

UNIVERSITÉ DE MONTRÉAL

ARCHITECTURE D'ESPACE D'APPRENTISSAGE COLLABORATIF
SUPPORTANT LA QUALITÉ DE SERVICE

DAVID TASSY

DÉPARTEMENT DE GÉNIE INFORMATIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE)

DÉCEMBRE 2002

© David TASSY, 2002



National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-81563-3

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

CE MÉMOIRE INTITULÉ :

ARCHITECTURE D'ESPACE D'APPRENTISSAGE COLLABORATIF
SUPPORTANT LA QUALITÉ DE SERVICE

présenté par : TASSY David

en vue de l'obtention du diplôme de : maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen composé de :

Mme. BOUCHENEB Hanifa, Ph.D., présidente

M. PIERRE Samuel, Ph.D., membre et directeur de recherche

M. QUINTERO Alejandro, Ph.D., membre et codirecteur de recherche

Mme. DUFRESNE Aude, Ph.D., membre

REMERCIEMENTS

Je voudrais exprimer mes remerciements à mon directeur de recherche, le professeur Samuel Pierre, pour sa disponibilité, son support et son encadrement durant mon cursus en maîtrise.

L'expression de ma reconnaissance va également à l'endroit de mon codirecteur de recherche, le professeur Alejandro Quintero, pour son dévouement, sa disponibilité, sa patience, son support et ses commentaires tout au long de ce projet. Mes remerciements vont ensuite à tous les membres du LARIM qui ont toujours été très disponibles pour m'aider si nécessaire, Fabien, Khaled, Doug et les autres.

Enfin, si ce mémoire a pu aboutir, c'est aussi grâce au soutien inconditionnel de mes parents et de mes amis à qui je désire exprimer ma gratitude pour leurs encouragements et leur support infailible.

RÉSUMÉ

Les laboratoires et les travaux d'expérimentation représentent un mode d'enseignement confortable, évolué, interactif et adaptable aux besoins des apprenants. Par ailleurs, les technologies se sont modernisées et permettent des applications très intéressantes. En considérant le potentiel de ces technologies, il est logique de vouloir les utiliser dans les environnements d'apprentissage. En effet, l'adaptation du concept de laboratoire et de tous ses composants au domaine informatique apparaît comme une alternative séduisante. Cependant, cette perspective induit forcément des contraintes au niveau de la manière d'ordonner l'apprentissage qui y est pratiqué. En effet, afin de reconstituer le plus fidèlement possible l'apprentissage traditionnel, il convient d'y inclure le concept de collaboration. Ainsi, un laboratoire qui serait créé sans possibilité d'offrir cette collaboration atteindrait vite ses limites. C'est dans cet esprit que le laboratoire virtuel développé au LARIM s'inscrit comme projet de recherche auquel nous tentons d'apporter notre contribution grâce à ce mémoire.

Le problème de la collaboration et de son utilisation pour l'enseignement est une donnée importante de notre recherche. En effet, la notion de travail en groupe que l'on retrouve dans un laboratoire, implique forcément celle de collaboration. Dans cette recherche, nous nous sommes donc appliqués à adapter la définition traditionnelle de ce concept en considérant toutes les contraintes inhérentes à l'utilisation des réseaux. Cette étape nous a conduit à constituer une grille d'analyse et de décomposition permettant de détailler les composants d'une architecture constituant une solution à ces problèmes. Cette spécification nous a permis aussi de confirmer le recours à des outils de coopération comme la visioconférence. En outre, comme nous l'avions imaginé au début de la recherche, le recours à ces outils entraîne forcément des problèmes de qualité de service. En effet, la qualité pédagogique, la qualité du contenu ainsi que celle des transmissions revêtent une importance prépondérante. L'apprentissage ne doit pas être

affecté par des faiblesses à ces niveaux et, de ce fait, il nous appartient donc de trouver les meilleures solutions.

Afin de rendre la plate-forme utilisable et de pouvoir l'imposer comme référence pour un laboratoire générique, il faut renforcer les aspects de collaboration et ceux ayant un lien avec la QoS. Au niveau informatique, c'est sur ce dernier point que nous avons concentré nos efforts pour améliorer l'ensemble. Ainsi, nous avons envisagé une architecture permettant de gérer la collaboration d'une part, mais d'appréhender d'autre part les problèmes de QoS qu'elle induit en responsabilisant l'utilisateur à ce niveau. En envisageant de différencier les flux en fonction des besoins, nous avons conjecturé qu'une adaptation à l'environnement et à la session avait une influence bénéfique sur les performances de l'ensemble.

Afin de tester notre modèle, nous avons eu recours au simulateur OPNET qui permet une analyse complète et précise des réseaux. Si l'idée de départ était de comparer les performances de différents contrats de QoS, allant d'un niveau faible à élevé, les premières manipulations nous ont poussé à adapter cette manière de voir les choses. En effet, nous avons préféré distinguer les contrats en fonction des applications auxquelles l'utilisateur aura recours dans sa session de laboratoire. Grâce aux fonctionnalités offertes par OPNET, nous avons pu, à partir de ces différenciations, montrer les améliorations de performance mesurées en termes de délai, de gigue et de pertes que cette méthode assure.

Les simulations menées sous OPNET nous ont permis d'affiner cette idée puisqu'il ressort que la solution la plus appropriée est une spécialisation des niveaux en fonction de l'état de collaboration requis : l'un orienté vidéo, l'autre voix et enfin le dernier permettant d'assurer les transferts HTTP. Les résultats obtenus pour ces cas sont relativement probants, notamment lorsque la charge dans le réseau augmente. Ainsi, nous pouvons affirmer que la différenciation doit permettre de donner une plus grande homogénéité à l'ensemble des échanges, sans privilégier l'un ou l'autre.

ABSTRACT

The laboratories and work of experimentation represent a comfortable, advanced, interactive and adaptable mode of teaching. Moreover, technical progress is consequent and allows very interesting applications. According to those elements we decided to use them in such environments. Indeed, the adaptation of the concept of laboratory and all its components, in the computer science field, seems a tempting alternative. However does this prospect induce constraints on the level in which we organize education in those environments. Indeed in order to recreate traditional education, we have to introduce the concept of collaboration. Thus, a laboratory designed without possibilities of collaboration would quickly become old-fashioned powerful. For this reason the virtual laboratory developed by the LARIM is a research project we try to improve in this study.

The problem of collaboration and its use for teaching is a significant data of our search. Indeed, the concept of work in group in a laboratory, implies for sure that of collaboration. Thus in this search we endeavoured to adapt the traditional definition of this concept by considering all the constraints inherent in the use of the networks. This stage led us to constitute a grid of analysis and decomposition allowing to detail the components of an architecture considering those problems. This specification also implies the use of co-operation tools like the videoconference. Moreover, the recourse to these tools inevitably involves problems of quality of service. Indeed, the quality of teaching, the quality of the contents as that of the transmissions have a dominating importance. In order to make the platform useful and to impose it like reference for a generic laboratory, it is necessary to optimize these parameters. At the computer level we focused on that point to improve the unit. Thus we considered an architecture allowing to manage collaboration, but also to apprehend the problems of QoS triggered by giving responsibilities to the user on this level. While planning to differentiate flows

according to the needs, we conjectured that an adaptation to the environment and the session had a beneficial influence on the performances of the unit.

In order to test our model, we had recourse to the simulator OPNET which allows a complete and precise analysis of networks. If the starting idea were to compare the performances of various contracts of QoS, the first handling pushed us to adapt this. Indeed, we preferred to distinguish the contracts according to the applications to which the user will have recourse in his session of laboratory. Thanks to the functionalities offered by OPNET we could, starting from these differentiations to show the improvements on the level performance of time, jitter and losses which this method ensures.

The results of our simulations made in this study are rather significant and convincing. Indeed, we made specific groups for the applications (HTTP, voice, video). Then we had to test our model by using different loads in the network. We could notice that the improvement caused by the differentiation is sensitive without special load, and it becomes even more significant when the number of users increases.

TABLE DES MATIÈRES

REMERCIEMENTS.....	iv
RÉSUMÉ.....	v
ABSTRACT.....	vii
LISTE DES ABRÉVIATIONS.....	xii
LISTE DES FIGURES.....	xiii
LISTE DES TABLEAUX.....	xv
 CHAPITRE I INTRODUCTION.....	 1
1.1 Définitions et concepts de base.....	1
1.2 Éléments de problématique.....	3
1.3 Objectifs de recherche.....	6
1.4 Plan du mémoire	6
 CHAPITRE II COLLABORATION DANS LES ENVIRONNEMENTS D'APPRENTISSAGE ÉLECTRONIQUE	 8
2.1 Apprentissage électronique.....	8
2.1.1 Définition et caractérisation	8
2.1.2 Complémentarité et améliorations	11
2.2 Laboratoires virtuels et télé-laboratoires.....	14
2.2.1 Laboratoires virtuels basés sur la simulation interactive	15
2.2.2 Laboratoires virtuels élaborés et interactifs	16
2.3 Modélisation de la collaboration.....	19
2.3.1 Caractérisation de la collaboration.....	19
2.3.2 Analyse des outils	21
2.3.3 Survol des environnements existants	24
2.3.4 Le concept de « Collaboratory »	27
2.4 Qualité de Service	29
2.4.1 La QoS dans les réseaux	29

2.4.2 La QoS pour un environnement collaboratif.....	32
2.4.3 La QoS pour une plate-forme d'apprentissage	35
CHAPITRE III ARCHITECTURE POUR UN ESPACE COLLABORATIF	38
3.1 Requis de l'architecture de collaboration.....	38
3.1.1 Dimensions expérimentales de l'apprentissage	38
3.1.2 Les protagonistes du laboratoire	39
3.1.3 Prise en compte de l'aspect de collaboration	42
3.2 Classes d'interaction de la collaboration.....	46
3.2.1 Coopération.....	47
3.2.2 Communication.....	51
3.3 Scénarii et architecture générale de collaboration	56
3.3.1 Scénarii les plus représentatifs.....	57
3.3.2 Architecture de collaboration.....	61
3.4 Préoccupations de qualité de service.....	68
3.4.1 Définition	68
3.4.2 La QoS dans le laboratoire virtuel	68
3.4.3 Solutions retenues	70
CHAPITRE IV IMPLÉMENTATION ET RÉSULTATS.....	71
4.1 Détails d'implémentation.....	71
4.1.1 Spécifications matérielles et logicielles	72
4.1.2 Environnement de réseau et protocoles.....	73
4.1.3 Implémentation de la QoS.....	75
4.2 Mise en oeuvre de l'environnement.....	79
4.2.1 Scénario d'utilisation	79
4.2.2 La plate-forme.....	80
4.3 Simulation et résultats.....	85
4.3.1 Définition du modèle	86
4.3.2 Plan d'expérience.....	90

4.3.3	Analyse des résultats	94
CHAPITRE V CONCLUSION.....		105
5.1	Synthèses des travaux et contributions	105
5.2	Limitations des travaux	106
5.3	Orientations de recherches futures	107
BIBLIOGRAPHIE		109

LISTE DES ABRÉVIATIONS

ATM	Asynchronous Transfer Mode
CSCW	Computer-Supported Cooperative Work
CVE	Collaborative Virtual Environment
DIVA	Développement Intégration et évaluation des environnements d'apprentissage
DSCP	Differentiated Services CodePoint
HTTP	HyperText Transfer Protocol
IETF	Internet Engineering Task Force
INVITE	Intelligent Distributed Virtual Training Environment
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
JETS	Junior Engineering Technological Society
LVE	Learning Virtual Environment
MPLS	Multiprotocol Label Switching
OPNET	Optimum Network Performance
QOS	Quality of Service
RMI	Remote Method Invocation
RPC	Remote Procedure Call
RSVP	Resource Reservation Protocol
RTP	Real Time Protocol
RTCP	Real Time Control Protocol
TOS	Type of Service
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

LISTE DES FIGURES

Figure 2.1 Interface du laboratoire de thermodynamique.....	16
Figure 2.3 Le concept de laboratoire virtuel de génie électrique en réseau.....	18
Figure 2.4 Composants de base d'un environnement d'apprentissage.....	25
Figure 2.5 Architecture du projet JETS.....	27
Figure 2.6 Contrôle de QoS.....	34
Figure 2.7 Échelle du taux de qualité.....	35
Figure 2.8 Trois aspects de la QoS dans un DLE.....	36
Figure 3.1 Composantes pédagogiques d'un environnement d'apprentissage.....	39
Figure 3.2 Les protagonistes ou acteurs.....	41
Figure 3.3 Place de la collaboration dans le processus d'apprentissage.....	43
Figure 3.4 Décomposition de la collaboration.....	44
Figure 3.5 Grille d'analyse et de structuration pour la collaboration.....	45
Figure 3.6 Besoins du coordonnateur.....	48
Figure 3.7 Besoins des élèves.....	49
Figure 3.8 Outils de coopération.....	51
Figure 3.9 Liaisons « unicast ».....	54
Figure 3.10 Liaisons « broadcast ».....	54
Figure 3.11 Liaisons « multicast ».....	55
Figure 3.12 Scénario de résolution d'exercices.....	58
Figure 3.13 Scénario de résolution d'exercices.....	59
Figure 3.14 Scénario de manipulation à distance.....	60
Figure 3.15 Scénario de manipulation à distance.....	62
Figure 3.16 Architecture globale de collaboration.....	64
Figure 4.1 Les protocoles des couches transport et application.....	75

Figure 4.2 Champ DSCP.....	77
Figure 4.3 Le JMenu du laboratoire.....	81
Figure 4.4 Interface de coopération	82
Figure 4.5 L'Interface de Coopération et le « chat »	84
Figure 4.6 Module de sécurité.....	85
Figure 4.7 Le réseau de simulation	87
Figure 4.8 L'application HTTP	88
Figure 4.9 L'application de voix.....	89
Figure 4.10 La visioconférence.....	90
Figure 4.11 Profils des applications <i>utilisateur</i> et <i>GroupeUtilisateurs</i>	93
Figure 4.12 Comparaison du temps de réponse pour <i>Utilisateur</i> et <i>Temoin</i>	95
Figure 4.13 Temps de réponse en fonction de la Congestion : <i>Temoin</i>	96
Figure 4.14 Temps de réponse en fonction de la congestion : <i>Utilisateur</i>	96
Figure 4.15 Comparaison du délai pour <i>Utilisateur</i> et <i>Temoin</i>	97
Figure 4.16 Délai en fonction de la Congestion : <i>Temoin</i>	98
Figure 4.17 Délai en fonction de la Congestion : <i>Utilisateur</i>	98
Figure 4.18 Pertes en fonction de la Congestion : <i>Temoin</i>	99
Figure 4.19 Comparaison du délai pour <i>Utilisateur</i> et <i>Temoin</i>	100
Figure 4.20 Délai en fonction de la Congestion : <i>Temoin</i>	101
Figure 4.21 Délai en fonction de la Congestion : <i>Utilisateur</i>	101
Figure 4.22 Pertes en fonction de la Congestion : <i>Temoin</i>	102

LISTE DES TABLEAUX

Tableau 4.1 Comparatifs des performances de délai pour <i>l'Analyse</i>	103
Tableau 4.2 Variations de délai pour le <i>Temoin</i>	103
Tableau 4.3 Variations de délai pour l' <i>Utilisateur</i>	104

CHAPITRE I

INTRODUCTION

Les avancées technologiques dans le domaine de l'information et de la communication permettent aujourd'hui d'envisager des applications insoupçonnées jusque là. En effet, l'utilisation d'infrastructures de réseaux performantes offre des possibilités intéressantes. Ainsi, l'évolution de la technologie Internet nous a conduit à un point où des centaines de millions de personnes ont accès à un nombre phénoménal de sources d'information et de services à travers le monde. De ce fait, le partage du savoir est entrain de s'universaliser et ce, dans tous les domaines. Cependant, il est parfois préférable d'essayer d'harmoniser un peu les efforts lancés de tous les cotés. Un développement qui ne serait pas pris en main pourrait en effet empêcher une optimisation des possibilités offertes. L'apprentissage est un domaine qui ne fait d'ailleurs pas exception. À cet effet, des professeurs, des chercheurs et de nombreux intervenants dans le milieu de l'enseignement se sont penchés sur le problème de l'enseignement virtuel comme alternative pertinente à un enseignement que nous qualifierons de plus traditionnel. Nous avons d'ailleurs pu constater la multiplication d'initiatives dans ce sens avec la création d'un grand nombre de laboratoires virtuels au cours de ces dernières années, laboratoires qui nous intéressent tout particulièrement dans ce projet de maîtrise. Dans ce chapitre d'introduction, nous présenterons d'abord quelques définitions et concepts de base nécessaires à la bonne compréhension de la problématique. Nous nous concentrerons ensuite sur la problématique qui guidera l'ensemble de la recherche avant de définir les objectifs à atteindre au cours de ce travail et le plan du mémoire.

1.1 Définitions et concepts de base

Si le concept de *laboratoire virtuel* n'est pas nouveau, voyons exactement ce qu'on entendra en l'évoquant. En effet, on peut trouver de nombreuses fois dans la littérature ce terme de laboratoire. Et même si les définitions ne sont jamais exactement

similaires, force est de constater que le fond est le même, que le but recherché est identique. À chaque fois, la notion *d'apprentissage à distance ou télé-apprentissage*, est centrale ; il est important d'insister sur ce terme car on distingue les SVE (shared virtual environments), CVE (collaborative virtual environments), et les LVE (learning virtual environments) qui sont destinés spécifiquement à l'enseignement. Cette caractéristique essentielle aura des conséquences importantes sur les pistes de réflexion puisque l'on se fixe comme objectif final de transmettre des connaissances. C'est cet *apprentissage* qui oriente tous les choix de conception et d'architecture de ces laboratoires.

Cependant, celui auquel nous ferons le plus fréquemment référence dans ce projet de maîtrise est celui développé spécifiquement au LARIM (Laboratoire de recherche en Réseautique et en Informatique Mobile). Un précédent projet de maîtrise, mené par Freddy Vé [1], a ainsi proposé une méthodologie concrète de conception de laboratoire. C'est d'ailleurs à partir de ce travail, des concepts mis en avant à cette occasion, que se basera cette recherche. Si la méthodologie de base est maintenant déployée, il reste cependant certains points à développer. La *collaboration* en fait partie.

On trouve ainsi toute sorte de laboratoires qui sont définis en détails dans certains articles [2]. Les fonctionnalités sont nombreuses et celui développé au LARIM en possède beaucoup. Dans ces plates-formes, on a réussi à implanter de nombreux outils qui permettent de recréer de façon virtuelle tout l'environnement d'un laboratoire réel.

Ainsi, dans le modèle proposé, un étudiant qui se connecte se voit offrir plusieurs options parmi lesquelles :

- remise à niveau et précisions concernant le cours ;
- évaluation des connaissances ;
- mise en pratique par des manipulations ;
- remise du rapport.

On peut donc retrouver toute la démarche habituellement suivie pour l'apprentissage par des laboratoires. Cependant, on ne peut limiter ces séances à ces quatre points seulement. En effet, ce serait un peu trop réducteur par rapport aux

multiples opportunités qu'offrent généralement des séances de travail en commun, qu'elles soient encadrées ou libres.

La *collaboration* est définie comme l'ensemble des processus et outils (*multimédia* ou autres) mis en œuvre pour permettre aux acteurs des laboratoires de communiquer entre eux, de s'échanger des informations et par conséquent de travailler en groupe afin de partager des savoir-faire et des connaissances. Ceci doit leur fournir des visions différentes et élargies des problèmes. La collaboration est donc ce qui permettra aux protagonistes de parvenir à leurs objectifs grâce à une mise en commun des efforts de travail.

Le « *Collaboratory* » [3] est un concept nouveau qui fait référence à un laboratoire virtuel à l'intérieur duquel les mécanismes de collaboration sont effectifs. Cependant, il nous incombe de lui donner une signification plus concrète en l'appliquant à cette recherche.

Enfin, la *qualité de service (QoS)* regroupe les critères que l'on choisit pour évaluer la performance de l'architecture. En effet, suivant les applications que l'on choisit d'utiliser (vidéo, chat, ...), différents paramètres sont plus ou moins pertinents à prendre en compte, comme par exemple : délai, gigue, pertes des paquets, engorgements, etc. Cependant, cette liste n'est pas exhaustive et nous devons dans un premier temps définir précisément ce dont nous aurons besoin dans notre étude.

1.2 Éléments de problématique

La plupart des environnements actuels qui ont déjà intégré de nombreux outils dans leur architecture n'ont souvent de collaboratif que le nom. En effet, quand on prend en détail les caractéristiques et les scénarii de déroulement de séance de laboratoire, on s'aperçoit que le terme *collaboration* est souvent utilisé comme mot passe partout. S'ils ont fréquemment les supports nécessaires pour de telles interactions, ils sont pourtant rarement employés comme ils pourraient l'être. Pourtant, la fusion des différents médias de l'information tels que la voix, les données et la vidéo pourrait

donner un outil très intéressant dans cette optique. Il est relativement séduisant de penser que l'utilisation de ces instruments pourrait accroître grandement les possibilités offertes aux étudiants. À partir de ce constat, le LARIM, dans son projet de développement d'un environnement d'apprentissage multi-utilisateurs, cherche donc à explorer des pistes crédibles dans ce domaine, qui permettraient d'affiner les objectifs pédagogiques. Ainsi, si on part bien évidemment de la simulation d'équipements déjà existants ainsi que des expériences en ligne propres aux différents laboratoires, il faut aussi rajouter des possibilités offertes par l'utilisation du multimédia, pour rendre par exemple les échanges plus vivants.

Si on n'essaye pas de pousser un peu plus loin la démarche de mise en place d'une méthodologie de conception d'environnement d'apprentissage virtuel, on pourrait passer à côté du scénario réel global de déroulement d'un laboratoire. Il ne faut donc pas négliger les interactions entre les différents acteurs humains présents dans le laboratoire. Et de manière à appréhender efficacement la notion d'apprentissage, il semble nécessaire de considérer que le travail à plusieurs a souvent bien des vertus. Chez les élèves qui l'apprécient, comme chez les professeurs d'ailleurs qui l'utilisent souvent, on constate de plus en plus le besoin de travailler en groupe. Or cette manière de travailler ne devrait pas être limitée par l'utilisation des technologies actuelles. Un étudiant qui désire travailler son laboratoire tout en restant chez lui doit être en mesure de retrouver tous les avantages que lui confère l'interactivité d'une séance traditionnelle. Ainsi, en cas de difficultés, d'incompréhensions ou de manipulations délicates, la plate-forme utilisée doit lui offrir la possibilité de faire appel à une aide extérieure. Le professeur ou chargé de laboratoire doit pouvoir lui prodiguer des conseils, être en mesure de répondre à ses questions ou d'évaluer le progrès du travail réalisé. De plus, il faut aussi que les étudiants puissent travailler ensemble, communiquer et échanger leurs idées à ce sujet. C'est dans ce domaine, dans la recherche d'une interaction plus poussée et plus conviviale, qu'il reste du travail à accomplir, et c'est à ce sujet qu'il sera intéressant de préciser le concept de « Collaboratory ».

La méthodologie d'intégration que préconise le LARIM pour les applications de son laboratoire prend en compte ces spécifications. Elle vise particulièrement à gérer au mieux les contraintes d'apprentissage, de collaboration et de qualité de service en veillant à ne pas les séparer et même au contraire à n'en faire si possible plus qu'une. Elle est composée de trois grandes parties :

- évaluation des besoins et la définition de tous les paramètres ;
- mise en place d'un modèle d'architecture ;
- élaboration de la plate-forme ;
- évaluation pédagogique.

L'évaluation des besoins et la définition des paramètres consistent à faire un inventaire le plus complet possible des besoins au niveau du matériel nécessaire et des interactions, mais aussi en terme de qualité de service et de faisabilité de projet. Il faut commencer par répertorier les protagonistes, les échanges, les performances et tout ce qui définira l'architecture de notre plate-forme. La mise en place du modèle touche à l'élaboration précise du scénario possible ainsi qu'aux solutions envisagées et leur faisabilité. L'élaboration, quant à elle, traite plus de la programmation à effectuer pour rendre effectifs les changements envisagés. Enfin, l'évaluation pédagogique tend à estimer si l'espace développé va permettre d'atteindre des objectifs pédagogiques pertinents. Ce projet de maîtrise concerne plutôt les deux premières parties de la démarche globale, notamment tout ce qui a trait à la qualité de service.

Cette démarche doit s'inscrire dans le processus de réflexion concernant les environnements virtuels, tout en n'omettant pas de se poser certaines questions essentielles aux objectifs que l'on se fixera. En effet, on peut se demander comment utiliser les travaux déjà réalisés, comment tirer profit des facilités déjà implantées pour profiter au maximum d'un échange d'information efficace entre les protagonistes du laboratoire ? Comment exploiter les infrastructures en place pour réaliser une architecture performante au niveau d'un espace d'apprentissage accessible, en insistant

notamment sur toute la dimension interactive et collaborative que l'on souhaite développer ?

1.3 Objectifs de recherche

L'objectif principal de ce mémoire est de modéliser la collaboration entre intervenants dans un environnement d'apprentissage virtuel en utilisant pour cela les infrastructures réseaux et tous les outils disponibles dans les laboratoires, tout en s'assurant de respecter des contraintes de QoS efficaces et pertinentes dans le cadre de ces plates-formes. De manière plus spécifique, nous visons les objectifs suivants :

- définir le problème de qualité de service en faisant ressortir les éléments intéressants qui peuvent s'appliquer simultanément à l'apprentissage, le multimédia et la collaboration ;
- modéliser une architecture et un scénario de « collaboratory » qui permettront d'assurer une collaboration optimale ;
- concevoir des outils collaboratifs respectant les contraintes de QoS définies auparavant ;
- intégrer les outils précédemment définis au sein de la plate-forme envisagée afin d'en utiliser les fonctionnalités ;
- analyser et évaluer la performance des solutions implémentées en vue de faire ressortir la meilleure.

Il serait par conséquent intéressant de développer cette partie en suivant la méthodologie déjà existante afin de l'implanter dans le projet et de lui conférer un rôle prépondérant

1.4 Plan du mémoire

À la suite de ce chapitre d'introduction, ce mémoire comprend quatre autres chapitres. Au Chapitre 2, nous présenterons une synthèse des laboratoires virtuels et des

questions de collaboration et Qualité de Service. Il est en effet assez intéressant de voir à quel niveau sont arrivés les laboratoires précédemment développés par différentes unités de recherche à travers le monde, et notamment faire l'inventaire de ce qui a été fait au niveau collaboration pour tenter de l'inclure dans ces plates-formes. Le problème de la Qualité de Service est introduit plus en détails dans le Chapitre 3 où l'on en définit les éléments pertinents. Dans ce chapitre, nous nous efforcerons de caractériser la QoS et notamment dans le cadre d'un environnement virtuel d'apprentissage. Ceci nous permet donc de définir une architecture et de la présenter en détaillant chacun de ses composants. Dans le Chapitre 4, nous concentrerons nos efforts sur deux sous-ensembles particulier, à savoir *l'Interface de coopération* élément central de notre architecture, et le *QoS manager* qui seront implémentés et dont les analyses de performances seront réalisées. Enfin, la conclusion fait une synthèse du travail qui a été accompli dans le cadre de ce projet de maîtrise, en mettant en évidence les principales contributions et suggestions de travaux futurs dans la perspective d'amélioration.

CHAPITRE II

COLLABORATION DANS LES ENVIRONNEMENTS D'APPRENTISSAGE ÉLECTRONIQUE

L'apprentissage électronique ou e-learning (electronic learning) a bénéficié de la multiplication des outils offerts par les nouvelles technologies de l'information et de la communication. L'avènement d'Internet a permis d'étendre ce phénomène. En effet, la mise à disposition d'outils relativement puissants et modernes peut maintenant se faire sans aucune restriction d'ordre géographique ou temporelle et bien souvent à moindre coût. De cette manière, le multimédia prend une place de choix dans les opportunités à offrir aux personnes intéressées par un tel mode d'enseignement. Avec les progrès réalisés dans ce domaine cette tendance devrait se préciser. Dans ce chapitre, nous présentons les concepts de base du paradigme du télé-apprentissage puis évaluons succinctement les avantages et les actuelles limitations. Ensuite, nous examinons la collaboration comme moyen de travailler dans des laboratoires virtuels. Enfin, nous analysons les aspects de la Qualité de Service (QoS) dans ces environnements.

2.1 Apprentissage électronique

Afin de bien comprendre la place qu'occupe le télé-apprentissage, voyons ce qui est déjà développé dans ce domaine. En effet, cette direction de recherche est déjà largement explorée. Nous allons donc tenter, dans cette section, d'en donner une définition pertinente le situant par rapport à l'enseignement traditionnel.

2.1.1 Définition et caractérisation

Afin de caractériser précisément cette forme d'apprentissage, il est important de commencer par la définir. En effet, nous devons voir en quoi elle rejoint l'enseignement

traditionnel et en quoi elle diffère. De cette manière nous pourrions obtenir une évaluation plus précise de ses besoins

L'apprentissage électronique se distingue bien évidemment en premier lieu de l'apprentissage traditionnel comme on l'entend [4]. En effet, c'est une évolution de ce que tout le monde a connu jusqu'à présent ou l'on se rendait à l'école pour apprendre. Et peu importe toutes les possibilités offertes par les cours dispensés dans les écoles, facultés et autres, la localisation géographique était l'une des contraintes majeures de cette situation. Ainsi, la qualité des cours qu'il nous était offert dépendait surtout de notre localisation plutôt que de nos envies réelles ou de nos capacités. La première génération de l'apprentissage à distance a tenté d'abolir cette contrainte. Cette première avancée était illustrée par un apprentissage qui utilisait les communications pour étendre, dans la mesure des possibilités de l'époque, les perspectives d'objectifs envisageables. Cette notion traditionnelle d'apprentissage à distance faisait surtout référence aux méthodes utilisant la vidéo et la transmission du savoir par ce média uniquement.

Avec l'avènement des nouvelles technologies et notamment de l'Internet, ce concept a trouvé un deuxième souffle. En effet, on a pu y rajouter l'interactivité, le multimédia, les examens sur demandes, les modules d'éducation programmés personnellement [5], etc. En fait, l'introduction de ces technologies a permis une plus grande automatisation du processus d'apprentissage. C'est donc une deuxième évolution que l'on a pu observer avec l'utilisation de ces outils. Cette piste est d'ailleurs très largement suivie pour la recherche dans ce domaine. En fait, ceci résulte du désir de rendre les élèves plus autonomes. Cependant, il ne faut pas se tromper, le terme automatisation fait référence à la systématisation et ce, au niveau de la mise en place des outils pédagogiques. Contrairement à ce que ce terme pourrait faire croire, c'est surtout une manière de rendre plus « humaine », plus personnelle la notion d'éducation. En effet, l'étudiant désireux de découvrir ou d'approfondir ses connaissances dans un domaine spécifique va se voir proposer un ensemble de cours et d'activités en rapport avec celui-ci, ayant des liens plus ou moins directs avec ses besoins. Il lui est aussi offert la possibilité de choisir de travailler uniquement les parties qu'il juge pertinentes à la

vue de ses acquis, de revenir dessus, d'insister ou même, le cas échéant, de ne pas les traiter s'il n'en ressent pas le besoin.

De telles capacités ont poussé les universités à développer cette nouvelle avenue d'éducation, que ce soit en parallèle avec ce qui se faisait déjà ou totalement à part entière. Certains auteurs ont même introduit le concept de *Cyberiversity* [5] pour tenter de décrire ce phénomène.

