible entre les différents Agents, voire une indépendance quasi totale, qui permet facilement de distribuer le modèle pour créer une architecture d'accueil pour collectif relativement centralisée.

L'originalité de ce modèle est d'utiliser un système de propagation de contraintes pour réaliser les Liens. Cette décomposition du système interactif et le mécanisme de propagation de contraintes permet de lier une Abstraction (développée en RendezVous) avec une Vue (développée aussi en RendezVous) sans avoir pensé à la dimension multi-utilisateur. Dans [Brinck 93], on trouve un exemple qui illustre cette liaison "a posteriori" entre un jeu de Tic-Tac-Toe utilisé comme Abstraction et un éditeur graphique utilisé comme Vue.

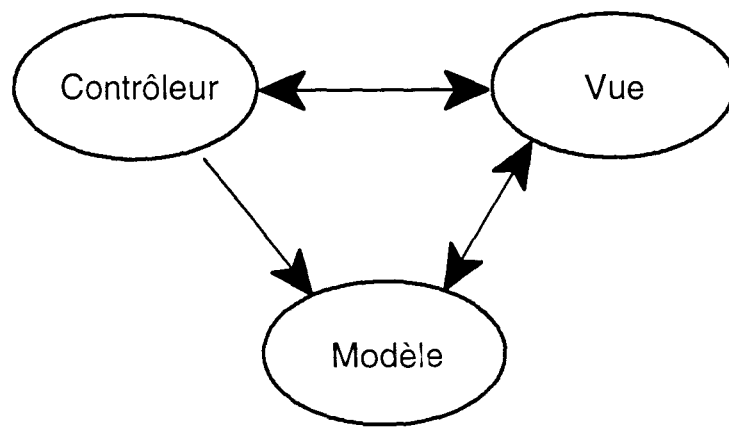


Figure 3 : Le modèle MVC

Le **modèle** contient les structures de données et les manipulations associées à l'information qui sera représentée à l'écran ;

La **vue** est la représentation externe du modèle. C'est par son intermédiaire que l'utilisateur perçoit les données associées au modèle et que le modèle rend compte de la modification de celles-ci à l'utilisateur.

Le **contrôleur** prend en charge les interactions de l'utilisateur réalisées par l'intermédiaire de la souris ou du clavier. Lorsque l'utilisateur manipule la Vue, le Contrôleur informe le Modèle des interactions faites. Ce dernier modifie alors son état et informe la Vue que les informations qu'elle affiche doivent être mise à jour.

La caractéristique intéressante dans le modèle MVC est la possibilité d'associer plusieurs Vues et Contrôleurs au même Modèle, alors qu'une Vue ne peut être associée qu'à un seul Modèle.

La modularité du modèle MVC permet une réutilisabilité des composants Vue et Contrôleur. Cette modularité et le faible niveau de couplage entre les différents agents sont intéressants si on veut répartir ce modèle sur différents sites. Cette constatation a permis de définir le modèle MVC étendu présenté ci-dessous.

2.3 - Le modèle MVC étendu

L'objectif de ce modèle est de supporter les interactions multi-utilisateur simultanées. La démarche repose sur une distribution du modèle MVC [Vilers 92][Croisy 91] : un exemplaire des composants Vue et Contrôleur par utilisateur, et un Modèle centralisé. Sur chaque site utilisateur, un objet mandataire est répliqué et joue le rôle de Modèle pour les Vues et Contrôleurs locaux. Ceci permet de conserver les relations qui existaient dans la triade MVC. Le mandataire qui représente le Modèle a pour rôle principal la redirection des messages venant de la Vue et du Contrôleur vers le Modèle Partagé. Ce Modèle Partagé traite les messages venant des différents mandataires.

Le mécanisme ALV prend bien en charge la dimension multi-utilisateur en offrant la mise à jour automatique des Vues lorsque l'Abstraction est modifiée, mais il est très pauvre en ce qui concerne la gestion de la coopération. Par exemple, il apparaît que la définition de l'algorithme de contrôle doit être prise en charge par l'Abstraction. Dans le cas du Tic-TacToe, si on définit un mécanisme de verrouillage pour gérer le tour de jeu, l'Abstraction doit contenir un composant qui exprime ce verrouillage. Nous pensons que ce malaise est lié à toute approche qui ne veut pas considérer dès le départ une application comme coopérative.

3 Conclusion sur le modèle multi-agent pour la conception de collecticiels

Nous voulons préciser ici quelles sont les caractéristiques d'un modèle multi-agent qui permettent ou permettront facilement de l'étendre pour qu'il permette la conception d'interface de groupe.

La séparation entre l'interface et l'application permet facilement de construire plusieurs interfaces utilisateur pour la même application en définissant un mécanisme d'accès concurrent à l'application. Ce principe de séparation offre trois démarches de distribution possibles [Dewan 92] :

- distribution de l'application : le noyau fonctionnel du système interactif est distribué ;
- distribution de l'interface utilisateur : l'interface utilisateur du système interactif est distribuée ;
- distribution des liens : le noyau fonctionnel et l'interface utilisateur sont distribués l'un par rapport à l'autre.

Concevoir un modèle multi-agent demande tout d'abord d'identifier les agents spécialisés qui vont prendre en charge le dialogue, puis d'identifier un modèle de distribution de ces agents, et enfin d'identifier un modèle de communication entre ces agents afin de prendre en compte la dimension multi-utilisateur.

La distribution de certains agents entraîne la création de conflits car le dialogue est à "plusieurs fils". C'est à dire que les agents associés à l'application vont devoir gérer plusieurs activités en parallèle et de plus celles-ci peuvent être conflictuelles. Dans Rendezvous, et MVC étendu, le problème est résolu par une séquentialisation des différentes activités. C'est à dire qu'une priorité est donnée à une activité. Ceci se traduit au niveau des agents par une sérialisation des événements.

En conclusion, l'approche multi-agent est intéressante pour la conception de système d'interface de groupe, car elle offre plusieurs possibilités dans la distribution des différents agents. Ils permettent une gestion simple (a priori) de la gestion des conflits engendrés par les actions simultanées de plusieurs utilisateurs, en proposant un filtrage sur les événements générés et traités par les agents.

Les modèles actuels ne prennent pas en compte certains enjeux des collecticiels qui doivent être adaptables à l'activité du groupe. Ce qui est nécessaire c'est d'adapter le comportement des agents à la nature même de l'activité, par la prise en compte de rôle dynamique, l'évolution de l'interface en fonction du rôle, la décomposition de l'activité en phase qui entraîne la modification du dialogue entre les utilisateurs et l'application, et enfin la prise en compte de plusieurs médias.

II - Un modèle d'architecture multi-agent pour système interactif de groupe

1 - Quels sont les objectifs du modèle en fonction des enjeux du collectif

Avant de présenter le modèle, il est important de définir ce que nous appelons activité et ses caractéristiques. Cette définition est proche de celle définie par le modèle Amigo [Pankoke 89][CoLearn 92b] et a déjà été abordée dans le chapitre III. Pour résumer, une activité va mettre en jeu un ensemble d'agents qui communiquent afin de mener à bien cette même activité. Elle est décomposée en phases, au sein desquelles chaque agent possède un rôle particulier. Certains messages vont cadencer les changements de phase. Les rôles sont dynamiques, c'est à dire que l'on peut changer de rôle durant une session.

Dans le paragraphe suivant nous proposons notre modèle d'architecture, basé sur la conception d'agents spécialisés qui prennent en charge le dialogue groupe application. Ce modèle repose sur la décomposition d'une activité selon Amigo, la création d'agents spécialisés dans la communication et la gestion d'un rôle, dans la gestion des styles d'interaction et de rétroaction, accompagné d'un mécanisme de délégation sémantique qui donne la dimension dynamique et paramétrable au système.

2 - Les composants du modèle

La figure 5 propose un schéma représentant les différents agents, en montrant les flux d'information et la relation de délégation.

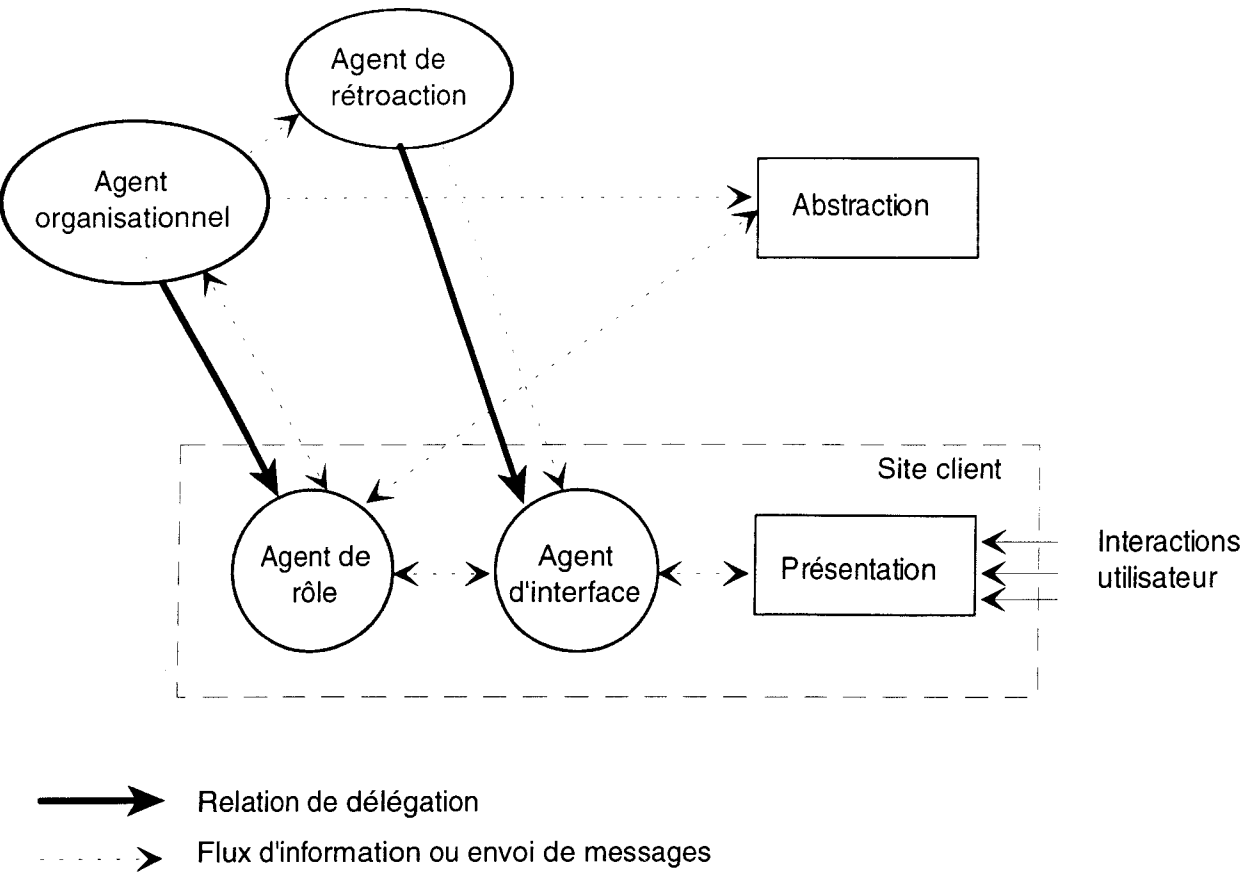


Figure 5 : Un modèle multi-agent pour les systèmes interactifs de groupe

Le composant Abstraction

Il contient l'ensemble des concepts et des fonctions associés au domaine. Il définit la sémantique de l'application (ceci correspond à l'Abstraction du modèle PAC).

Le composant Présentation

Ce composant est celui qui va permettre à l'utilisateur d'interagir sur l'activité (de la faire évoluer dans le temps). Il prend en charge l'aspect visuel associé à l'activité coopérative : représentation des objets abstraits de l'Abstraction et représentation du rôle de l'utilisateur.

L'agent organisationnel

Son rôle est d'avoir une vision globale sur l'activité afin d'être une entité de décision par rapport au groupe. Nous le nommons Agent Organisationnel car il définit le comportement des groupes dans une organisation. Il doit donc être capable de s'adapter à celle-ci : c'est lui qui définit la malléabilité structurelle (voir chapitre III).

Ses connaissances :

Il connaît l'ensemble des rôles possibles des utilisateurs, les droits associés à chaque rôle dans une phase précise de l'activité en fonction de l'état de celle-ci (droit

de créer un objet, de demander une ressource), l'ensemble des politiques qui permettent de gérer les conflits, et enfin la politique active.

Lien de communication :

Il communique avec les Agents de Rôle associés aux utilisateurs, et reçoit les messages de ceux-ci. Il communique avec l'Abstraction lorsque la politique appliquée agit comme un filtre afin de gérer les accès concurrents à l'Abstraction. Il communique aussi avec l'Agent de Rétroaction lorsqu'il agit comme un filtre par rapport aux modifications du mécanisme de rétroaction.

Processus de traitement

Il prend en compte l'arrivée et le départ d'un utilisateur, en instanciant un Agent de Rôle pour celui-ci, ou bien on peut le considérer comme une agence qui a la possibilité d'envoyer un Agent de Rôle sur le site client afin qu'il prenne en charge le dialogue entre l'utilisateur et l'agence. Il gère les messages qui interviennent sur l'état ou la phase de l'activité, en appliquant la politique courante de gestion de l'activité. Ceci peut entraîner des envois de messages à l'Abstraction ou l'Agent de Rétroaction lorsque les décisions prises les concernent.

Agent de Rétroaction

Il définit quand et comment doivent être répercutées les actions d'un utilisateur sur les autres sites. Il est défini pour contrôler le niveau de rétroaction entre les participants. Par exemple, si un utilisateur commence à tracer un rectangle, cet agent décide si tous les mouvements doivent être visibles par tous les utilisateurs ou pas. Dans ce cas, il s'agit de définir si on fonctionne dans un mode Wysiwis strict ou un mode Wysiwis relâché. Il ne filtre pas lui-même les événements générés par la Présentation, mais il délègue cette responsabilité à l'Agent d'Interface. L'Agent de Rétroaction fournit un filtre à l'Agent d'Interface de telle façon que celui-ci ne fournisse aux autres Agents d'Interface que les messages qui sont nécessaires pour le respect du mode de rétroaction. Ce système de spécification de la rétroaction à l'extérieur des applications, nous permet de prévoir une adaptation de la rétroaction aux besoins du groupe sans intervenir sur l'Abstraction et la Présentation.

Ses Connaissances

Il possède des filtres qui permettent de définir les messages qui doivent être diffusés en fonction du mode de rétroaction choisi. Actuellement, nous percevons deux modes de fonctionnement : Wysiwis strict et Wysiwis relâché. Nous pensons que des modes intermédiaires peuvent être définis, mais notre expérience ne nous permet pas une spécification plus fine. Ces filtres vont être associés à des objets d'interaction relativement générique tel que les barres de défilement, les boîtes de dialogue etc.

Ceci implique que la création de nouveaux objets d'interaction entraîne la mise à jour de l'Agent de Rétroaction.

Lien de communication

Il communique naturellement avec l'Agent d'Interface afin de lui fournir les filtres qui lui sont nécessaires. Il reçoit des messages de la part de l'Agent Organisationnel pour les modifications du mode de rétroaction. Ces changements de mode peuvent être décidés par un utilisateur, ou bien par d'autres agents, par exemple pour une adaptation de la rétroaction en fonction des performances du réseau de communication (si la bande passante du réseau est trop étroite, adapter la rétroaction peut diminuer la quantité d'information qui transite sur ce même réseau). Ces changements de mode peuvent concerner tout le groupe, ou une instance de rôle particulière si on veut que chaque utilisateur puisse adapter son interface. Dans ce cas le filtre agit dans le sens Agent d'Interface Présentation.

Le processus de traitement

A la réception d'un changement de mode, l'Agent de Rétroaction fournit les nouveaux filtres aux Agents d'Interface, ou à l'Agent d'Interface concerné dans le cas où l'on donne le choix à l'utilisateur de modifier son niveau de rétroaction (ce qu'il perçoit des autres).

Agent de rôle

Cet Agent de Rôle est un mandataire auquel l'Agent Organisationnel délègue une partie de ses connaissances. Il existe un seul exemplaire par site client donc par utilisateur. Il est l'interface en terme de communication entre les agents associés à l'utilisateur (contenu dans un site client) et les autres sites clients, l'Agent d'Organisation et le composant Abstraction. Pour les composants du site client, il est perçu comme un agent de communication avec les composants extérieurs.

Ses connaissances

Il possède un mécanisme de stockage des informations associées à un rôle. Cette connaissance lui est fournie par l'Agent Organisationnel en terme de droits lors de sa création, et est dynamiquement modifiable grâce à la relation de délégation qui existe entre cet agent et l'Agent Organisationnel. Cette information consiste en un ensemble de scripts qui expriment les droits accordés à l'utilisateur en fonction de la phase et de l'état de l'activité, et qui contrôle la validité des interactions utilisateurs.

Lien de communication

Il communique avec l'Agent Organisationnel qui l'a créé, ainsi que les agents de rôle associés à la même activité représentant les autres utilisateurs. Il communique aussi avec l'Abstraction afin de diffuser les messages consécutifs aux interactions utilisateur, et aussi pour diffuser à l'Agent d'Interface les messages venant de l'abstraction.

Processus de traitement

Il traite les messages relatifs aux changements d'état ou de phase de l'activité qui lui sont fournis par l'Agent Organisationnel. Ceci a pour conséquence d'envoyer des messages à l'Agent d'Interface qui vont correspondre à la liste des droits qui sont donnés à l'utilisateur.

Agent d'interface

Cet agent prend en charge les interactions locales de l'utilisateur. Les interactions locales sont toutes celles qui n'entraînent pas de communication avec l'Abstraction, et qui n'ont aucune répercussion sur les changements de phase et d'état. Par exemple, le dimensionnement d'une fenêtre, l'ouverture d'un menu sont des interactions locales. L'agent d'Interaction traite ces interactions locales en fonction du style d'interaction choisi : les styles d'interaction définissent la façon dont on gère la rétroaction de groupe. Si on décide un fonctionnement en Wysiwis strict, l'Agent d'Interaction va filtrer toutes les interactions utilisateur et les diffuser au Agent d'Interaction lui correspondant sur les autres sites. Si au contraire on décide une relaxation spatiale, l'Agent d'Interaction ne diffusera pas les interactions utilisateur. Son deuxième rôle est d'adapter les messages venant de l'Abstraction pour qu'ils soient compréhensibles par la Présentation (similaire au rôle du contrôleur dans PAC), mais aussi une adaptation des messages liés à la gestion de l'activité : ces messages sont ceux qui permettront d'exprimer les droits d'un utilisateur en fonction de son rôle par une représentation visuelle gérée par la Présentation. Cette adaptation entre les représentations gérées par l'Agent de Rôle est donc faite par l'Agent d'Interface en tenant compte des modalités offertes par l'environnement local. Par exemple, si la station utilisateur possède une carte son, on pourra générer des sons pour indiquer un changement au niveau de l'interface, au lieu d'utiliser un artefact visuel pour attirer l'attention de l'utilisateur. Ceci n'est pas sans rappeler l'utilisation des échos audio dans GroupDesign [Karsenty 94]. Le choix de cette modalité peut être géré automatiquement par l'Agent en fonction de la configuration matérielle, définit statiquement à l'avance dans l'agent ou encore définit par l'utilisateur lui-même par modification des filtres fournis par l'Agent de Rétroaction (interface adaptative).

Ses connaissances

Il contient des tables de correspondances qui lui permettent de transcrire les droits donnés à l'utilisateur en terme compréhensible par le composant Présentation (à

rapprocher du contrôleur du modèle PAC). Ces tables sont modifiables dynamiquement par l'utilisateur en choisissant parmi des comportements par défaut ou imposées par le système (bibliothèques d'Agents d'Interface).

Il contient l'ensemble des filtres qui lui permettent de respecter le mode de rétroaction choisi. Ces filtres lui sont fournis par l'Agent de Rétroaction et sont dynamiquement modifiables grâce à la relation de délégation existant entre les deux agents.

Lien de communication

Il communique avec l'Agent de Rôle pour lui transmettre les messages qu'il ne sait pas traiter. Par exemple, tous les messages susceptibles de modifier l'état d'une activité sont transmis.

Il communique avec les Agents d'Interface qui lui correspondent dans les autres sites pour principalement prendre en charge la rétroaction de groupe, par l'intermédiaire des fonctionnalités de communication offertes par l'Agent de Rôle.

Il communique avec le composant Présentation afin de lui transmettre l'information pour représenter le rôle, et aussi pour adapter l'information provenant de l'Abstraction

Le processus de traitement

Il traite les messages associés à la description des droits des utilisateurs, et les interprètent en envoyant au composant Présentation des messages qui vont lui permettre d'adapter l'interface utilisateur en fonction du rôle.

Il reçoit des messages de la part de la Présentation et les interprète en fonction d'une typologie : les messages qui sont susceptibles de modifier l'activité en terme d'état ou de phase sont redirigés vers l'Agent de Rôle ; les messages pour l'Abstraction sont adaptés et diffusés à l'Agent de Rôle ; les messages associés à la rétroaction sont diffusés aux Agents d'Interface lui correspondant sur les autres sites par l'intermédiaire des capacités de communication incluses dans l'Agent de Rôle après avoir été filtrés pour respecter le mode de rétroaction courant.

3 - La relation de délégation

A travers la description des différents agents, cette relation apparaît comme une délégation de pouvoir, de prise de décision. Elle repose sur la diffusion de connaissances entre les agents, soit à la création de ceux-ci ou bien par une mise à jour dynamique en cours de session. La prise de décision est organisée sous forme hiérarchique. Par exemple, si les Agents de Rôle possèdent suffisamment de connaissance pour prendre les décisions par rapport à la réception d'un message, ils le

font. Sinon ils le transmettent à l'Agent Organisationnel. Dans le cas de l'Agent de Rétroaction, il ne s'agit que d'un filtrage de messages pour respecter un mode de rétroaction. Ce filtrage se fait en émission lorsque le mode correspond au désir du groupe (l'interface de chaque participant fonctionne dans le même mode de rétroaction). Ce filtrage se fait en réception lorsque les utilisateurs ont la possibilité de choisir le niveau de rétroaction associé à l'activité des autres.

Cette prise de décision peut être :

- un refus de traiter le message, qui correspond par exemple à une action non autorisée pour l'utilisateur concerné ;
- une décision de routage vers l'Abstraction si le message lui est adressé et qu'il peut être traité par l'Abstraction sans générer de conflit ou d'incohérence ;
- une décision de routage vers l'Agent Organisationnel lorsqu'il s'agit d'un message lié à la gestion de l'activité, comme le début d'un vote ou la demande d'une ressource partagée ;
- un routage ou non d'un message reçus en respect d'un mode de fonctionnement.

La relation de délégation peut être utilisée à tout moment pour modifier l'Agent de Rôle : par exemple pour le changement de rôle. Ces changements peuvent s'opérer dynamiquement sur demande d'un utilisateur ayant un rôle lui permettant de changer les rôles. Il en est de même pour les Agents d'Interface qui peuvent recevoir de nouveaux filtres de la part de l'Agent de Rétroaction.

4 - Les flux de messages et le filtrage

Nous définissons sept classes de messages :

- MAB : les Messages d'Abstraction sont destinés au composant Abstraction pour modifier les données du domaine ;
- MMP : les Messages de Mise à jour de la Présentation qui permettent le maintien de la cohérence entre l'Abstraction et la Présentation, mais aussi qui sont générés par l'Agent de Rôle pour décrire la représentation visuelle du rôle de l'utilisateur local ;
- MA : les Messages de gestion de l'Activité permettent de séquencer l'activité en terme de début, de fin, d'exprimer des requêtes en vue de changement de phase ou d'état, ou bien pour exprimer le partage d'une ressource (demande de contrôle, donner le contrôle, relâcher le contrôle) ;

- MP : les Messages de Présentation associés aux événements locaux générés par la Présentation (déplacement d'une fenêtre, ouverture d'un menu), mais aussi les messages qui permettent de mettre à jour la Présentation en fonction de la rétroaction de groupe (griser une icone, ouverture d'une menu) ;
- MR : les Messages de Rétroaction qui permettent la mise en place de la rétroaction de groupe entre les différents utilisateurs ;
- MGD : les Messages de Gestion de Délégation qui permettent la mise à jour des Agents de Rôle et des Agents d'Interface;
- MCR : les Messages de notification de Changement de Rétroaction.

Après avoir défini ces différentes classes de messages, nous voulons décrire le cheminement des messages entre les différents agents. Ceci nous amène à redécrire plus précisément les processus de traitement associés à chaque agent en utilisant les classes de messages et en s'attachant à mieux définir le rôle de chaque agent en terme de filtrage.

Il est difficile de décrire toutes les situations possibles car le nombre d'agent et le nombre de classe de messages permettent de définir un grand nombre de possibilités. Nous nous restreignons à la description de deux situations pertinentes pour les collecticiels : le partage de ressources et la prise en compte de la rétroaction de groupe.

Le partage d'une ressource

Cet exemple peut correspondre à l'algorithme de contrôle correspondant au partage d'un télépointeur, ou bien au partage d'une ressource physique qui possède des contraintes d'exploitation comme un magnétoscope.

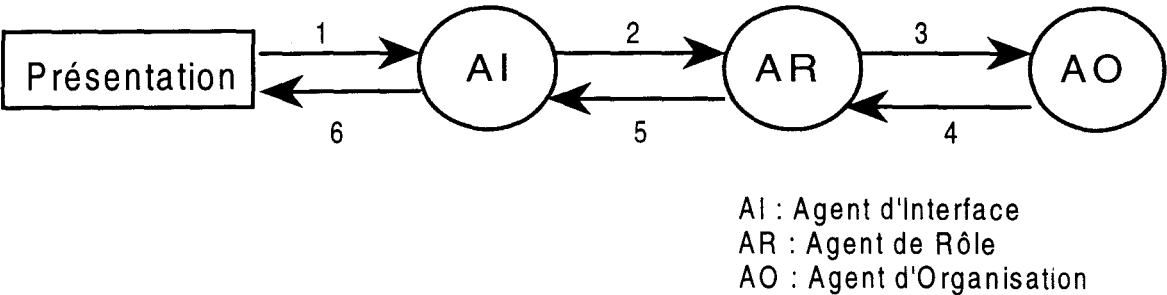


Figure 6 : Le cheminement des messages

Le cycle de 1 à 6 correspond à une demande de ressource qui aboutit. Nous décrivons maintenant chaque phase du cycle, en typant les messages échangés et en expliquant le rôle de chaque agent.

(1) L'utilisateur demande la ressource en "clickant" sur une icône ou en choisissant un item dans un menu. Un message de type MP est reçu par l'AI. Nous ne prenons pas ici en compte la rétroaction de groupe.

(2) L'AI traite ce message en générant un message générique de demande de ressource de type MA. Il adapte le message pour qu'il soit interprétable par l'AO.

(3) L'AR traite le message de type MA en vérifiant si le message est valide. C'est à dire qu'il vérifie si l'utilisateur a le droit de générer ce type de message, et s'il est cohérent avec l'état et la phase courante de l'activité. Un changement de phase ou d'état peut intervenir à tout moment et modifier les droits associés au rôle. Si le message est refusé on passe dans la phase 5 sinon le message est redirigé vers l'AO pour traitement.

(4) L'AO traite le message en appliquant la politique courante. Dans notre cas on accorde la ressource, ce qui a pour conséquence de modifier la phase et l'état de l'activité pour le groupe. Ceci entraîne l'envoi d'un message de type MA à chaque AR.

(5) L'AR interprète ce changement de phase et d'état en générant des messages vers l'AI afin de préciser les droits associés au rôle dans la phase et l'état courant, et aussi un message de réponse répondant à la demande de ressource (accepté ou refusé). Dans le cas où l'on vient de la phase 3 (demande non accordée), les messages correspondant au droit de l'utilisateur ne sont pas envoyés.

(6) L'AI traduit les messages reçus en messages compréhensibles par la Présentation pour une mise à jour de l'interface utilisateur.

Cet exemple illustre les relations existant entre l'Agent d'Organisation et l'Agent de Rôle. Il montre comment la communication entre les deux agents est faible : 1 message pour changer d'état et de phase.

La prise en compte de la rétroaction de groupe

Nous prenons le système dans l'état suivant : l'AI possède un filtre correspondant au mode de rétroaction qui lui a été fourni par l'AR à sa création.

La Présentation envoie un flot de messages qui correspondent aux interactions de l'utilisateur comme les déplacements souris, les déplacements d'une barre de défilement, un clic sur une icône etc.

L'AI utilise son filtre afin de trier les messages qui se révèlent nécessaires à la gestion de la rétroaction de groupe. Il les diffuse aux autres AI par l'intermédiaire des

fonctionnalités de communication fournies par l'AR. Les AI qui reçoivent ces messages MR les diffusent à la Présentation.