L'apprentissage peut être vu sous l'aspect scénario [6], c'est-à-dire en distinguant l'organisation globale de l'enseignement offert, où plutôt en se concentrant sur les outils mis à disposition [7]. En se basant sur les prérogatives imposées aux élèves, on peut distinguer trois modèles de télé-apprentissage [6] :

- le modèle d'apprentissage indépendant du temps ;
- le modèle d'apprentissage distribué simultanément ;
- le modèle d'étude indépendant.

Dans le premier, les activités traditionnelles habituellement suivies en classes se combinent avec des études à la maison ou au travail et, à partir de là, un dialogue perpétuel s'installe entre le formateur et les apprenants, et les apprenants entre eux. Dans le second modèle, les apprenants peuvent voir et entendre leur formateur et peuvent donc à tout moment établir le dialogue avec celui-ci. Typiquement, ce modèle utilise satellite, télévision par câble ou téléphonie vidéo et maintenant tous les outils offerts par l'informatique, notamment l'Internet et le travail en réseau. Enfin, dans le dernier modèle l'apprenant travaille seul avec le sujet d'étude dans une séquence de modules organisés. On utilise alors souvent des CD-ROM. On doit cependant ajouter à cette façon de travailler des séances régulières avec un conseiller, ce qui représente un pas en arrière puisqu'on réintroduit la contrainte géographique. Toutefois, quelle que soit la façon envisagée, il reste des composantes inévitables. Dans tous les modes de télé-apprentissage, une dimension critique est la part donnée au contact qui est permis entre les étudiants et d'autres personnes (le formateur, l'équipe d'instruction, le tuteur, d'autres apprenants) dans le cours. Ceci est vrai d'ailleurs, quelle que soit la forme

d'instruction considérée. Par exemple, dans une situation de face à face, le formateur peut permettre aux étudiants de poser un grand nombre de questions et stimuler considérablement l'interaction dans la classe, alors que d'autres ne le feront pas du tout. Cependant, dans l'enseignement traditionnel, ces périodes d'échanges sont limitées au seul temps des cours (à moins bien sûr que le professeur se rende disponible aux heures non prévues à cet effet). Dans le cas de l'apprentissage électronique par contre, l'opportunité pour des contacts directs s'en trouve considérablement augmentée grâce notamment à l'utilisation des télécommunications. Cependant, ceci peut amener à un énorme flot de messages presque ingérable pour le professeur, au vu de l'investissement que cela nécessiterait pour lui (d'où la nécessité de bien appréhender le nombre de participants et les échanges à considérer).

2.1.2 Complémentarité et améliorations

La principale avancée que l'apprentissage à distance procure par rapport à l'approche traditionnelle, est de repousser les frontières de distance et les limitations sur la taille du public cible choisi. En d'autres termes, la présence physique de l'étudiant dans une pièce particulière n'est plus requise. Mais au niveau pédagogique, cette nouvelle approche n'apporte en fait rien de vraiment nouveau. L'apprentissage électronique au contraire diffère grandement de cette voie basée uniquement sur la télévision. Voyons quelques points intéressants à ce propos et notamment les principales différences :

- L'interactivité due au multimédia: l'étudiant contrôle, avec les outils mis à sa disposition (on pensera notamment à la souris et à ses clicks si puissants...) le flot d'information et de données qu'il va rencontrer. Ainsi, chaque étudiant peut réaliser des lectures différentes, et ce, à différents moments ou lieux. Ceci s'oppose à la méthode traditionnelle dans laquelle tous les étudiants doivent travailler à une heure déterminée et parfois même dans un lieu spécial si les contraintes de réception l'obligent.

- L'absence de contrainte de rythme : Les élèves fastidieux et long dans leur apprentissage peuvent utiliser le contrôle qu'ils ont sur la rapidité du flot d'informations pour ralentir le rythme si tel est leur besoin. Et pareillement, des élèves capables d'apprendre beaucoup plus vite ne sont plus pénalisés par une certaine lenteur qui pourrait conduire à les ennuyer.
- La modularité : En effet, la méthode d'apprentissage devient adaptative avec ces possibilités. Chaque étudiant peut choisir les points importants pour lui, sans se soucier des parties qu'il aurait préalablement déjà intégrées dans une formation antérieure.

En plus des institutions préexistantes pour l'éducation à distance tel que les « Open Universities », beaucoup d'universités traditionnelles s'intéressent de plus en plus à l'apprentissage à distance et notamment le télé-apprentissage. D'abord, parce que les universités ont toujours besoin de plus en plus de publics pour suivre leur cours et cette manière est une bonne façon d'élargir à l'infini leurs possibilités de recrutement. Les universités s'orientent alors vers un mode dual d'Enseignement. Ce point est non négligeable au niveau des administrations des universités. En effet, force est de constater que cela représente un important bénéfice par rapport à ce qui se faisait auparavant. Mais il serait exagéré et erroné de restreindre la motivation des universités à ce seul point. En effet, l'hypothèse implicite justifiant ces changements est réellement plus noble et a une explication pédagogique. Il ressort à priori que le développement de ces nouvelles techniques d'apprentissage et l'ajout du télé-apprentissage à l'enseignement traditionnel va rendre les activités des universités plus efficaces et productives. Mais, est-ce vraiment le cas ? Que faut-il faire pour le rendre comme tel ?

En tout cas, il est important que ces modifications apportent de la valeur ajoutée à la situation existante. On conjecture alors que la qualité et donc le rendement des cours en seront fortement améliorés. Pour regarder si le cours est vraiment amélioré, il faut prendre en compte un ensemble de paramètres plus ou moins objectifs. Cela va des résultats (en terme de grade par exemple) à la satisfaction personnelle des étudiants. Par

ailleurs, un paramètre souvent négligé, le temps que l'étudiant doit investir dans l'apprentissage, est relativement intéressant à prendre en compte. La recherche a montré que l'apprentissage basé sur l'utilisation de l'informatique motive fortement les étudiants à passer plus de temps à l'étude de ses cours, et notamment au traitement de certaines tâches anciennement un peu rébarbatives ou à l'exécution de travaux rendus plus intéressants [7] . Mais, ce n'est pas là le seul profit que l'élève perçoit au niveau de son emploi du temps. Il est aisé de comprendre tout le confort que peut ressentir un étudiant qui peut travailler à son propre rythme. Il est donc évident qu'une telle solution est un progrès non négligeable quand il peut tout réaliser à sa manière avec en plus la possibilité de créer de multiples connexions lui permettant d'établir le dialogues avec les autres personnes confrontées aux mêmes problèmes que lui (grâce aux mails, tableaux noirs et vidéo conférence). Les études de satisfaction relevées dans [8] vont même jusqu'à parler « d'enthousiasme » constaté quand il s'agissait d'effectuer les tâches demandées. Ces recherches insistent aussi sur le gain brut que le recours à l'informatique et à ces nouvelles technologies peut apporter. Il ressort en effet que l'utilisation de tels outils pouvait réduire la nécessité du temps consacré à l'apprentissage de 2/3 par rapport à un apprentissage conventionnel.

En résumé, l'apprentissage électronique dans l'enseignement supérieur doit être utilisé afin d'offrir une valeur ajoutée et une qualité ajoutée à ce qui existe déjà, et notamment en optimisant l'utilisation du temps de l'étudiant. Mais ce n'est pas là les seules améliorations que cette méthode d'apprentissage peut apporter.

L'étudiant se retrouve en effet confronté à des responsabilités accrues. Ainsi, à moins de considérer des séances de laboratoire à heures fixes et prédéfinies, l'apprenant est livré à lui-même, il choisit de travailler quand il veut. Cette possibilité peut apparaître très séduisante au premier abord, et notamment pour les premiers concernés elle implique aussi une rigueur et une discipline plus importantes.

Grâce au recours à ces technologies, on constate que les apprenants peuvent manipuler des outils de plus en plus efficaces et en rapport avec ce que l'on trouve après

dans le monde du travail. En effet, les environnements concernés sont souvent équipés des dernières avancées technologiques. Mais cette caractéristique est tout simplement liée à la définition intrinsèque du télé-apprentissage et au pourquoi de son introduction. Cependant, ce n'est pas le seul élément intéressant justifiant ces technologies en tant que tel. En effet, de par leur nature évolutive, ces outils permettent une grande adaptabilité. Si nécessaire, l'instructeur peut décider d'apporter des modifications relativement facilement. Face à la lourdeur des démarches à mettre habituellement en jeu pour pouvoir ne serait ce qu'apporter une toute petite amélioration, la possibilité offerte de tout modifier d'un seul coup est très séduisante. Il est en effet possible de modifier rapidement l'orientation des cours et activités si elle devient un peu obsolète à un moment donné. Voilà donc autant de points qui justifient la prolifération de laboratoires dispensant des cours virtuels et l'utilisation de l'apprentissage électronique.

2.2 Laboratoires virtuels et télé-laboratoires

L'apprentissage a trouvé un souffle nouveau grâce à Internet. Les communications entre formateurs et enseignants en sont grandement facilitées, et cela est très sensible au niveau universitaire avec l'utilisation tellement répandue des e-mails. Mais, le point le plus marquant est sûrement la possibilité de s'affranchir de considérations, aussi bien temporelles que géographiques, que confère ce puissant média. D'ailleurs, nous nous sommes vite aperçus que nombres d'universités utilisent l'informatique à des fins éducationnelles poussées. Ce phénomène est d'autant plus vrai dans le domaine scientifique où la part expérimentale est primordiale. Les apprenants retirent beaucoup de bénéfices de ce mode d'enseignement. Les universités qui axent leur développement sur ce phénomène l'ont bien compris, puisque l'on a vu apparaître énormément de laboratoires de simulations disponibles à partir d'Internet. À ce propos, nous verrons dans les sous-sections suivantes deux exemples de télé-laboratoires qui illustrent bien nos propos.

2.2.1 Laboratoires virtuels basés sur la simulation interactive

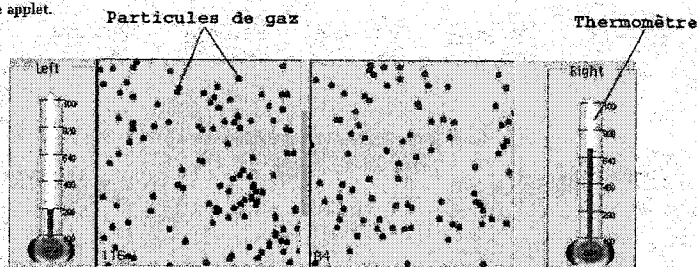
Cette forme de laboratoire est la plus répandue sur le web. Leur nombre s'est particulièrement accru avec l'apparition de la technologie java. Ils consistent en général en une page web intégrant un ou plusieurs applets java. Ces petites applications confèrent à ce type de laboratoire, leur caractère virtuel. En effet, c'est là que se trouvent les modules interactifs permettant de faire les simulations ou les tests décrits dans le scénario d'apprentissage proposé dans ces laboratoires. Afin d'illustrer ces mots, nous avons choisi de présenter un laboratoire de thermodynamique portant sur l'étude des gaz et de leur équilibre. Ce laboratoire a été développé au département de sciences de l'université d'Oregon. Il permet de vérifier les propriétés thermodynamiques de ces gaz selon les conditions de température. Il consiste notamment à contrôler l'état d'excitation des particules de gaz selon la température de l'environnement dans lequel ils évoluent. La Figure 2.1 illustre l'interface de l'applet java simulant les deux chambres dans lesquelles sont appliquées les conditions particulières de température. En jouant sur les paramètres, comme cela est illustré à la Figure 2.2, on pourra voir les différents comportements spécifiques. En effet, dans cette deuxième partie de la page Internet du laboratoire, il est proposé à l'apprenant d'observer différentes comparaisons pour chaque situations. Malheureusement, ces essais sont limités car l'interactivité se limite aux choix proposés par les concepteurs de ce laboratoire. Si ces environnements sont assez ludiques et plaisants, ils manquent pourtant de consistance. En effet, de par leur nature même, la simulation encadrée, ils sont assez limitatifs et n'ont pas un intérêt pendant très longtemps. Cependant, nous les retiendrons car ils sont représentatifs des laboratoires que nous voulons montrer et ils constituent une très bonne base pour des travaux futurs.

En parcourant le web, nous nous apercevons que ceux ne sont pourtant pas les seuls présent, et que beaucoup d'efforts sont réalisés pour augmenter l'interactivité des laboratoires disponibles comme cela est illustré aux figures 2.1 et 2.2.

THERMODYNAMIC EQUILIBRIUM

Important Notes:

- The way that caching seems to work in Netscape or Internet Explorer (IE) is that when you leave one thermodynamics experiment to go to another, the other one is still "running" in the cache. That means that over time the experiment will run slower on your machine. We therefore strongly recommend that you bookmark this page and then restart either Netscape or IE to do more of the experiments. We realize this is inconvenient but there is little we can do about it. If your machine speed is less than 150 MHz you will almost certainly have to do this.
- Hitting the reload button while the applet is running will generally cause aberrant behavior. Don't hit the reload button.
- As with any JAVA applet, it's a good idea to flush your cache periodically.
- We have replaced the multi-colored balls with black dots which makes the applet much less CPU intensive. It should run well on most platforms now under Netscape 3.0 and higher and IE 4.0
- A screen resolution of 800x600 is needed
- Here is a snapshot of the applet.



Read the instructions and then go to the links at the bottom of this page to activate various kinds of experiments that can be done with this applet.

Figure 2.1 Interface du laboratoire de thermodynamique

Experimental Instructions

This applet is designed to simulate the diffusion process which occurs when gases of different temperatures are mixed. To activate the mixing click one time on the red vertical bar that separates the two chambers. After a few moments, the bar will turn green and the gases will start to mix and share their energy. The counters in the respective chambers indicate the number of particles in that chamber. The thermometers will start to change temperature as the mixing process is occurring. Eventually both chambers will reach the same temperature.

In order for this simulation to properly run - its important that the user not be doing other activities on the machine as this will interfere with the timing aspects of the experiment.

Different initial conditions are setup in a series of experiments and questions related to that experiment can be found underneath the apparatus.

 [Do the First Experiment](#)

 [Do the Second Experiment](#)

 [Do the Third Experiment](#)

 [Do the Fourth Experiment](#)

 [Do the Fifth Experiment](#)

 [Do the Sixth Experiment](#)

 [Do the Seventh Experiment](#)

Figure 2.2 Interface du laboratoire de thermodynamique (suite)

2.2.2 Laboratoires virtuels élaborés et interactifs

Contrairement aux premiers types de laboratoires que nous avons présenté, les laboratoires virtuels dans un environnement distribué offrant notamment la collaboration ne sont pas légion et sont bien souvent encore à l'état de prototype. Ainsi, si les

interfaces sont beaucoup plus évoluées que dans la sous-section précédente, nous pouvons pourtant regretter que la collaboration n'y soit pas totalement implantée. C'est dans cet esprit que nous présentons le laboratoire National E.O Lawrence de Berkeley en Californie. En effet, en se rendant sur le site de l'Université on a accès au laboratoire afin d'y conduire les expériences normalement prévues à partir du site de l'Université, avec possibilité de collaboration à distance, grâce notamment au partage de ressources. Cependant, afin d'assurer un rendu maximal de ces expériences les responsables ont identifié certains éléments essentiels : contrôle et surveillance à distance, téléprésence, cahier de laboratoire en ligne, plate-forme centrale compatible, accès à partir de sites repartis. Dans ces points nous retiendrons particulièrement ceux de téléprésence et cahier de laboratoire. En effet, c'est à partir de ces deux concepts que la collaboration peut être mise en place. Cet exemple permet de mettre en exergue les efforts réalisés pour s'assurer de reproduire fidèlement la réalité tout en maintenant en plus les composantes de ces laboratoires.

Nous pouvons d'ailleurs aussi citer le laboratoire de physique de Télé-université (Montréal, Québec) qui est un environnement d'apprentissage très évolué offrant un grand nombre d'expériences relativement réalistes. Ce télé-laboratoire intègre les fonctionnalités de télé-expérimentation et de télé-mesure en réseau. La Figure 2.3 illustre le concept d'un télé-laboratoire. On peut voir comment, à travers une plate-forme de télécommunications appropriée, dotée d'un système de gestion dédié et d'un certain nombre d'interfaces de logiciels et de matériels, il est possible à des utilisateurs géographiquement dispersés, d'accéder à des ressources disponibles dans plusieurs sites distants. Il est donc ainsi possible de miser sur l'accessibilité des ressources matérielles et humaines distribuées pour réaliser des expérimentations autrement irréalisables dans un contexte local. Le premier rôle joué par chacune des composantes d'un environnement virtuel de d'apprentissage, de formation et de recherche utilisant un réseau est celui de la communication entre les participants (apprenants, formateurs et chercheurs) d'une part, et entre les apprenants et l'objet de la formation à travers le réseau informatique d'autre part. Par exemple, la tablette graphique peut permettre à

l'apprenant, au formateur ou au chercheur d'écrire ou de dessiner naturellement. De la même manière, il sera possible d'accéder au montage électronique du site " A " à partir du poste de visioconférence " B " ; dans ce cas, l'oscilloscope réel connecté au montage du site " A " est considéré comme virtuel au site " B ".

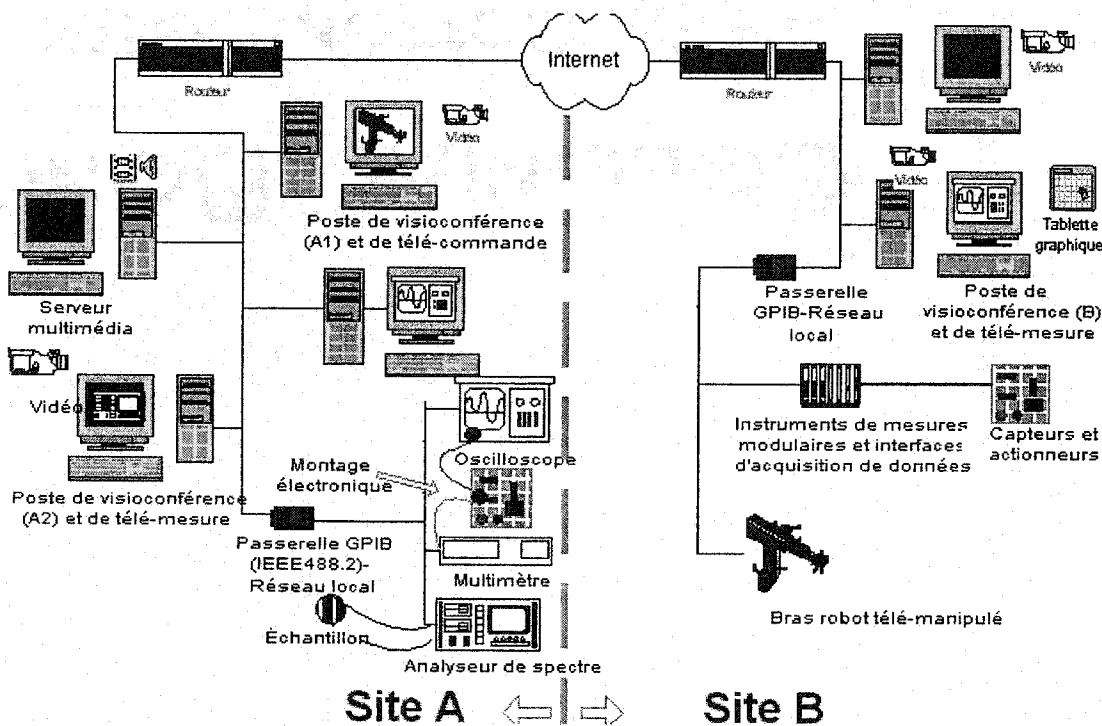


Figure 2.3 Le concept de laboratoire virtuel de génie électrique en réseau

Dans cette Figure 2.3, nous pouvons observer le rôle et la place de tous les acteurs du laboratoire. Cet exemple montre comment un apprenant jouant le rôle de "contrôleur" peut télécharger une application du site Web, l'installer sur son propre ordinateur et l'exécuter. L'apprenant (*contrôleur*) sera alors immédiatement connecté au serveur d'application. Ainsi, il peut participer à une séance de travail pratique interactive synchrone. Toutes ses actions peuvent être observées par les autres membres de l'équipe (*observateurs*) pendant toute la durée de la réalisation de la tâche spécifique. Ces *observateurs* doivent au préalable, eux aussi, tout comme l'apprenant *contrôleur*,

télécharger l'application correspondante. Notons que dans ce cas, le formateur peut être *observateur* et il peut interagir, comme chacun des apprenants pourraient le faire, avec tous les membres de l'équipe de réalisation du travail pratique à l'aide de moyens de communication appropriés. Si ces fonctionnalités sont séduisantes elles n'en sont pas moins limitées. En effet, dans les exemples proposés sur le site [9] les manipulations se résument pour l'instant à des manipulations simplistes sur les opérateurs logiques pour lesquelles les communications sont très limitées.

2.3 Modélisation de la collaboration

Puisque nous avons vu que le télé-apprentissage vise une reconstitution de l'apprentissage traditionnel en utilisant les technologies actuelles, il est normal que tout ce qui compose ce dernier s'y retrouve. Et puisque tout le monde s'accorde à dire que le travail en commun est bien plus bénéfique que celui réalisé seul, nous nous attendons donc à retrouver ce phénomène dans l'apprentissage électronique. C'est pour cela que nous avons décidé de nous pencher sur la collaboration et la place qu'elle pourrait prendre dans cette forme d'apprentissage.

2.3.1 Caractérisation de la collaboration

La collaboration sur le Web est un domaine de recherche très large [10], incluant un grand nombre de possibilités comme la transmission des connaissances, la représentation du savoir, la figuration par objets, la création d'objets et toute autre alternative qui permet le partage d'information ou la fabrication de systèmes partagés. La distinction entre les modes synchrones et asynchrones est quelque peu arbitraire dans le sens où il n'y a pas de distinction absolue et précise entre ces deux extrêmes. Dans certains cas, les participants reçoivent des « réponses » de ses interlocuteurs très rapidement. Dans les autres cas, les réponses peuvent être considérablement retardés (on parle d'heures ou de jours). Actuellement, la plupart des systèmes actifs sont pourtant asynchrones, cela signifiant que les retours ne sont pas immédiats. La raison est simple.

En effet, les clients parlent avec des serveurs qui sont passifs, donc les clients ne peuvent pas utiliser les serveurs pour parler directement avec d'autres clients. La collaboration synchrone (proche du temps réel) commence à être développée un peu plus efficacement en dépit des difficultés rencontrées jusqu'à présent. Dans cette section nous nous attacherons seulement à décrire ce qu'est la collaboration, et la forme qu'elle prend sur Internet et les réseaux. Commençons donc par identifier certains domaines dans lesquels la collaboration peut s'avérer utile :

- système de conférences ;
- organisation d'informations ;
- système de vote ;
- création collaborative ;
- marchés virtuels ;
- enseignement.

Établissons à présent à quels niveaux il est important de considérer celle-ci. En effet, de nombreux paramètres sont à prendre en compte avant de pouvoir l'utiliser efficacement. Il faut donc bien cibler les besoins de l'outil que l'on veut mettre en place:

- synchrone ou asynchrone ;
- même (ou différent) heure ou même (ou différent) lieux ;
- active ou passive (qui initie et qui attend ?) ;
- unidirectionnelle, bidirectionnelle ou multidirectionnelle ;
- distribuée ou centralisée ;
- persistante ou éphémère ;
- privée, groupe ou publique ;
- simple lecture, lecture/écriture ou simple écriture.

Le laboratoire sur lequel nous travaillons est un Environnement Collaboratif Virtuel (CVE). La technologie des CVEs tente de transformer les réseaux informatiques

en espace de travail à l'intérieur duquel les participants ont la possibilité d'interagir socialement. L'utilisation d'avatars permet de les représenter et ces avatars peuvent communiquer avec le monde virtuel ou les autres utilisateurs connectés par l'utilisation de différents médias comme l'audio, la vidéo, les supports graphiques ou les textes [11]. Les CVEs sont le fruit de travaux sur la réalité virtuelle et les communautés de CSCW (Computer-Supported Cooperative Work) qui convergent vers des intérêts communs. Dans ces communautés, les CVEs représentent l'extension naturelle des technologies de réalité virtuelle pour un seul utilisateur que l'on a élargi à des groupes de personnes. En effet, les participants sont bien souvent dispersés physiquement et communiquent simplement à l'aide de réseaux informatiques. Ils permettent ainsi de recréer en partie les relations sociales qui sont souvent négligées avec de telles technologies.

Il doit être possible de déployer à grande échelle ces CVEs pour tirer partie au maximum des supports en temps réel et les faire partager à un plus grand nombre. Ces environnements sont destinés à des groupes entiers, et la communication entre eux peut se baser sur plusieurs formes d'architectures :

- Client/Serveur : Chaque application communique seulement avec le serveur qui est responsable de les retransmettre aux autres ;
- Unicast « peer-to-peer »: Chaque programme client individuel envoie ses informations directement à un autre programme client ;
- Multicast « peer-to-peer » : Similaire au précédent sauf que l'information est envoyée à tout le groupe plutôt qu'à un seul individu.

Enfin, il faut s'assurer que dans un CVE chaque participant voit au même moment le même contenu à l'écran que ses collaborateurs.

2.3.2 Analyse des outils

Il y a beaucoup d'aspects envisageables pour la télécollaboration. Par exemple, les e-mails, mailing-lists, et news-groupes peuvent être considérés comme des outils de

collaboration. En effet, l'un des acteurs soumet son document à ses collaborateurs qui se chargent d'apporter des modifications et de lui transmettre (à lui ou à tous) la nouvelle version. Dans les faits, cette manière de procéder a été très souvent utilisée avec succès pour développer la majorité des standards et protocoles de l'Internet, et elle est d'ailleurs toujours largement utilisée. Cependant, ces interactions ne se font pas en temps réel et la collaboration qui en résulte est clairement inférieure à une rencontre face à face. De cette manière on n'exploite pas toutes les possibilités offertes. Pourtant la technologie au niveau informatique et notamment en ce qui concerne les réseaux et l'Internet en est maintenant à un point où la collaboration en temps réel est possible. Cette collaboration peut prendre la forme d'applications partagées, comme les « tableaux blancs » (pour lesquels on peut partager des textes, des images et des schémas) ou la forme de communications directes grâce à une combinaison de l'audio, de la vidéo et du texte.

Dans cette section nous nous efforcerons donc de présenter les outils déjà mis en place par des entreprises pour répondre aux exigences de la collaboration. Et si nous ne prétendons pas être exhaustifs, le but est de se faire une bonne idée des qualités et limitations de chacun [11] :

- **Conférence (Netscape):** un outil de collaboration avec audio, tableau blanc, transfert de fichier, chat, et moteur de recherche Web collaboratif. Malheureusement, il n'y a pas de vidéo et de véritable application partagée. Sa force est d'être compatible avec une grande majorité des systèmes d'exploitation. Sa principale faiblesse est qu'il se limite à une interaction à deux personnes au plus.
- **CU-SeeMe (white Pine):** Plate-forme surtout célèbre pour ses compétences au niveau vidéoconférence, CU-SeeMe offre aussi des outils de collaboration incluant un tableau blanc, un support de transfert de fichiers et un espace de discussion. A l'opposé de Conference ou Netmeeting, il supporte les conférences multi-utilisateurs. Cependant, à l'instar du précédent, il n'offre pas d'applications partagées en temps réel.

- **Habanaro (NCSA):** Habanaro est un outil de collaboration basé sur Java et qui fournit une structure pour partager des objets Java avec des collaborateurs repartis et ce, sur Internet. Il vient avec un tableau blanc, un éditeur de texte, un chat et un moteur de recherche collaboratif. Malheureusement, il ne supporte pas vraiment la vidéoconférence.
- **ICP (Vocaltec) :** Célèbre pour son application de partage de documents. En effet, les changements réalisés apparaissent presque instantanément sur les autres écrans considérés. ICP vient avec 6 modules : tableaux blancs, écran de chat multi-utilisateurs, transfert de fichier, audio, moteur de recherche collaboratif et partage d'application. La seule faiblesse est qu'un utilisateur qui veut éditer un document particulier doit absolument avoir l'application appropriée.
- **NetMeeting (Microsoft) :** NetMeeting est un outil très puissant et hautement recommandé en matière de collaboration. Il se distingue de la plupart des autres applications par sa capacité à permettre aux utilisateurs le partage d'applications. Ces outils sont peut-être les seuls gratuits qui fournissent audio, vidéo, transfert de fichier, chat, partage de documents et d'applications, et un tableau blanc. Malheureusement, le principal problème de NetMeeting est sans doute sa non fiabilité.
- **Sound IDEAS (Newstar):** Logiciel très efficace en matière de vidéoconférence, utilisé par de nombreuses entreprises comme la NASA. La qualité audio est relativement bonne (juste un peu moins qu'une conversation téléphonique). Tous les utilisateurs peuvent parler avec cet outil. Sound IDEAS donne la priorité aux paquets de données qu'il envoie à travers l'Internet avec la prédominance pour ceux concernant la voix, ce qui permet de garder les conversations ininterrompues.

Voilà donc un échantillon d'outils utilisé ça et là qui supportent la collaboration et se basent même principalement dessus. On pourra aussi citer TVM, PowWow, MBONE ou VoxPhone et sûrement de nombreuses autres. Cependant, s'ils semblent

efficaces pour des tâches particulières, ils ne représentent pourtant pas exactement ce que nous recherchons car ils n'ont pas pour but premier l'enseignement.

2.3.3 Survol des environnements existants

Le premier élément qu'il paraît intéressant de détailler est l'environnement INVITE (Intelligent Distributed Virtual Training Environment project). Le projet INVITE s'insère dans la structure de l'Information Society Technology (IST), Programme de la Commission Européenne. Elle a commencé en février 2000 et se poursuivra pendant presque 3 ans. L'objectif principal de ce projet [12] est de construire une plate-forme synchrone pour le télé-apprentissage capable d'interférer avec des systèmes intelligents de gestion. Pour atteindre ce but, les sous-objectifs suivants sont définis :

- identification et définition des besoins collaboratifs pour l'apprentissage ;
- développement d'un système intégré et distribué incluant notamment des agents intelligents, et avatars de représentation des utilisateurs et activités ;
- évaluation du prototype dans différents contextes d'enseignement ;
- recherche de résultats sur le phénomène social de l'apprentissage dans les environnements virtuels.

Pour la construction de cet environnement, un modèle virtuel sera utilisé. L'interface utilisateur sera mise en valeur par le recours à de la vidéo 2D et l'utilisation de mondes en 3D. Afin d'offrir des services d'enseignement asynchrone, INVITE intègre des systèmes asynchrones et autres supports audio/vidéo. De même, tous les outils de gestion de fichiers et données seront utilisés afin de rendre cette plate-forme la plus efficace possible. Enfin, les interactions entre utilisateurs seront réalisées grâce à des chats écrits ou parlés et des traductions en ligne. Ces utilisateurs sont pris en charge par des agents mobiles intelligents permettant des recherches personnalisées et

l'organisation des informations personnelles. On détaille l'organisation des composants principaux de cet environnement collaboratif virtuel dans la Figure 2.4 [12] :

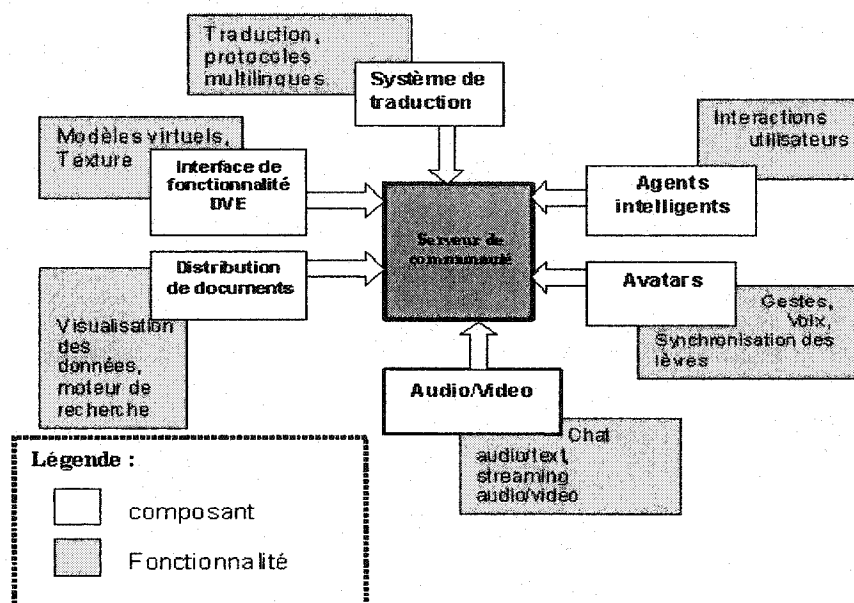


Figure 2.4 Composants de base d'un environnement d'apprentissage

De nos jours les agents intelligents sont utilisés pour de nombreuses tâches. Dans le projet INVITE, ils sont utilisés pour un grand nombre de fonctions comme la gestion des avatars, prise en main des objets, communications mobiles et collaboration. Mais, ils seront aussi utilisés comme « guides », gardes pour les recherches ou « reminders ». Le langage utilisé est XML (plutôt que HTML de temps en temps) car il gère les textes multilingues pour ce qui touche à leur traduction et leur traitement. On a recours au KQML afin d'assurer la communication entre agents (sous JAVA). Enfin, les modes Multi-Utilisateurs sont possibles grâce à l'utilisation de VIP et ISTP (applications en temps réel). INVITE a donc pour principal objectif de développer un environnement d'apprentissage à distance collaboratif. Si le projet est concluant, il devrait s'imposer grâce à ses fonctionnalités et sa technologie. Les étudiants doivent avoir l'opportunité de

participer en temps réel à toutes les activités et ce, d'une manière la plus active possible, les agents étant utilisés pour faciliter cela.

Un deuxième environnement semble aussi intéressant à la vue de toutes les possibilités offertes. C'est celui que l'on retrouve dans [13] qui traite de la plate-forme développée à partir du langage Java. Le constat de départ ayant poussé à développer JETS a été assez simple. En utilisant des moteurs de recherches sur le Web, les utilisateurs peuvent maintenant lancer des applications à partir du réseau. En envisageant les choses du point de vue informatique en réseau, la prochaine étape des environnements collaboratifs serait le partage en temps réel de ces applications. Afin de réaliser cela, les concepteurs de JETS ont développé ce système générique, multi-utilisateurs et collaboratif qui permet le partage en temps réel d'applets Java sur Internet. Voyons tout d'abord les besoins essentiels d'une application multi-utilisateurs, besoins auxquels JETS doit être en mesure de répondre. Pour assurer que ce système est pleinement collaboratif, et notamment en rendant possible l'utilisation du multimédia, il y a au moins trois recommandations à suivre:

- consistance des données ;
- contrôle d'accès aux applications ;
- transfert fiable de données et messages.

Les applets java

Le recours à ce langage informatique et notamment à son mode particulier de programmation est relativement facile à expliquer. En effet, les applets permettent à l'utilisateur de ne pas trop déployer d'effort (l'essentiel du travail d'installation est réalisé par l'applet elle-même qui s'installe seule sur la machine). De plus, la petite taille est un gros avantage et enfin les mesures de sécurité qu'imposent Java sont forts utiles pour un tel environnement. C'est à partir de ces besoins et constatations sur les avantages (et faiblesses) que pouvait conférer Java que l'architecture a pu être mise en place. JETS est un système client/serveur qui est supposé supporter efficacement le « multithread » du fait de l'utilisation de multiples sous-serveurs capables de gérer

chacun une application. Le gain de traitement est ainsi amélioré. Le principe de cette architecture est détaillée à la Figure 2.5.

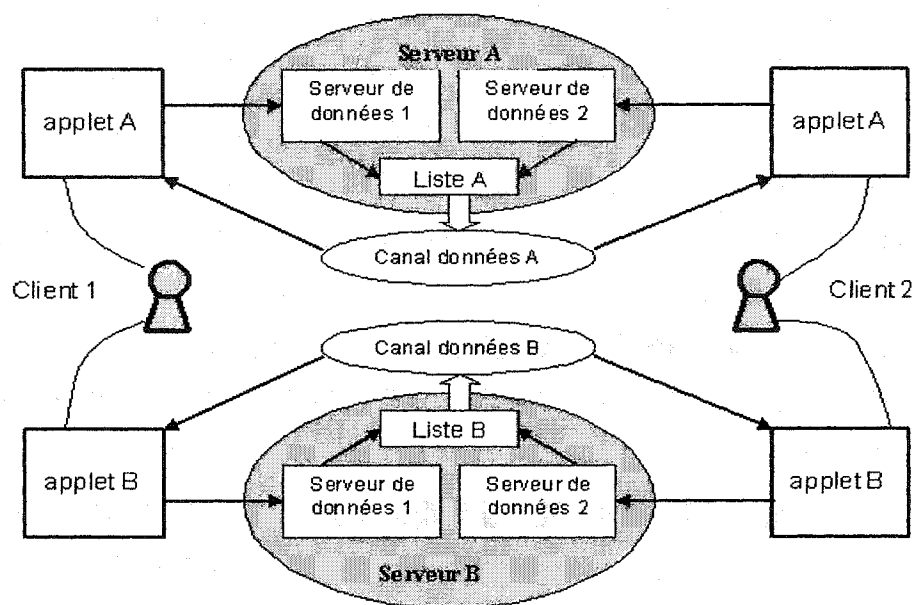


Figure 2.5 Architecture du projet JETS

Si le modèle est séduisant, il reste tout de même quelques points à préciser. En effet, pour obtenir l'inter-connectivité optimale, il reste un certain nombre de problèmes. Comment gérer le manque de qualité de service d'Internet et notamment les pertes de paquets ? L'utilisation de Java, permet-elle de fournir une fiabilité acceptable et une qualité suffisante pour le partage des applications audio et vidéo en temps réel ?

2.3.4 Le concept de « Collaboratory »

Le problème avec les outils utilisés et notamment la collaboration est que leur utilisation n'est pas un phénomène naturel. Il faut bien au départ qu'il y ait un initiateur pour la proposer. Et c'est à cet effet que les CSCL (Computer Supported Collaborative Learning) ont été développés. Mais cette idée n'est pas nouvelle, Dorneich [3] a

introduit en effet le concept de « Collaboratory » que nous tenterons de mettre à profit dans le cadre de cette recherche. Il désigne en premier lieu un environnement virtuel qui utilise les nouvelles technologies comme média de communication pour des utilisateurs qui ne sont pas réunis, du moins au niveau géographique. Bien entendu cette idée n'est pas novatrice puisque c'est justement ce à quoi tous les environnements étudiés tendent à devenir mais il est tout de même remarquable que ce concept tente d'aller plus loin. En effet, la mise en place d'un « Collaboratory » suppose une remise en question totale de la manière d'envisager l'enseignement.

Les laboratoires virtuels sont des supports efficaces puisqu'ils offrent une grande autonomie. L'utilisateur est en mesure de choisir exactement ce dont il a besoin et de prendre le temps qu'il faut (aux moments qu'il aura choisi opportun) pour accomplir son travail. Mais ce n'est pas la seule chose puisqu'il doit avoir la possibilité de progresser en n'étant pas tout seul. Le travail à plusieurs est hautement recommandé puisqu'il offre la possibilité de confronter les idées de chacun et pousse à l'émulation au sein du groupe. De plus, les interactions avec le formateur doivent aussi constituer une option envisageable. Ce dernier devient, dans ce mode d'enseignement et cet environnement, le superviseur capable d'aider les apprenants si le besoin s'en fait sentir; il doit surtout être en mesure de les guider pour leur progression dans les activités proposées dans le laboratoire. Dans [3], l'auteur propose son propre modèle de « Collaboratory », le CLEOS (Collaborative Learning Environment for Operational System) dans lequel les étudiants collaborent pour apprendre théorie et pratique de systèmes opérationnels, le domaine d'étude étant les sciences physiques expérimentales comme la spectroscopie. Ce concept de « Collaboratory » inspirera notre modèle de collaboration

Comme nous venons de l'établir, l'ensemble des outils utiles pour la mise en place d'une plate-forme pleinement collaborative est disponible. En effet, un cahier de travail est déjà disponible, un chat est implémenté, et certaines études nous permettent d'envisager raisonnablement le recours à la vidéo pour compléter cet arsenal. C'est pourquoi nous n'avons donc pas à nous concentrer sur le développement de moyens de communication ; il importe plutôt de voir comment les réunir pour obtenir ce que nous

recherchons vraiment. Ainsi, il apparaît plus intéressant de se focaliser sur le scénario que devra respecter notre laboratoire. Il faut donc s'attacher à faire l'inventaire de toutes les interactions utiles, ainsi que la manière de les agencer pour les utiliser le plus efficacement possible. Cette recherche devant se faire en ne perdant pas de vue que nous serons confrontés à de nombreuses contraintes au niveau de leur mise en application et notamment sur la qualité des communications que nous pourrions obtenir.

2.4 Qualité de Service

Dans les articles déjà évoqués, on a vu apparaître le problème de la Qualité de Service (QoS) dans tous les environnements que l'on a rencontrés. La qualité de services est une préoccupation majeure pour l'Internet et les organisations en réseaux en règle générale. Les environnements collaboratifs ne font en effet pas exception car ils font partie intégrante de cet ensemble et leur importance est grandissante. Mais avant toute spécification propre à ce domaine, il est intéressant de revenir à la définition première de la qualité de service et voir exactement de quoi il s'agit.

2.4.1 La QoS dans les réseaux

Au début des mises en réseau, on avait uniquement le souci de l'acheminement des paquets, mais avec la croissance des réseaux, la globalisation et la stabilité à obtenir, les problèmes se sont multipliés. Dans le cas du « Best Effort » ou aucune QoS constante ne peut être garantie et il n'y a pas de priorité pour certains types de trafic, les problèmes peuvent être durs à gérer. Il faut donc concentrer les efforts sur une définition cohérente et une implémentation efficace.

La Qualité de Service est généralement assimilée à la discrimination des services, autrement dit à la définition de classes différenciées de services. Mais cela signifie aussi garantir un service et pour cela réserver des ressources. Pour implémenter ces classes ou garantir des ressources, il faut définir une ou plusieurs politiques sur les nœuds du réseau permettant d'implémenter la Qualité de Service demandée, en utilisant divers

mécanismes (traffic shaping, contrôle d'admission, gestion de la congestion, etc.). Enfin, pouvoir associer une classe de service et garantir des ressources à un certain trafic implique de pouvoir prédire le comportement du réseau. Pour cela, il est nécessaire de s'entendre sur la définition de critères de mesures ou métriques afin de pouvoir sélectionner le meilleur chemin, lorsque des chemins multiples existent (routage) et vérifier que les demandes en QoS sont satisfaites. En définitive, la QoS englobe tous les mécanismes permettant de différencier les types de trafic, ceux-ci pouvant être classés et administrés différemment à travers le réseau.

Mais, avant de parler de Qualité de Service, il convient de s'interroger sur les critères qui caractérisent le service dans le réseau permettant de délivrer un service de bonne qualité.

Critères de Qualité de Service dans les réseaux

Que ce soit dans les réseaux de télécoms ou l'Internet, il est important de voir exactement les critères servant à évaluer la Qualité de services ! La caractérisation de la QoS dans les réseaux est généralement exprimée par les critères suivants :

- Délai: temps écoulé entre l'envoi d'un paquet par un émetteur et sa réception par le destinataire. Le délai tient compte du délai de propagation le long du chemin et du délai de transmission induit par la mise en file d'attente des paquets dans les systèmes intermédiaires.
- Gigue : variation du délai de bout en bout.
- Bande passante ou débit maximum: taux de transfert maximum pouvant être maintenu entre deux points terminaux.
- Disponibilité : taux moyen d'erreurs d'une liaison.
- Pertes : le nombre de paquets qui ne sont pas retransmis correctement au cours de la transmission.

Mais ceux ne sont pas les seuls car, suivant le domaine considéré, leur nature et leur nombre peuvent être très variables.

Facteurs d'analyse

L'effort essentiel auquel s'attacher en qualité de service est donc de rendre optimaux les critères qu'on décide de prendre en compte quand il s'agit d'évaluer les performances du réseau. Ainsi, les problèmes à résoudre consistent à raccourcir le délai pour rendre l'interactivité la plus confortable possible, réduire la gigue afin d'avoir des flux synchronisés, compenser les pertes de paquets, etc. A cet effet, on peut envisager différentes stratégies pour pallier ces problèmes suivant la façon d'aborder le réseau. Ils sont composés d'un ensemble de routeurs et de liaisons de transmission :

- les liaisons possèdent des caractéristiques de délai, débit maximum et disponibilité ;
- les routeurs peuvent avoir un impact significatif sur le délai, la gigue et la disponibilité.

Du point de vue du transport, la stratégie peut consister à augmenter la bande passante disponible, améliorer les techniques de transport (le support lui-même ou les techniques de multiplexage). En abordant le problème sous l'angle du « bout en bout », il peut être judicieux d'adapter les terminaux au traitement du temps réel ou de contrôler la qualité pas les opérateurs de réseau. Enfin, en se focalisant directement sur la stratégie réseau, on essaye d'améliorer le routage et la communication (processeurs plus puissants ou algorithmes plus performants) ou de définir des règles de priorité. Dans ce contexte, offrir un service de bonne qualité consiste à fournir un service avec un délai et une gigue minimum ainsi qu'une disponibilité et un débit maximum, assurant un service "best-effort" convenable pour tous les types de trafic. La Qualité de Service, quant à elle, peut-être interprétée comme une méthode permettant d'appliquer un traitement préférentiel à un certain type de trafic et d'augmenter ainsi le niveau de qualité des critères caractéristiques de cette catégorie de trafic. Dans cette optique, implémenter la Qualité

de Service via des classes différenciées de service, consiste à implémenter des mécanismes tels que:

- modifier l'algorithme de choix du routeur suivant, pour choisir un chemin conforme au niveau de service souhaité ;
- modifier les algorithmes de mise en file d'attente et de rejet des paquets pour garantir les ressources ;
- modifier les caractéristiques du flux de trafic à l'entrée dans le réseau pour le rendre conforme au profil négocié.

Mesure de la Qualité de Service

Compte tenu des différentes méthodes pour implémenter la Qualité de Service à travers le réseau, il est important de pouvoir mesurer l'efficacité de celles-ci. Cela revient à mesurer les performances du réseau. Le comportement du réseau est basé sur une interaction complexe entre le trafic obéissant aux règles du protocole TCP et des routeurs possédant des fonctionnalités élaborées de gestion de la QoS, rendant la tâche extrêmement complexe. La compréhension de ce comportement et de l'influence des divers mécanismes de la QoS est essentielle pour le développement d'outils fiables. Les critères généralement utilisés tels que la disponibilité, le délai, la gigue et le taux de pertes de paquets, sont difficiles à mesurer du fait du caractère aléatoire des pertes de paquets et du temps indéterminé passé dans les files d'attente sur tous les routeurs qui constituent le chemin entre une source et une destination. Pour établir une vérification de la QoS ou pouvoir comparer les QoS fournies par différents réseaux, il faut pouvoir comparer ces observations.

2.4.2 La QoS pour un environnement collaboratif

Nous nous intéresserons dans cette section, au problème posé par le niveau de contrôle requis pour la Qualité de Service pour des applications ayant rapport avec un environnement collaboratif. Un système de collaboration possède toujours un espace

partagé qui est composé d'une ou plusieurs fenêtres en fonction du niveau de collaboration souhaité. Chaque fenêtre est ainsi le support pour les applications audio, vidéo ou de données qui seront utilisées pour coopérer par les utilisateurs et assurer une communication efficace entre eux. Dans des projets comme UARC (Upper Atmospheric Research Colloratory) ou le "Medical Collaboratory Project" [14], les protagonistes ont besoin de partager des données et des applications alors qu'ils ne se trouvent pas du tout au même endroit ni au même moment. Les flots correspondants aux médias vont donc transiter sur le réseau. Ils sont donc sujets au délai, aux pertes de paquets, etc. Ainsi, les communications peuvent être caractérisées par une certaine qualité. Pour cela, il est assez judicieux de faire appel aux paramètres suivants :

- **Latence** : Pour une collaboration correcte, des actions comme la discussion entre participants ou la modification dans l'espace de travail partagé doivent pouvoir être assurés avec une latence suffisamment faible ;
- **Gigue** : Les flots audio et vidéo doivent être transmis avec une faible gigue afin d'assurer une qualité continue ;
- **Perte de paquets** : Typiquement, les flots audio et vidéo peuvent supporter une certaine perte au niveau des paquets transmis. Mais des données ne peuvent supporter les mêmes pertes ;
- **Asynchronisme** : Pour une communication efficace, la fenêtre de transfert de données et les flots de vidéo doivent être synchronisés. Par ailleurs, quand on mêle vidéo et audio, il faut pouvoir assurer aussi une synchronisation entre les deux médias ;
- **Taux de transfert** : Pour toutes les applications en temps réel, il est important que le taux soit le plus élevé possible afin d'assurer un confort pour l'utilisateur.

Tous ces paramètres ont chacun leurs propres spécifications, mais il faut aussi tenir compte de leur dépendance. En effet, ils interagissent entre eux, et toute spécification sur l'un d'eux peut avoir des répercussions sur les autres. Il est d'ailleurs important de mettre des priorités si l'on veut gérer correctement ce problème. Quand une

nouvelle session de collaboration est initiée, le contrôle de Qualité de Service implique la surveillance des différents paramètres pendant que les paquets sont transmis. Il faut ensuite essayer d'arranger cela si la qualité est mauvaise afin de respecter les contraintes. On trouve différents modules [15] pour gérer chaque média et assigner les propriétés, comme l'illustre la Figure 2.6.

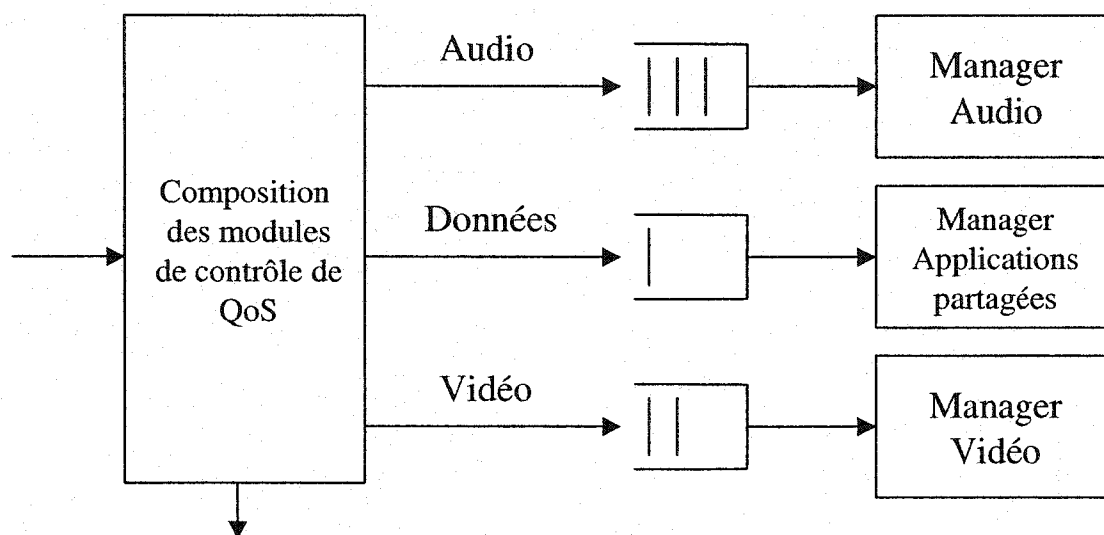


Figure 2.6 Contrôle de QoS

Dans le contexte des applications collaboratives, il est intéressant de présenter des résultats qui touchent à la QoS et au multimédia. En effet, le multimédia est une composante essentielle de la collaboration. De nombreuses applications existent pour transmettre la vidéo et l'audio entre les utilisateurs. Et comme bien souvent les plateformes développées le sont sur Internet, il y a très peu de garantie de Qualité de Service. Dans tous les cas, les applications voient leur qualité diminuer pendant une session multimédia. De plus, la qualité décroît quand les ressources reçues arrivent à saturation au niveau de l'ordinateur client. Il ressort alors qu'il y a deux niveaux à considérer [15] : Le niveau réseau et le niveau utilisateur.

Alfano et Sigle ([15]) préconisent de se placer au deuxième niveau en se basant notamment sur la perception qu'il peut avoir des services offerts par l'environnement

considéré. Nous voyons d'ailleurs une échelle de valeur intéressante pour tester cela à la Figure 2.7.

Niveau	Affaiblissement	Qualité
5	Imperceptible	Excellent
4	Perceptible, pas ennuyeux	Bon
3	Légèrement ennuyeux	Convenable
2	Ennuyeux	Pauvre
1	Très ennuyeux	Mauvais

Figure 2.7 Échelle du taux de qualité

Cette manière de réfléchir est relativement intéressante si on ne perd pas de vue le sujet de recherche de ce mémoire, les environnements d'apprentissage collaboratifs étant en effet hautement tributaires du jugement que peuvent se faire les utilisateurs de ce qui est développé. Voyons d'ailleurs les écrits qui existent déjà à propos de la qualité de service dans le domaine du télé-apprentissage.

2.4.3 La QoS pour une plate-forme d'apprentissage

La présence grandissante du multimédia devrait augmenter ou même remplacer à terme les modes traditionnels d'interactions sociales et économiques. Cette évolution est aussi vraie dans le domaine de l'éducation. Les environnements d'apprentissages répartis (DLE, Distributed Learning Environment) conçus à partir des technologies de l'information et des communications sont considérés comme des solutions possibles, mais il faut tout de même réussir à les implémenter efficacement. Pour que les objectifs éducatifs soient atteints et respectés, les DLEs doivent être en mesure de fournir certains services [16] :

- interactivité, engagement ;
- recours aux groupes ;
- parallélisme avec le monde réel (pas seulement de simulations) ;
- place centrale de l'étudiant ;
- accès à l'éducation en tout temps et en tout lieu.

La réceptivité est essentielle pour assurer un sentiment fort d'interaction dans les réseaux. Il faut en effet que le délai entre la requête et la demande soit raisonnable. La formation de groupes implique un partage sûr et sérieux des ressources du laboratoire, et ce, quel que soit le nombre d'utilisateurs concurrents. La concurrence est donc une préoccupation essentielle au même titre que celle de temps réel et de sécurité.

Rentabilité Abstractions pour étudiants et tuteurs, Groupes, Interaction, Entrée vers le monde réel, Interfaçage
Intergiciel Sécurité, Disponibilité, Concurrence, Contrôle, Reproduction
Infrastructure Réseau, Systèmes opérants, Serveur et client Plate-forme

Figure 2.8 Trois aspects de la QoS dans un DLE

Le rendement au niveau Qualité de Service est une clé du succès de ces plates-formes. Cette notion doit être étoffée pour qu'elle soit pertinente dans le contexte de

l'apprentissage électronique. Le DLE doit donc proposer une palette de propriétés systématiques, décrites à la Figure 2.8.

Ainsi le service que les applications requièrent peut être décrit par certains indices tels largeur de bande, délai convenable, fiabilité, gigue minimum, etc. Il est ainsi important d'avoir un contrôle renforcé du trafic. Les contraintes citées sont donc dictées par le type de ressources nécessaires comme les vidéoconférences basées sur le multimédia, etc. Nous nous concentrons sur ce problème de QoS qui constitue un des points d'intérêt majeur de ce mémoire.

CHAPITRE III

ARCHITECTURE POUR UN ESPACE COLLABORATIF

Le terme le plus intéressant et qui émerge pour désigner un espace de collaboration est sans conteste celui de « collabortory ». Ce concept s'inscrit parfaitement dans la démarche initiée au Laboratoire de recherche en réseautique et informatique mobile (LARIM). Ce chapitre vise à définir une architecture conceptuelle de collaboration prenant notamment en compte les exigences de Qualité de service (QoS) qu'un tel environnement d'apprentissage doit satisfaire. Après avoir spécifié le requis de l'architecture de collaboration, nous définirons les classes d'interaction de cette collaboration, suivi des scénarii et de l'architecture générale de collaboration. Nous terminerons le chapitre par la prise en compte des préoccupations relatives à la qualité de service dans ces environnements d'apprentissage.

3.1 Requis de l'architecture de collaboration

Afin de traiter le problème exposé dans le premier chapitre de ce mémoire, la première étape consiste bien évidemment à poser les bases pour la définition de l'architecture. Cela passe notamment par une analyse détaillée des besoins de la plateforme afin de proposer une solution adaptée à la situation.

3.1.1 Dimensions expérimentales de l'apprentissage

Pour l'enseignement des sciences, il est essentiel de pouvoir prolonger les exposés théoriques par de la pratique et de l'expérimentation. En effet, il est de notoriété que les élèves apprennent plus rapidement grâce à cela. Il faut donc mettre à la disposition de ces derniers :

- des cours biens conçus en fonction d'objectifs clairs ;
- des exercices conformes aux objectifs des cours et permettant de tester les connaissances ;

- un accès à des bases de données pertinentes et utiles.
- des simulations (expériences déjà réalisées avec applications spécifiques) ;
- des expériences pouvant se réaliser à distance par l'intermédiaire d'une infrastructure de réseaux.

La Figure 3.1 intègre les composants pédagogiques d'un tel environnement d'apprentissage virtuel :

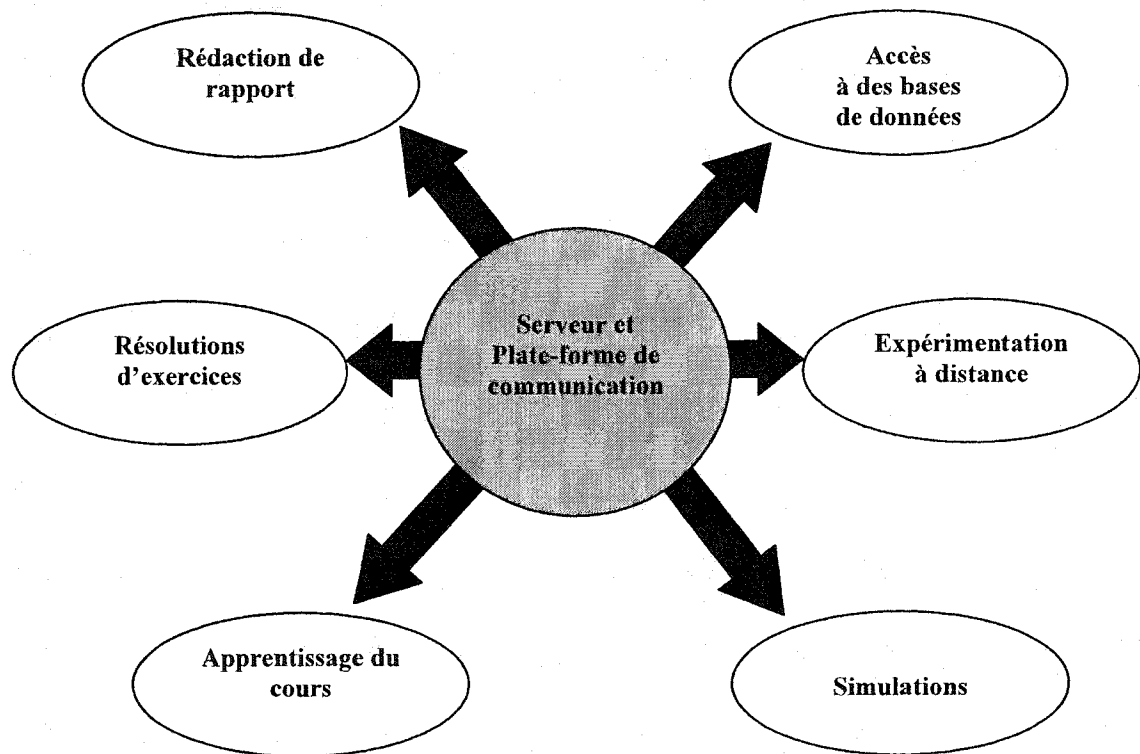


Figure 3.1 Composantes pédagogiques d'un environnement d'apprentissage

3.1.2 Les protagonistes du laboratoire

L'inventaire réalisé dans le paragraphe précédent n'est pas suffisant s'il est pris seul. En effet, pour se faire une idée exacte des besoins d'un tel laboratoire, il faut aussi

répertorier les protagonistes. La collaboration envisagée dans ce mémoire implique donc forcément plusieurs personnes et plusieurs entités. Ainsi, suivant les travaux à effectuer, on peut avoir besoin de s'adresser à une ou plusieurs personnes, et cela sera fait en distinguant les rôles de chacune d'entre elles. Il faut donc définir ces entités en prenant en compte leur rôle et leurs fonctions. Pour le problème qui nous concerne nous avons déterminé trois acteurs distincts : Coordonnateur, Elèves et Groupes. Décrivons donc exactement ce que représentent ces trois entités.

- **Le Coordonnateur (C):** Ce terme sert à désigner le professeur, le chargé de laboratoire ou toute autre personne qui gère le processus d'apprentissage. Sa principale fonction consiste à superviser et coordonner l'ensemble des ressources, services et outils du laboratoire. C'est lui qui prendra les décisions visant à répartir les efforts en disant quoi faire et quand le faire. Il doit être capable de répondre aux questions, de fournir des explications sur les expériences ou d'évaluer le travail des étudiants.
- **L'Etudiant (E):** C'est l'entité de base du laboratoire. C'est en effet celle qui est la plus simple et autour de laquelle tout doit s'articuler. Le processus d'enseignement doit se focaliser sur elle. Tous les efforts et tous les outils seront développés pour satisfaire au mieux ses besoins. Il doit avoir accès à tous les outils, toutes les fonctionnalités et un maximum de données lui permettant de progresser dans son apprentissage. Par ailleurs, il doit être en mesure de communiquer avec tout le monde.
- **Le Groupe (G):** Le groupe désigne un regroupement d'élèves (le nombre n'est pas connu car il est fonction du laboratoire). C'est l'entité la plus singulière. En effet, si elle n'a pas d'existence physique au même titre que le coordonnateur ou l'élève elle n'en demeure pas moins essentielle. Le groupe est l'essence même de la collaboration. Au niveau de l'architecture, nous verrons qu'il a une existence (il faut recourir à des

protocoles particuliers pour gérer les groupes). Par ailleurs, la notion de groupe implique forcément celle de *leader* (notion que nous développerons par la suite) puisqu'il faudra forcément un représentant concret pour la symboliser lors des échanges à l'intérieur du laboratoire. Ce dernier est chargé de s'exprimer au nom de l'ensemble des élèves d'un groupe.

Ces trois entités composent donc l'ensemble des intervenants répertoriés. C'est à partir de ces trois là que nous pourrons tout détailler, expliquer et mettre en oeuvre par la suite. L'architecture de collaboration sera donc développée pour répondre aux besoins de ces trois types d'« intervenants » illustrés à la figure 3.2 .