Les changements de mode de rétroaction se font à la suite d'une décision d'un utilisateur possédant les droits suffisant. C'est l'AR qui configure la Présentation (par l'intermédiaire de l'AI) pour que l'interface utilisateur contienne une fonction de changement de mode. Lorsque cette fonction est activée (par exemple par le choix dans une liste) l'AI traduit le MP reçu en message de type MCR. Celui-ci est transmis à l'AO pour vérifier si le changement de mode est possible en fonction de l'état de l'activité. Ensuite le message MCR est transmis à l'AR pour mise à jour des différents AI en utilisant la relation de délégation.

5 - Apport du modèle

Avantages du modèle

Il offre un modèle de communication riche entre les divers agents qui permet de réaliser des architectures centralisées, répliquées voir hybrides. Ce mode de communication pourrait nous permettre d'implémenter l'algorithme ORESTE [Karsenty 94] en spécialisant l'Agent de Rôle ou d'Interface en lui permettant de faire l'analyse des messages et de pouvoir réaliser les commutations en lui donnant le descriptif des actions commutables. L'interface utilisateur est paramétrable en jouant sur le comportement de l'Agent d'Interface.

L'Agent de Rétroaction fournit une possibilité d'adaptation de la rétroaction de groupe aux besoins du groupe. Nous avons identifié ce besoin dans les phases de formation aux outils Co-Learn qui nécessitent une rétroaction de grain fin alors que l'utilisation qui suit peut se faire avec une rétroaction de plus gros grain. L'avantage de ce modèle d'architecture est de permettre une définition dynamique de la rétroaction. Ceci nous permettra d'évaluer et de comparer le comportement du même outil avec des rétroactions différentes. De plus cette séparation de la rétroaction de groupe de l'application elle-même va nous permettre d'aborder le problème des interfaces adaptables par l'utilisateur et de mesurer leur utilité dans le contexte de travail de groupe.

Le modèle permet la conception de systèmes malléables capables de prendre en compte les problèmes de politique de gestion (algorithme de contrôle). Il offre la possibilité de les changer dynamiquement et d'en redéfinir de nouvelles sans remettre en cause totalement le système. Ce qui nous semble intéressant c'est d'essayer de coupler notre modèle avec des modèles multi-agent existant dans le monde mono-utilisateur, afin de voir si les agents que nous proposons sont génériques.

Inconvénients

Nous ne possédons pas pour l'instant de formalisme qui nous permette de décrire simplement l'activité en terme de rôle, de phase et de droit. Peut être certaines pistes sont à explorer dans les travaux de Palanque qui proposent une formalisation de l'algorithme de contrôle à partir des réseaux de Pétri [Palanque 91].

6 Conclusion

Notre modèle d'architecture, nous a servi dans la réalisation des outils de l'environnement Co-Learn décrits au chapitre IV. Ces implémentations ne font pas apparaître l'Agent d'Interface et l'Agent de Rétroaction car elles imposent un mode de rétroaction statique et une forte dépendance de l'Agent de Rôle avec la Présentation utilisée.

L'utilisation de ce modèle générique, nous a permis de définir trois types d'architecture, qui pour nous redéfinissent une nouvelle discussion sur les problèmes des architectures pour les collecticiels [Lantz 86][Lauwers 90]. Ces trois types d'architectures permettent de définir le contenu des composants Présentation et Abstraction en fonction de critères dépendant des caractéristiques du collecticiel : persistance de l'état d'une activité, accueil des nouveaux arrivants, caractéristique de la station clientes (voir chapitre IV, §3.2). En partant du même modèle générique, et en fonction de critères il nous est possible de réaliser des outils sans modifier la réalisation du système de gestion de rôle.

L'Agent d'Organisation dans notre implémentation du modèle est unique et centralisé. Cette approche nous permet de faire plus facilement évoluer l'Agent d'Organisation sans remettre en cause les composants logiciels présents sur les sites clients. Une réalisation distribuée de cet agent est possible en utilisant les apports des univers multi-agent distribués en l'intelligence artificielle qui offrent des mécanismes de prise de décisions répartis. Avant de travailler sur cette piste, nous voulons estimer les apports de ce type de réalisation, en fonction des contraintes d'exploitation, de gestion et des performances du système.

La description de nos agents fait ressortir des besoins en terme de filtrage d'événements, et de diffusion de ces mêmes événements.

Nous n'avons fait aucune hypothèse sur la distribution du composant Abstraction. Celui-ci pourra être répliqué sur chaque site utilisateur, ou être centralisé sur un site serveur. Cette distribution aura une répercussion sur le mécanisme de diffusion des messages au niveau de l'Agent de Rôle, sur le maintien de la cohérence entre les répliques de l'Abstraction, mais ne modifiera en rien le principe exposé dans le modèle.

Des solutions existent déjà parmi les systèmes à objets distribués et des systèmes comme Isis qui offrent différents niveaux de diffusion d'information [Birman 90]

III - Quelques recommandations

1 - Les architectures de communication

Les modèles d'architectures présentés dans le début de ce chapitre nécessitent pour leurs mises en oeuvre, une architecture de communication avec des fonctionnalités identifiées comme la diffusion de messages. Dans les boîtes à outils comme GroupKit et RendezVous, cette architecture repose soit sur des mécanismes de diffusion de messages ou bien sur des mécanismes de communication entre processus distribués. Il est important que les futures architectures de communication prennent en charge le transport de données multimédia en mode diffusion. Il faut aussi que ces architectures intègrent la gestion de périphériques physiques tels que les ponts d'audioconférences et de visioconférences ainsi que des mécanismes de synchronisation entre ces différents médias.

Il nous semble aussi important que ces architectures de communication offrent des niveaux de qualité de services. Par exemple, la qualité de service de la couche de transport permettra d'adapter certaines fonctionnalités de hauts niveaux des collecticiels comme la rétroaction de groupe. Les niveaux de service englobent aussi des mécanismes de détection de dysfonctionnements. Dans les systèmes mono-utilisateur, une panne réseau ne concerne qu'un utilisateur, alors que dans les collecticiels la "disparition" d'un participant peut avoir des répercussions sur l'activité du groupe. Il est donc important que l'architecture de communication offre des mécanismes de notifications qui permettront aux composants logiciels qui les utilisent de mieux informer l'utilisateur sur l'état du système.

Les niveaux de service fournis par l'architecture de communication doivent être indépendants de l'architecture physique utilisée. Si on veut obtenir des collecticiels utilisables sur une large échelle, il est important que les collecticiels puissent être hétérogènes.

L'architecture de communication doit aussi prendre en compte les problèmes d'optimisation des coûts de communication. La prise en compte des coûts est intéressante car elle a des conséquences sur l'architecture de communication. Par exemple dans Venue (collecticiel développé par Digital), on trouve la notion de "relay" qui permet de prendre en charge la communication entre les stations d'un site local et la communication vers les autres sites distants. Ce concept introduit une nouvelle façon de gérer la diffusion entre les différentes stations impliquées dans l'utilisation d'un collecticiel.

Transport multimédia**Synchronisation****Mécanismes de notification sur la qualité de services et les dysfonctionnements****Hétérogénéité****Recherche d'efficacité dans les mécanismes de diffusion**

2 - Les systèmes de fenêtrage

Les besoins au niveau des systèmes de fenêtrage sont directement liés à la dimension multi-utilisateur et à l'utilisation de nouvelles métaphores.

La dimension multi-utilisateur fait ressortir des besoins en terme de multi-curseur pour une gestion plus facile des interactions multi-utilisateur simultanées et un renforcement de la téléprésence. Ces possibilités de multi-curseur peuvent être aussi utilisées pour réaliser des outils de télépointage qui permettent de prendre en charge la gestuelle. De nouveaux objets interactifs ou widgets apparaissent comme les barres de défilement multi-utilisateur. Nous pensons que ces objets interactifs doivent intégrer des nouveaux comportements qui permettent de plus facilement définir la rétroaction de groupe : par exemple créer des objets d'interaction "fantôme" qui sont capables de simuler à distance la manipulation de leur original. La boîte à outils de GroupKit fournit des menus qui respectent le Wysiwis strict, des boîtes de dialogue qui apparaissent sur plusieurs écrans simultanément et qui évoluent en même temps.

Les nouvelles dimensions métaphoriques comme la métaphore de pièce expriment des besoins en terme d'interface 3D (trois dimensions)[Card 87][Benford 93][Bly 93]. En effet, l'utilisation de ces univers 3D permet d'immerger l'utilisateur dans un monde virtuel. Il est donc nécessaire de définir de nouvelles manipulations sur cet environnement, de réfléchir sur la signification du curseur dans ce nouvel univers, sur la manipulation des documents et leur représentation. La question que nous nous posons est de réfléchir en quoi la troisième dimension intégrée dans les systèmes d'interaction peut nous aider à mieux gérer les interactions de groupe et le travail coopératif en général. Ou en quoi les exigences du travail coopératif nécessitent l'utilisation d'interface 3D ?

Mieux prendre en compte la dimension multi-utilisateur dans le système de fenêtrage

En quoi les environnements d'interaction en trois dimensions peuvent aider à la réalisation de collecticiels ?

3 - Comment prendre en compte la dimension sociale ?

Nous avons besoin d'intégrer des mécanismes génériques de gestion des objets sociaux : il n'y a plus un utilisateur mais un groupe d'utilisateur qui existe dans une organisation avec des responsabilités et des droits.

Conclusion

Les SGIU gèrent l'interface utilisateur dans un dialogue Homme machine. Tout au long de cette thèse, nous avons identifié plusieurs enjeux associés à la conception des collecticiels [Roseman 92][Lauwers 90][Lu 93][Croisy 94b], qui vont permettre de définir les futurs SGIU, peut être les SGIG (Système de Gestion d'Interface de Groupe), ou les SGCI (Système de gestion de Communication Interpersonnelle).

Les SGIG vont devoir s'intéresser à :

- l'interface homme-homme qui intègre la communication humaine et la collaboration au sein d'un groupe [McGrath 90][Schmidt 91, 93] ;

- l'interface homme-machine qui définit le dialogue entre un humain et un dispositif physique. Elle doit plus précisément définir de nouveaux mécanismes de représentation visuelle et de perception auditive, plus généralement multimédia pour rendre compte à un utilisateur de l'activité des autres utilisateurs et de la présence des autres (rétroaction de groupe et téléprésence)[Gaver 92][Buxton 92][Clark 91].

CONCLUSION

Ces travaux sur le collecticiel temps réel appliqués au champs éducatif ont permis de participer à la construction d'un environnement maintenant utilisable dans des situations réelles. Ce CSCL (Computer Supported Collaborative Learning) est actuellement à la base d'un projet financé par le CNET sur les interfaces 3D nommé Space [Chaillou 94]. Ce projet veut appréhender l'apport de ce type d'interface pour la conception des environnements coopératifs, en réfléchissant sur les nouvelles métaphores qui permettront de renforcer la téléprésence [Buxton 92]. Pour ceci, il utilise l'environnement Co-Learn, l'architecture de communication, le mécanisme de gestion de rôle et le modèle multi-agent proposé au chapitre VI [Croisy 94a].

D'autres projets sont aussi en cours de négociation qui mettent plus en jeu le problème de l'évaluation : proposition de projet pour le quatrième framework du programme DELTA financé par la Communauté Européenne. Ce type de projet devrait valider les caractéristiques intrinsèques de l'environnement Co-Learn, à savoir la malléabilité fonctionnelle et structurelle. Il est aussi évident que l'un des objectifs de ce projet serait de découvrir de nouveaux usages de l'environnement.

Les travaux de cette thèse ont permis de définir un nouveau modèle d'architecture qui met en évidence le besoin d'extraire de l'application la gestion coopérative de l'application [Croisy 92], et la rétroaction de groupe. Cette dimension permet de définir la malléabilité fonctionnelle qui permet de prendre en compte les différences interpersonnelles. Ce modèle multi-agent pose certains problèmes liés aux architectures d'accueil des collecticiels temps réel : les besoins en communication multimédia et l'intégration dans ces architectures de concepts appartenant au CSCW comme la notion de groupe. Cette thèse va permettre de mieux définir les objectifs d'un projet nommé OSACA, qui vise à définir une plate-forme d'accueil générique pour les collecticiels. Ce projet fait parti des activités de Ganymède qui est un pôle régional de recherche concertée sur la communication avancée, retenue au contrat de plan Etat-Région 1994-1998.

Un autre objectif sur lequel l'auteur de cette thèse souhaite continuer son travail, est la conception d'une boîte à outil qui permette de développer rapidement de nouveaux applicatifs. Ceci passe par une extension de la plate-forme Co-Learn, pour qu'elle prenne en compte l'ensemble des agents définis dans le modèle multi-agent décrit au chapitre VI. Ces applicatifs vont essayer de mettre en oeuvre de nouveaux paradigmes ou métaphores, pour mieux prendre en compte la métacommunication (par exemple la gestuelle) [Nakajima 93], et augmenter la téléprésence [Buxton 93]. D'autres pistes vont être mise en oeuvre pour permettre l'utilisation dans l'environnement Co-Learn de documents vidéos, et la création de nouveaux outils pour répondre à des besoins spécifiques de l'apprentissage coopératif. Par exemple, un éditeur

graphique multi-utilisateur orienté objet, est en cours de réalisation dont l'objectif est d'essayer de permettre l'articulation de points de vue et la prise en compte du conflit socio-cognitif¹.

¹L'articulation de point de vue et le conflit socio-cognitif sont des caractéristiques nécessaires à l'apprentissage coopératif. L'objectif de cet éditeur est de montrer comment un outil informatique peut mettre en oeuvre ces caractéristiques définies par les Sciences Cognitives. Le chapitre II donne une définition de ces deux notions.