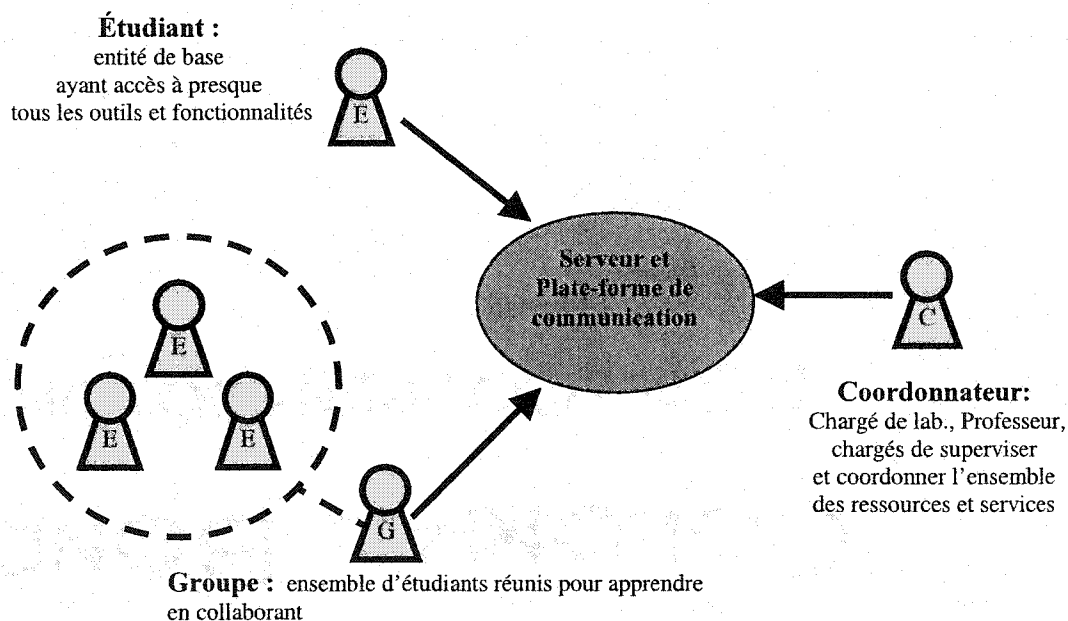


Figure 3.2 Les protagonistes ou acteurs

Le rôle de Leader

Après avoir défini les intervenants du laboratoire et avant de voir plus en détails les scénarii de collaboration revenons sur une notion primordiale, celle de leader. Au

premier abord, on peut se poser la question de savoir si ce concept a une réelle utilité. Pourtant, si cela peut paraître un peu paradoxal, le recours à ce « protagoniste » vise bien à simplifier le tout. En effet, si l'un des élèves du groupe ne possède pas de statut particulier, s'il n'a pas de prérogatives spéciales et étendues par rapport aux autres, il est évident que les échanges avec le coordonnateur ou entre groupes ne pourront se faire correctement. Grâce à ce leader, qui a un rôle de relai, on réduit toutes les communications à des communications au niveau de complexité équivalent à celui de communication entre deux interlocuteurs simplement. De cette manière, seule une personne envoie le flot de données. Dans le même temps, le professeur, les élèves de l'autre groupe et ceux du groupe du leader ont exactement le même statut et reçoivent le même flot grâce à l'utilisation de liaisons « multicast ».

Par ailleurs, à l'intérieur d'un groupe, si au cours de la discussion, les élèves de ce groupe ne sont plus d'accord avec le leader, il faut interrompre le transfert vers l'extérieur pour rediscuter et redéfinir la position commune. Après cette concertation intra-groupe (via le « chat » ou des liaisons « multicast » réduites), un nouveau leader chargé de s'exprimer au nom de l'ensemble est nommé.

3.1.3 Prise en compte de l'aspect de collaboration

Comme dans de nombreux autres problèmes scientifiques, la base de la résolution du problème qui nous concerne repose sur une identification précise et complète de l'ensemble des tenants et aboutissants. Ici puisque nous devons être en mesure de proposer un modèle d'architecture collaboratif [18] pour le laboratoire virtuel, nous allons suivre cette démarche. De cette manière, nous serons en mesure d'identifier les besoins exacts pour la collaboration ainsi que tous les outils et composants nous permettant de proposer une architecture efficace et répondant à toutes les exigences d'un tel environnement.

Avant de définir les bases de ce modèle, il est peut-être judicieux de commencer par trouver la place exacte que l'on doit donner à la collaboration dans le processus d'apprentissage. C'est cette situation qui est illustrée à la Figure 3.3.

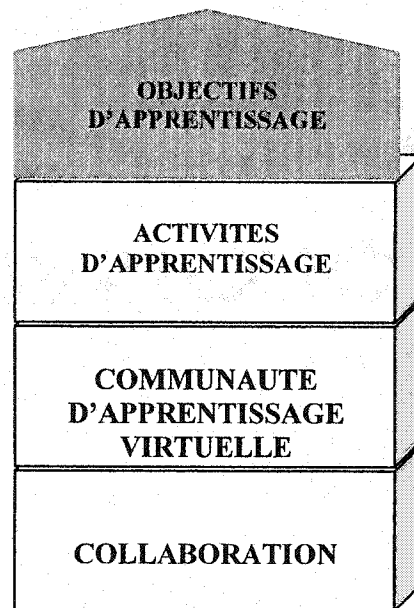


Figure 3.3 Place de la collaboration dans le processus d'apprentissage

La collaboration peut être appréhendée sous l'angle de deux classes d'interactions complémentaires : la *coopération* et la *communication*. Commençons donc par définir ces deux concepts dont les liens avec la collaboration sont illustrées à la Figure 3.4.

Coopération

Par *coopération*, on entend le processus de partage et de mise en commun des outils et des connaissances permettant une optimisation du travail de groupe.

Communication

La *communication* est l'ensemble des échanges qui peuvent s'opérer au sein du laboratoire et conventions de paroles que l'on peut retrouver.

Ainsi, il est plus aisé de comprendre tout ce qu'implique le mot *collaboration* comme on souhaite l'utiliser dans notre travail. De même, on comprend mieux la place qu'elle prend et pourquoi elle est étudiée à ce niveau de l'apprentissage.

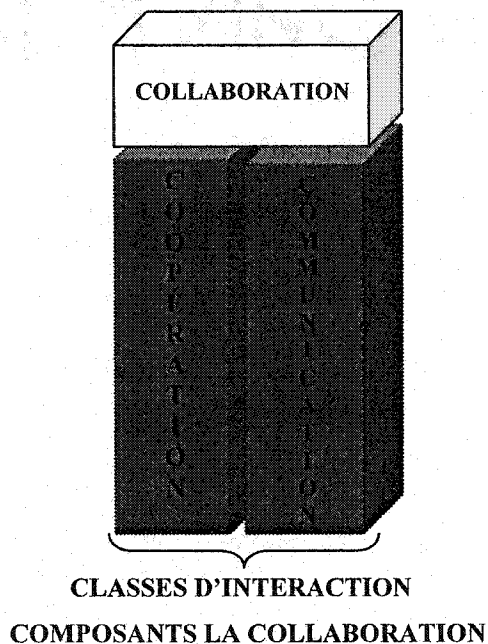


Figure 3.4 Décomposition de la collaboration

Ces notions seraient incomplètes si nous n'y associons pas non plus la coordination qui joue un rôle essentiel dans l'ensemble du processus. Cette dernière est la répartition des efforts, ainsi que la gestion des ressources et participants. C'est en quelque sorte le ciment qui permet la cohésion entre l'ensemble des étapes du processus. C'est l'élément essentiel qui, bien que n'ayant pas d'existence concrète, apporte toute l'organisation et l'ordre nécessaires au bon développement de la méthode mise en place. Cependant, nous ne l'étudierons pas en détails dans ce mémoire car nous ne la considérons pas comme une classe d'interaction au même titre que les deux précédentes.

Le modèle conceptuel peut encore être amélioré si on ne se contente pas de cette classification verticale. Ainsi, chaque classe d'interaction peut être analysée à la lumière de 3 niveaux sous jacents : le niveau fonctionnel, le niveau structurel et le niveau technologique. Cette décomposition permet de détailler dans les classes d'interactions en précisant graduellement la base de chacune d'elle. Ainsi, le niveau fonctionnel est purement conceptuel puisqu'il décrit ce que chaque classe d'interaction est censée représenter. Ensuite, le niveau structurel permet de définir les moyens à mettre en œuvre pour réaliser ces objectifs. Enfin, le niveau technologique présente les solutions techniques adéquates permettant de supporter tous les outils utilisés pour réaliser les fonctionnalités de notre plate-forme.

De cette manière, il va donc ressortir une grille d'analyse et de structuration pour la collaboration qu'il nous incombe donc de compléter en détails dans la suite de notre travail, grille exposée à la Figure 3.5.

	COOPERATION	COMMUNICATION
FONCTIONNEL		
STRUCTUREL		
TECHNOLOGIQUE		

Figure 3.5 Grille d'analyse et de structuration pour la collaboration

Chaque partie de cette matrice est donc à préciser, conceptuellement dans un premier temps puis de façon plus concrète en identifiant notamment les éléments qu'il

faut insérer et définir pour arriver à une analyse la plus pertinente possible. Tentons d'abord de dresser un rapide inventaire des différents éléments de cette grille d'analyse. D'un point de vue fonctionnel, la coopération décrit les paradigmes d'interaction qui mettent en avant les interactions enseignant /apprenants (utilisateurs) dans les classes. Au niveau structurel, la coopération désigne la palette d'outils auxquels on a recours pour assurer une réponse conforme à ces exigences. Enfin, du point de vue technologique, celle-ci fait référence aux différentes technologies susceptibles d'être utilisées pour implémenter ces outils et ce, au niveau de l'interface et du processus.

Dans le même ordre d'idées, voyons à quoi correspond la communication. Du point de vue fonctionnel, cette classe d'interaction décrit les conventions de langage qui sont présentes dans le processus d'apprentissage du laboratoire. Au niveau structurel, la communication désigne les exigences quantitatives et temporelles à considérer dans les différents échanges envisagés au cours de la démarche. Enfin, en se plongeant plus profondément dans les couches et en allant jusqu'au niveau technologique, on trouve l'ensemble des protocoles nécessaires pour gérer les communications simples et de groupes. En poussant le raisonnement un peu plus loin, nous pourrions à la limite utiliser cette décomposition pour traiter le cas de la *coordination*. Cependant, étant donné que nous avons décidé placer la *coordination* à un niveau plus général, nous nous contenterons de l'évoquer en la considérant, comme évoqué précédemment, en tant qu'élément fédérateur de l'ensemble. Après ce rapide tour d'horizon assez général de notre modèle conceptuel, attachons nous à décrire plus en détails chaque éléments de la matrice en insistant sur les parties que nous voudrions développer dans le futur.

3.2 Classes d'interaction de la collaboration

Puisqu'il a été démontré que la collaboration faisait partie intégrante du processus d'apprentissage et que l'on conjecture par ailleurs qu'elle se décompose en deux classes d'interactions différentes permettant de la décrire complètement, nous allons voir dans la suite ce que représentent en détails ces deux classes : coopération et communication.

3.2.1 Coopération

Afin de bien représenter la coopération, il sera intéressant de la décomposer pour en voir toutes les subtilités. Nous examinerons donc les trois différents niveaux (fonctionnel, structurel et technologique) définis précédemment.

a) Niveau fonctionnel

Nous décrivons ici ce qu'est la coopération dans son ensemble, et pour cela nous suivrons l'ordre des niveaux d'analyse déjà évoqués, dont le premier est le niveau fonctionnel. En effet, avant de voir ce qui est possible comme solution technique et les outils que l'on peut proposer aux utilisateurs, il est important de préciser les besoins, soit le niveau fonctionnel de la coopération. En effet, on peut ainsi définir l'ensemble des interactions entre participants, voire les différentes actions que chacun d'eux peut avoir besoin de réaliser. Il faut avoir une vision claire et précise de tous les tenants et aboutissants de ces interactions.

Pour en donner des définitions les plus correctes et précises, faisons cet inventaire en fonction du rôle de chacun dans le laboratoire. Ainsi, nous séparerons le coordonnateur et les élèves. Ces deux entités ont des rôles distincts à jouer que nous illustrons aux figures 3.6 et 3.7. Le coordonnateur a donc un certain nombre d'exigences auxquelles la plate-forme doit être en mesure de répondre. Son rôle d'éducateur et de guide ne doit pas être altéré par la spécificité de l'outil d'apprentissage. En effet, pour réaliser cette classification, on s'inspire très largement des besoins que pourrait avoir le professeur dans une classe traditionnelle.

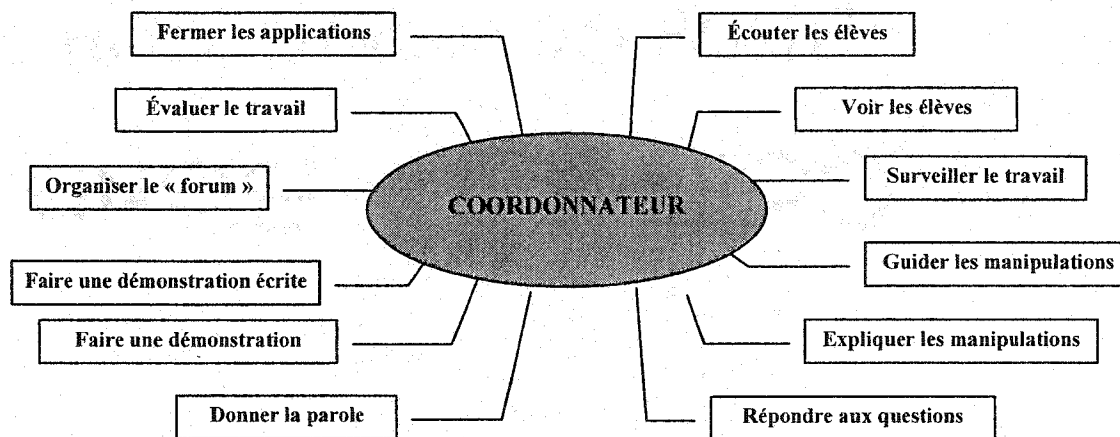


Figure 3.6 Besoins du coordonnateur

En plus de toutes ces possibilités, il est bien évidemment essentiel de rajouter les particularités dues à l'utilisation du multimédia et de la mise en réseau des connaissances et du savoir-faire. Ainsi, il faudra pouvoir lui assurer par exemple : de voir et écouter les élèves, de surveiller le travail, de l'évaluer, mais il faut aussi pouvoir faire des démonstrations, organiser un « forum » de discussion, etc. Si ces actions peuvent sembler quelconques, il faut aussi prendre en considération le fait qu'elles doivent pouvoir s'adapter au support technique proposé. Ainsi, l'évaluation du travail requiert une transmission des droits et un partage des applications, et chaque action aura son exigence à ce niveau.

Cette démarche est aussi celle suivie pour établir l'inventaire des actions auxquelles pourront être confrontés les élèves lorsqu'ils évolueront dans le laboratoire virtuel. Voyons donc toutes les possibilités offertes par la plate-forme à ces derniers. Là encore, nous nous apercevons que tout ce que l'élève doit être en mesure de réaliser est largement inspiré de ce auquel il est habitué dans une classe traditionnelle. En effet, pour le travail en groupe, il doit le faire en association avec ses camarades et donc doit pouvoir montrer son cahier ou écrire sur celui du groupe, mais encore une fois, il ne faut pas oublier que l'utilisation de l'informatique et de la réseautique apporte une contrainte supplémentaire : comment assurer cette délégation de droit avec sécurité. De même il

faut que ses échanges avec le professeur soient efficaces (prises de notes, remise du rapport, etc.).

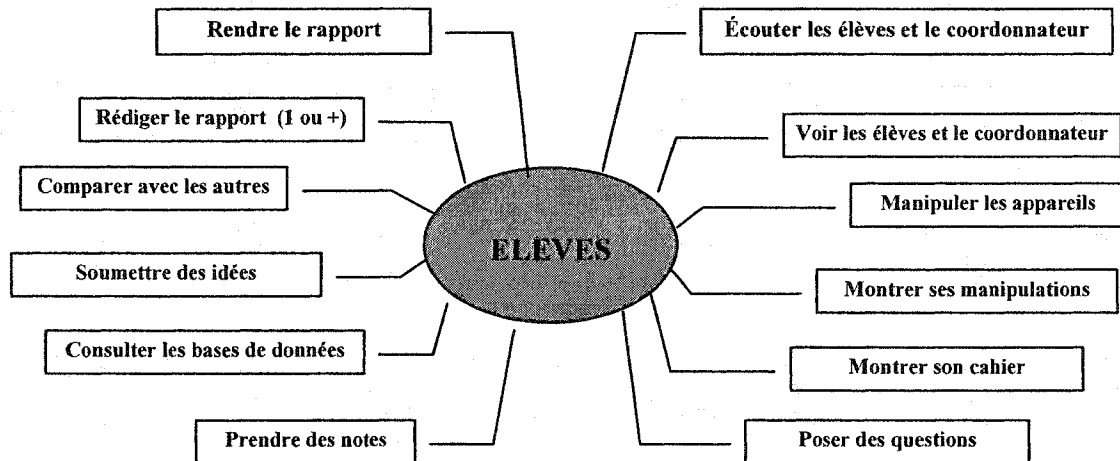


Figure 3.7 Besoins des élèves

Les besoins du groupe

Le groupe n'est pas détaillé ici, car il est assimilé aux élèves qui le constituent. C'est pour cela que l'on ne lui donnera pas d'autres prérogatives et qu'on ne détaillera pas plus ses besoins. En effet, nous considérerons que ce sont les mêmes.

b) Niveau structurel

Après avoir énuméré l'ensemble des actions que l'on peut imaginer et rencontrer dans le laboratoire virtuel pour coopérer, il faut aussi définir les outils permettant de le faire. Il s'agit donc de construire le niveau structurel de la coopération. Pour cela, en fonction de nos besoins, nous allons présenter une palette de possibilités envisageables. En effet, il nous faut définir un ensemble d'outils à mettre en oeuvre pour assurer une interaction productive entre les protagonistes (E et C).

L'outil de base pour les échanges entre les utilisateurs pourrait être le « chat ». C'est un outil simple mais efficace pour assurer la discussion entre tous et le contact permanent. De plus, il permet de voir qui est effectivement présent dans le laboratoire au moment où un utilisateur donné souhaite travailler.

Puisque les élèves et le coordonnateur doivent être en mesure de montrer un travail ou de faire des démonstrations, nous pensons immédiatement au cahier blanc. Ainsi, ce dernier doit être un support de réflexion et de travail pour les uns comme les autres, et tous doivent être en mesure de le faire partager. Les technologies actuelles permettent d'envisager un travail à distance qui ne serait cependant pas privé totalement de contact audio et vidéo. En effet, l'utilisation de l'image et du son peut donner une dimension supplémentaire à l'enseignement à distance. En le rendant plus attractif, il serait beaucoup plus facile de l'imposer comme standard pour le futur. C'est pour cela que le recours à la visioconférence est clairement conseillé pour une coopération efficace. La figure 3.8 synthétise les outils de coopération pouvant être mis à la disposition des apprenants à des fins de collaboration. Le design et l'utilisation simultanée de tous ces outils imposent un grand nombre de contraintes de QoS. Pour que le recours à ces médias soit efficace, il faut en effet assurer que tout sera mis en oeuvre pour n'avoir aucune limitation due à la technologie.

c) Niveau technologique

Les outils individuels de coopération que nous avons décrit dans la sous-section précédente constituent le support et l'essence de la coopération. Il nous incombe de décrire les technologies utilisées pour l'implémentation. Cependant puisque la description détaillée sera faite dans le chapitre suivant selon les besoins, nous nous limiterons ici à une description succincte dans ce paragraphe. Nous nous contenterons donc de nommer les protocoles utilisables et prévus. Au niveau de la couche transport, le recours à UDP et TCP, selon les applications considérées, est vivement recommandé. Pour la couche application chaque outils a sa technologie. Ainsi nous recommandons l'utilisation de HTTP pour le cahier, RPC pour la pile, RMI pour le « chat » ou encore RTP/RTCP pour le « chat audio » ou la visioconférence.

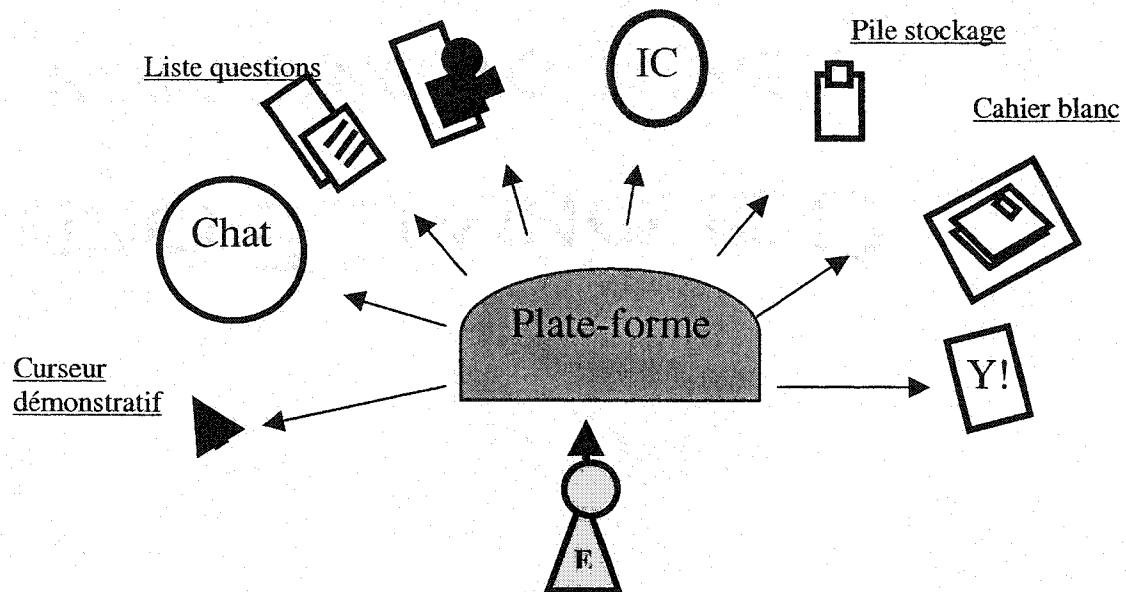


Figure 3.8 Outils de coopération

3.2.2 Communication

Comme pour le cas de la coopération, la méthode suivie pour la description du concept associé à la communication est très simple. Nous suivrons les trois différents niveaux (fonctionnel, structurel et technologique) définis précédemment.

a) Niveau fonctionnel

A nouveau, il est essentiel de débiter par le niveau fonctionnel car c'est lui qui nous donne la définition la plus précise et concrète de la communication. De ce point de vue, la communication correspond à l'ensemble des conventions qui vont régir et impliquer les échanges d'informations, données et paroles entre les différents protagonistes du laboratoire. En effet, il faut définir l'ensemble des actions qui seront menées par les intervenants afin d'obtenir des échanges profitables et bénéfiques. Un concept qui résume relativement bien cette idée est celui qui apparaît dans la littérature sous le terme de « Speech Act » [19] dont la traduction littérale est « acte de la parole ». Il décrit bien ce que l'on sous-entend par communication à ce niveau (même si le mot

parole peut paraître un peu restrictif pour notre sujet vu la diversité des échanges permis par les outils). Voyons donc les différentes situations où on va retrouver ces composantes du « Speech Act » à l'intérieur du laboratoire et du processus d'apprentissage. Pour effectuer cette classification, là encore, nous nous sommes directement inspirés de ce que l'on retrouve dans les classes et séances de travaux pratiques traditionnels. Pour la plupart, ces actions sont communes à toute communication entre personnes :

- Questions : Que ce soit aux coordonnateur ou aux autres élèves, à tout moment, un apprenant est susceptible de poser des questions, nécessaires à sa compréhension ou mise en pratique expérimentale ;
- Réponses : Un dialogue s'installe au cours de la séance de laboratoire ou tout le monde est apte à aider l'autre en répondant à ses interrogations ;
- Explications : Une fois les manipulations effectuées, si le résultat n'est pas celui escompté, il faut que l'un des interlocuteurs soit en mesure d'expliquer son point de vue ou que le coordonnateur montre ce qu'il attend de l'élève ;
- Argumentation : Lors des discussions sur un projet, à l'intérieur d'un groupe, les apprenants vont discuter pour essayer de convaincre les autres du bien fondé de leurs idées ;
- Mise en hypothèses : Toujours lors de la préparation du travail, il faut souvent mettre en hypothèse avant de pouvoir le partager avec les autres et de les tester ;
- Justification : Pour appuyer ses hypothèses il faut absolument apporter des arguments recevables et en rapport avec le cours ;
- Concertation : Avant de présenter le travail ou la manipulation et d'élire le leader chargé de parler au nom du groupe, les élèves doivent se mettre d'accord.

Enfin, il reste les tâches qui incombent plus spécifiquement au coordonnateur du fait de son rôle de superviseur :

- Reformulation : Afin de rendre les explications plus claires pour l'ensemble des élèves, il peut avoir besoin de tout ré expliquer dans des termes plus précis ;
- Commentaires : A la fin du processus, il doit être en mesure d'apporter des commentaires sur le travail effectué avant de l'évaluer.

b) Niveau structurel

Après avoir décrit les différentes actions envisageables pour expliquer au mieux ce qu'était la communication (« Speech Act », [19]) , déterminons maintenant les contraintes structurelles qui concernent cette classe d'interaction. En effet, il est essentiel de remarquer que l'utilisation des réseaux rend les échanges légèrement plus complexes que la voie traditionnelle. Ainsi, il ne faut plus considérer un seul mode de transmission. Suivant le nombre d'utilisateurs, suivant les moments de connexions, les communications seront classées dans différentes catégories. Pour bien décrire toutes les possibilités, nous nous intéresserons à deux sortes de considérations : quantitatives et temporelles.

Considérations quantitatives

Nous classerons les différentes communications en fonction du nombre de participants, de la présence de groupe ou non, etc. Pour ce faire, nous irons des communications les plus simples à réaliser vers les plus complexes :

La communication la plus simple à ce niveau est celle qui ne met en jeu que deux interlocuteurs: on parlera alors de liaison « unicast » illustrée à la figure 3.9.

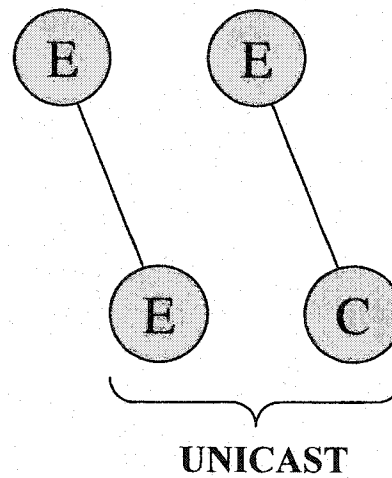


Figure 3.9 Liaisons « unicast »

Considérons ensuite le cas où le professeur veut s'adresser à tous (ou du moins permettre à tous d'avoir un accès éventuel à ce qu'il « dit »). Il utilisera alors une liaison « broadcast » illustrée à la figure 3.10 .

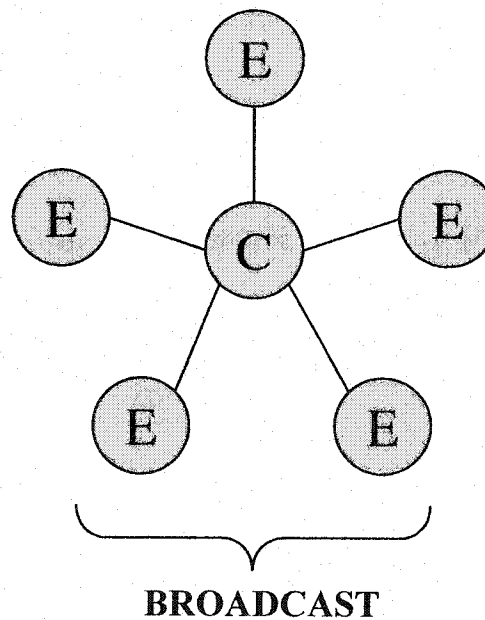


Figure 3.10 Liaisons « broadcast »

Enfin, en se plaçant au niveau des groupes, on peut avoir des ensembles d'envois restreints (nombre limité de personnes, 1 groupe, 2 groupes, etc.). Il s'agit alors de liaisons « multicast » illustrées à la Figure 3.11.

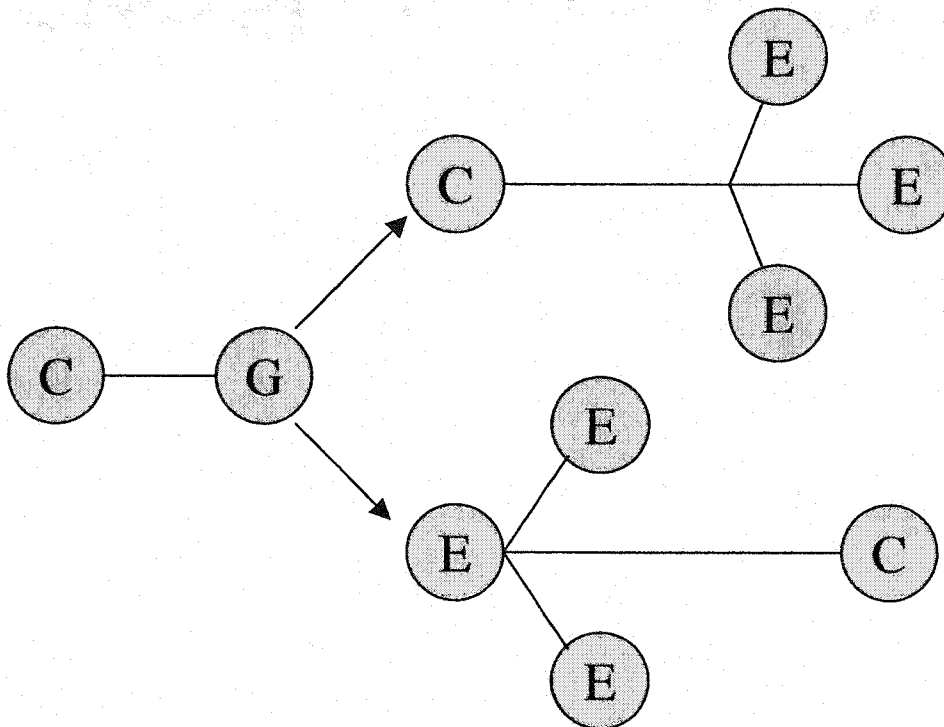


Figure 3.11 Liaisons « multicast »

À ce niveau, il faut vraiment faire très attention à bien percevoir dans quel ensemble d'échange il faut se placer. En effet, tout s'organisera de manière bien différente si on se trouve dans un groupe seulement ou dans des ensembles plus vastes.

Considérations temporelles

Ici aussi, le but recherché sera déterminant dans la manière d'organiser l'architecture et la forme des communications. Il faut donc être vigilant lors de la mise en place des solutions souhaitées. En effet, suivant les applications requises, nous devrons envisager trois niveaux de communication distincts : asynchrone, synchrone et temps réel [20]. Ces trois modes différents seront alors classés par ordre de complexité croissante au niveau des contraintes.

- Asynchrone : Cela fait référence aux applications pour lesquelles les interlocuteurs ne sont pas simultanément présents (typiquement les mails ou les forums de discussion) [21]. Si cette méthode paraît la plus simple à prime abord, elle n'est pourtant pas dénuée d'intérêt. Et même bien au contraire puisqu'elle pourrait être la plus intéressante car c'est celle qui rajoute le plus de possibilité par rapport à l'enseignement traditionnel. Elle a la chance d'être bien complémentaire avec le manque habituel (on s'affranchit de la nécessité temporelle) ;
- Synchrone : Pour cette méthode, les applications sont démarrées en même temps (origine identique) [22] avec en plus la même base de temps et on progresse en parallèle avec son interlocuteur (chat) ;
- Temps Réel : Par rapport à la classification précédente, il faut rajouter la contrainte plus forte d'interaction. En effet, un « signe » de l'interlocuteur est nécessaire pour la progression (visioconférence).

c) Niveau technologique

Pour la communication nous devons aussi préciser le niveau technologique. Pour cette classe d'interaction, l'architecture doit notamment être capable de gérer les communication « multicast » ou « broadcast ». Le point important de notre analyse est le choix de la couche réseau utilisée. En effet, une étude du protocole Ipv6 nous permet de conclure qu'elle est parfaitement adaptée à nos besoins. Les différentes formes d'échange envisagées sont pris en charge par le protocole lui même. Nous nous baserons donc sur cette technologie pour réaliser ce projet.

3.3 Scénarii et architecture générale de collaboration

Après avoir vu tous les éléments de la plate-forme, après avoir classé tous les besoins, il reste une étape pour expliquer au mieux les choix faits dans l'architecture globale. Il s'agit là aussi d'un inventaire, mais d'une autre forme : répertorier les différents scénarii que l'on peut rencontrer. En effet, afin de bien comprendre le rôle

exact de chaque module, afin de comprendre pourquoi on a recours à certaines parties, il est intéressant de décrire le déroulement d'une étape de l'apprentissage.

3.3.1 Scénarii les plus représentatifs

Voyons ce qui se passe en détails lors de la résolution d'exercices et la manipulation à distance.

a) Résolution d'exercices

Pour la résolution d'exercices, nous considérerons qu'un nombre n d'élèves se voient proposés des exercices d'application du cours, et ce, de manière synchrone ou asynchrone. En considérant que le coordonnateur est présent lorsque l'élève fait sa séance (le cas asynchrone est abordé par la suite), l'outil de communication de base est le « chat ». Grâce à lui, l'élève (E) pose toutes les questions qu'il veut (figure 3.12 et figure 3.13).

Voyons le premier cas de figure : la question nécessite juste une réponse personnelle pour Ei . Dans ce cas, le professeur (C) a trois moyens pour répondre. En effet, en fonction de la complexité, il pourra choisir d'entrer en contact avec Ei grâce à trois outils distincts:

- Simple => « Chat »
- Démonstration simple => Visioconférence
- Démonstration avec exemples => Cahier de démonstration.

Le mieux demeure bien entendu de parvenir à coordonner deux ou même tous les outils simultanément (d'où l'importance de la coordination). Il serait en effet très séduisant d'envisager la combinaison des trois ou le C pourrait par exemple faire les gestes qui seraient ensuite suivis de mouvement (curseur) sur le cahier.

Le deuxième cas de figure (Figure 3.13) est simplement un peu plus long mais pas tellement plus complexe. En effet, si C trouve que la question de Ei est suffisamment intéressante et profitable pour l'ensemble de la classe, il peut décider de

proposer une réponse commune (la classe étant interrompue ou non). Comme précédemment, cette réponse pourra se faire sous les trois formes permises par les outils. La différence résiderait alors dans le fait qu'on utilise de la diffusion « broadcast » au lieu de l' « unicast ».

Cependant, ce ne serait pas la seule amélioration par rapport au cas précédent. C'est en effet maintenant qu'intervient la pile de stockage permettant d'appréhender de façon efficace le problème de l'asynchronisme. Pour permettre aux élèves n'étant pas présent à l'heure du laboratoire d'avoir accès aux mêmes fonctionnalités, on envisage de bâtir une liste de questions-réponses sur le modèle d'un forum de discussion. C ayant trouvé la question pertinente, il l'édite sur la liste qui est accessible par tous et en tout temps. Les élèves auront alors accès à une liste de questions-réponses (les réponses pouvant prendre trois formes différentes) en ligne. Le résumé de ce déroulement est illustré à la Figure 3.12.

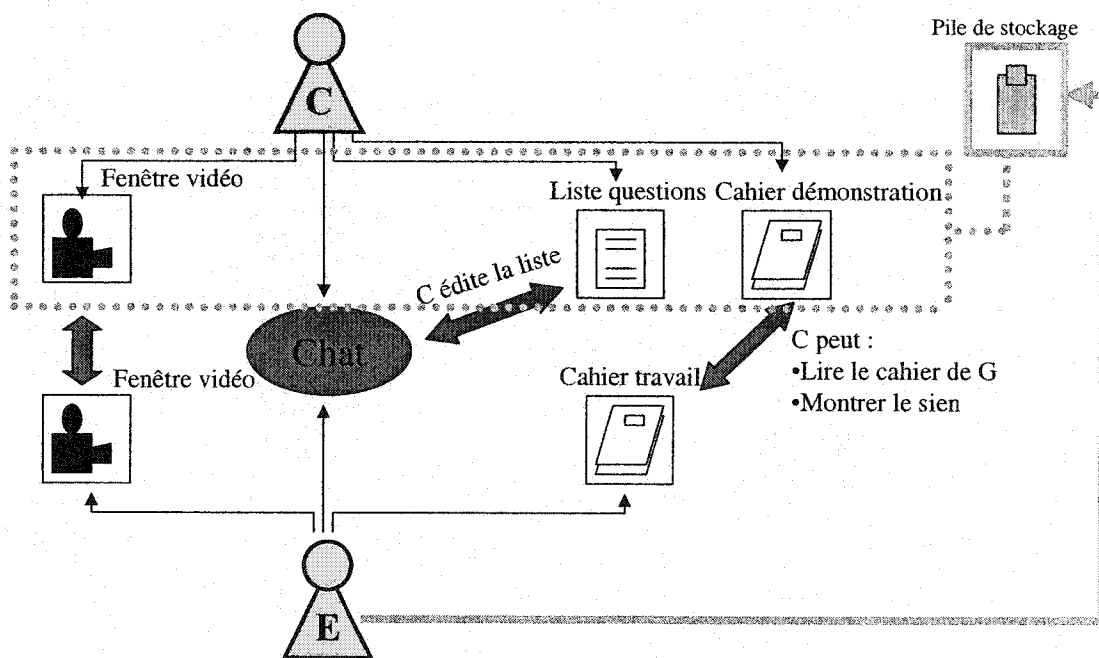


Figure 3.12 Scénario de résolution d'exercices

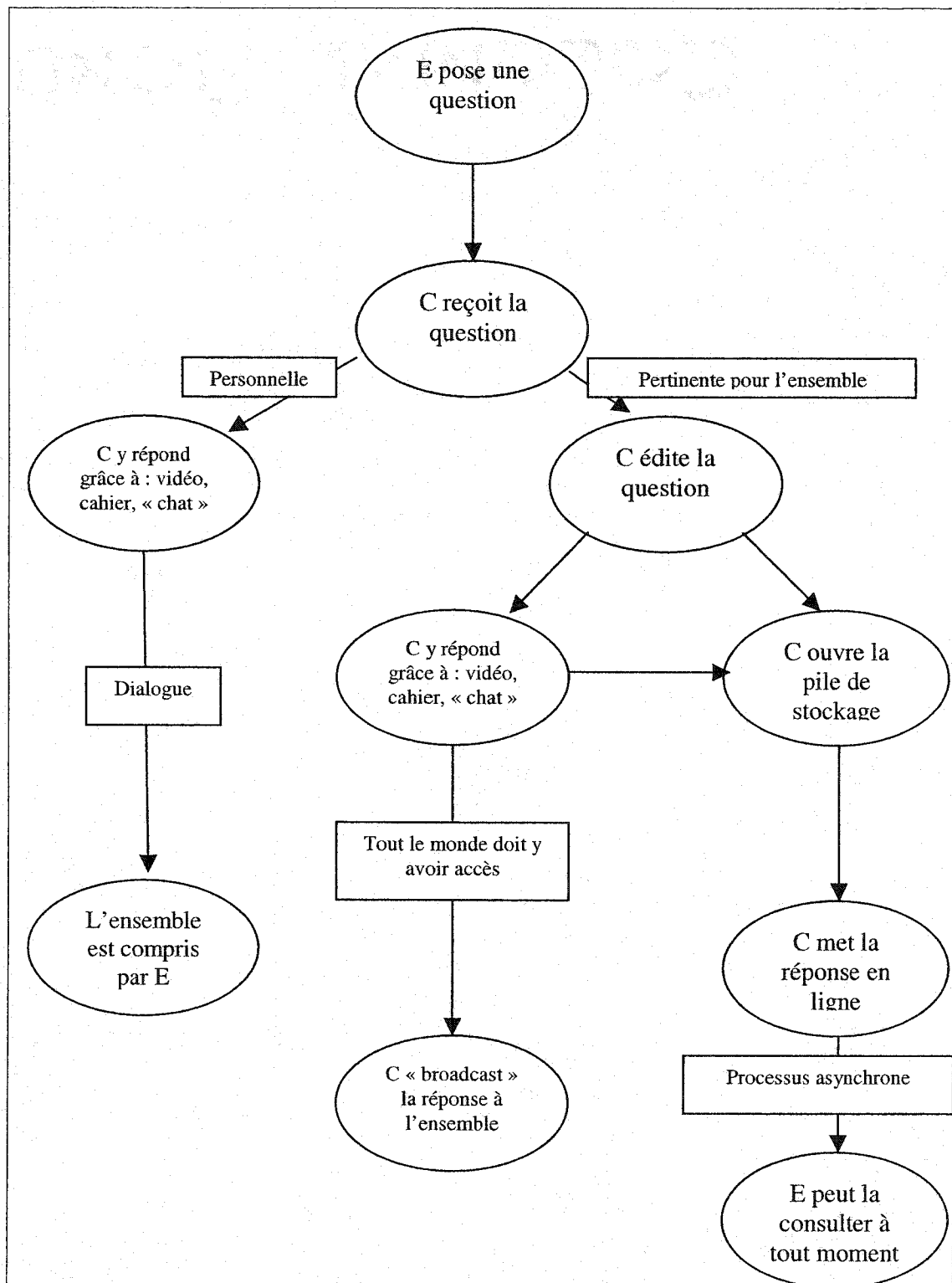


Figure 3.13 Scénario de résolution d'exercices

b) Manipulation

Le deuxième scénario (Figure 3.14 et Figure 3.15) le plus représentatif (et le plus différent du premier) concerne les manipulations. Il est assez complémentaire du premier car bien dissociable et illustre donc les points que ne faisait pas le précédent. Nous présenterons dans ce cas de Figure un scénario où l'essentiel des échanges se passent à l'intérieur d'un groupe

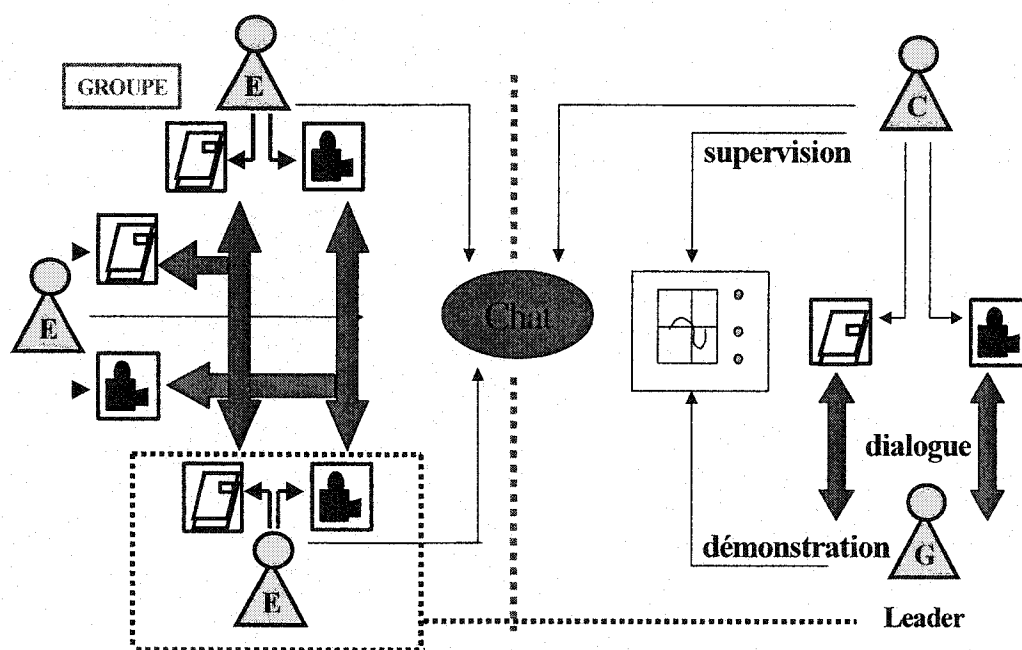


Figure 3.14 Scénario de manipulation à distance

Il est intéressant de noter comment le travail à plusieurs est organisé dans le laboratoire. Comme lors de la résolution d'exercices, l'outil principal est le « chat ». On utilise la possibilité de restreindre celui-ci à un nombre limité (un groupe en l'occurrence). Grâce à cet outil, les étudiants peuvent soumettre leurs idées et discuter facilement. La complexité éventuelle sera traitée à l'aide de la visioconférence ou du cahier.

A ce niveau du processus, tous les élèves d'un groupe ont exactement le même statut. Mais par la suite, il va leur falloir élire le leader. Et après consultation au sein du

groupe, après les manipulations, c'est ce dernier qui est chargé de faire la démonstration à C (il parle avec C comme pour les exercices). Il prend ensuite le contrôle de l'appareil à manipuler et fait la démonstration à C. Toujours grâce au principe de la diffusion « multicast », les autres élèves du groupe peuvent suivre, au même titre que C, ce qui se déroule. Si nécessaire, C prend à son tour le contrôle de l'appareil pour apporter les modifications qui s'imposent. Ce scénario est illustré à la Figure 3.15 où l'enchaînement des étapes est détaillé.

3.3.2 Architecture de collaboration

Dans la sous-section précédente, nous nous sommes attachés à détailler au maximum ce que représentait la collaboration. Cela avait pour but de préparer et comprendre l'architecture de l'espace collaboratif que nous souhaitons développer. C'est cette architecture que nous présentons donc dans cette sous-section en tentant d'y faire ressortir les points intéressants et remarquables ayant trait à l'analyse effectuée jusque là. Dans cette architecture, il ressort deux sous-ensembles assez différents. En effet, le premier (séparé du reste par des pointillés) représente la partie client. On se place dans une architecture du genre client-serveur. Ce client représente l'ensemble des éléments que l'on retrouve en local « sur la machine », par opposition avec le reste où tout est commun à l'ensemble des utilisateurs du laboratoire.

Le deuxième sous-ensemble (séparé à l'aide de traits gras) est surtout remarquable par sa fonction. En effet, il touche à tout ce qui est QoS. Puisque c'est un point important de notre recherche, il est normal de lui donner une place conséquente dans notre architecture. Cet ensemble dont le *QoS manager* tient la place centrale est chargé de s'assurer que tout est fait pour que la qualité de service soit optimale. Dans ce qui suit, nous allons voir plus en détails chaque module constituant l'architecture globale proposée à la Figure 3.16. Voyons donc à quoi correspondent les différents modules en insistant notamment sur les rapports entre tous. Nous tenterons aussi de bien insister sur leur place dans le processus de collaboration.

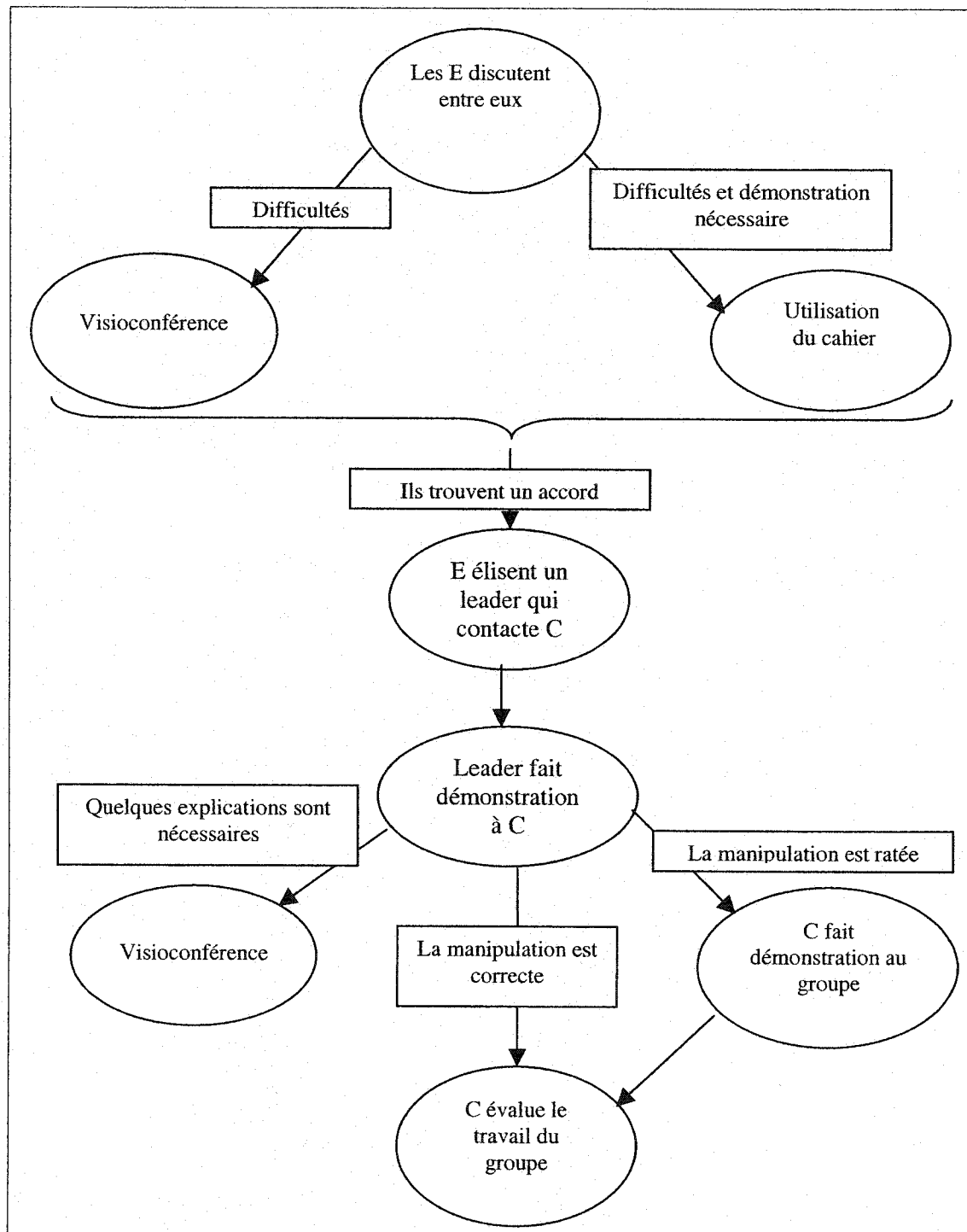


Figure 3.15 Scénario de manipulation à distance

Interface Graphique Utilisateur : Premier contact de l'utilisateur avec le laboratoire, c'est à partir de là que l'utilisateur a accès aux différentes parties du laboratoire. L'interface graphique est accessible à partir du net (d'où la suite qui se basera sur IP) faisant la relation entre le contenu et l'utilisateur. Une fois dessus, l'utilisateur peut aussi avoir accès aux données du laboratoire s'il a été reconnu comme ayant droit.

Module d'Admission : Une fois que l'utilisateur est arrivé sur le site du laboratoire, il doit s'enregistrer (afin de voir quelles parties lui sont accessibles, quelle place est la sienne. C'est grâce au module d'admission que l'utilisateur peut rentrer effectivement dans le laboratoire. Ce module établit donc le lien entre la partie graphique et la partie plus invisible du laboratoire ; il fait également le bilan des sessions en cours et des sessions passées, donne accès aux travaux déjà réalisés. Par ailleurs, il fait le lien avec le module de sécurité.

Module Sécurité : C'est ce module qui permet la reconnaissance de l'utilisateur. Grâce à lui, on peut assurer la confidentialité des différentes applications et données (il faut que tout utilisateur puisse être sûr de son travail et que personne n'y aura accès). Il faut aussi qu'il y ait une grande confidentialité lorsqu'un groupe rend son travail au Coordonnateur. Relié directement au module d'admission, c'est le module de sécurité qui gère plus spécifiquement les requêtes d'identification. Ce module a une influence directe sur la QoS (il peut entraîner des regains de temps par exemple) ce qui justifie son lien avec la partie QoS dans l'architecture.

Interface de Coopération : C'est le module central permettant de mettre en place une coopération et, par la suite, une collaboration efficace. Tout en assurant ce niveau de collaboration, il est aussi utilisé pour gérer la QoS (sûreté, vitesse, définition de l'image, débit...) et les contraintes qu'elle induit (en responsabilisant notamment l'utilisateur qui choisira le niveau de QoS souhaité selon les applications) ainsi que la mise en groupe des étudiants.

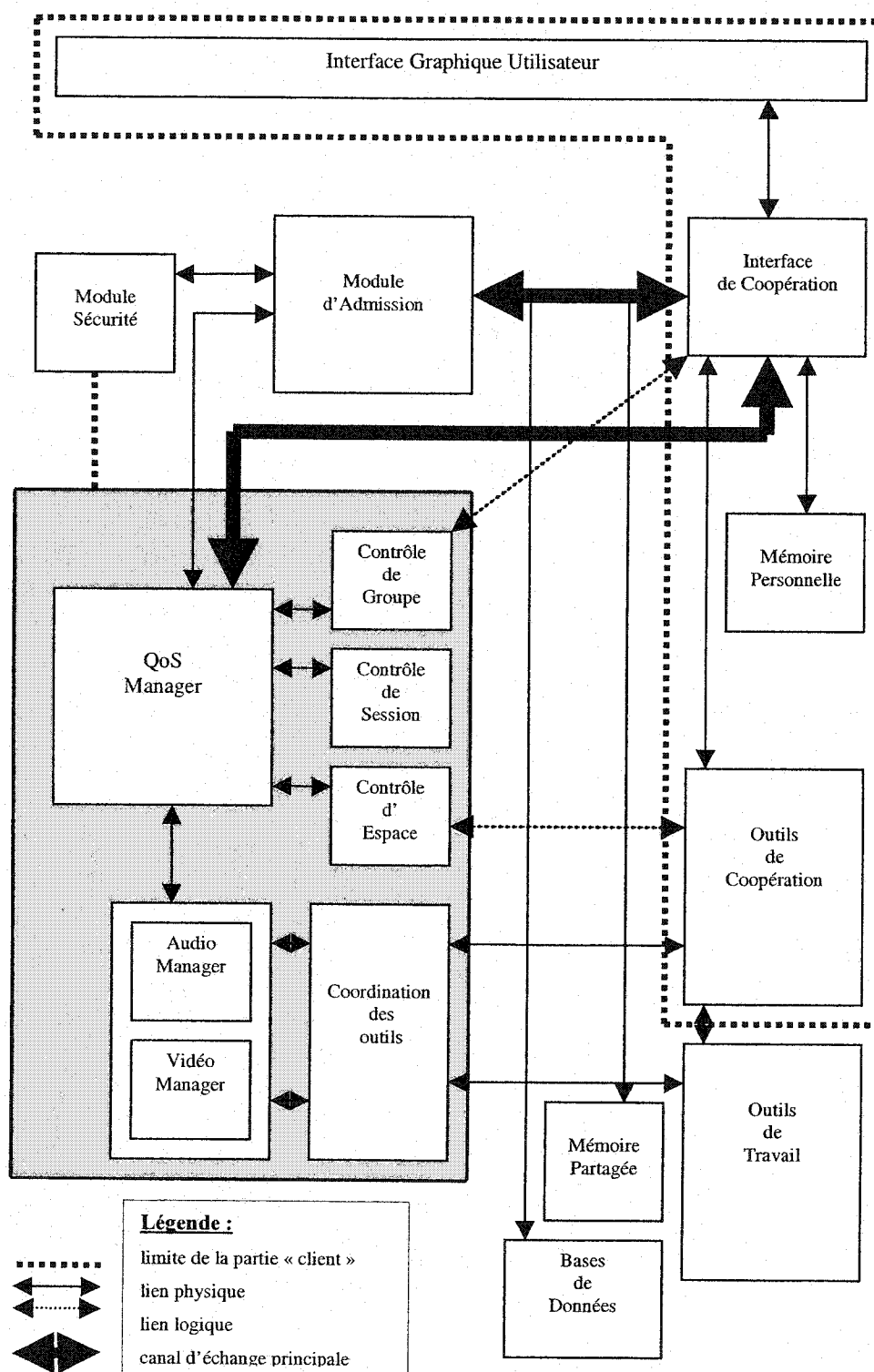


Figure 3.16 Architecture globale de collaboration

Il fait le point sur la session en cours, les problèmes à résoudre, les utilisateurs connectés, les possibilités de groupes, les outils à utiliser et permet donc de choisir la QoS. On peut aussi envisager qu'il y ait des formes différentes pour les élèves et le coordonnateur. C'est le module qui a le plus de lien avec les autres modules car c'est à partir de lui que l'utilisateur joue sur la QoS, en exprimant ses souhaits au *QoS manager*. Il peut avoir accès aux outils de coopération en sélectionnant ceux qui sont les plus utiles pour lui suivant son laboratoire. C'est lui aussi qui va donner la possibilité de gérer la mémoire personnelle. C'est grâce à ce module l'utilisateur exerce son contrôle sur l'ensemble du processus.

QoS Manager : C'est le module de base de la « partie QoS ». Il est chargé de faire le lien entre l'utilisateur (par l'intermédiaire de l'Interface de Coopération) et les différentes composantes de QoS (Session, Espace, Groupe). En effet, il est chargé de collecter les informations transmises par chacun sur l'état de la plate-forme et du trafic pour tous ces paramètres afin d'en informer l'utilisateur. Mais l'interaction se fait aussi dans l'autre sens puisque c'est lui qui traduit les souhaits de l'utilisateur pour la session en cours.

Contrôle de Session : On a défini au cours de ce chapitre la coordination comme le niveau de transition entre les deux principales classes d'interaction que sont la coopération et la communication. C'est à partir du module de Contrôle de Session que la coordination peut être respectée. En effet le module de contrôle de session s'occupe de la définition des rôles (Coordonnateur, Elèves, leader) et prérogatives de chacun. Il faut aussi veiller à ce que cela en soit ainsi tout au long de la session et qu'il n'y ait pas de modifications intempestives.

Contrôle d'Espace : Au niveau de la communication et de l'espace de travail, il existe diverses options. Pour éviter un mélange ou une confusion dans les applications et le travail, il est essentiel de gérer tout cela. Il faut par exemple contrôler qui peut utiliser

une application donnée à un instant donné. C'est justement à ce niveau que le contrôle d'espace intervient. C'est lui qui attribue les ordres de transmission de priorité et veille à ce que tout soit respecté.

Gestion de Groupes : Alors que le module précédent était plutôt consacré à la dimension temporelle (niveau structurel) de la communication, celui-ci permet de gérer la dimension quantitative. En effet, tout l'intérêt de la collaboration repose sur la possibilité du travail à plusieurs. Il faut donc assurer cette fonctionnalité et ce, grâce à ce module qui est chargé de répertorier les groupes. C'est aussi grâce à, lui que la notion de leader est supportée.

Audio Manager : Il est chargé d'assurer les transferts efficaces des flots audio dans le laboratoire. Il permet de déterminer si on peut les transmettre, quand, à qui et avec quelle qualité.

Vidéo Manager : Il a les mêmes prérogatives, au niveau vidéo, que le module précédent. Il a cependant une importance toute particulière, car c'est à ce niveau que toute la différence peut se faire par rapport à une plate-forme non collaborative. C'est pour cela que son efficacité doit être maximale.

Outils de Travail : Les outils de travail sont ceux qui ont plus trait à l'expérimentation. On entend par cela les outils spécifiques à chaque laboratoire : oscilloscope, ohm-mètre, calculatrice, etc.

Outils de Coopération : C'est l'ensemble des outils décrits au paragraphe précédent. Cependant, contrairement aux précédents, ils n'ont rien de spécifiques car ils sont mis en place uniquement pour la collaboration. Dans ces outils nous retiendrons donc : le curseur de contrôle, la pile et la liste de questions, le « browser », le « chat », le « chat

audio » et enfin l'application de visioconférence. Tous ces outils sont des applications génériques choisies en fonction des programmeurs du laboratoire.

Coordination des Outils : Encore une fois, par la présence de ce module, toute l'importance de la coordination est soulignée à ce niveau. En effet, pour qu'il n'y ait pas d'interaction stérile, il faut une unité capable de synchroniser l'ensemble et d'assurer une utilisation en parallèle de tous les outils.

Mémoire partagée : La mémoire partagée est la mémoire commune de l'ensemble des intervenants du laboratoire. C'est dans celle-ci par exemple que le professeur pourra stocker l'ensemble de son forum de discussion dans lequel les élèves viendront tirer les réponses dont ils ont besoin. C'est aussi dans cet espace qu'on aura les mémoires de groupes lorsque les élèves travailleront sur des projets communs. Elle doit donc être en relation avec la mémoire personnelle.

Mémoire Personnelle : Ce module permet à chacun de stocker l'ensemble des données dont il peut avoir besoin dans le futur. C'est aussi ici que seront gardées toutes les informations personnelles.

Bases de Données : Les bases de données permettent aux élèves d'avoir accès à un ensemble très étendu d'informations sur des expérimentations déjà existantes suivant les applications souhaitées, mais c'est aussi là que se trouveront les cours et encyclopédies mis à la disposition des intervenants. Contrairement à la mémoire partagée, elle ne présente pas la possibilité de renouvellement dynamique et d'interactivité.

3.4 Préoccupations de qualité de service

Dans l'architecture globale à la Figure 3.16, nous avons pu noter qu'une partie des modules était plus spécifiquement consacrée au contrôle de la QoS. Ce point est relativement important car il représente l'élément central de distinction de notre recherche à propos du laboratoire virtuel et du modèle d'architecture d'espace collaboratif. En effet, en développant ce modèle architectural, nous devons nous attacher à observer et respecter scrupuleusement des contraintes précises de QoS. Revenons d'abord rapidement sur la définition de la Qualité de Service.

3.4.1 Définition

Lorsque nous parlons de QoS, nous faisons référence à la mesure de la réaction du réseau aux variations de trafic et l'efficacité de ce réseau à gérer l'ensemble des ressources. Cela correspond à une tentative de définition et de quantification des caractéristiques et propriétés des services spécifiques offerts par le réseau. Les indicateurs les plus souvent utilisés de Qualité de Service sont : la gigue, le délai, la probabilité d'erreurs, le débit, etc. Une telle classification demeure très générale et ne peut donc convenir pleinement à notre problème.

3.4.2 La QoS dans le laboratoire virtuel

Il est d'abord utile de faire remarquer que les systèmes collaboratifs synchrones ont des contraintes très importantes par rapport à d'autres applications plus communes. Pourtant si on considère la plus simple d'entre elles (la navigation sur le web) [13] force est de constater que les exigences sont déjà importantes. Beaucoup d'utilisateurs ont déjà expérimenté des navigations pour lesquelles le chargement des pages s'éternisait. Or, vu comme cela peut être pénible, il est facile d'imaginer le désagrément causé par des phénomènes semblables lorsqu'en plus il y a collaboration entre différentes personnes. Il nous revient alors de répondre aux questions suivantes : Comment définir une QoS

cohérente pour les environnements d'apprentissage à distance ? Et comment l'appliquer à l'environnement d'apprentissage constituant les laboratoires virtuels ?