BIBLIOGRAPHIE

[Abowd 90]

G.D.Abowd, "Agents : Communicating Interactive Processes", Proceedings of Human Computer Interaction INTERACT'90, D.Diaper et al. (editors), Elsevier Sciences Publishers B.V (North-Holland), IFIP, 1990, pp.143-148

[Anzieu 68]

D.Anzieux, J.Y.Martin, "La dynamique des groupes restreints", Presses Universitaires de France, 1968

[Bannon 91]

L.J. Bannon, K. Schmidt, "CSCW : Four Characters in Search of a Context", Studies in Computer Supported Cooperative Work, J.M. Bowers and S.D. Benford (Editors), Elsevier Science Publishers B.V. (North-Holland), 1991, pp.3-16

[Barme 95]

L.Barme, A.C.Derycke, "Computer Supported Audio Conferencing System : a Groupware Agent", soumis à INTERACT'95

[Barme 94]

L.Barme, "Audioconférence assistée par ordinateur : gestion et contrôle", Document de travail des journées IHM94, Lille le 8 et 9 décembre 1994, à paraître

[Baldeschieler 93]

J.E. Baldeschieler, T. Gutekunst, B. Plattner, "A Survey of X Protocol Multiplexors", ACM SIGCOMM, 1993, pp 16

[Benford 93]

S.Benford, A.Bullock, N.Cook, P.Harvey, R. Ingram and O.K.Lee, "From rooms to cyberspace : models of interaction in large virtual computer spaces", Interacting with Computers, vol.5, n°2, june 1993, pp.217-238

[Ben-Shaul 93]

Ben-Shaul, I.Z. Kaiser, G.E. Heineman, G.T, "An Architecture for Multi-User Software for Multi-User Software development Environments", Computing Systems, Vol.6, N°2, Spring 1993, pp.65-103

[Birman 90]

Birman, K., Cooper, R., Joseph, T., Marzullo, K., Makpangou, M., Kane, K., Schmuck, F., and Wood, M., "The Isis System Manual. Technical report", The Isis Project, Cornell University, Septembre 90

[Bly 93]

S.A.Bly, S.R. Harrison, S.Irwin, "MediaSpace : bringing people together in a video, audio and computing environment", Communication of the ACM, january 1993, vol.36, n°1, pp.28-47

[Bhandaru 90]

N.Bhandaru, W.B.Craft, "Architecture for supporting goal-based cooperative work", in Multi-user Interfaces and Applications, S.J.Gibbs and A.A.Verrijn-Stuart Editors, Elsevier Sciences Publishers B.V (North Holland), IFIP, 1990, pp.337-354

[Bourdon 93]

Bourdon T., "Structures de communication et d'organisation pour la coopération dans un univers multi-agents", thèse Paris VI, février 1993

[Brinck 92]

Brinck, T. and Gomez,L.M., "A Collaborative Medium for the support of Conversation Props", in Proceedings of the conference on Computer Supported Collaborative Work (CSCW'92), pp.171-178

[Buxton 92]

W. A. Buxton, "Telepresence : Integration Shared Task and Person Spaces", Proceedings of Graphics Interface'92, 11-15 may, Vancouver, 1992, pp.123-129

[Card 87]

S.K.Card, A.Henderson, "A Multiple, Virtual-Workspace Interface to Support User Task Switching", ACM, 1987

[Cardozo 93]

E.Cardozo, J.S.Sichman, Y.Demazeau, "Using the Active Object Model to Implement Multi-Agent Systems", Proceedings of the IEEE Int.l Conf. on Tools with AI, Boston, Massachusetts, nov. 1993, pp.70-77

[Carlier 94]

Carlier P., "Evaluation d'Architectures de Collecticiels dans l'Environnement Unix et X-Window", mémoire de DEA, laboratoire Trigone et laboratoire d'informatique fondamentale de Lille, septembre 1994

[Carre 89]

B.Carré, "Méthodologie orientée objet pour la représentation des connaissance, concepts de point de vue, de représentation multiple et évolutive d'objet", thèse en informatique, LIFL Université des Sciences et Technologies de Lille, 1989

[Chaillou 94]

Chaillou C., Wibaux L., Barme L., "Une expérience d'interface 3D exploitant la notion de pièce", Actes des Sixièmes Journées sur l'Ingénierie des Interfaces Hommes-Machines IHM'94, Lille les 8-9 décembre, 1994

[Clark 91]

Clark, H. Brennan, S., "Grounding in Communication", in Perspective on Socially Shared Cognition, American Psychological Association, Washington DC, 1991, pp 127-149

[Clément 94]

Clément D., "Fondements théoriques de l'analyse de l'activité coopérative", Document de travail du Workshop GANYMEDE sur le travail coopératif et la communication avancée, Lille le 3 juin 1994, 6 pages

[CoLearn 94a]

Author - Sall M., Viéville C., "Standart and norms", deliverable ?, à paraître

[CoLearn 94b]

Author - Sall M., "User Manual for the Real Time Tele-Teaching Tool", DELTA Project D2005, Deliverable 21B, mars 94

[CoLearn 93a]

Author - Croisy Pascal & Viéville Claude, "User manual of real-time multimedia conference", DELTA Project D2005, Deliverable 24, octobre 93

[CoLearn 93b]

Author - Software de Base S.A., "First CPI version", DELTA Project D2005, Deliverable 19, septembre 1993

[CoLearn 93c]

Author - TH Darmstadt, "Programmers Reference Manual for MM Object Update & Retrieval", deliverable 20B, septembre 1993

[CoLearn 93d]

Author - Politecnical University of Madrid (DIT-UPM), "Reference Manual of Communication Services", DELTA Project D2005, Deliverable 18B, octobre 1993

[CoLearn 93e]

Author - M.Sall, "User Manual for the Real Time Tele-Assistance Tool", DELTA Project D2005, Deliverable 22B, septembre 1993

[Colearn 93f]

Author - C.Viéville, "User Manual for the Asynchronous Multimedia Conferencing System", DELTA Project D2005, Deliverable 23B, septembre 1993

[CoLearn 92a]

Author - Trigone, "Software of the common object oriented toolbox", DELTA Project D2005, Deliverable 15, septembre 1992

[CoLearn 92b]

Author - Trigone, Polytechnical University of Madrid, Télésystèmes, "Specification of the Common Object Oriented Toolbox", DELTA Project D2005, Deliverable 3, juin 1992

[CoLearn 92c]

Author - Trigone, Open University, Polytechnical University of Madrid, "Representation Models for Collaborative Educationnal Situations and Collaborative Learning Activities", DELTA Project D2005, Deliverable 4, juin 92

[Coutaz 90]

J. Coutaz, "Interface homme ordinateur conception et réalisation, Edition Dunod Informatique, 1990

[Croisy 94a]

Croisy P., "Modèle multi-agent et conception d'applications coopératives interactives ", Actes des Sixièmes Journées sur l'Ingénierie des Interfaces Homme-Machine IHM'94, Lille les 8-9 décembre, 1994

[Croisy 94b]

Croisy P., "Architecture logicielle des collecticiels", réunion du groupe de travail n°8 du GDR PRC Communication Homme Machine, GT-SCOOP, Lyon, 1994

[Croisy 94c]

Croisy.P, Clément.D, Barme.L, "Co-Learning at a distance : results from first trials", EDMEDIA'94 Worlf Conference on Multimedia and Hypermedia, 29-30 juin, Vancouver, Canada, AACE press, 1994, p.641

[Croisy 92]

P.Croisy, A.Derycke, "Un modèle conceptuel de système interactif pour une application coopérative", Actes des Quatrièmes Journées sur l'Ingénierie des Interfaces Homme-Machine IHM'92, 30 novembre et 1-2 décembre 1992, pp.151-156

[Croisy 91]

Croisy P., "Architecture d'un système pour le travail coopératif : l'aspect interface multi-utilisateur dans un environnement objets", Mémoire de DEA, Université des Sciences et Technologies de Lille, juillet 91, 56 pages

[Crowley 90]

Crowley,T., Baker,E., Forsdick,H., Milazzo,P. and Tomlinson,R., "MMConf: an infrastructure for building shared applications", in Proceedings of the conference on Computer Supported Cooperative Work CSCW'90, 1990, pp

[Davies 93]

R.N.Davies, J.C. Jennings, "The multimedia teleschool : telematics in a corporate learning environment", in Proceedings of TELETEACHING'93, IFIP, Trondheim, Norway, August 20-25, G.Davies and B.Samways (eds), North-Holland, 1993, pp183-190

[Decouchant 94]

D.Decouchant, "Rétroaction de groupe et édition coopérative de documents structurés", Document de travail des sixièmes journées sur l'ingénierie des Interfaces Hommes-Machines IHM'94, Lille, 7 et 8 décembre, à paraître

[Decouchant 93]

D.Decouchant, V.Quint, M.Riveill et I.Vatton, "Alliance : A Cooperative, Structured, Distributed Document Editor", Rapport de Recherche n°20, Bull-IMAG, Grenoble, juin 1993

[Decouchant 92]

D.Decouchant, V.Quint, I.Vatton, "L'édition coopérative de documents avec Griffon", Document de travail des quatrièmes journées sur l'ingénierie des Interfaces Hommes-Machines IHM'92, Décembre 92, Telecom Paris 92 S 004, pp 137-141

[Derycke 95]

A.C.Derycke, C.Smith, L.Hemery, "Metaphors and intercatons in virtual environments for open and distance education", article soumis à ED_MEDIA'95

[Derycke 94a]

A.Derycke, L.Barne, "Multimedia in the computer supported conference", BIWIT'94 conference, IEEE, AFCET, february 7-9, Cepadues edition, Toulouse, pp.287-297

[Derycke 94b]

A.Derycke, "Le CSCW au delà de l'IHM : Taxonomie et Dimension Sociale", présentation réalisée aux journées du groupe de travail GT SCOOP (GT8) du PRC Communication Homme Machine les 13 et 14 juin 1994 à Lyon

[Derycke 93a]

A.C.Derycke, P.Croisy, P.Vilers, "Computer Supported Cooperative Learning : a real-time multimedia approach", Proceeding of ED-MEDIA'93, Orlando, FL, june 23-26, AACE Editors, 1993

[Derycke 93b]

A.C. Derycke, C. Chaillou, L. Dumesges, "La troisième dimension dans les interfaces de collecticiels", Document de travail des cinquièmes journées sur l'ingénierie des Interfaces Hommes-Machines IHM'93, Lyon le 19-20 octobre 1993, pp 81-86

[Derycke 93c]

A.C.Derycke, A.R.Haye, "Participative modelling and design of collaborative distance learning tools in Co-Learn project", Teleteaching'93, G.Davies and B.Samways (Editors), Elsevier Science Publishers B.V. (North-Holland), IFIP, 1993, pp191-200

[Derycke 93d]

A.C.Derycke & C.Viéville, "Real time Multimedia Conferencing System and Collaborative Learning", NATO Advanced Workshop, Segovia, Spain, April 24-27, 1993. A paraître, F.Verdejo, S.Cerri (eds), ASI Series, Berlin, Springer Verlag, 23 pages

[Derycke 90]

Derycke A., Viéville C., Vilers P., "Cooperation and communication in Open Learning : the CoCoNut project", Proceedings of the WCC'90 IFIP Conference, Sidney, July, in Computer in Education, A.Lc Dougall and C.Dowling (editors), North-Holland, pp. 956-962

[Dewan 90]

Dewan, P. "A Tour of the Suite User Interface Software", in the Proceedings of the ACM Third Annual Symposium on User Interface Software and Technology (UIST'90), 1990, pp.57-65

[Dewan 91]

Dewan,P. and Chouldhary, R. "Primitives for Programming Multi-User Interfaces", in User Interface Software and Technology (UIST'91), 1991,pp.69-78

[Dewan 92]

Prasun Dewan et al, "Critical Issues : User Interfaces to Distributed Systems", Engineering for Human Computer Interaction, J.Larson and C.Unger (editors), Elsevier Science Publishers B.V. (North Holland), IFIP, Panel, 1992, pp.415-417

[Dugerdil 90]

Dugerdil Ph., "Programmation par objet", Collection Informatique, Edition Dunod, 1990

[Easterbrook 93]

S.M.Easterbrook, E.E.Beck, J.S.Goodlet, L.Plowman, M.Sharple and C.C.Wood, "A survey of Empirical Studies of Conflict", In Cscw : cooperation or conflict ?, ed. S.Easterbrook, London, Springer-Verlag, pp.1-68

[Edigo 88]

Edigo, C., "Videoconferencing as a technology to support group work : a review of its failures", Proceedings of the Conference on Computer Supported Cooperative Work CSCW'88, 1988, pp.13-24

[Ellis 89]

Ellis C.A., Gibbs S.J., "Concurrency Control in Groupware Systems", in Proceedings of the ACM SIGMOD'89 Conference on the Management of Data, Seattle Wash., May 2-4, 1989

[Ellis 90]

Ellis, C., Gibbs, S. J., & Rein, G., "Design and use of a group editor", Proceedings of the JFIP TC2/W2.7 Working conference in Engineering for Human-Computer interaction, Napa Valey, 21-25 Aout 1989, Editing by G.Cockton North Holland, 1990, pp. 13-25

[Ellis 91]

Ellis, C., Gibbs, S. J., & Rein, G., "Some Issues and Experiences", Communication of ACM, vol.34, n°1, january 1991, pp39-?

[Falhen 93]

Falhen et al., "A space based model for user interaction in shared synthetic environments. Proceeding of INTERCHI'93, IFIP, vol.2, n°2, 1993, pp.115-129

[Fitzgerald 91]

P.F. Fitzgerald, N.Y. Rosson, L. Uljon, "Evaluating alternative display sharing system architectures", IEEE, 1991, pp.145-157

[Ganymede 93]

Pôle Régional de Recherche concertée sur la Communication Avancée,
Programme Scientifique proposé pour le Contrat de Plan Etat-Région (1994-1998), "Activité Coopératives et Communication Avancée", mai 1993

[Gaver 92]

W. Gaver, " The Affordances of Media Spaces for Collaboration", Proceedings of CSCW'92, November, 1992, pp.17-24

[Gibbs 89]

S.J.Gibbs, "LIZA : An Extensible Groupware Toolkit", CHI'89 Proceedings, ACM, May 1989, pp 29-35

[Goldberg 84]

Goldberg A., Robson D., "Smalltalk 80 : the language and its implementation", Addison Wesley, 1983

[Greenberg 92]

Greenberg,S., Roseman,M., Webster,D. and Bohnet, R., "Human and Technical Factors of Distributed Group Drawing Tools", Interacting with Computers, 4(3), 1992, pp 364-392

[Grudin 88]

Grudin J., "Why CSCW Applications fail: Problem in the Design and Evaluation of Organizational Interfaces", in Proceedings of the ACM Conference on Computer Supported Collaborative Work (CSCW'88), Portland, September 26-29, 1988, pp.85-93

[Gust 88]

Gust P., "SharedX : X in a distributed group environment", in the Second Annual X Technical Conference, MIT, 1988

[Haake 92]

Haake, J.M Wilson, B., "Supported Collaborative Writing of Hyperdocuments in Sepia", In Proceedings of CSCW'92 Sharing Perspectives, ACM, Toronto, Canada, October 31 to November 4, 1992, pp.138-146

[Hill 92]

Ralph D.Hill, "The Abstraction-Link-View paradigm: using constraints to connect user interfaces to applications", ACM , Mai 1992

[Hiltz 90]

Référence à compléter

[Hoogstoel 94]

Hoogstoel F., Thèse de l'Université des Sciences et Technologies de Lille, à paraître

[Johansen 88]

"Groupware : computer Support for Business Teams", Series in Communication Technology and Society, Everett M.Rogers and Frederick Williams (editors), 1988, Chapitre 2, pp 12-44

[Karsenty 94a]

Karsenty A., "GroupDesign : un collecticiel synchrone pour l'édition partagée de documents", Thèse présentée à l'Université de Paris-Sud

[Karsenty 94b]

Karsenty A., "Le collecticiel : de l'interaction homme-machine à la communication homme-machine-homme", Technique et science informatiques, vol 13, n°1, 1994, pp.105-127

[Karsenty 94c]

A.Karsenty, "Interfaces sonores pour collecticiels", Actes des Sixième Journées sur l'Ingénierie des Interfaces Homme-Machine IHM'94, Lille les 8-9 décembre, 1994

[Karsenty 93]

Karsenty A., Beaudoin-Lafon M., "An Algorithm for Distributed Groupware Applications", in Proceedings of the 13th IEEE International Conference on Distributed Computing Systems ICDCS'93, Pittsburgh, Pennsylvania, May 25-28, 1993

[Kaye 92]

Kaye, A.R., "Collaborative Learning through Computer Conferencing", NATO ASI series, vol.90, Springer Verlag, 1992

[Kreifelts 93]

Kreifelt, T. Hinrichs, E. Woetzel, G., "Sharing Todo Lists with a distributed task manager", In DeMichelis et al (eds) Proceedings of ECSCW'93, Milan, Italy, September 13-17, Dordrecht, Netherland, Kluwer Academic Publishers, pp. 31-45

[Lantz 86]

Keith A.Lantz, "An Experiment in Integrated Multimedia Conferencing", CSCW'86, Décembre 1986, pp.267-275

[Lauwers 90]

Lauwers J.C., Joseph T.A., Lantz K.A., and Romanow A.L., "Replicated Architectures for Shared Window Systems", In Proceedings of the Conference on Office Information Systems", Boston, 1990, April 25-27, pp.249-260

[Lu 93]

I.M.Lu, M.M.Mantai, "Managing Design Ideas with a Shared Drawing Tool", Interacting with Computer, vol.5, n°1, mars, march 93, pp.79-114

[Malone 87]

Malone, T.W., Grant, K.R., Turbak, F.A., Brobst, S.A., and Cohen, M.D., "Intelligent Information Sharing Systems", Communications of the ACM, vol.30, n°5, may 1987, pp.390-402

[McGrath 90]

McGrath, J.E., "Time matters in groups", in Galegher J. et al (eds.) Intellectual teamwork : Social and technological foundations of cooperative work, Lawrence Erlbaum Associates, publishers, pp.23-62

[Ménadier 91]

J.P.Meinadier, "L'interface utilisateur : pour une informatique plus conviviale", Dunod, 1991

[Miller 92]

D.S.Miller, J.G.Smith & M.J.Muller, "TelePICTIVE: Computer-Supported Collaborative GUI Design for Designers with Diverse Expertise", Proceedings of the ACM Symposium on User Interface Software and Technology", California, November 15-18, 1992, pp.151-160

[Mugny 84]

Mugny G., "Le développement social de l'intelligence", Ed. Peter Lang, 1984

[Myers 90]

Myers G.E., Myers M.T., "Les bases de la communication humaine, une approche théorique et pratique", McGraw-Hill, Editeurs

[Navarro 93]

L. Navarro, W.Prinz and T. Rodden, "CSCW requires open systems", Computer Communication, vol.16, n°5, may 1993, pp.288-297

[Nakajima 93]

A.Nakajima, "Telepointing issues in desktop conferencing systems", Computer Communications, vol.16, n°9, Butterworth-heinemann Ltd, september 1993, pp.603-610

[Palanque 91]

Bastide R., Palanque P., "Modélisation de l'interface d'un logiciel de groupe par objets coopératifs ", Document de travail IHM'91, pp.1-10

[Pankoke 89]

Pankoke-Babatz, U. "Applications of AMIGO Work and Future Developments", In Computer Based Group Communication. The Amigo Activity Model, Pankoke-Babatz, U.(ed) Ellis Horwood, 1989

[Pattel 93]

Dorad Patel and Scott D. Kalter, "Low overhead, loosely coupled communication channels in collaboration", Proceedings of the Third European Conference on Computer-Supported Cooperative Work - ECSCW'93, 13-17 September, Milan, 1993, pp.203-218

[Patterson 90]

John F. Patterson, Ralph D. Hill, Steven L. Rohall, "Rendezvous: An Architecture for Synchronous Multi-User Application", CSCW'90 Proceedings, Octobre 1990, pp 317-328

[Patterson 91]

John F. Patterson, "Comparing the Programming Demands of Single-User and Multi-User Applications", UIST'91, Novembre 1991, pp 87-94

[Pinsky 92]

Pinsky, L., "Concevoir pour l'action et la communication", Collection Sciences pour la communication, Peter Lang, Berne, 1992

[Pondy 93]

Pondy "Organizational conflict : concept and models.", tiré de [Easterbrook 93] page 31

[Pozzi 94]

S.Pozzi, A.Tancredi, F.Tisato, "An integrated environment for distance education supporting multiple interaction styles", in Proceedings of ED-MEDIA'94, Vancouver, BC, Canada, June 25-30, T.Ottman and I.Tomek (eds), AACE Press, 1994, pp.440-455

[Prinz 93]

Prinz, W., "TOSCA : providing organisational information to CSCW applications to CSCW applications", In DeMichelis et al (eds) Proceedings of ECSCW'93, Milan, Italy, September 13-17, Dordrecht, Netherland, Kluwer Academic Publishers, pp.139-154

[Ravindran 92]

Ravindran, K. Prasad, B., "Communications Structures and paradigms for distributed Conferencing Applications", In 12th International Conference on Distributed Computing System, IEEE, Yokhama, Japan, 1992, pp.598-605

[Rodden 91]

T.Rodden, G.Blair, "CSCW and Distributed Systems : the Problem of Control", Proceedings of the Second European Conference on Computer-Supported Cooperative Work ECSCW 91, 1991, pp.49-64

[Rouzé 93]

Rouzé F., "Description d'un télécurseur multi-utilisateur utilisant la vidéo", mémoire de DEA, Laboratoire Trigone et Laboratoire d'Informatique Fondamentale de Lille, septembre 1993, 38 pages

[Roseman 94a]

Roseman M., "Design of a Real-Time Groupware Toolkit", Thesis for degree of master of Science, University of Calgary, feb. 1993

[Roseman 94b]

Roseman M., Greenberg S., Yibarek S., "GroupKit User's Manual", August 1994

[Roseman 92]

Roseman, M., Greenberg, S., "A Groupware Toolkit for Building Real Time Conferencing Applications", CSCW 92 Proceedings, Novembre 1992, pp 43 50

[Schmidt 91]

Schmidt, Kjeld, "Riding a tiger, or Computer Supported Cooperative Work", ECSCW'91 Proceedings, Amsterdam, 1991, pp.1-16

[Schmidt 93]

Schmidt, Kjeld, "Mode and Mechanisms of Interaction in Cooperative Work", Rapport projet ESPRIT 2065 COMIC, 7 septembre 1993, Riso National Laboratory, Denmark, 85 p

[Seguin 92]

Seguin J., Guivarch J., Morin T., et Guillot Y., " Télé-Amphi : un service de communication de groupe", L'Echo de la Recherches n°149, CNET

[Shlaer 88]

S.Shlaer & S.J.Mellor, "Object-Oriented Systems Analysis : Modeling the World in Data", Yourdon Press Computing Series

[Smith 94]

C.Smith, "Usability audit", Deliverable D29a, Project D2005, Co-Learn, Delta reports, 75 p.

[Stefik 87a]

Stefik, M., Bobrow, D. G., Foster, G., Lanning, S. & Tatar, D., "WYSIWIS revised: Early experiences with multiuser interface". ACM Trans Office information Systems, 5(2), Avril 1987, pp. 147-167

[Stefik 87b]

Stefik, M., Bobrow, D. G., Kahn, K., Lanning, S. & Suchman, L., " Beyond the chalkboard : Computer support for collaboration and problem solving in meetings", Communication of ACM, 30(1), 1987, pp. 32-47

[Tatar 91]

B.G.Tatar, G.Foster, D.G.Bobrow, "Design for conversation : lessons from the Cognoter", In Computer Supported Collaborative Work and Groupware, edited by Saul Greenberg, Harcourt Brace Jovanovich, Publisher, 1991, pp.55-79

[Viéville 95]

C.Viéville, "Structuring conversation in asynchronous communication systems to support collaborative learning at a distance.", article soumis à ED-MEDIA'95

[Viéville 94a]

C.Viéville, F.Hoogstoel, P.Croisy, A.Derycke, " A distributed architecture for cooperative systems supporting public-access", rapport interne Trigone, 12 pages

[Viéville 94b]

C.Viéville, "Accessibily of the CoLearn server", tiré de [CoLearn 94a], octobre 94, à paraître

[Viéville 90]

C.Viéville, A.Derycke, P.Vilers, "Architecture of a Collaborative System using Smalltalk and Unix", EUUG Spring'90 Procceedings, Munich, 23-27 April, 1990, pp.89-98

[Vilers 92]

Vilers P., "Etude d'un système de téléconférence temps réel : application à un jeu de rôle en gestion de production", Thèse de Doctorat soutenue à l'Université de Lille I en février 1992 (Laboratoire Trigone)

[Vigotsky 85]

Vigotsky L., "Pensée et langage", Editions sociales 1985

[Winograd 86]

Winograd, T.A. "langage/action perspective on the design of cooperative work",
proceedings de la conférence CSCW'86, Austin, Texas, ACM, 1986, pp.203-220

ANNEXE I

PRESENTATION DU PROJET

CO-LEARN

Les Membres du Consortium Co-Learn



Télésystèmes, Paris : Le contractant principal au sein du consortium et aussi responsable des outils de télé-enseignement et de la télé-assistance temps-reel.



Laboratoire TRIGONE, Institut CUEEP, Université de Lille I : Responsable du développement des outils de conférence temps réel multimedia et de télé-conférence assistée par ordinateur.



Software de Base, Madrid : Responsable de différents aspects de l'architecture de réseaux, de l'ergonomie et des outils d'édition.



IET, Open University, Milton Keynes : Responsable de l'organisation et de l'évaluation des expérimentations des logiciels de Co-Learn sur treize sites pilotes en France, Espagne, Portugal et Ecosse.



Technische Hochschule Darmstadt : Responsable du développement de la base de données multimedia support de la médiathèque.



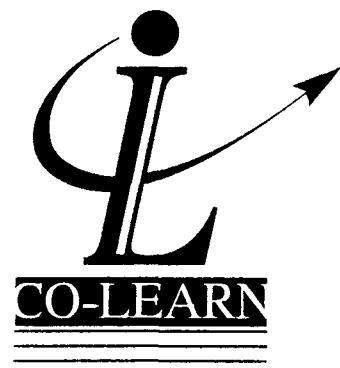
Dpto. de Ingeniería de Sistemas Telemáticos (DIT), Universidad Politécnica de Madrid: responsable de l'architecture du réseau Co-Learn.



INESC, Universidade de Aveiro: responsable du développement des outils d'édition, et de certains aspects de l'ergonomie du système.

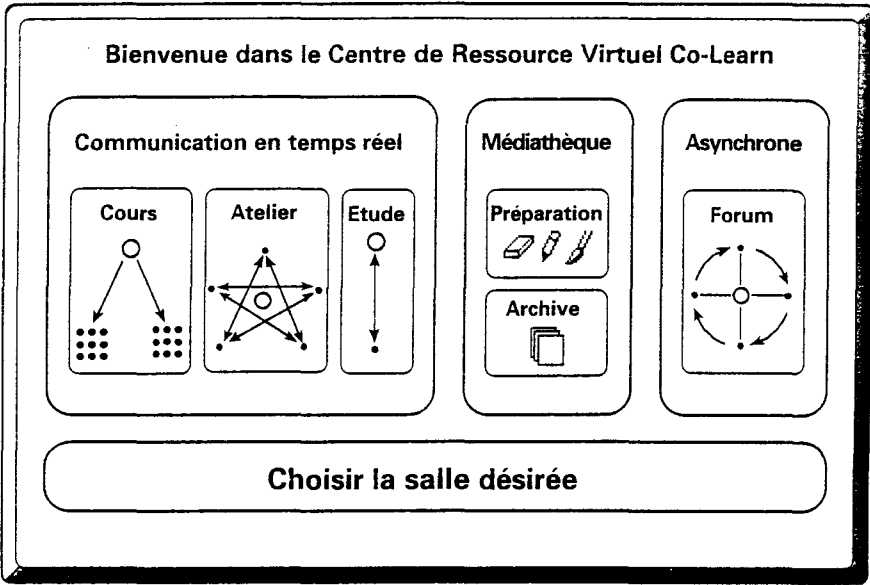
Le programme DELTA

Delta signifie *"Developing European Learning through Technological Advance"*. C'est un programme de la CEE de 3 ans qui a démarré en janvier 1992, avec un budget total de 54,5 millions d'écus. Il y a 22 projets retenus dans le programme, impliquant 174 institutions venant des douze états de la communauté et de quatre états de l'EFTA. L'objectif général de DELTA est de développer et de tester les usages des réseaux télématiques pour l'enseignement et la formation à distance en Europe.



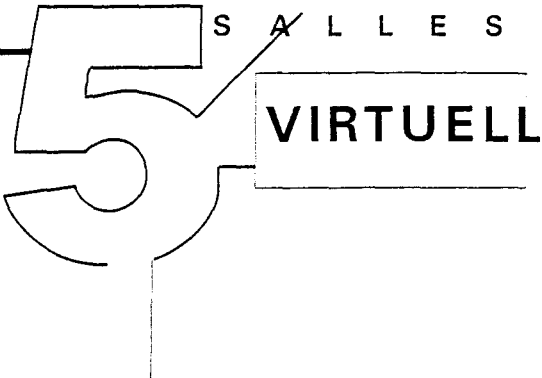
Co-Learn est un projet de recherche et développement appartenant au programme DELTA de la CEE.