L'un des éléments de réponse le plus pertinent à mon avis, est qu'il faut se baser avant tout sur la subjectivité de l'utilisateur. Il nous incombe donc d'insister sur la qualité de contenu, la qualité pédagogique et enfin la qualité des transmissions au niveau des médias. Les deux premiers points sont vérifiés en amont au moment de la conception, partie propre à chaque domaine d'étude. Les concepteurs sont chargés d'assurer un contenu le plus complet possible répondant pour le mieux aux objectifs pédagogiques. Des experts vérifient alors que la qualité de l'enseignement est conforme à ce que l'on peut attendre dans un tel cours. Au niveau des médias, il faut se tourner d'avantage vers le côté technologique du laboratoire. En effet, au niveau de la conception, il faut prévoir tous les modules susceptibles de prendre en charge chaque partie de la QoS. Par ailleurs, il faut aussi veiller à ce que le système mis en place puisse répondre en temps réel aux modifications de trafic et s'adapter aux caractéristiques particulières de chaque situation rencontrée. Ainsi, en plus des éléments de QoS communs cités dans les paragraphes précédents, il faut aussi tenir compte des solutions envisagées plus spécifiquement pour les problèmes suivants : engorgement du réseau, synchronisation des outils, mise à jour, stockage des données si mise en ligne, etc. Il faut donc préciser un peu plus l'élément de réponse apporté jusque là en se concentrant toujours sur le confort de l'utilisateur (l'interface de coopération est chargée d'assumer ce rôle en familiarisant l'utilisateur avec cette question).

Par ailleurs, les applications n'ont pas toutes les mêmes exigences. Typiquement, le « chat » a moins de besoin en bande passante que les vidéos en temps réel. Il faut donc veiller à distinguer chaque classe d'application (manager vidéo, manager audio, différentes classes d'outils). Enfin, le dernier élément intéressant provient de la grande interactivité de nos applications, ce qui implique donc que nous prenions en compte la prédominance du canal de retour et du délai.

3.4.3 Solutions retenues

Voyons donc les critères qui nous semblent intéressant pour notre problématique et ce que nous retenons. Pour les applications audio et vidéo qui se rapportent plus spécifiquement à la visioconférence, nous effectuerons les tests et analyses pour :

- la gigue ;
- les liaisons sans coupures ;
- le flot régulier ;
- les conversations sans échos.

En ce qui concerne les outils comme le cahier de laboratoire, nous pourrons nous consacrer à la mesure de la cohérence et des pertes. Enfin, en ce qui concerne les communications à proprement parler, nous nous intéresserons aux considérations quantitatives et temporelles définies à la section 3.2. Ce problème doit être pris en charge par les modules de gestion des groupes et gestion de diffusion de l'architecture globale.

CHAPITRE IV

IMPLÉMENTATION ET RÉSULTATS

Comme tout modèle conceptuel, l'architecture de collaboration proposée pour la plate-forme d'apprentissage permet d'étendre la portée des laboratoires virtuels. La prochaine étape dans le processus de déploiement consiste pour nous à implémenter un « outil » supportant la collaboration. Ainsi, nous devrions parvenir à intégrer ces modules sur la plate-forme existante. Le grand défi que pose la conception réside dans la définition du *QoS Manager*, qui nous permettra de jauger le modèle établi jusque là. Dans cette perspective, ce chapitre permet de présenter le travail fait en exposant la démarche suivie. Nous y présentons tout d'abord le travail effectué pour réaliser l'interface de coopération, élément central de notre architecture. Par la suite, nous nous tournerons vers la simulation qui est nécessaire pour l'évaluation du système et notamment en ce qui concerne les processus de QoS. Finalement, nous analyserons les résultats de nos simulations à la lumière des objectifs que nous nous étions fixés.

4.1 Détails d'implémentation

Pour effectuer cette implémentation, le langage le plus approprié était définitivement Java. En effet, il est parfaitement adapté au développement d'interface graphique, ce qui constitue la première étape de notre travail. Par ailleurs, et ce n'est pas négligeable, le reste des travaux réalisés jusque là dans le projet du laboratoire l'a été en Java. Il est donc normal, pour assurer la continuité et la pérennité de ce projet, de choisir une solution permettant l'intégration de ce travail à l'ensemble déjà existant. Ceci est la base de notre raisonnement de conception et le point de départ pour le travail à effectuer. Cependant, cela est bien insuffisant et il faut encore spécifier bien des points pour compléter notre analyse.

4.1.1 Spécifications matérielles et logicielles

Dans le chapitre précédent, nous avons introduit de nombreux concepts faisant référence à des outils, des applications et des processus liés à l'informatique. Tous ces éléments qui composent la base de notre architecture sont à définir à la lumière des objectifs que nous imposent les besoins d'apprentissage, ainsi que des contraintes inhérentes aux technologies utilisées pour les mettre en place. Que ce soit pour les outils de coopération, qui proviennent de projets antérieurs par exemple ou pour les protocoles à utiliser, il nous incombe d'en faire l'inventaire et les spécifications nécessaires à la bonne compréhension de la recherche en cours. L'ensemble des outils génériques a été conçu (et sera complété par la suite) de manière à garder une certaine compatibilité, quel que soit le laboratoire sur lequel cette interface sera implantée. En effet, la console des outils de base est une Applet Java [23] qui implémente pour le moment quatre outils : un chronomètre, un cahier, une calculatrice et un « chat », tous réalisés lors de projets antérieurs. C'est à nous maintenant de le compléter avec notamment l'interface de coopération qui aura pour rôle celui d'une nouvelle « console d'outils » mais de manière totalement générique et transposable à tous les laboratoires.

L'adaptation de tous ces outils au contexte spécifique du laboratoire se fait lors du téléchargement de l'Applet. En d'autres termes, de par le caractère générique que nous voulons donner à notre projet, nous nous assurons que tout laboratoire possède une interface d'appel propre, susceptible de réaliser cette mission, et qui comprend ce genre d'Applet Java [24]. Ceci symbolise donc l'implantation de la palette d'outils utile à la coopération, et donc à la collaboration par extension et caractérise aussi les fonctionnalités qui sont associées à ces outils. Ainsi, l'utilisateur aura à sa disposition un ensemble d'outils tout usage, sans avoir la perception des différences d'implantation de deux outils quelconques. A cet effet, les outils ont été conçus à l'aide de paramètres à accès public, peu importe le langage d'implémentation. Le réglage de ces outils est confié à l'interface qui devra s'adapter à n'importe quel laboratoire spécifique. A ce propos, l'expérience de programmation, ainsi que les préalables inhérents à ce projet favorise le langage Java qui a déjà été utilisé maintes fois dans des cas similaires.

4.1.2 Environnement de réseau et protocoles

L'implémentation correspond à la création de l'interface au niveau informatique. Ce niveau d'abstraction ne comprend pas la définition des protocoles nécessaires à la réalisation de toutes les tâches imputées à une telle interface. Cette étape du développement global n'est pourtant pas négligeable. C'est grâce à une définition correcte que tout ce qui est censé être implanté pourra marcher lorsque l'ensemble sera terminé. Ainsi, pour rendre le projet réalisable et exploitable par tous, il faut s'attacher à trouver les meilleurs protocoles.

L'un des objectifs principaux de ce projet, de par sa définition même, est de rendre le laboratoire accessible au plus grand nombre. Cette plate-forme n'aura un intérêt réel que si elle est disponible partout et en tout temps (évidemment ce cas de figure est optimal). Il apparaît donc logique de partir de l'hypothèse de base que le support de ce laboratoire sera l'Internet. Par conséquent, le protocole approprié qui s'impose est l'Internet Protocol (IP). Par ailleurs, et c'est à ce niveau que le travail de conception et de définition réalisé en aval était primordial, la collaboration implique bon nombre d'autres contraintes à prendre en compte. Il faut donc considérer les spécificités que nous avons introduites et définies au chapitre précédent. Citons donc celles qui ont une influence à ce niveau, à savoir :

- communications « multicast » ;
- utilisation fréquente du temps réel ;
- gestion de la QoS.

Le protocole IPv6 (IP version 6 par opposition à la version 4) s'impose à ce niveau [25]. Il est en passe de devenir le support principal de l'Internet grâce à ses nombreux atouts et toutes les fonctionnalités bien adaptées aux nouvelles technologies. En effet, son format étendu d'entête est prévu spécifiquement pour gérer le « multicast », la QoS et les applications en temps réel. Voilà pourquoi nous avons porté notre choix sur ce protocole pour le laboratoire. Cependant, il faut émettre une réserve à cette description idyllique. Dans les faits, IPv6 n'est pas complètement opérationnel, ou du

moins pas encore. Il est toujours en voie de développement à l'heure actuelle. Nous ne pourrions donc pas l'utiliser effectivement. Ainsi, l'implémentation et l'expérimentation se feront sur IPv4 seulement. Nous conjecturons donc que dans l'absolue le recours à IPv6 est parfaitement approprié à notre projet mais, pour des raisons techniques, nous ne pourrions y avoir recours. En outre, si les tests sont concluants avec IPv4, nous pourrions penser qu'il le sera d'autant plus avec IPv6.

Puisque cette question est réglée, passons au niveau de la couche supérieure du modèle OSI, à savoir la couche transport. A ce niveau, notre rôle est d'autant plus important que les possibilités sont plus nombreuses. Les choix que nous effectuerons auront donc une incidence directe sur les performances de notre plate-forme. Revenons à la partie qui nous intéresse spécifiquement, à savoir la collaboration et plus précisément la communication. En effet, il nous reste à détailler le niveau technologique de cette classe d'interaction. Selon les outils utilisés, il faut choisir le protocole adéquat. En effet, chaque outil a ses spécificités (charge, temps, bande passante nécessaire...). Il est donc préférable d'avoir recours à une différenciation. Ainsi, bien que TCP soit très fiable et relativement efficace, il ne conviendra pas à tous les outils. Sa fiabilité est un avantage certain pour le « chat » et surtout le cahier. Pour ce dernier, des pertes seraient catastrophiques et remettraient tout le processus de collaboration en cause. Pourtant, si cette fiabilité a ses avantages, elle a aussi un « coût ». En effet, pour l'assurer, le protocole a recours à des messages d'« acknowledgement » qui ont une forte tendance à congestionner le trafic. Par ailleurs, ces échanges de messages de confirmation sont trop pénalisants pour le temps réel. Ils induisent des retransmissions qui n'auront plus leur place dans les flots et, avec le multimédia, le nombre de paquets est très important. Dans ce cas, le recours au protocole UDP sera donc préférable. En outre, même si ce protocole n'exclut pas des pertes de paquets, nous privilégierons la régularité et la continuité dans la transmission. Ainsi, pour la visioconférence ou le « chat-audio », nous choisirons UDP. Tous ces choix sont illustrés à la Figure 4.1.

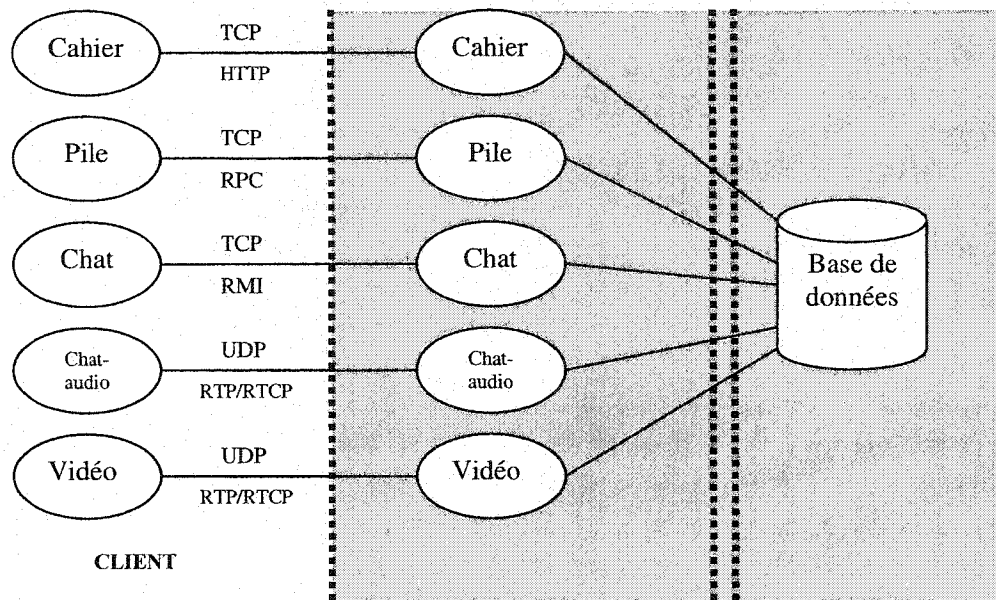


Figure 4.1 Les protocoles des couches transport et application

4.1.3 Implémentation de la QoS

L'autre aspect important de ce projet de recherche concerne comme nous l'avons vu précédemment la qualité de service qui est spécifiquement pris en charge par le *QoS Manager* de l'architecture conceptuelle. Nous avons établi précédemment que l'utilisateur doit être en mesure de choisir le niveau de service adapté à l'utilisation qu'il souhaite faire de la plate-forme. Ainsi, et puisqu'il nous est impossible d'implémenter l'architecture dans son ensemble, nous nous concentrerons sur cette partie : la possibilité de fixer une classe de service grâce au *JRadioButton* prévus à cet effet. Tachons donc maintenant de comprendre quelles sont les options possibles pour gérer cette fonctionnalité.

Dans le choix d'une solution acceptable pour le *QoS Manager*, il faut tenir compte des possibilités technologiques que nous offre le réseau considéré. En effet, il serait mal venu de proposer des solutions qui ne seront pas applicables et déployables le plus largement possible. A la lecture de nos recommandations, nous pourrions penser

que l'ATM serait parfaitement approprié (des classes de service y sont déjà prédéfinies). Cependant, cette technologie n'est pas assez répandue et il apparaît même plutôt que IP prend le dessus définitivement. Par conséquent, une solution tenant compte d'ATM ne serait pas pertinente, et la solution envisagée doit être adaptée à cet état de fait. Notre choix s'est donc plutôt porté sur les normes proposées par l'Internet Engineering Task Force (IETF) : modèle d'architecture Integrated Service (*IntServ*) ou Differentiated Service (*DiffServ*). Avec ces alternatives, l'IETF a cherché à répondre efficacement au problème de la QoS dans l'Internet qui devient primordial de nos jours. Ainsi, il est donc ressorti ces deux modèles particuliers. Nous allons rappeler les principes de base de ces architectures, car par la suite nos simulations porterons dessus. En effet, il nous faut déterminer la méthode la plus adaptée à notre problème en fonction des résultats obtenus. De cette manière, nous chercherons à obtenir le *QoS Manager* le plus efficace possible. Présentons donc succinctement le fonctionnement du modèle de différenciation de services.

Architecture DiffServ

Le modèle DiffServ [26] [27] de QoS IP permet, à l'instar de ATM, de différencier des classes de service. Cela peut donc permettre à l'utilisateur de spécifier des « contrats » pour certains niveaux à choisir dans une palette de possibilités prédéfinies : *Best effort*, *excellent effort*, *Standard*, *interactive voice*, *interactive video*, etc. Pour chaque classe certaines spécificités sont prévues [28] : délai moyen, pertes minimales, etc. Le type de service est indiqué explicitement à l'intérieur de chaque paquet IP [29]. La sélection peut être faite par le client ou les « nœuds frontières » (border nodes). Au niveau des nœuds intérieurs au réseau (les nœuds du cœur), il y a l'interprétation qui est réalisée pour fournir les différents types de services. Cette sélection qui peut s'effectuer par deux entités différentes implique deux solutions pour notre QoS Manager. En effet, on peut envisager les deux manières, à savoir traitement au niveau du client ou envoi d'un message d'avertissement au nœud frontière qui sera chargé d'appliquer le choix. Le traitement différencié des flux de paquets se fait en

fonction du code d'accès aux services différenciés (DSCP) [28], lequel se trouve dans le champ de bits « Type of Service » (TOS) de l'entête IP, comme cela est illustré à la Figure 4.2. Le champ DSCP indique le traitement que doit suivre le paquet.

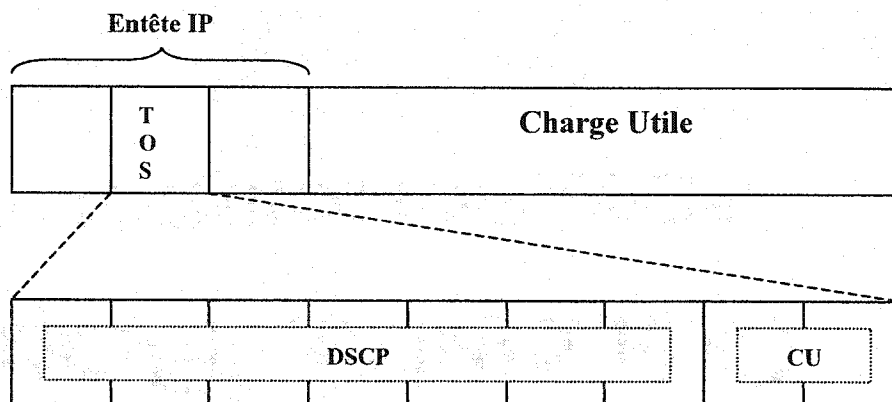


Figure 4.2 Champ DSCP

Ensuite, en fonction de ces codes, les routeurs DiffServ adoptent des comportements de commutation de proche en proche (PHB) adaptés. Ces PHB sont par exemple l'« Expedited Forwarding » (EF) pour un service assurant un faible ratio de perte de paquet, un faible délai, et une faible gigue, ou l'« Assured Forwarding » (AF) pour lequel on dénombre quatre classes AF qui obtiennent des quantités de ressources différentes dans les routeurs du « backbone ». Cependant, c'est aussi en cela que réside une partie du problème qui nous amènera à opter pour la simulation. En effet, le réseau doit être en mesure de supporter ces interprétations, notamment avec des routeurs spécialisés. Par conséquent, ce que nous retiendrons et que nous mettrons en application lors des simulations, sera la différenciation des services opérée grâce au champ du TOS. En effet, cette fonctionnalité est offerte par le simulateur que nous envisageons d'utiliser, à savoir OPNET (Optimum NETwork performance). En outre, pour les solutions envisageables, c'est aussi celle qui correspond le mieux à nos attentes. Comme nous l'avons vu, nous envisageons deux manières de procéder éventuelles. Selon la première, on fixe le TOS au niveau de l'utilisateur, car c'est lui qui en cochant le *JCheckBox* adéquat

déclenche le processus qui met les bits appropriés dans le TOS. L'autre possibilité serait que, par le choix du *JChecBox*, l'utilisateur envoie un message au routeur qui prend en compte cette requête et applique le traitement souhaité à tout le flux suivant, tant que le choix n'est pas changé. A priori, cette solution serait privilégiée par la suite. Cependant, grâce à l'outil de simulation que nous utiliserons, nous allons d'abord pouvoir comparer ce modèle et ses performances à l'architecture *IntServ* par exemple. Voyons donc succinctement les fondements de ce modèle maintenant.

Architecture IntServ

Dans ce cas ci, et à l'inverse du modèle *DiffServ* où le problème de la QoS est traité au niveau de la couche trois (OSI), le traitement est réglé au niveau supérieur. Chaque flot de la couche quatre va informer le réseau de ses besoins en QoS et celui-ci devra répondre en conséquence [31]. C'est à ce dernier que revient la responsabilité de décider s'il peut traiter ou non les flux de manière spécifique. Dans ce modèle, il existe aussi plusieurs sortes de classes de service différentes. Il s'agit ici du *service garanti* et du *service contrôlé*. On définit ces deux classes qui s'appuient sur une particularité de l'*IntServ*, à savoir le protocole RSVP (Ressource reservation Protocol) qui permet l'établissement de flux avec garanties de QoS dans les réseaux datagrames. Par ailleurs, il est aussi configuré pour pouvoir gérer le « multicast », ce qui serait éventuellement intéressant pour notre projet.

A la vue de ces éléments, nous retiendrons donc qu'il existe plusieurs possibilités pour implémenter un service de QoS pour notre *QoS Manager*. Cependant, de par les possibilités au niveau de la QoS qu'elle offre, nous nous intéresserons par la suite à une architecture particulière, le modèle *DiffServ*, en tentant d'évaluer au mieux ses possibilités dans le cadre de notre recherche. Mais, nous prendrons bien soin de comparer cette technologie aux autres, afin d'obtenir la meilleure solution possible.

4.2 Mise en oeuvre de l'environnement

La première étape de notre travail consiste donc à implémenter les éléments de notre architecture afin d'en évaluer les performances. En effet, il faut valider la démarche suivie et pouvoir montrer qu'elle apporte une contribution effective au projet initial. L'implémentation réalisée pour l'interface de coopération ainsi que le reste du laboratoire, a été réalisée dans un environnement de programmation Java avec *Jbuilder*. Cependant, avant de voir le résultat de l'implémentation à proprement parler, commençons par faire un point de ce que réalise exactement la plate-forme. Rappelons le déroulement du scénario que rencontre un utilisateur qui vient faire une session de laboratoire. Nous pourrons alors voir les programmes plus en détails et le travail réalisé.

4.2.1 Scénario d'utilisation

Voyons donc en détails la démarche que fait un apprenant. Ce résumé nous permettra de comprendre les besoins pour la plate-forme et donc les choix réalisés. Ainsi il commence à se connecter à Internet pour pouvoir accéder au site. Une fois cette connexion réalisée, l'utilisateur demande la possibilité d'ouvrir une session. Le *module d'admission* qui était sollicité doit donc contacter le *module de sécurité* chargé de l'identification. Il fait donc la demande d'authentification. Si celle-ci est acceptée, alors le laboratoire pourra éventuellement lancer *l'interface de coopération*, dans le cas où l'utilisateur en fait la demande. A partir de ce moment, ce dernier a accès à toutes les fonctionnalités : outils, mémoires, module de QoS, etc. L'ensemble des modules fait le point sur les caractéristiques du trafic (quels flots, quelle QoS, quels groupes, quels rôles, ...) et de la session. Toutes ces informations sont traitées par le *QoS Manager* qui envoie toutes ces données à *l'interface de coopération*. L'apprenant est donc amené à faire un état des lieux. Il peut faire les choix appropriés pour le laboratoire, ses partenaires, les outils dont il a besoin et le niveau de QoS souhaité. Il transmet au *QoS Manager* ses souhaits dans ce domaine, qui lui même transfère ces données aux modules concernés (gestion de groupes, de sessions et d'espace). Chacun de ces

modules sera chargé de faire respecter l'ordre pour la partie qui lui est échue lors du reste de la session. Il faut ensuite éventuellement que tout soit recalculé s'il y a des changements pendant la séance (nouvel entrant, apprenant qui sort, etc.). Ensuite, une fois la session entamée, et en fonction de l'utilisation qui est faite des outils, les *vidéo* et *audio manager* seront sollicités pour accorder l'ensemble.

4.2.2 La plate-forme

Afin de rendre le projet plus concret, il est important de quitter le niveau conceptuel. C'est dans cette optique, que nous avons implémenté avant toute autre chose ce que nous qualifierons de « coquille informatique » de l'architecture. Ainsi, grâce à l'environnement de programmation Java *Jbuilder* [23], nous avons pu donner corps à ce qui avait été proposé au chapitre précédent. Il nous fallait montrer plus précisément à quoi correspondait tous les modules et la place qu'ils occupent dans la plate-forme. Comme nous l'avons déjà précisé précédemment, l'Interface de coopération est au centre de cet ensemble. Il est donc normal que tout passe par elle. Pour avoir accès à tous les modules, la première étape est cette interface. C'est ainsi qu'on choisit les outils par exemple. Il est aussi possible de sélectionner le niveau de QoS grâce à elle. Ainsi nous avons développé l'interface, puis, nous l'avons intégrée à la partie déjà existante à l'aide du menu du laboratoire.

a) L'interface de coopération

L'interface de coopération sera contenue dans un *Jframe* accessible à partir du *Jmenu* déjà développé dans le projet du laboratoire virtuel. Nous voyons ce *Jmenu* qui est illustré à la Figure 4.3. Ce menu offre la possibilité d'accéder aux différentes laboratoires et outils à l'aide des onglets prévus à cet effet.

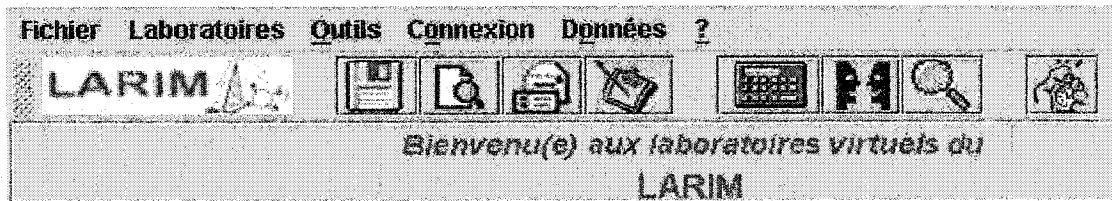


Figure 4.3 Le JMenu du laboratoire

Le *Jframe* global devra contenir quatre sous-parties, chacune consacrée à une tâche bien spécifique : bilan de session, gestion des outils, gestion des groupes et gestion de la QoS. Grâce à la fonction permettant le placement voulu disponible dans la bibliothèque Java, nous pouvons obtenir l'organisation souhaitée. Ainsi, le *BorderGridLayout* nous permet d'obtenir la « géographie » illustrée à la Figure 4.4. Afin de mieux comprendre exactement le rôle de cette interface précisons un peu ce que sont toutes ces zones, au niveau fonctionnel mais aussi informatique. En effet, il est intéressant de comprendre le rôle de chacune ainsi que la méthode de programmation utilisée pour les obtenir.

Bilan de session : Cette première zone sert de résumé pour l'utilisateur. En effet, il doit pouvoir y trouver toutes les informations générales concernant la session en cours. Il est assez intéressant par exemple de se voir présenter toutes les informations à propos de celle-ci : nom du laboratoire, présence et nombre de groupes, connaissances requises, manipulations proposées, avancement dans le processus, etc. Afin de réaliser cette fonction, le choix le plus simple permis par le langage *Java*, choix que nous avons retenu, est le *JTextField* non éditable.

Gestion des outils : Grâce à cette partie, l'utilisateur se voit offrir la possibilité de sélectionner les outils de collaboration (ainsi que ceux de travail) qu'il juge utile pour la séance qu'il doit accomplir. Nous avons donc décidé de placer à l'intérieur d'un *JPanel*, un groupement de *JCheckBox* qui permet de faire la sélection appropriée. En ajoutant un

eventListener associé à un *Jbutton*, la méthode de choix est très simple. Ainsi, après avoir coché tous les outils désirés, l'utilisateur a juste à appuyer sur le bouton qui lancera l'outil (ou plusieurs si nécessaire). Il est à noter qu'à tout moment dans la séance, l'utilisateur peut fermer un outil ou en ouvrir un autre s'il le désire. Le déclenchement d'un outil est illustré à la Figure 4.5 avec l'ouverture du « chat ».

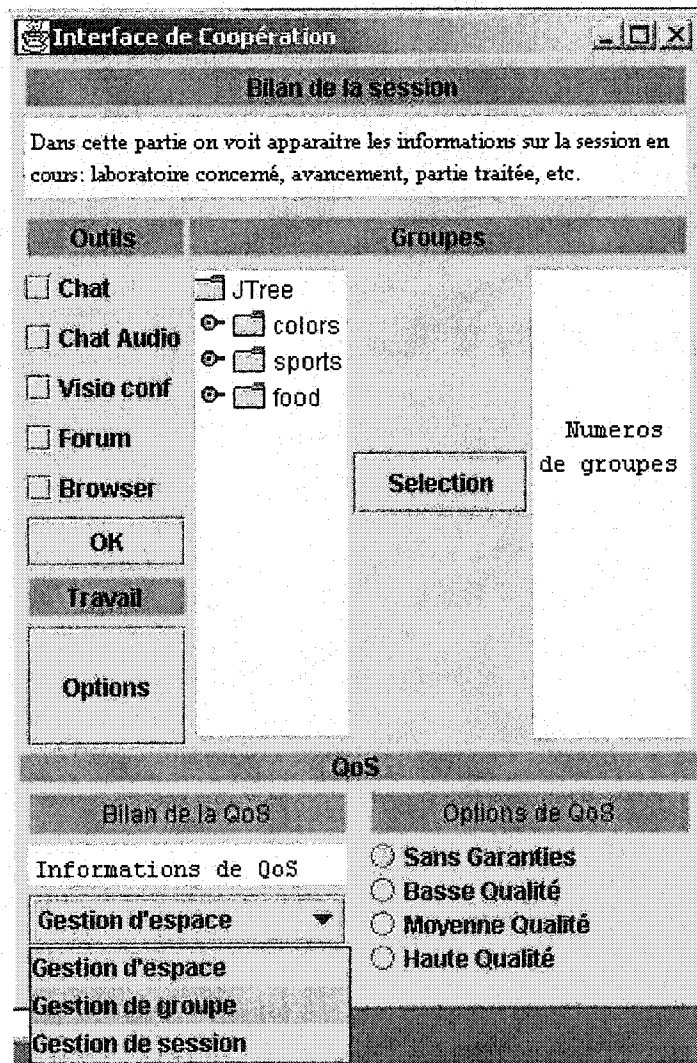


Figure 4.4 Interface de coopération

Gestion de groupes : Grâce à ce panneau l'utilisateur peut avoir accès à tout ce qui concerne les groupes. En effet, il est important pour lui de voir qui sont les personnes qui font le même laboratoire que lui au même moment. Ainsi, il peut décider de les contacter et de lancer éventuellement le processus de collaboration. En plaçant deux *JTree* côte à côte et séparés par un *Jbutton*, on peut sélectionner les partenaires de groupes et les mettre dans son groupe. Ainsi, tous les groupes formés pour un laboratoire apparaissent dans le *JTree* placé dans la zone *EST* (*BorderGridLayout*).

Gestion de QoS : Dans ce dernier *JPanel*, l'utilisateur se voit offrir la possibilité de gérer la QoS. En fait, si bien sûr il ne joue pas complètement dessus, cette partie lui fait simplement prendre conscience de ce que cela représente. Ainsi, avec un groupement de *JRadioButton* (« best effort », moyenne qualité, haute qualité) relativement simple, il doit choisir la qualité qu'il voudra pour l'adapter aux applications dont il a besoin. Par défaut, ce champ est fixé au « best effort » traditionnel de l'Internet.

Nous avons donc fait le tour de cette *interface de coopération* en voyant tous ses composants. En programmant ce *Jframe*, parallèlement à tout ce qui avait déjà été fait pour le laboratoire virtuel, le travail obtenu est illustré à la Figure 4.4 quand on sélectionne *collaboration* sur le *JMenu* de base.

b) Intégration des parties existantes

Le deuxième travail de la partie consacrée à l'implémentation consiste à intégrer tout ou partie des outils des autres composants du laboratoire. Il nous a fallu bien évidemment commencer par rendre l'interface de coopération disponible à partir de la plate-forme de base. Ensuite, le travail consistait à faire la même chose pour les outils. Le premier est le « chat » développé par un des membres de l'équipe de projet. Ainsi, en implémentant la fonction nécessaire, on peut tout à fait le lancer en cochant la boîte adéquate des *jcheckbox*. Ceci est illustré à la Figure 4.5. Nous devons ensuite nous tourner vers la visioconférence. L'outil choisi à priori est celui de visioconférence développé actuellement au LARIM dans l'une des composantes du laboratoire.