Le Consortium en charge du projet conçoit un environnement de type "Collecticiel" pour l'apprentissage coopératif à distance, en utilisant des micro-ordinateurs connectés à un réseau RNIS. L'environnement Co-Learn inclura différents outils pour la communication en temps réel et asynchrone via des textes, des graphiques ou de la parole, ainsi que des outils pour la préparation, le partage et l'annotation de didacticiels ou de supports pédagogiques multimédia.



Co-Learn contribue au développement du concept d'enseignement à distance de troisième génération, dans lequel la communication entre, et parmi, les apprenants, tuteurs et personnes ressources, dans un centre de ressource virtuel, est un élément clef du processus d'apprentissage.

**Les Caractéristiques de
l'Environnement Co-Learn**



Télé-enseignement en temps réel

Pour des cours à distance entre un à plusieurs sites, en conjonction avec des supports de cours (transparents) qui ont été au préalable téléchargés sur les stations de travail distantes. L'enseignant peut contrôler la présentation sur les "tableaux noirs" distants et désigner différentes parties des écrans via un télépointeur. Après la phrase de cours magistral, les apprenants peuvent poser des questions par l'intermédiaire de canaux vocaux contrôlés par le serveur.

Conférence multimédia en temps réel

Pour un petit groupe de participants délocalisés (jusqu'à 5 ou 6), utilisant la métaphore de la salle de réunion virtuelle. Chaque participant possède sur sa station de travail un espace privé pour ses documents et il peut interagir avec un espace public commun dans le cadre d'une audioconférence synchrone. Un canal B du RNIS est utilisé pour la voix, le second pour l'échange des données. Quant les réseaux le permettent (débit large bande) la visioconférence est également possible.

Télé-assistance en temps réel

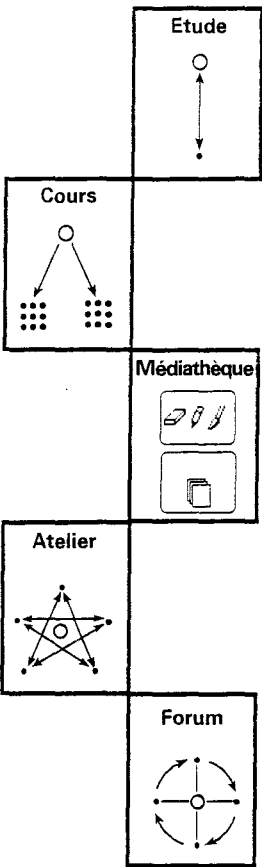
Ceci permet des connexions directes entre un site centre d'assistance et jusqu'à quatre stations "assistées". L'écran centre du tuteur ou conseiller sera capable de représenter l'état de chacun des écrans distants et le tuteur pourrait aider un apprenant (prise de main à distance) ou superviser jusqu'à quatre apprenants simultanément.

Une médiathèque électronique

C'est une banque de données multimédias contenant les supports pédagogiques informatisés (texte, images, didacticiels, hypermédia). L'enseignant ou le formateur peuvent, avec les outils spécialement développés dans Co-Learn, rendre disponible ces documents dans les différentes salles virtuelles Co-Learn

Forum

C'est un outil de télé-conférence assistée par ordinateur utilisant l'échange de messages textuels en temps différé: il est spécialement conçu pour supporter des activités d'apprentissage coopératif telles que des études de cas où la résolution collective de problèmes en tenant du jeu de rôle de chaque participant. Il est prévu de pouvoir attacher des annotations sonores et des fichiers graphiques aux messages.



Spécifications de Base

Les caractéristiques minima des stations de travail élève capable de supporter les outils Co-Learn sont les suivantes :

- Ordinateur PC, MS/DOS, 386, 25 MHz, SX ou DX
- 8 Mo RAM, au moins 120 Mo de disque dur.
- Un écran VGA (14 pouces ou plus)
- Souris et environnement Windows 3.1
- Une carte d'accès RNIS OST
- Un accès de base RNIS (S0)
- Un logiciel de réseau TCP/IP
- Une carte son compatible MPC Windows 3.1
- Un microphone
- Enceintes acoustiques

Certains autres équipements peuvent être nécessairement dépendants des conditions d'utilisation (salle de classe, amphithéâtre).

Ceci permettra l'utilisation des logiciels de communication individuelle ou de groupe, l'accès à la base de données contenant la médiathèque, l'annotation sonore, la communication vocal au travers d'un téléphone connecté au réseau RNIS ou par une ligne téléphonique standard supplémentaire. Des coûts de licence pour les logiciels pourront être requis pour certains des outils de Co-Learn.

Version Améliorée

- Conférence audionumérique**
Pour obtenir un son de qualité au travers d'un pont d'audioconférence numérique (BP : 7 KHZ) les équipements suivants seront nécessaires :
- une carte audio-multiplexeur H 221 de Prescom
 - un micro-casque adapté.

Ceci devrait aussi permettre aux sons issus d'une ligne téléphonique normale (BP : 4 KHZ) d'être mélange dans le pont rendant ainsi possible l'association d'un expert externe accédant par le réseau téléphonique normal.

- Son amplifié**
Pour le son amplifié, nécessaire pour le télé-enseignement en amphithéâtre, il faudrait ajouter :
- un préamplificateur/mixeur bas de gamme
 - un ou plusieurs microphones
 - un amplificateur de puissance et des enceintes acoustiques.

- La télé-écriture et les facsimilés**
Co-Learn supportera la télé-écriture via une tablette à digitaliser compatible Windows 3.1 et la saisie de documents sur un scanner.



Les Sites Pilotes Co-Learn

Institute of Educational Technology, Open University, Milton Keynes
Responsable de la planification globale des expérimentations *in situ* et des activités d'évaluation.

ECOSSE

L'évaluation est conduite en collaboration avec l'Institute of Computer Based Learning à l'Université Heriot-Watt (Edinburgh).

British Telecom, Aberdeen, la Network Services Agency à Inverness, et Highlands and Islands Enterprise coordonnent les essais dans les îles Orkneys, Shetland, Western, et Skye, en collaboration avec Kirkwall Community College, ACE-HI Invergordon, l'Université d'Aberdeen.

FRANCE

Evaluation conduite conjointement par le CUEEP, Université de Lille, et l'ENSSAT, Université de Rennes.

Université des Sciences et Technologies de Lille (U) Formation d'ingénieurs, math, physique

Neutron Lab, Archamps Formation au management

GREP, Rennes Enseignement de la langue japonaise

des Pays de la Loire, Nantes Formation aux nouvelles technologies dans les PME

Ecole d'Architecture de Nantes (Laboratoire CERMA) Cours de DEA conjoint avec l'Université de Grenoble

PORTUGAL

INESC, Université d'Aveiro Formation de médecins à la radiologie à Coimbra et Aveiro

ESPAGNE

L'Evaluation est conduite en collaboration avec la GATE, à l'Université Polytechnique de Madrid.

La Communauté de Madrid Formation permanente visant à développer l'employabilité

Université Polytechnique de Madrid Enseignement de 3ème cycle en ingénierie

Ministère de l'Education et de la Formation des Maîtres

Treize organisations ou institutions de formation de la communauté européenne sont associées au Consortium dans la définition et la mise en place des logiciels et serveurs Co-Learn. Le retour attendu des essais *in situ*, conduits avec divers publics cibles, et dans différentes matières, jouera un rôle déterminant dans le développement des prototypes Co-Learn et dans la réalisation du système final.



Membres du Consortium

Télesystèmes, Paris

Dominique Ray

tel: +331 46 14 51 78 fax: +331 46 14 5311
email: dr@telesys-innov.fr

CUEEP, Université de Lille 1

Alain Derycke

tel: +33 20 43 49 21 fax: +33 20 43 69 67
email: derycke@trigone.citilille.fr

Software de Base, Madrid

Gonzalo Sanchez Dueñas

tel: +341 803 2194 fax: +341 803 8555
email: gonza@softbase.es

Open University, Milton Keynes

Tony Kaye

tel: +44 908 653782 fax: +44 908 653744
email: a.r.kaye@open.ac.uk

Technische Hochschule Darmstadt

Hans-Jürgen Hoffmann

tel: +49 6 1511 63410 fax: +496 1511 66648
email: hoffmann@pu.informatik.th-darmstadt.de

INESC, Universidade de Aveiro

Joaquim Arnaldo Martins

tel: +351 34 20173 fax: +351 34 381128
email : jam@inesca.pt

Universidad Politecnica de Madrid

Encarna Pastor

tel: +341 336 7328 fax: +341 543 2077
email: encarna@dit.upm.es

France

• La CRCI des Pays de la Loire, Nantes

Christophe Parmentier

Tel: + 33 40 73 32 14 Fax: + 33 40 73.79 97

• L'Ecole d'Architecture de Nantes

Jean Pierre Peneau Tel: + 33 40 59 43 24 Fax: + 33 40 59 16 70

• Le Centre National d'Enseignement à Distance (CNED), Rennes

Isabelle Le Dilly

Tel: + 33 99 25 13 52 Fax: + 33 99 38 43 89

• L'Université des Sciences et Techniques de Lille-Flandres-Artois

Alain Derycke

Tel: + 33 20 58 11 11 Fax: + 33 20 58 11 10 Email: derycke@trigone.citilille.fr

• Neurope Lab, Archamps

Denis Servant / Richard Collin

Tel: + 33 50 31 56 20 Fax: +33 50 31 56 30 Email: collin@neuropeiab.unige.ch

SITES PILOTES

Portugal

• FUNDETEC, Aveiro / Hospital Universidade de Coimbra

Antonio M.M. Sousa Pereira

Tel: + 351 34 20173 Fax: +351 34 381128 Email: asp@inesca.pt

Scotland

• British Telecom, Aberdeen

Colin Pavey

Tel: +44 224 75 2258

Voice mail: +44 426 989346

Fax: +44 244 212451

Spain

• GATE, Universidad Politecnica de Madrid (UPM)

Manuel Fernandez de Villalta

Tel: + 341 336 6185 Fax: + 341 336 6173

Email: manuel.fernandez@gate.upm.es

• PNTIC, Ministerio de Educacion y Ciencia, Madrid

Carlos San Jose

Tel: + 341 377 1530 Fax: + 341 268 0709

Email: carlos.sanjose@scb.rediris.es

• Consejería de Educación, Comunidad de Madrid

Paloma Fernandez Gómez

Tel: + 341 580 4553 Fax: + 341 580 4555



ANNEXE II

SPECIFICATION DE

L'OUTIL DE TELEPOINTAGE

ET DE

L'OUTIL DE VOTE

SUIVANT LA

METHODE OOA DE SHLAER & MELLOR

SPECIFICATION DU TELEPOINTEUR

Modèle d'information

Cet outil est utilisé pour montrer les objets de la conversation (les objets sur lesquels on parle) visualisés grâce à l'outil de visualisation collectif. Il est aussi utilisé pour faire de l'annotation textuelle. L'outil de télépointage est toujours associé à un transparent, lui même associé à un document (ce dernier est contenu dans l'outil de visualisation collectif).

Le transparent est toujours placé sur la fenêtre correspondant à l'outil de visualisation collectif (fenêtre publique). Le télépointeur est un outil partagé qui obéit à des règles de partage.

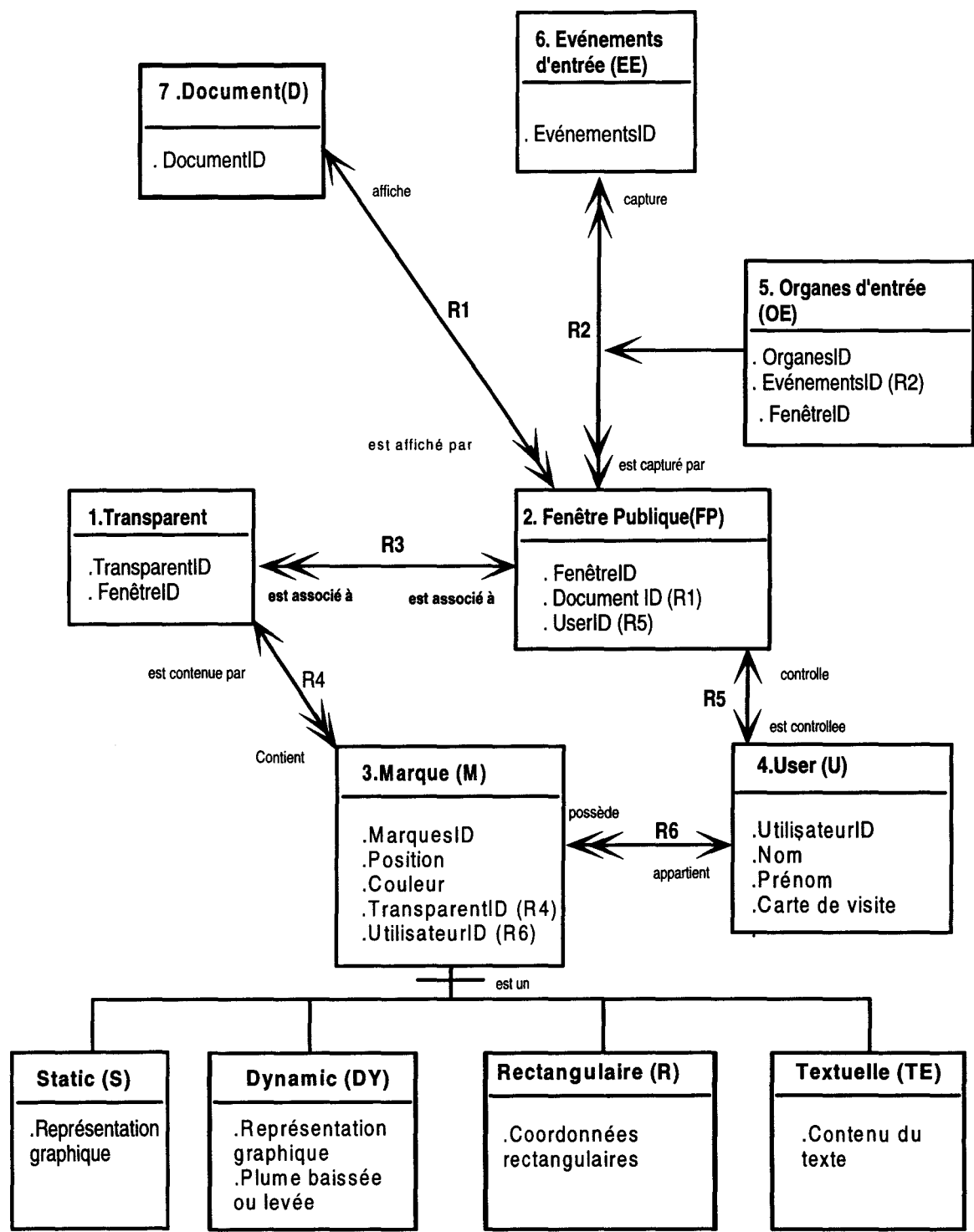
Le document doit être visualisé sur toutes les stations de telle façon à respecter le concept de Wysiwis.

Le télépointeur est utilisé par la personne qui en possède le contrôle, pour désigner certaines parties du document. Le transparent mémorise les marques déposées par les utilisateurs. L'utilisateur possédant le contrôle de l'outil peut effacer toutes les marques contenues par le transparent.

L'aspect visuel du télépointeur change en fonction de la marque choisie :

- marque statique : elle sert à désigner un objet qui a une représentation graphique bien délimitée comme une icône. Utiliser ce type de marque correspond à déposer une marque sous forme de "main" sur le transparent ;
- marque rectangulaire : elle sert à désigner un objet ou plusieurs objets qui non pas de représentation graphique bien définie (par exemple une image). Dans ce cas un rectangle apparaît sur toutes les stations utilisateur ;
- marque de désignation dynamique : elle simule le déplacement de la main devant une surface. Les utilisateurs visualisent le déplacement d'un curseur, qui a une forme de main. Cette "main" a la possibilité de laisser une trace sur le transparent (ligne de couleur) ;
- marque textuelle : elle sert à déposer du texte sur le transparent à l'endroit où on le désire.

Diagramme de structure



Définition des classes

Transparent : c'est un objet associé à la fenêtre publique.

Fenêtre publique : elle visualise une partie de l'information partagée (document) par le groupe et respecte le concept de Wysiwis.

Marque : correspond à l'aspect visuel du télépointeur. Nous avons défini quatre types de marque : statique, dynamique, rectangulaire, textuelle.

Utilisateur : correspond à l'utilisateur qui possède l'outil et qui peut déposer des marques. Il est identifié par le système et possède un nom et un prénom, ce qui permet pour les autres utilisateurs d'identifier qui possède l'outil.

Organes d'entrée : ce sont les périphériques utilisés par l'utilisateur pour créer les marques.

Événements d'entrée : événements Windows générés par les organes d'entrées.

Documents : ce sont tous les documents multimédia qui peuvent être visualisés à l'aide de l'outil de visualisation collectif.

Les trois dernières classes ne sont pas décrites complètement. Elles apparaissent pour expliciter certaines relations.

Définition des relations

R1 : un Document est affiché par une Fenêtre Publique.

R2 : un Événement d'entrée est capturé par la Fenêtre Publique.

R3 : un Transparent est associé à une Fenêtre Publique.

R4 : un Transparent contient des Marques.

R5 : un Utilisateur contrôle la Fenêtre Publique.

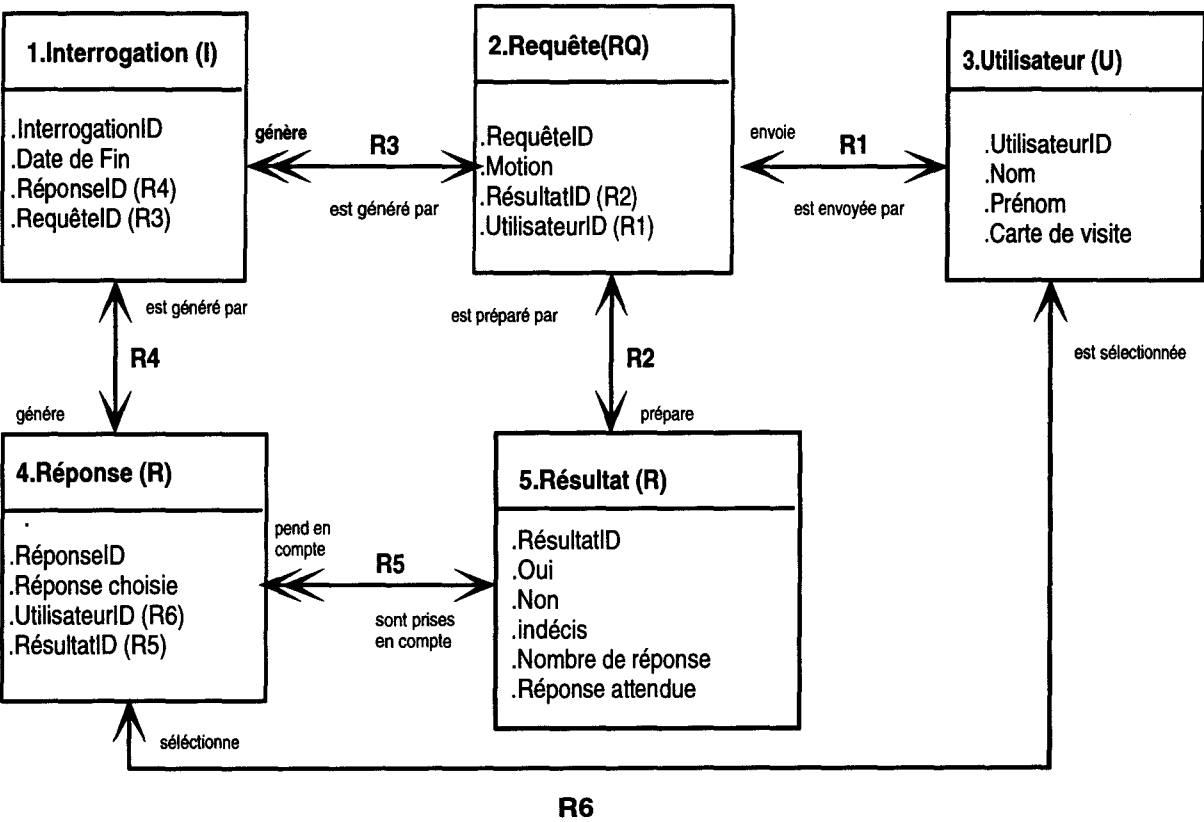
R6 : une Marque appartient à un Utilisateur.

SPECIFICATION DE L'OUTIL DE VOTE¹

Modèle d'information

L'outil de vote permet d'envoyer une requête aux autres utilisateurs présents dans la conférence. Cet utilisateur est appelé l'initiateur du vote. Une requête correspond à une question (motion du vote) à laquelle les utilisateurs peuvent répondre par oui, non ou je ne sais pas. Dans tous les cas, ils doivent répondre avant la fin du vote, qui est fixée par l'initiateur. Chaque utilisateur reçoit cette requête sous la forme suivante : elle contient la question qui est posée et les réponses possibles des utilisateurs. L'utilisateur vote en choisissant parmi les réponses possibles généralement en sélectionnant un bouton. Ce vote est alors pris en compte par l'objet Résultat. L'initiateur peut décider si les résultats intermédiaires doivent être visibles ou non. Il peut aussi permettre aux utilisateurs de modifier leur vote. Enfin, il peut arrêter à tout moment la procédure de vote. Le résultat s'énonce comme suit : le nombre de oui, le nombre de non, le nombre d'indécis, le nombre de réponses, le nombre de réponses encore attendues. Pour finir, l'initiateur peut choisir si le vote est anonyme ou pas.

Diagramme de structure



¹Cette outil a été spécifié et réalisé par Claude Viéville Ingénieur d'étude au laboratoire Trigone

Définition des classes

Utilisateur : une personne qui est identifié par le système. Chaque utilisateur possède un nom et un prénom. On lui associe une carte de visite qu'il peut modifier quand il le veut.

Requête : quand un utilisateur initie un vote, il définit une requête. Il remplit un formulaire contenant la motion du vote et la date de fin. Chaque membre de la conférence est concerné par le vote.

Interrogation : c'est le formulaire reçu par tous les utilisateurs. Il contient la motion du vote, des précisions sur la fin du vote et les boutons qui correspondent aux réponses possibles.

Réponse : elle contient le choix de l'utilisateur (oui, non, indécis) et des informations sur son identité.

Résultat : cet objet va prendre en compte l'ensemble des réponses afin de produire le résultat du vote : nombre de oui, de non, d'indécis, de votants, de non votants.

Définition des relations

R1 : un Utilisateur peut envoyer une requête (initier un vote).

R2 : la Requête prépare un objet Résultat qui va traiter les résultats.

R3 : une Requête génère une Interrogation pour tous les participants.

R4 : une Interrogation propose des réponses possibles aux utilisateurs, qui doivent en choisir une. On peut dire qu'une interrogation génère une Réponse.

R5 : une Réponse est traitée par l'objet Résultat.

R6 : un Utilisateur sélectionne une réponse parmi celles proposées par l'Interrogation.

ANNEXE III

REALISATION DE

L'OUTIL DE TELEPOINTAGE

ET DE

L'OUTIL DE VISUALISATION COLLECTIF

L'objectif de cet annexe est de donner un aperçu sur la réalisation de deux outils fortement liés que sont le télépointeur et l'outil de visualisation collectif. La description de la réalisation du télépointeur est légèrement différente en terme de classes, par rapport à la spécification donnée en annexe 2 (prise en compte des caractéristiques de l'environnement Windows).

Tous les développements ont été réalisés en C++ en utilisant le compilateur Borland et la librairie OWL (Object Window Library) fournit avec l'environnement Borland.

Le télépointeur

Cet outil est utilisé pour montrer les objets de la conversation (les objets sur lesquels on parle) visualisés grâce à l'outil de visualisation collectif. Nous avons déjà décrit ses fonctionnalités dans le chapitre 5 et aussi dans l'annexe 2.

L'outil télépointeur est réalisé par un ensemble de classes C++. La classe correspondant à la fenêtre principale se nomme TToolBar, et hérite de la classe TWindowCoop que nous avons défini au chapitre IV figure 9. Cette classe TToolBar doit son nom à la représentation visuelle qui lui est associée. Elle correspond à une palette d'icônes permettant d'accéder aux différents outils qui définissent les fonctions du télépointeur.

L'ouverture de la conversation DDE se fait avec les paramètres suivant : le nom de l'application est *telep* et *l'item* (ou topic) nécessaire est une chaîne de caractères au choix. Les paramètres nécessaires pour activer les liens chauds sont BCMDE et SCMDE. Ce mécanisme de communication DDE est pris en charge par la classe TWindowCoop.

La figure 1 donne une description simplifiée de la classe TToolBar.

L'outil télépointeur doit afficher sur la station où il se trouve, les marques créées sur les autres postes distants. Pour ceci, il reçoit des messages DDE qu'il traite dans la fonction de traitement des messages nommé Exec(). Cette fonction membre est héritée de la classe TWindowCoop, et doit être définie pour répondre aux comportements spécifiques de l'outil. La figure 2 donne une vision simplifiée de cette fonction membre, et décrit le traitement associé à quelques messages : messages liés à l'affichage des marques, et des messages liés à la gestion de la ressource (vouloir, rendre la ressource).

```

#include "ddesvr.h"           // gestion de la conversation DDE
#include "stockage.h"         //stockage des marques
#include "telep.h"           //définition des messages Windows pour l'échange entre
                             // telepointeur et fenêtre publique
#include "teleprc.h"         //ressources Windows
#include "icontab.h"         //gestion dynamique de la palette d'icônes
#include "cmd.h"             //analyseur syntaxique de commande DDE
#define DDENAME "telep"     //paramètre ouverture conversation DDE

class TToolBar : public TWindowCoop
{
private:
    FARPROC EmProc;
    HPEN BrushPen;
    PTState State;           //stockage de l'état de l'outil
    BOOL Drawing, Owner;
        // Owner = vrai, il existe un possesseur de l'outil
    char room[20];           //nom de la pièce où est engagé l'outil
    BOOL active_telep;       //définit si le télépointeur est actif
    HICON icur;              // identifie l'icône pour la désignation dynamique
    char UserName[20];       // nom de l'utilisateur local
    char OwnerName[20];      //nom du possesseur de l'outil
    ...

public:
    HWND WysiwisHandle;      //sauvegarde l'identifiant de la fenêtre publique
    TToolBar(PTWindowsObject AParent,LPSTR ATitle,char *r);
    virtual ~TToolBar();
    ...
    // messages envoyés par la fenêtre publique
        //la désignation vient de commencer
    virtual void WMLButtonDown_Wysiwis (TMessage& Message)=
        [ WM_FIRST +WM_LBUTTONDOWN_WYSIWIS];
        //la désignation vient de finir
    virtual void WMLButtonUp_Wysiwis (RTMessage)=[WM_FIRST + M_LBUTTONUP];
        //la fenêtre publique vient de finir son réaffichage : il faut afficher les marques
    virtual void WMPaintWysiwis (RTMessage)=[WM_FIRST + WM_PAINT_WYSIWIS];
        // fonction de reception de l'identifiant de la fenêtre publique
    virtual void WMInit(RTMessage)=[WM_FIRST + WM_INIT_TELEP];

    virtual BOOL Exec(char *);           //fonction de gestion des messages DDE

    void BroadcastBuffer(char *);        //fonction d'envoi de messages DDE
};

```

Figure 1 : La classe TToolBar

```

BOOL TToolBar:: Exec(char *cmd)
{
...
char *param;
// liste des messages DDE traités
static char *CmdList[]={ "owner","text","rect","putmark","trabegin","trato","traend","dynbegin",
    "dynto","dynend","user","showwant","hidewant","showrel","hiderel","showtake",
    "hidetake","showgive","hidegive","gom","seltool","add","remove",
    "mode","hidemode","setmode","color" };

static DDECmd cmds(CmdList,27);
cmds.SetCmdString(cmd);           //analyseur syntaxique de messages DDE
while (c=cmds.NextCmd()){
    switch (c){
        case 1: //[owner(auser)]           auser devient le possesseur de l'outil
        case 2: //[text (atext,X,Y,acolor)] affiche atext à la position x y
        case 3: //[rectangle(x,y,w,h,color,pensize)] affiche un rectangle
        case 4: //[putmark (X,Y)]           affiche une marque sous forme de main
        case 5: //[tracebegin(X,Y,color,penSize)] début de désignation avec trace en x y
        case 6: //[trato(x,y)]             déplace le télépointeur à la position x y
        case 7: //[traend()]                la désignation est terminée
        case 8: //[dynbegin(X,Y)]           début de désignation sans trace en x y
            WIS_DC = GetDC(WysiwiHandle);
            hdc= CreateCompatibleDC(WIS_DC);
            hBuf = CreateCompatibleBitmap(WIS_DC,32,32);
            X = atoi(cmds.NextParam()); Y = atoi(cmds.NextParam());
            //stockage du bitmap correspondant à la position courante du télépointeur
            SelectObject(hdc, hBuf);
            BitBlt(hdc, 0, 0, 32, 32, WIS_DC, X,Y, SRCCOPY);
            DrawIcon(WIS_DC,X,Y,icur);     //affichage de la marque
            DeleteDC(hdc);
            break;
        case 9: //[dynto(X1,Y1,...,Xn,Yn)] déplace le télépointeur successivement des positions X1,Y1 à Xn,Yn
        case 10: //[dynend()]                la désignation est terminée
        case 11 : //[user(username, toolID, toolboxID)]
        case 12 : //[showwant()]             montrer l'icône correspondant à l'action de demander
            WantOK=TRUE;
            index = add(&toolTab[0],index,wantIndex);
            MoveWindow(HWindow,Attr.X,Attr.Y,0,0,TRUE);
            break;
        case 13 : //[hidewant()]             cacher l'icône demander le télépointeur
        case 14 : //[showrel()]              montrer l'icône pour rendre le télépointeur
        case 15 : //[hiderel()]              cacher l'icône pour rendre le télépointeur
        case 16 : //[showtake()]             montrer l'icône pour prendre le télépointeur
        case 17 : //[hidetake()]             cacher l'icône pour prendre le télépointeur
        case 18 : //[showgive()]             montrer l'icône pour donner le télépointeur
        case 19 : //[hidegive()]             cacher l'icône pour donner le télépointeur
            index = concat(&toolTab[0],index,giveIndex);
            MoveWindow(HWindow,Attr.X,Attr.Y,0,0,TRUE);
            break;
        case 20 : //[gom()]                  supprime toutes les marques
    ...}
    }
return TRUE;
}