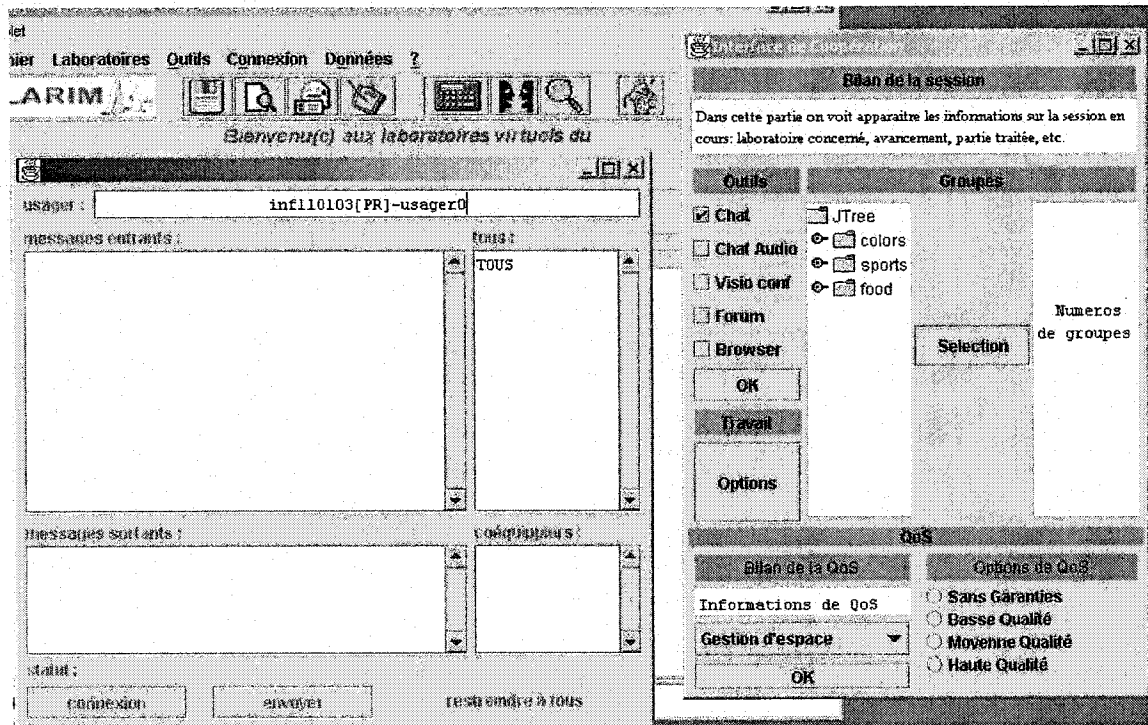


Figure 4.5 L'Interface de Coopération et le « chat »

c) Le module de sécurité

Dans les versions successives du laboratoire virtuel, il existait déjà des modèles de module consacré à la sécurité et l'identification. Cependant, afin de donner une idée exacte de la forme de l'architecture, nous avons implémenté un module de sécurité à part. Il est accessible à partir du *JMenu* général du laboratoire. En programmant ce module, on peut faire ressortir le processus complet décrit dans le scénario vu dans la sous-section précédente. Pour le réaliser, nous avons utilisé un *JPanel* très simple qui est activé quand on choisit l'option connexion (qui est la seule disponible au départ) du *JMenu*. Le résultat est illustré à la Figure 4.6.

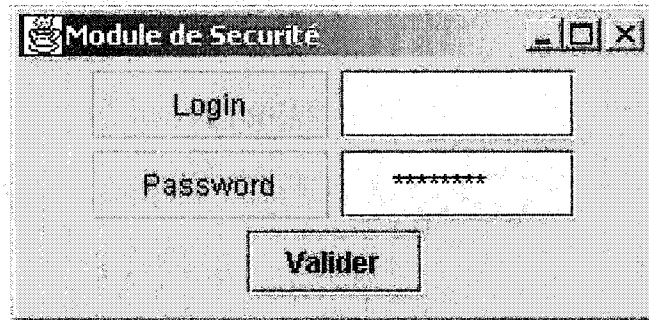


Figure 4.6 Module de sécurité

4.3 Simulation et résultats

L'exposé mené jusqu'à présent fait ressortir une contrainte d'ordre technique pour la réalisation du modèle d'architecture. L'implémentation, telle que nous l'avons réalisée, est en effet limitée car le module permettant de gérer la différenciation des classes de service ne pourra être testé. Le recours au modèle *DiffServ*, bien que très intéressant au niveau possibilités s'accompagne pourtant d'une restriction. En effet, il est impossible de le tester à notre échelle. Pour pouvoir mener des tests concluants sur la performance de l'infrastructure, il faut que tout le réseau supporte les spécifications du modèle. Ainsi, si nous envisagions par exemple d'obtenir plusieurs classes de service spécifiques en fixant le champ du DSCP de l'entête IP au code du niveau désiré (cela est tout à fait réalisable), encore faut-il que les différents nœuds du réseau soient capables de les interpréter. Malheureusement, les réseaux à notre disposition ne nous permettent pas d'observer ce phénomène dans son ensemble et sur la globalité du réseau. Cela nous empêcherait donc de tester le modèle proposé et par conséquent de le valider. Nous avons décidé de nous tourner vers une solution alternative, la simulation.

Il est séduisant d'envisager un outil nous permettant d'observer la réponse du réseau en fonction de la charge du trafic, des applications utilisées, du nombre d'utilisateur et surtout du niveau de QoS choisi. En effet, il nous faut tester au moins deux sortes différentes de « contrats » pour la QoS que l'utilisateur est susceptible de

choisir. Il nous est donc demandé de tester s'il y a une amélioration des paramètres d'évaluation lorsque nous passons de l'un à l'autre. Il faut donc définir totalement une démarche de simulation pour appréhender ces difficultés. Afin de bien comprendre notre cheminement, voyons le cadre exact de la simulation, telle que nous l'avons envisagée.

Le simulateur utilisé est OPNET (Optimum Network Performance) [32]. Il nous permet d'effectuer tous les tests désirés sur une topologie de réseau bien particulière. Ainsi, nous allons analyser la différence de traitement des flux selon la classe de service sélectionnée, pour des réseaux normaux supportant *DiffServ*. De cette façon, nous obtiendrons une cohérence bien utile dans les résultats, car nous pourrons mener tous les tests sur la même architecture. En fixant la topologie une fois pour toute, il ne nous reste plus par la suite qu'à définir le trafic, à le dupliquer et à le faire transiter sur le réseau.

4.3.1 Définition du modèle

La première étape de la simulation consiste bien évidemment à définir et détailler les paramètres de l'étude. Nous savons déjà que, pour évaluer la pertinence de nos choix, nous utiliserons les mesures concernant : le délai, la gigue et les pertes. Cependant, ce n'est pas tout ce qu'il faut connaître pour compléter cette étude. Nous devons aussi déterminer les éléments du support, censé modéliser au mieux l'architecture réelle d'une session de laboratoire. Pour cela, nous avons défini notre réseau ainsi que les applications et le trafic.

a) Le réseau

Commençons donc par sélectionner la topologie appropriée [33]. Au cours de cette étude nous « privilégions » un utilisateur. Cela ne signifie pas qu'il sera traité de façon particulière pour les performances offertes, mais simplement c'est sur lui que se focaliseront les mesures. Il sera baptisé *Utilisateur* à la Figure 4.7. Nous mettrons aussi un apprenant *Témoin* à ses côtés afin de comparer tous les résultats avec une mesure étalon. Afin de modéliser au mieux le laboratoire virtuel, sachant que nous l'avons pris comme hypothèse au chapitre précédent, nous utiliserons une architecture client-serveur.

Ainsi, l'un des bouts du réseau est un serveur chargé de traiter tous les flux envoyés par les différents utilisateurs. Il y a donc les deux bouts qu'il faut maintenant relier par une topologie modélisant notamment l'Internet (*Internet*). Pour cela, nous utilisons l'*IP Cloud* de OPNET, sur lequel les routeurs d'accès sont branchés (ce sont eux qui sont chargés de l'interprétation des classes de service aux frontières du réseau dans la réalité), *Routeur1* et *Routeur3*. Ainsi, nous avons créé le canal prioritaire qui forme la base de l'étude. Cependant, étant donné que nous devons évaluer les performances du modèle au niveau QoS, il est évident que la congestion est un facteur critique que l'on doit prendre en compte. Nous devons donc rajouter d'autres utilisateurs (*GroupeUtilisateurs*) qui modéliseront le nombre d'utilisateurs souhaité. Pour cela, nous utilisons les LAN 10B-T du simulateur OPNET qui permettent de dimensionner le trafic à notre volonté. Nous le relierons ensuite au nuage Internet à l'aide d'un routeur d'accès, *Routeur2*. Ainsi, la topologie obtenue est censée coller le plus possible à la réalité, et elle est illustrée à la Figure 4.7.

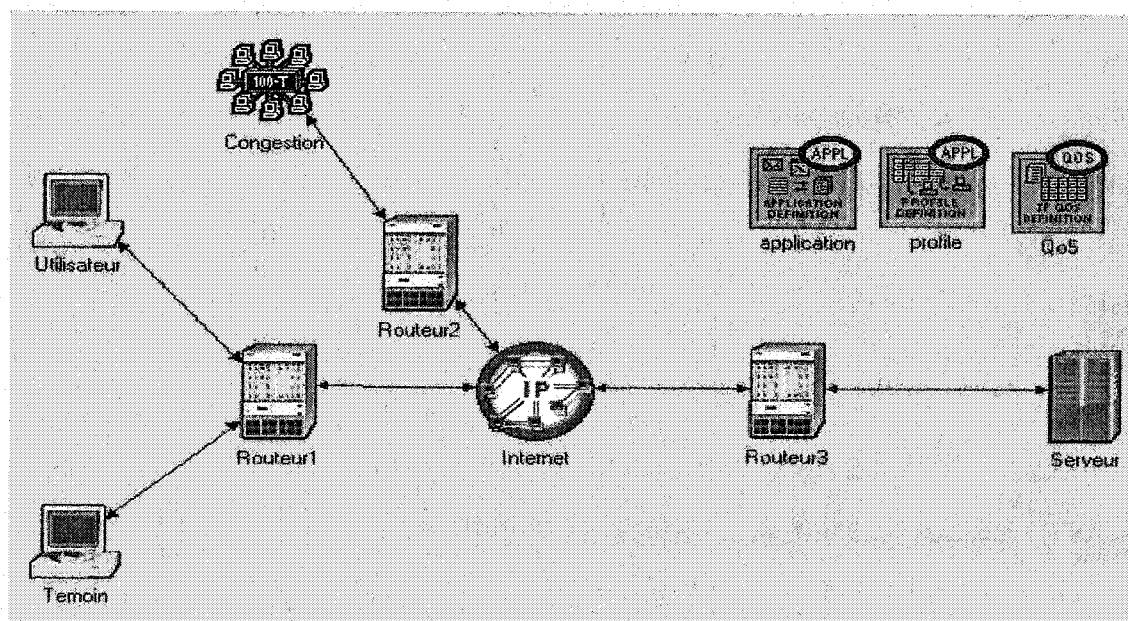


Figure 4.7 Le réseau de simulation

L'ensemble des nœuds du réseau est relié par des liaisons point à point basiques permettant de représenter un réseau quelconque. Puisque la topologie générale est totalement définie, il nous reste maintenant à définir le trafic.

b) Le trafic

Définissons maintenant le trafic [34] qui sera véhiculé sur le réseau présenté précédemment. Nous ne modélisons pas le trafic exact présent dans le laboratoire. Cependant, nous nous attachons à coller le plus possible à la réalité. Pour cela, nous avons basé nos modèles sur une étude de *trace file* trouvés sur Internet [35]. Puis, nous ne prenons que les applications les plus caractéristiques retenues pour la coopération. Ainsi, nous avons recours à la vidéo, la voix et le transfert de fichiers HTTP. La première étape consiste alors à les définir en détails.

HTTP : Grâce à cette application, nous prenons en compte les recherches sur le web (apprentissage du cours) et les échanges par « chat ». Cette application correspond à des transferts de données http classiques. Il faut faire attention à envoyer une première page ne contenant que l'entête puis les autres peuvent suivre, paramètre fixé dans *l'Application Configuration*. La configuration la plus simple est illustrée à la Figure 4.8 où nous pouvons observer les tableaux de configuration ainsi que la forme basique du signal observé.

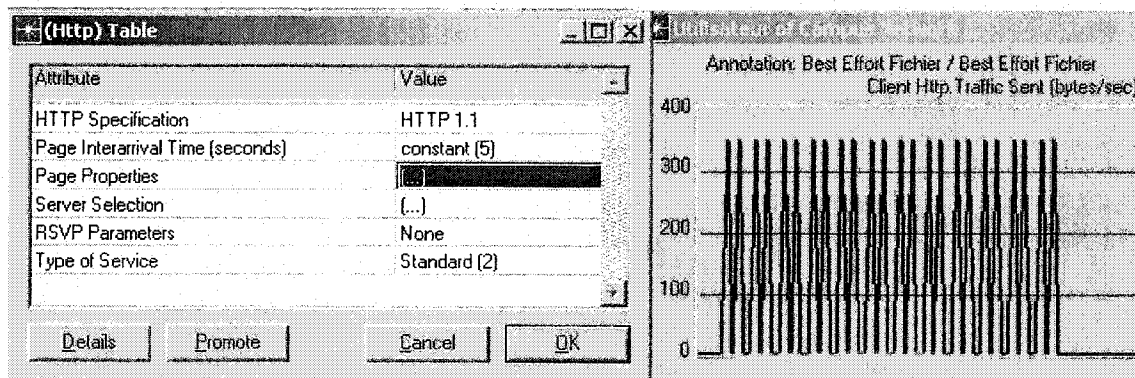


Figure 4.8 L'application HTTP

Nous réglerons le temps de transfert quand nous préciserons les paramètres de simulation. Pour cette application, nous fixons deux niveaux différents de QoS (*Best Effort Fichier* et *Standard Fichier*) basés sur le champ TOS.

Voix : Cette application sert à modéliser le « chat audio ». Pour celle-ci, nous avons recours au modèle standard qui stipule les paramètres suivants : temps de silence, temps entre les silences ou encore codage utilisé. Nous pouvons visualiser les réglages et la forme du signal à la Figure 4.9.

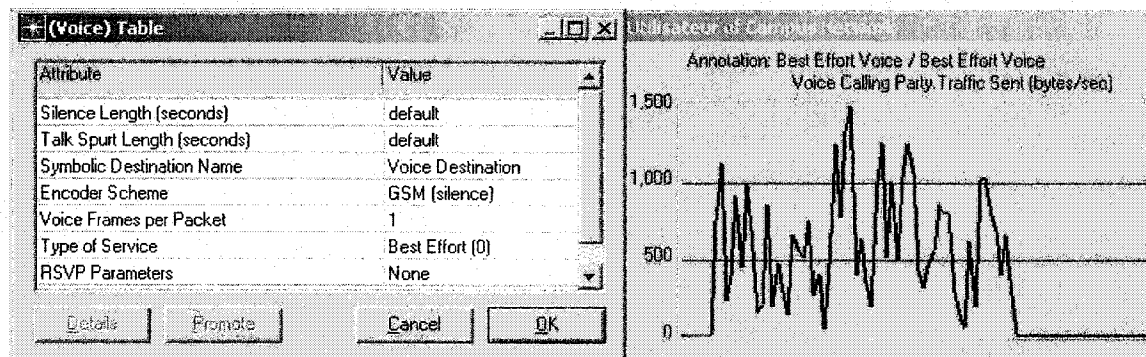


Figure 4.9 L'application de voix

Comme précédemment, les durées d'émission sont fixées lors de la simulation. Par ailleurs, nous utilisons dans ce cas des niveaux de TOS différents (*Interactive Voice* et *Best Effort Voice*).

Vidéo : la dernière application utile pour notre modèle est la visioconférence. Il faut ici préciser la taille des *frames* ainsi que les temps *interframes*. Les paramètres choisis sont ceux trouvés sur Internet et correspondant à des applications classiques de visioconférence. Tout cela est illustré à la Figure 4.10. Pour la vidéo, nous prendrons aussi deux TOS différents (*Interactive Video* et *Best Effort Video*) afin de former les différents contrats.

Nous avons donc fini de définir les applications que nous utiliserons dans cette simulation. La dernière étape concerne la QoS. En effet, il faut que le réseau soit adapté à ce traitement. Pour cela, il faut régler dans le *QoS Configuration* le format de « queuing » désiré et supportant les TOS. Ainsi, tout est prêt pour lancer la simulation. Il ne nous restera plus qu' à préciser les profils exacts des applications, lors de l'exposé du plan d'expérience.

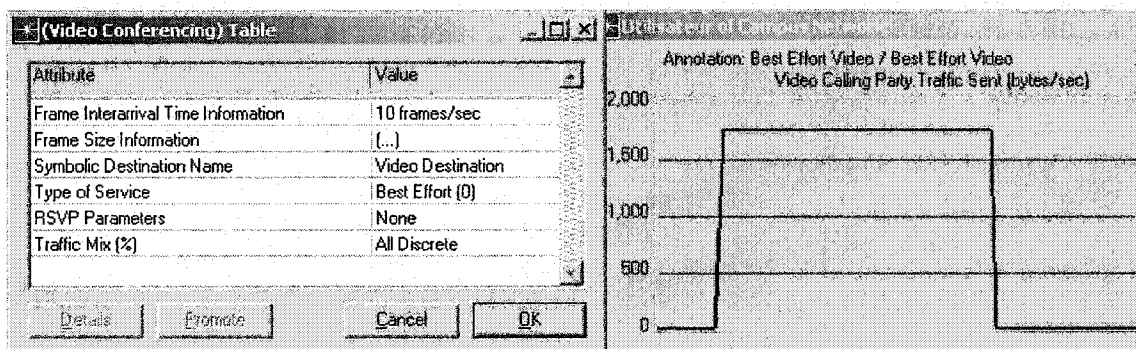


Figure 4.10 La visioconférence

4.3.2 Plan d'expérience

Cette section commence par élaborer les métriques utilisées pour l'évaluation de performance. Elle présente ensuite le modèle de charge pour les applications et discute des conditions réseau. Elle se termine en présentant le scénario d'analyse suivi durant les expériences.

a) Métriques

Comme nous l'avons précisé précédemment, nous utilisons trois métriques de performance dans notre évaluation, que nous comparerons pour chaque utilisation modélisée :

- le *délai bout à bout* mesure l'évolution de la réponse transmise à l'utilisateur ;
- la *gigue* mesure la variation de ce temps de réponse durant l'ensemble de la session ;

- la *perte de paquets* permet de quantifier les transmissions qui échouent.

Étant donné un nombre N de participants et une sélection particulière du contrat de service par l'*utilisateur*, deux principaux facteurs influencent ces métriques :

- le nombre N de participants à la session ;
- le contrat de service requis à choisir parmi les trois possibles : *videoprio*, *voiceprio* et *fichierprio*.

Le choix de ces différents contrats a représenté l'une des premières difficultés de la phase de simulation au niveau de la préparation. Il s'agissait en effet, de définir des regroupements permettant une différenciation efficace des flux. A priori, il pourrait sembler logique de vouloir mettre une qualité de service maximale pour toutes les applications et ce, dans tous les cas (*Interactive multimedia*, *Interactive voice*, *Standard fichier*). Cependant, les premiers tests menés dans cette direction n'ont pas été concluants. En mettant un *Utilisateur* avec un tel niveau, face à un *Temoin* avec des classes qualifiées de faibles (tout à Best Effort), nous avons constaté qu'il n'y avait aucune régularité dans les résultats. Nous avons en effet obtenu des résultats dissemblables en réalisant l'expérience plusieurs fois avec un *seed* différent (option de OPNET permettant de changer aléatoirement les calculs effectués dans une simulation). À Chaque fois, seule une application (*fichier*, *voice* ou *video*) était privilégiée sans que ce ne soit toujours la même. Il nous était donc impossible de tirer des conclusions pertinentes. Étant parti de l'hypothèse que le niveau de service à choisir devait l'être en fonction des applications, nous avons décidé de faire trois regroupements distincts qui se focalisent chaque fois sur une application particulière. Aussi, nous avons pris :

videoprio= (*Interactive Video*, *Best Effort Voice*, *Best Effort Fichier*) ;

voiceprio= (*Best Effort Video*, *Interactive Voice*, *Best Effort Fichier*) ;

fichierprio= (*Best Effort Video*, *Best Effort Voice*, *Standard Fichier*) ;

etalon= (*Best Effort Video*, *Best Effort Voice*, *Best Effort Fichier*).

Nous faisons alors une étude comparative de ces regroupements et de leur utilisation en deux volets :

- l'évaluation des métriques est réalisée de manière primaire pour obtenir un étalon et ce, sans appliquer trop de trafic au réseau ;
- l'évaluation des métriques est ensuite mesurée en fonction du nombre de participants. Nous considérons quatre *GroupeUtilisateurs* différentes correspondants à 10, 25, 50 et 100 utilisateurs dans le nombre de « workstation » du LAN *GroupeUtilisateurs*.

Ainsi, pour l'ensemble de la simulation, il faudra réaliser un total de vingt sessions différentes (cependant peut être que celle correspondant au regroupement *etalon* est inutile suivant les résultats de la première mesure à ce niveau). Par ailleurs, en annexe nous présentons aussi les résultats de comparaison simple des différents niveaux pour chaque application.

b) Modèle de la charge

Si les applications ont été définies dans la sous-section précédente, il nous reste encore à voir exactement comment celles-ci seront agencées pour modéliser au mieux une session du laboratoire. Afin d'obtenir des temps de simulation raisonnable et de pouvoir obtenir par conséquent un nombre minimum de résultats pour mener la comparaison, nous modéliserons une session par trente minutes de trafic constant. Durant ces trente minutes, il est important d'avoir une succession pertinente des flots des différentes applications. Ainsi, en configurant dans le module *profile* les temps de déclenchement et durées d'exécution, nous obtenons le profil des trafics présenté à la Figure 4.11.

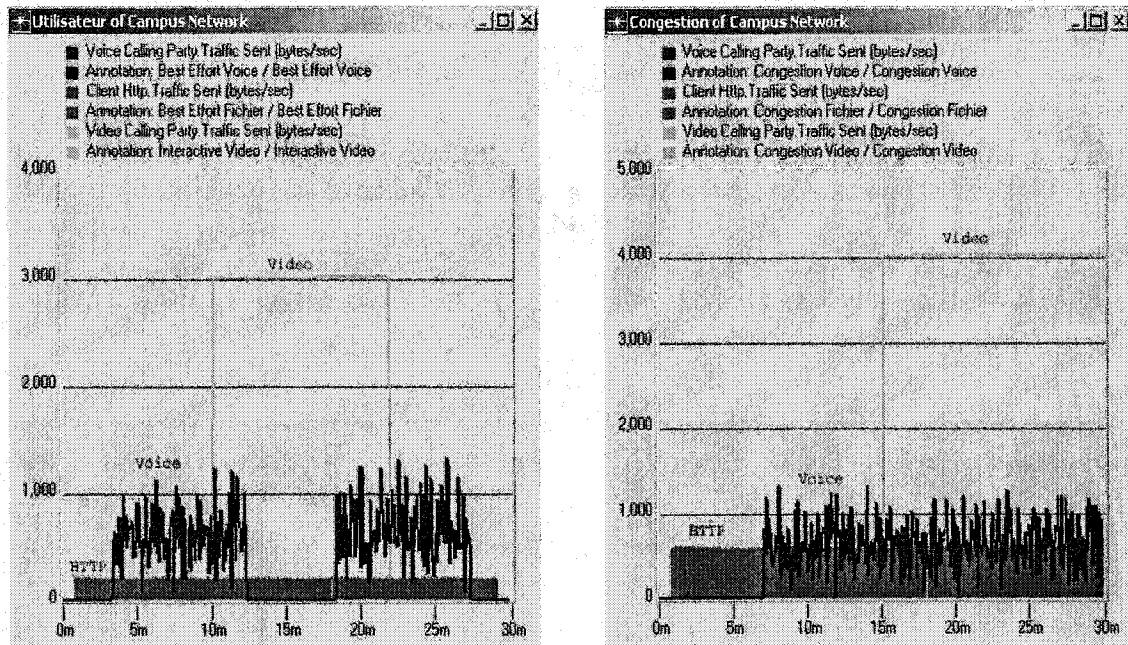


Figure 4.11 Profils des applications *utilisateur* et *GroupeUtilisateurs*

Ce profil est appliqué à l'*Utilisateur* ainsi qu'au *Temoin*. Pour le *GroupeUtilisateurs*, nous prendrons une succession simplifiée des trois applications. En effet, il y aura *Best Effort Fichier*, *Best Effort Voice* puis *Best Effort Video*, les uns après les autres, comme cela est représenté à la Figure 4.11.

c) Scénario

Il nous faut donc maintenant lancer les simulations en suivant un ordre logique et ordonné permettant une comparaison optimale des résultats obtenus sur les métriques choisies. Nous commençons donc par lancer les quatre scénarios (*Scénario_etalon*, *Scénario_videoprio*, *Scénario_voiceprio*, *Scénario_fichierprio*) dans leur configuration la plus simple. Le *Temoin* a le profil *etalon* et l'*Utilisateur* a les quatre autres. Cette première série de mesures devrait permettre de jauger tout de suite les différentes performances des contrats choisis et de voir s'ils sont pertinemment choisis. Dans un deuxième temps et pour chacun d'eux, nous referons les mesures en augmentant le

nombre d'utilisateurs. Ainsi, nous nous ferons une bonne idée de l'influence de la congestion sur la performance du modèle que nous proposons.

4.3.3 Analyse des résultats

La réalisation du plan d'expérience exposé précédemment a permis d'obtenir un certain nombre de résultats concernant la performance. Dans ce qui suit, nous analysons successivement les mesures de délai, de gigue et de pertes pour les trois applications que nous avons choisi d'étudier, à savoir : http, voix et vidéo. Pour les présenter, nous nous attachons notamment à souligner l'influence du niveau de service choisi et de la congestion dans le réseau.

a) L'application *fichier*

Que ce soit au niveau du temps de réponse par objets ou des pertes, les résultats sont relativement similaires et caractéristiques. La Figure 4.12 expose les résultats comparatifs pour deux applications identiques traitées avec et sans qualité de service. Celle nommée *Utilisateur* correspond à une application *Standard* alors que le *Temoin* est *Best Effort*. Il apparaît clairement que, dans le premier cas, le temps de réponse est amélioré de manière significative. Si les profils généraux sont identiques (influence légère de la *video* qui provoque un léger ralentissement de la réponse), les chiffres sont assez éloquentes. En effet, en appliquant le traitement standard on assure un gain de 20% en moyenne. Cette augmentation ne paraît pas extrêmement significative que ça (0,065 s), force est de constater que relativement elle n'est pas négligeable non plus et que par conséquent la différenciation semble relativement bénéfique pour l'utilisateur. En outre, cette tendance se confirme et devient même plus évidente lorsqu'on augmente la charge réseau en ajoutant du trafic *GroupeUtilisateurs*. La différence devient d'autant plus importante que le nombre d'utilisateurs augmente.

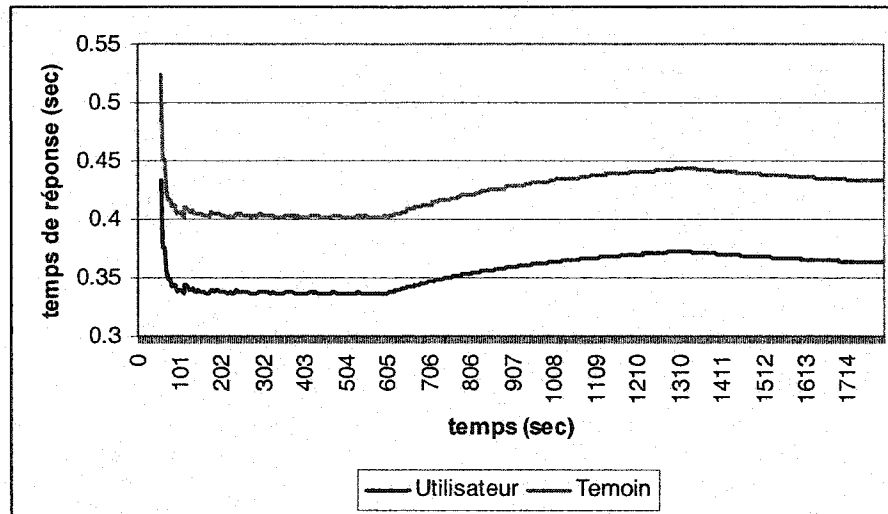


Figure 4.12 Comparaison du temps de réponse pour *Utilisateur* et *Témoin*

En effet, la Figure 4.13 laisse clairement apparaître que, si même pour l'Utilisateur il y a une dégradation des performances, elle reste toute relative comparée à celle observée chez le *Témoin*. Dans le premier cas, il y a au maximum une augmentation de 22% de la dégradation, alors que dans le cas *Témoin* cette augmentation est de l'ordre de 500 % (avant saturation qui plus est). Au niveau brut, les temps de réponses vont jusqu'à 3 secondes, et même 5 secondes à la saturation. En effet, il est important de noter que, dans le cas *Témoin*, l'introduction de *GroupeUtilisateurs video*, les performances se dégradent de telle manière que cela aboutit à de la saturation et une courbe de réponse totalement discontinue. Cela s'explique par le fait que, à partir de cet instant, nombre de requêtes http ne pourront plus être traitées. Ce phénomène est illustré à la Figure 4.14. La courbe *Analyse* correspond au scénario sans groupes d'utilisateurs supplémentaires

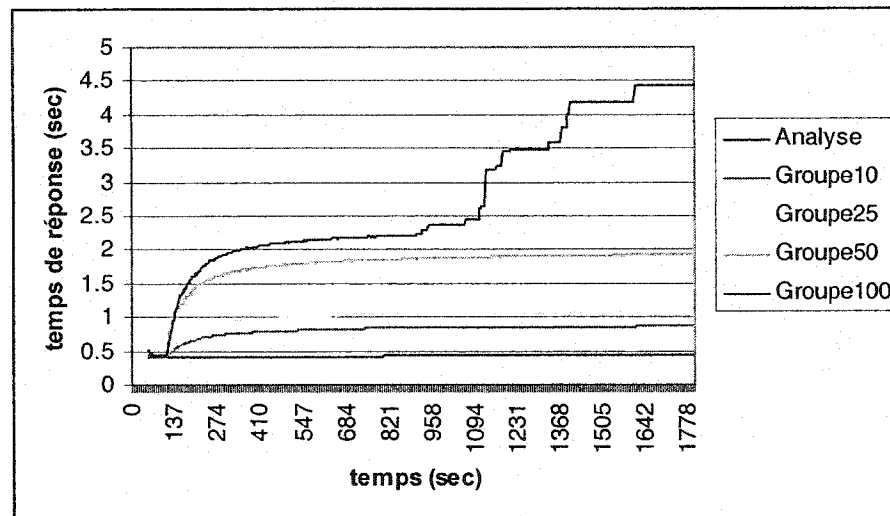


Figure 4.13 Temps de réponse en fonction de la Congestion : *Témoin*

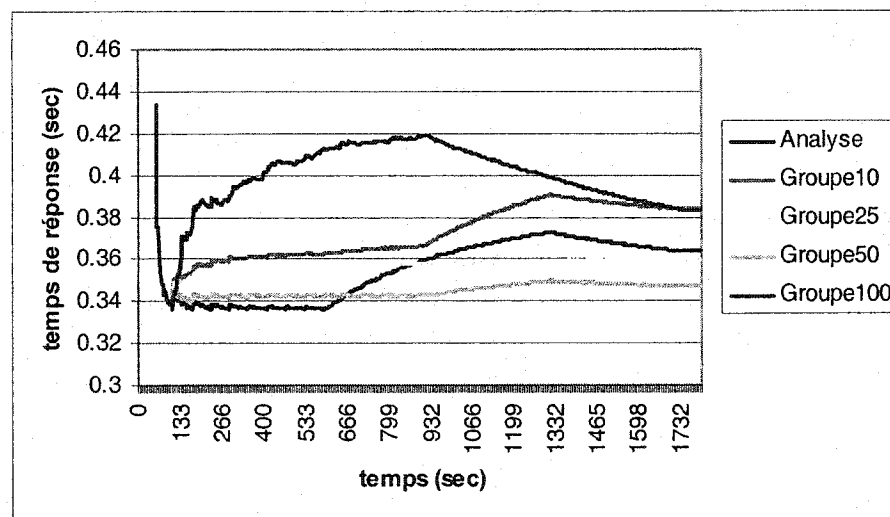


Figure 4.14 Temps de réponse en fonction de la congestion : *Utilisateur*

Il ressort donc que l'introduction de flux différenciés représente un avantage certain si on cherche à privilégier les applications http. Pour être complet, il nous apparaît important de préciser que, dans ce cas, les performances des applications *video* et *voice* sont comparables à celle des *Témoin* dans les cas se référant à l'étude de chacune de ces applications, que nous voyons dans la suite.

b) L'application *voice*

Voyons maintenant quels sont les changements provoqués au niveau de la voix grâce à une analyse des résultats correspondant au scénario *voice*. En effet, dans ce cas aussi, l'application d'un traitement particulier, en l'occurrence le recours à la classe de service *Interactive voice* a une incidence sur les performances des applications liées à la voix comme celle du « chat audio ». La première constatation est que, comme dans le cas du http, la comparaison du délai montre un écart entre les deux manières de traiter la voix. En effet, il est intéressant de voir, cela est illustré à la Figure 4.15, l'amélioration que cela induit même sans congestion. Le gain en terme de délai est de l'ordre de 2,3 %, ce qui est moins notable que dans le cas précédent. Cela correspond d'ailleurs à une amélioration de simplement 0,0025 seconde, ce qui peut apparaître négligeable. Cependant, l'analyse ne doit pas s'arrêter là. Voyons maintenant l'influence de la congestion. Pour cela, nous nous referons aux Figures 4.16 et 4.17 qui seront expliquées en partie, à la lumière de la Figure 4.18.