```

Figure 2 : La fonction membre de traitement des messages

Le fonctionnement de cet outil étant associé à un outil de visualisation, nous avons donc réalisé un mécanisme simple de communication entre ces deux outils. Pour gérer l'affichage des marques sur la fenêtre publique, le GDI (Graphique Device Interface) ou l'interface graphique de Windows a besoin de l'identifiant de la fenêtre (paramètre défini par le noyau Windows). Pour ceci, l'application télépointeur doit recevoir un message Windows (WM_INIT_TELEP) lui indiquant l'identifiant de fenêtre de l'outil de visualisation qui lui est associé. La connaissance de cet identifiant permettra à l'outil télépointeur de capturer les messages associés à la fenêtre publique lorsque le télépointeur sera actif (fonction SetCapture() du GDI Windows). La figure 3 donne une vision du fichier de déclaration servant à spécifier les messages échangés entre les deux outils.

```
//les messages Windows utilisés pour la communication entre l'outil de visualisation
//collectif et le télépointeur
#define WM_LBUTTONDOWN_WYSIWIS WM_USER +50 // permet de spécifier au télépointeur
// que la désignation vient de commencer
#define WM_PAINT_WYSIWIS WM_USER+51 // la fenêtre publique vient de se réafficher, elle
// demande au télépointeur de réafficher les marques
#define WM_ACTIVE_TELEP WM_USER+52 // permet de spécifier à la fenêtre publique que le
// télépointeur est actif ou inactif
#define WM_INIT_TELEP WM_USER+53 // permet au télépointeur de recevoir l'identifiant de la
// fenêtre publique
...
```

Figure 3 : Les messages utilisés pour la communication entre l'outil de visualisation collectif (fenêtre publique) et le télépointeur

Nous avons jusqu'à présent décrit le comportement de l'outil télépointeur pour les stations distantes. Pour la station où se trouve l'utilisateur qui génère les marques, le comportement de l'outil est pris en charge par plusieurs classes qui réalisent chacune une fonctionnalité de l'outil. Elles héritent toutes de la classe abstraite nommée TTelTool, qui définit le comportement générique de chaque classe représentant une fonctionnalité. La figure 4 donne une description de cette classe.

```

class TTelTool
{
protected:
    ...
    PTState State;          sauvegarde l'état courant de l'outil
    ...
public:
    ...
    TTelTool (LPSTR IconName);
    ~TTelTool();
    // la désignation commence (suite au message WM_LBUTTONDOWN_WYSIWIS reçu)
    void MouseDown ( HWND AWindow, int X, int Y, PTState AState);
    // la désignation est en cours (capture d'événements souris)
    void MouseMove (int X, int Y); )
    // la désignation est terminée
    void MouseUp ();

    //fonctions de rendu visuel à redéfinir pour chaque sous classe
    virtual void DrawBegin(int , int) {};
    virtual void DrawTo (int, int) {};
    virtual void DrawEnd () {};

    HCURSOR GetCursor() { return Cursor; };      // curseur associé à la fonctionnalité courante
    HICON GetIcon() { return Icon; };            // icône associée à la fonctionnalité courante
    virtual void Display(PTState AState) {};     // affichages des marques
    virtual void Delete() {};                   // suppression des marques
};

```

Figure 4 : La classe TTelTool

La réalisation du télépointeur repose sur une possibilité offerte par l'environnement Windows, de capturer tous les messages souris de n'importe quelle fenêtre, connaissant son identificateur. Le mécanisme de communication entre le télépointeur et la fenêtre publique permet de communiquer l'identifiant, et le début de la désignation. Ceci permet au télépointeur de piéger les événements souris associés à la fenêtre publique de telle façon à construire la représentation visuelle associée à la désignation courante. Ceci permet aussi de générer des messages DDE pour les répliques de l'outil télépointeur sur les sites distants, pour qu'elles puissent reproduire le phénomène (voir figure 6, pour une illustration de la construction et de l'envoi du message DDE). La figure 5 donne une vision du code correspondant à la capture des événements souris.

```

void TTelTool::MouseDown (HWND AWindow, int X, int Y, PTState AState)
{
// début de la désignation
POINT mark;
mark.x=X;
mark.y=Y;
Window = AWindow;
State = AState;
SetCapture(State->TToolBar_Handle); // capture de événements souris associés à la
fenêtre publique
Pen = CreatePen(PS_SOLID, State->penSize, State->penColor);
DC = GetDC(Window);
OldPen = SelectObject(DC, Pen);
DrawBegin(mark.x,mark.y); // construction visuelle locale et distribution du message
// DDE [Trabegin(x,y)]
}
//-----
void TTelTool::MouseMove (int X, int Y)
{
// traitement des déplacements souris capturés
POINT mark;
mark.x= X;
mark.y= Y;
ClientToScreen(State->TToolBar_Handle,&mark);
ScreenToClient(Window,&mark);
DrawTo(mark.x,mark.y); // construction visuelle locale et distribution du message [Trato(x,y)]
}
//-----
void TTelTool::MouseUp()
{
// Fin de la désignation
DrawEnd();
SelectObject(DC,OldPen);
ReleaseCapture(); // arrêt de la capture
ReleaseDC(Window, DC);
DeleteObject(Pen);
}

```

Figure 5 : Les trois fonctions membres qui gèrent la capture des événements souris

```

void TDynamicDesignation :: DrawBegin (int X, int Y)
{
...
Points->add(* (new TPoint(X, Y,State->brushColor,State->penSize))); //mémorisation de la marque
MoveTo(DC, X, Y); // action locale
// contruction du message DDE [trabegin(X,Y,aColor,penSize)
itoa(X,buf1,10); itoa(Y,buf2,10);
ltoa(State->brushColor,buf3,10); itoa(State->penSize,buf4,10);
sprintf(buf, "[trabegin(%s,%s,%s,%s)]",
        buf1,buf2,buf3,buf4);
//envoi du message
(State->TToolBar_Instance)->BroadcastBuffer(buf,NULL);
}

```

Figure 6 : Construction et envoi d'un message DDE

L'outil de visualisation collectif ou fenêtre publique

Cet outil est fortement couplé avec le télépointeur. Il affiche les documents qui vont correspondre aux objets de la conversation (les objets sur lesquels on parle). La description sera moins complète que la précédente : le mécanisme de communication DDE est le même que pour l'outil de télépointage car il hérite tous les deux de la même classe TWindowCoop. Nous allons plutôt insister sur les contraintes imposées pour qu'il y ait un bon fonctionnement avec l'outil de télépointage. Nous illustrons donc, non pas le développement complet de l'outil de visualisation, mais plutôt le squelette de toute application respectant le concept de Wysiwis, et voulant fonctionner avec le télépointeur.

La communication avec le télépointeur

La figure 7 illustre l'initialisation de la communication avec le télépointeur. Le comportement associé à la fenêtre publique a été défini dans une classe que nous nommons TTelWindow. La fenêtre publique diffuse le message WM_INIT_TELEP contenant en paramètre l'identifiant de fenêtre de telle façon que le télépointeur puisse récupérer l'identifiant de la fenêtre, afin de capturer les messages souris en mode télépointage.

```
void TTelWindow::SetupWindow()
{
    TWindowCoop::SetupWindow();
    ...
    // envoi du message WM_INIT_TELEP avec en paramètre l'identificateur de fenêtre
    if (GetModuleHandle(telep) == NULL) MessageBox(HWindow, telep_mes, "", MB_OK);
    else SendMessage(0xFFFF, WM_INIT_TELEP, (WORD)HWindow, MAKELONG(Topic, 0));
    ...
}
```

Figure 7 : Initialisation de la communication avec le télépointeur

A chaque fois que l'on active ou désactive le mode télépointage, le télépointeur diffuse le message WM_ACTIVE_TELEP. Ce message possède en paramètre l'état du mode. En fonction de cet état, la fenêtre publique adapte le traitement des messages souris qu'elle reçoit. Si on se trouve en mode télépointeur, la fenêtre publique diffuse le message WM_LBUTTONDOWN_WYSIWIS pour indiquer au télépointeur que la désignation a commencé. Dans le cas contraire, la fenêtre publique interprète ces messages pour son propre compte, afin de définir son comportement. La figure 8 montre le code que doivent contenir toutes applications fonctionnant avec le télépointeur.

```

#include "telep.h"          // identificateurs des messages de communication avec le télépointeur
#define DDENAME "wysiwis" // paramètre de conversation DDE
...
class TTelWindow:public TWindowCoop
{public:
    ...
    BOOL active_telep;    // mémorise l'état du mode de télépointage actif ou inactif
    TTelWindow (PTWindowsObject AParent, LPSTR ATitle, char *);
    TTelWindow::~TTelWindow();
    ...
    virtual void SetupWindow();
    virtual void WMLButtonDown (TMessage& )=[ WM_FIRST + WM_LBUTTONDOWN];
    virtual void WMActiveTelep (RTMessage )= [ WM_FIRST + WM_ACTIVE_TELEP];
    virtual void WMPaint(RTMessage)=[WM_FIRST + WM_PAINT];
    ...
    void Exec();    // boucle de traitement des messages
};
//-----
// fonction de traitement des messages souris
void TTelWindow::WMLButtonDown (TMessage& Mes)
{
    if (active_telep)        //cas ou le mode télépointage est actif
        SendMessage(0xFFFF,WM_LBUTTONDOWN_WYSIWIS,0,Mes.LParam);
    else
    {    // le mode télépointage est inactif. L'application définit son propre comportement
        ... }
    }
//-----
// traitement du message WM_ACTIVE_TELEP. Il permet la mise à jour de l'état du mode
void TTelWindow::WMActiveTelep(RTMessage Message)
{
    if(Message.LParam)
    {
        active_telep = TRUE; // le télépointeur devient actif
        ... }
    else
    {
        active_telep=FALSE; // le télépointeur devient inactif
        ... }
    }
//-----
// fonction de gestion de l'affichage: lorsque la fenêtre publique se réaffiche, il faut que le télépointeur
//réaffiche les marque. Elle envoie donc le message WM_PAINT_WYSIWIS
void TTelWindow::WMPaint (RTMessage)
{
    RECT Rect; PAINTSTRUCT ps;
    BeginPaint(HWindow,&ps);
    ...
    // demande de réaffichage des marques
    SendMessage(0xFFFF,WM_FIRST + WM_PAINT_WYSIWIS,0,0L);
    EndPaint(HWindow,&ps);
}

```

Figure 8 : Le squelette d'une application Windows fonctionnant avec le télépointeur

L'outil de visualisation collectif lui-même

L'explication complète du développement de cet outil apporterait peu de chose à la compréhension. Cet outil possède comme tout outil Co-Learn une boucle de traitement des messages semblable à celle décrite pour le télépointeur à la figure 2. La construction et l'envoi des messages DDE sont identiques à ceux illustrés pour le télépointeur en figure 6. La plus grande difficulté dans la réalisation de cet outil a été la gestion de l'affichage de la fenêtre principale, et des fenêtres filles qui permettent de donner la dimension multi-document à l'outil (vision simultanée de plusieurs documents). Dans l'environnement Windows, ce mécanisme existe déjà avec les applications MDI (Multiple Documents Interface). Il ne nous a pas été possible d'utiliser ce mécanisme, car certains comportements sont gérés au niveau du noyau Windows.

Une deuxième difficulté (non pas par la complexité, mais par le temps passé) a été l'affichage dans une fenêtre Windows de fichier au format Bitmap (Facile), WMF (Windows MetaFile) (déjà un peu plus difficile) et enfin du format Write (beaucoup plus complexe). Nous n'avons pas utilisé la technologie OLE, car en début 92, le compilateur Borland ne fournissait pas d'encapsulation de ces mécanismes. Il aurait fallu travailler au niveau de l'API Windows, avec peu d'information sur les mécanismes. Une nouvelle version de cet outil devrait voir le jour en début 95 en utilisant cette technologie OLE.

La troisième difficulté était la distribution des documents (fichier au format Bitmap, WMF et Write) sur toutes les stations, et la réalisation du mécanisme de copier coller collectif. Nous avons utilisé une solution pas très élégante, mais qui nous a permis d'obtenir rapidement des résultats et une grande fiabilité du mécanisme (nécessaire sur RNIS). Le logiciel TCP que nous utilisons offre la possibilité de partager un disque distant se trouvant sur un serveur Unix proposant les services NFS. Les documents sont téléchargés à partir de ce disque distant à chaque début de session, ou en cours de session (parfois avec des temps de chargement un peu long). Le mécanisme de presse papier collectif utilise ce disque distant en créant un fichier temporaire, accessible en lecture pour toutes les stations. Le document contenu dans le fichier peut alors être visualisé sur toutes les stations. Nous avons utilisé des petits algorithmes de compression fournis en standard sous Windows pour les formats Bitmap, afin de réduire les temps de transfert. Cette solution aurait dû être remplacée par les services de la couche de communication fournis par nos partenaires d'UPM (Université Polytechnique de Madrid). Malheureusement, les performances et la fiabilité de ce service, ne nous ont pas satisfaits.