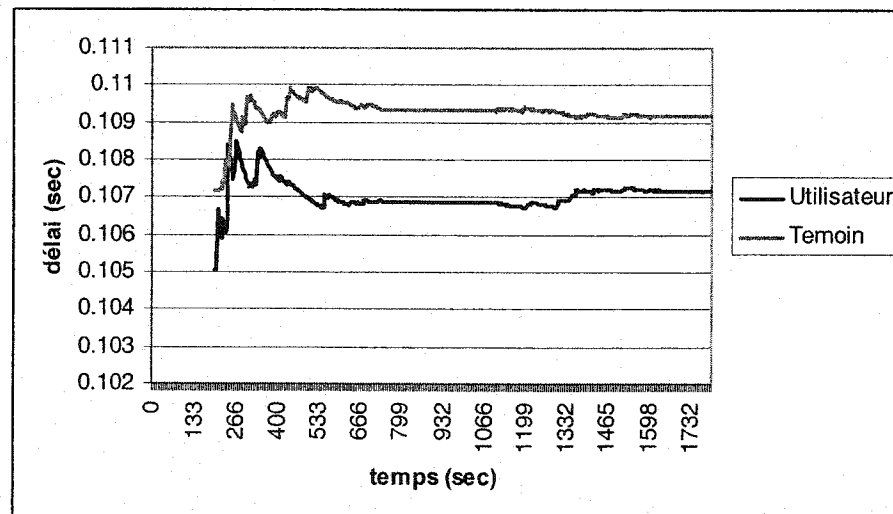


Figure 4.15 Comparaison du délai pour *Utilisateur* et *Témoin*

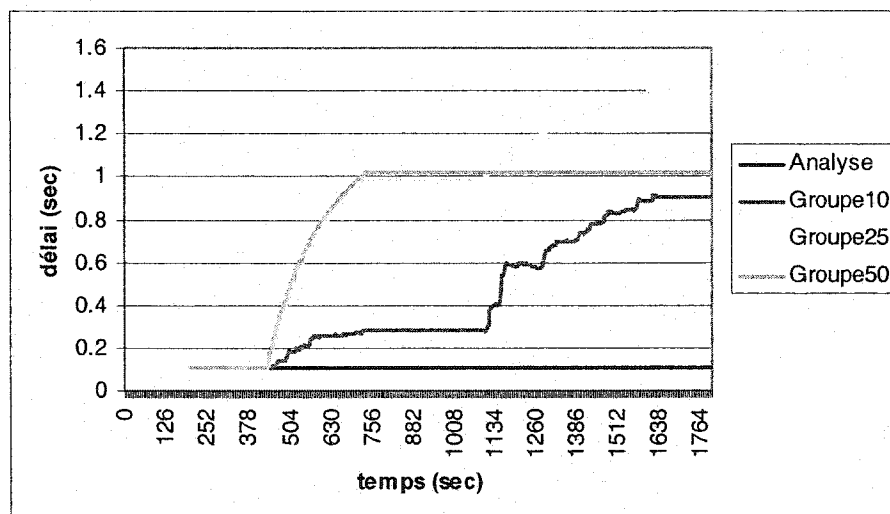


Figure 4.16 Délai en fonction de la Congestion : *Temoin*

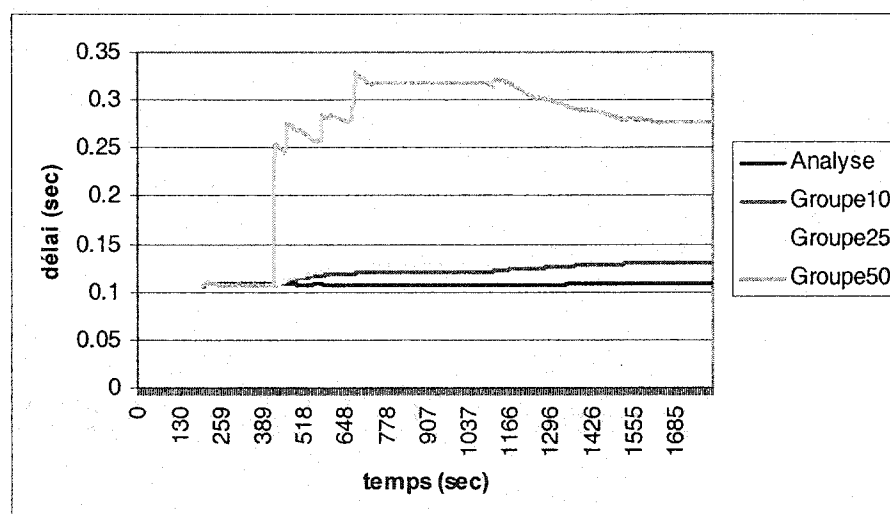


Figure 4.17 Délai en fonction de la Congestion : *Utilisateur*

Le premier fait marquant est le phénomène comparable à ce que nous avons déjà vu dans le cas précédent, à savoir une limitation de la dégradation pour l'*Utilisateur*, comparée à celle du *Temoin*, quand la Congestion augmente. En effet, si bien sûr il y a une dégradation, elle est limitée (de l'ordre de 195 % pour *GroupeUtilisateurs50*). Nous remarquons que *GroupeUtilisateurs100* n'apparaît pas car la saturation empêche toute

comparaison). Comme nous pouvons le constater, cela se traduit au niveau des courbes de la Figure 4.18 par une absence de paquets reçus à partir d'un certain niveau. En effet, à un seuil fixé, les pertes sont trop importantes, comme cela nous l'a d'ailleurs été confirmé par la fenêtre prévue à cet effet dans la console de simulation. Pour *GroupeUtilisateurs10* et *GroupeUtilisateurs25*, le phénomène est assez limité (12% et 18%). Cet ordre de grandeur est comparable pour la gigue.

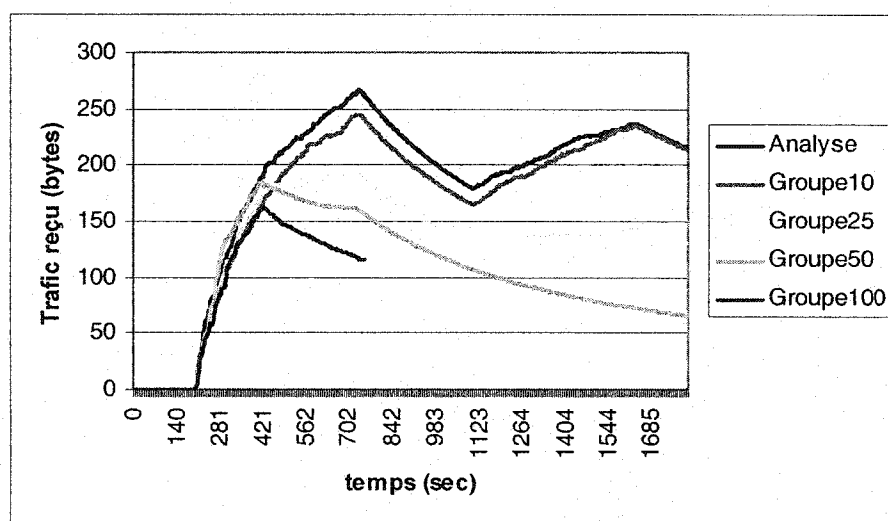


Figure 4.18 Pertes en fonction de la Congestion : Temoïn

Quand nous nous penchons sur le cas *Temoïn*, la tendance générale est similaire. Cependant, les proportions sont beaucoup plus importantes, alors que pour l'*Utilisateur* ce phénomène reste limité (de l'ordre de la demi seconde). Ici les valeurs sont tout autres, du fait notamment des pertes beaucoup plus importantes. Sur la Figure 4.18, nous nous apercevons qu'elles commencent à *GroupeUtilisateurs25*. Dans ce cas, il y a une saturation évidente pour *Congestion50* à partir de l'introduction de *GroupeUtilisateurs video*. Le nombre de paquets reçus diminue alors fortement. Dans ce cas, les dégradations relatives maximales sont de l'ordre de 740% et 1295%. Les conclusions sont donc identiques au cas précédent.

c) L'application *video*

Le dernier aspect de nos simulations touche à l'évaluation d'une différenciation des classes de service pour les applications vidéos. En effet, ces applications constituent le dernier volet de la collaboration et des outils de coopération utilisés pour la gérer. Notre analyse suivra le même plan que ce qui a été fait jusque là. Le premier point concerne en effet la comparaison en *Utilisateur* (Interactive multimédia) et le *Temoin* (Best Effort). Comme nous pouvons le constater à la Figure 4.19, la différence est évidente. Il y a une amélioration de l'ordre de 0,15 seconde, ce qui correspond à un gain relatif de 96%.

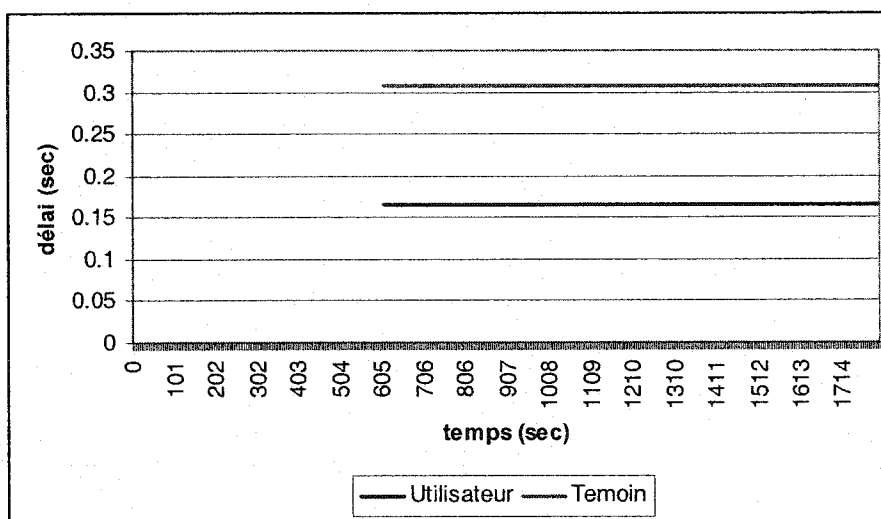


Figure 4.19 Comparaison du délai pour *Utilisateur* et *Temoin*

Voyons maintenant l'évolution en fonction de la congestion. Dans cette partie, en appuyant notre analyse sur une étude des Figures 4.20 et 4.21, nous observons le phénomène décrit pour la voix, à savoir l'augmentation du délai en fonction du nombre d'utilisateurs de manière moins conséquente pour l'*Utilisateur* que pour le *Temoin*. La particularité observée entre *GroupeUtilisateurs10* et *GroupeUtilisateurs100* (délai moins important dans le second cas) provient uniquement du fait que, dans le deuxième cas, à partir d'un certain seuil, il apparaît des pertes de plus en plus importantes en fonction de

la congestion. Cette interprétation s'appuie sur l'observation de la Figure 4.21 dans la quelle nous constatons une diminution du nombre de paquets reçus par l'expéditeur. L'analyse est d'ailleurs renforcée par les statistiques fournies par la console de simulation qui indique que, pour des raisons de congestion, certains paquets sont rejetés.

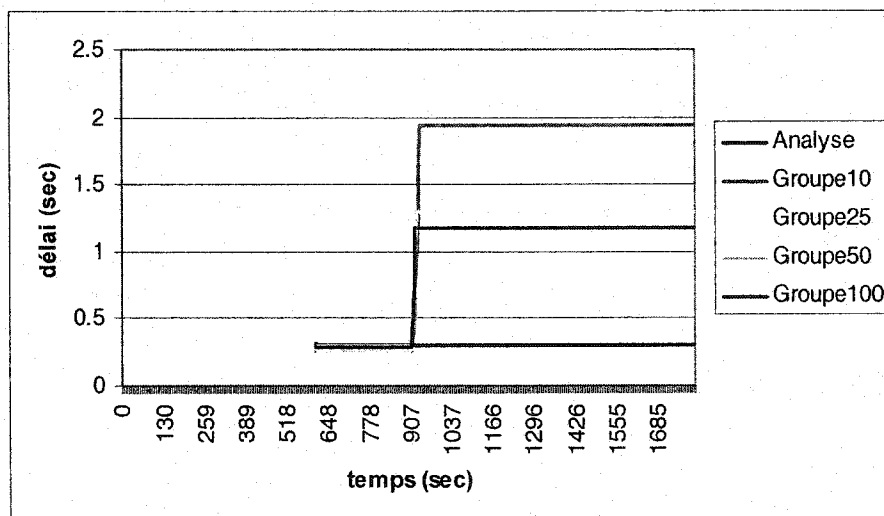


Figure 4.20 Délai en fonction de la Congestion : *Temoin*

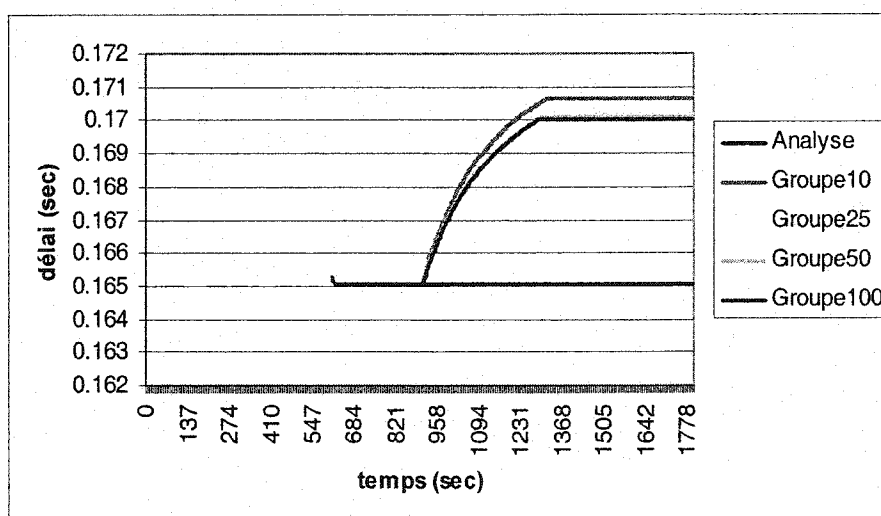


Figure 4.21 Délai en fonction de la Congestion : *Utilisateur*

Cependant, nous pouvons tirer certains chiffres significatifs à partir de l'analyse des courbes. Ainsi, pour l'*Utilisateur*, la dégradation est de l'ordre de 0,02 seconde soit 3%, tandis que pour le *Temoin* cela passe à 1,5 secondes soit 530% de dégradation. Encore une fois, il apparaît donc que l'application de la différentiation apporte un bénéfice dans les conditions normales d'utilisation, mais elle permet aussi de prévenir une augmentation trop importante des performances quand la charge dans le réseau devient trop conséquente. Au niveau de la gigue, les conclusions sont identiques.

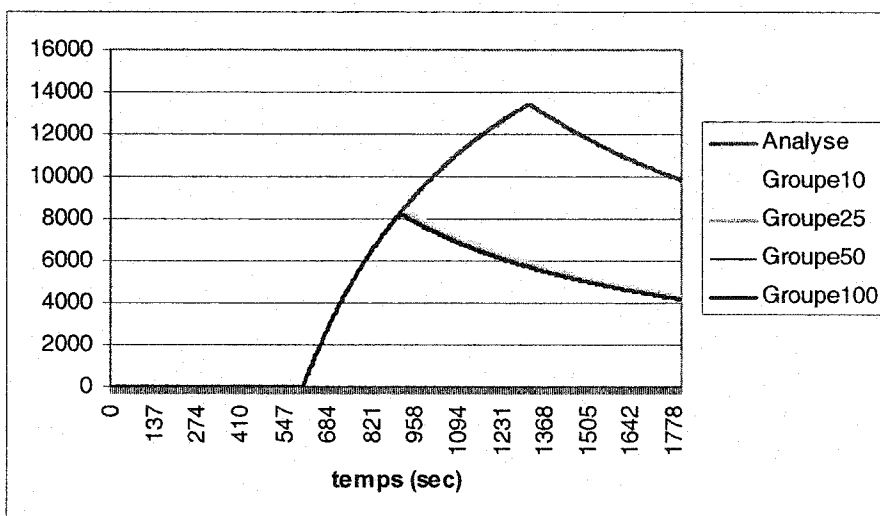


Figure 4.22 Pertes en fonction de la Congestion : *Temoin*

d) Synthèse des résultats

Afin de récapituler toutes ces observations, nous avons consigné tous ces résultats dans les tableaux 4.1 à 4.3, ce qui permet d'avoir une vision globale des phénomènes décrits précédemment.

Tableau 4.1 Comparatifs des performances de délai pour l'Analyse

	HTTP	VOICE	VIDEO
Augmentation Brute (sec)	0.0654	0.0019	0.1421
Augmentation Relative (%)	19.4	1.85	86.1

Ces tableaux, qui sont tirés des courbes obtenus à partir des simulations, permettent de dresser la comparaison des variations de performances selon le contrat de service choisi, les applications considérées et la congestion dans le réseau. Ainsi, le Tableau 4.1 fait le résumé de toutes les différences entre une application privilégiée et la même qui ne l'est pas.

Tableau 4.2 Variations de délai pour le Témoin

		HTTP	VOICE	VIDEO
Analyse/ GroupeUtilisateurs10	Augmentation Brute (sec)	0.4155	0.7991	1.6273
	Augmentation Relative (%)	97	740	530
Analyse/ GroupeUtilisateurs25	Augmentation Brute (sec)	0.6127	1.2941	0.9920
	Augmentation Relative (%)	143	1197	323
Analyse/ GroupeUtilisateurs50	Augmentation Brute (sec)	1.4376	0.9083	0.8691
	Augmentation Relative (%)	335	841	283
Analyse/ GroupeUtilisateurs100	Augmentation Brute (sec)	1.79 (saturation)	N/A	0.8742
	Augmentation Relative (%)	418 (saturation)	N/A	285

Dans les Tableaux 4.2 et 4.3, nous présentons tous les résultats obtenus au niveau du délai (la notation N/A indique que le résultat n'est pas disponible pour cause de saturation). Ainsi, en montrant la dégradation observée selon la congestion pour le *Témoin* et l'*Utilisateur*, nous pouvons nous faire une idée du bénéfice que la

différentiation de service peut entraîner. En effet, étant donné que sans congestion les résultats de l'*Utilisateur* sont meilleurs que ceux du *Temoin*, et sachant que la dégradation est plus importante dans le deuxième cas, il est clair que la différence sera accrue avec une forte congestion.

Tableau 4.3 Variations de délai pour l'*Utilisateur*

		HTTP	VOICE	VIDEO
Analyse/ GroupeUtilisateurs10	Augmentation Brute (sec)	0.0250	0.0134	0.0056
	Augmentation Relative (%)	7.5	11.8	3.4
Analyse/ GroupeUtilisateurs25	Augmentation Brute (sec)	0.0197	0.0196	0.0052
	Augmentation Relative (%)	5.9	18.2	3.1
Analyse/ GroupeUtilisateurs50	Augmentation Brute (sec)	0.0052	0.2100	0.005
	Augmentation Relative (%)	1.6	195	3.0
Analyse/ GroupeUtilisateurs100	Augmentation Brute (sec)	0.0752	N/A	0.005
	Augmentation Relative (%)	22	N/A	3.0

CHAPITRE V

CONCLUSION

Le présent mémoire a porté sur le développement d'une architecture d'espace collaboratif permettant notamment de gérer la Qualité de Service. Nous effectuons une synthèse de nos travaux dans la première section de ce dernier chapitre. La deuxième section énonce les limitations relatives à nos méthodes ainsi qu'à leur implémentation. Enfin, nous terminons ce chapitre en énonçant dans une troisième section les travaux futurs qu'il serait possible d'entreprendre afin de poursuivre les efforts entrepris dans cette recherche, dans le but de les appliquer éventuellement à d'autres problèmes similaires.

5.1 Synthèses des travaux et contributions

Parmi les prérogatives et contraintes que nous nous étions fixées au début de notre recherche, l'une des plus importantes consistait à intégrer notre travail à un projet déjà existant afin d'y apporter une petite contribution en accord avec ce qui avait déjà été fait et de ce qui se fait actuellement. Tout cela, en ne perdant pas de vue le fait que le but premier est d'adapter le concept de laboratoire aux technologies offertes de nos jours. A cet effet, nous avons défini des pistes privilégiées pour améliorer le concept de laboratoire virtuel. La première spécificité concerne la collaboration. Il nous a fallu la définir en faisant la synthèse de l'apprentissage virtuel, de la collaboration traditionnelle, et des technologies maintenant disponibles en informatique pour renforcer ces concepts. Une analyse détaillée de ces paramètres nous a permis de caractériser explicitement la collaboration. En effet, nous l'avons défini sous la forme de deux classes d'interactions, à savoir : coopération et communication. Par ailleurs, afin de rendre cette décomposition complète, nous y avons ajouté la coordination qui permet d'assurer une démarche efficace en joignant l'ensemble. De plus, pour faciliter aussi la mise en place de

solutions adaptées, nous nous sommes efforcés de compléter la décomposition en niveaux d'analyse. Un des aspects originaux de ce mémoire demeure la définition d'une interface de coopération qui assure un contrôle de tous les aspects envisagés, et notamment la Qualité de Service, en ayant recours à des mécanismes de communication pour optimiser le processus de collaboration. L'utilisation des outils que nous avons définis en est la meilleure illustration. Nous avons en effet tenté de trouver une solution au fait que les outils entraînent forcément des problèmes de qualité.

La première étape de l'intégration de la QoS et de son contrôle est, à notre avis, la sensibilisation de l'utilisateur. L'interface de coopération a donc été implémentée à cette fin. En effet, si elle permet de contrôler l'ensemble des outils ou de gérer les groupes de l'utilisateur, elle donne aussi une part prépondérante à la gestion de la QoS. Grâce aux fonctionnalités prévues à cette fin, l'utilisateur est amené à prendre un certain nombre de décisions lui permettant d'adapter le média de communication à ses besoins. C'est dans cette optique que nous avons cherché à différencier certaines classes de service.

Les simulations menées sous OPNET nous ont permis d'affiner cette idée puisqu'il ressort que la solution la plus appropriée est une spécialisation des niveaux en fonction de l'état de collaboration requis : l'un orienté vidéo, l'autre voix et enfin le dernier permettant d'assurer les transferts HTTP. Les résultats obtenus pour ces cas sont relativement probants, notamment lorsque la charge dans le réseau augmente. Ainsi, nous pouvons penser que la différenciation doit permettre de donner une plus grande homogénéité à l'ensemble des échanges, sans privilégier l'un ou l'autre.

5.2 Limitations des travaux

En plus de l'implémentation de l'interface de coopération qui s'avère concluante, nous avons mené les simulations permettant de conforter les hypothèses sur le choix des différentes classes de service pour gérer la collaboration. Cependant, les simulations menées avec OPNET sont conformes à la réalité, nous ne sommes pas sûr à cent pour cent que ce sera le cas avec la configuration choisie. Nous conjecturons par

extrapolation, et aux vues des résultats obtenus avec OPNET en général, que leur pertinence est assez élevée. Par ailleurs, il est peut être intéressant et envisageable de vouloir transposer ces manipulations à un réseau réel, ce qui permettrait de valider totalement le travail effectué dans la recherche. En effet, l'une des limites constatées dans le simulateur OPNET concerne la dépendance entre les applications. Après une analyse des résultats et notamment des courbes fournies au niveau délai et gigue, nous avons constaté que la relation entre les applications n'est pas totalement prise en compte. Les effets de l'un sur l'autre sont relativement nuls pour ne pas dire inexistantes. Nous pouvons donc penser, que dans la réalité, il y aurait quelques différences. Par ailleurs, il serait pertinent aussi de vouloir tester dans un réseau global supportant le modèle *DiffServ* afin de confirmer ces observations.

Par ailleurs, dans l'élaboration du modèle, dans la recherche de solutions et dans la phase de tests, force est de constater que nous avons omis un aspect du développement. En effet, nous n'avons pas pris en compte la notion de coût. Il serait judicieux de tenir compte des possibilités de déploiement afin d'éviter de proposer des solutions qui seraient totalement farfelues ou impossibles à réaliser avec des moyens limités. À notre avis, cette vision des choses doit avoir une importance non négligeable car le choix des solutions est fortement lié aux moyens disponibles. Alors, même si nous avons tenté de nous placer tout au long de l'étude à un niveau permettant de ne pas rendre les choses trop complexes, peut être faudrait-il maintenant s'y consacrer.

5.3 Orientations de recherches futures

L'une de nos principales préoccupations était de respecter une intégration progressive des différents outils, notamment notre Interface et nos modules. En ce sens, nous avons essayé de suivre la démarche suivie par le LARIM dans sa proposition de laboratoire virtuel générique. Après cette petite pierre apportée à l'édifice, il est important de la compléter. Comme dans beaucoup de travaux de recherche, l'implémentation ne peut être exhaustive puisque nous avons préféré nous concentrer sur

certaines modules caractéristiques. Ainsi, il apparaît que ce travail peut être enrichi en développant certains outils dont l'intégration sera similaire à celle du chat par exemple.

Par ailleurs, la base de réflexion entamée au niveau QoS doit pouvoir être achevée. Étant donné que les regroupements se sont avérés efficaces, il ne reste plus qu'à implémenter complètement le module qui gèrera cette fonctionnalité. Une fois ce travail effectué, il s'agira alors de l'intégrer à la plate-forme et notamment à l'interface de coopération. De plus, l'architecture envisagée dans le chapitre 3 proposait d'autres fonctionnalités pour la gestion de la QoS et ces pistes sont donc à explorer.

Enfin, l'une des dernières pistes, mais non des moindres, qui s'offre à nous est la collaboration avec d'autres laboratoires ou centres de recherche. À ce sujet, il est intéressant d'envisager la mise en commun des efforts faits dans ce sens au sein du projet global DIVA, auquel le LARIM participe.

BIBLIOGRAPHIE

- [1] F. Vé, “Méthodologie de conception de laboratoires virtuels et cadre d’expérimentation”. Mémoire de Maîtrise, École polytechnique de Montréal, Canada, 2001.
- [2] C. Bouras, V. Triantafillou, T. Tsiatos. “Aspects of a Collaborative Learning Environment using Distributed Virtual Environments”. ED-MEDIA 2001, Tampere, Finland, June 25-30 2001. pp. 173-178.
- [3] M.C. Dorneich, P.H. Jones. “The design and implementation of a learning Collaboratory”. Conference on Systems, Man, and Cybernetics, Volume 2, IEEE International, 2000. pp 1146 –1151.
- [4] A. Johannsen, W. Van Diggelen, G.J. De Vreede, H. Kremer. “Effects of Video Communication and Telepresence on Cooperative Telelearning Arrangements”. Proceedings of the 33rd Annual Hawaii International Conference on , 2000. pp. 417 –426.
- [5] E.N. Houstis, A. Joshi, S. Weeraawarana, A. Elmagarmid. “Internet, Education, and the Web”. 5th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE'96) 1996.
- [6] B. Collis. “Tele-learning in a Digital World: The Future of Distance Learning”. London: International Thomson Publications, 1997.

- [7] A.W. Bates. "Technology: Open Learning and Distance Education". Routledge, Londres, 1995.
- [8] D.N. Yarbrough. "A Comparative Analysis of Student Satisfaction and Learning in a Computer-Assisted Environment Versus a Lecture Environment". Journal on Excellence in College Teaching, 2001. pp.129-147
- [9] « Télé-Laboratoires » <http://www.licefteluq.quebec.ca/telelab> , 2001.
- [10] G.M. Olson, T.W. Malone, J.B Smith. "Coordination Theory and Collaboration Technology". Hardcover Edition, 2001.
- [11] G. Hall, J. Vaisey. "A survey of Web-Based Telecollaboration Tools". Telelearning NCE-Project 3.1, 1998.
- [12] C. Bouras, G. Hornig, V. Triantafillou, T. Tsiatsos. "Architectures Supporting e-Learning Through Collaborative Virtual Environments: The Case of INVITE". IEEE International Conference on Advanced Learning Technologies-ICALT 2001, August 2001, pp. 13-16.
- [13] S. Shirmohammadi, N.D. Georganas. "JETS: a Java-Enabled Telecollaboration System". Multimedia Computing and Systems '97. International Conference on Proceedings, IEEE, 1997. pp. 541 –547.
- [14] A.G. Mathur, A. Prakash. "A protocol Composition-Based Approach to QoS Control in Collaboration Systems". Proceedings of MULTIMEDIA, 1996. pp. 62-69.

- [15] M. Alfano, R. Sigle. "Controlling QoS in a Collaborative Multimedia Environment". Proceedings of HPDC-5, 1996. pp. 340-342
- [16] C. Allison, H. Lawson, D. McKechnan, A. Ruddle. "Quality of Service Issues in Distributed Learning Environments". Conference on Advanced Learning Technologies, 2001. Proceedings. IEEE International, 2001. pp. 29 –32.
- [17] S. Shirmohammadi, J.C. Oliveira. "Applet_Based Telecollaboration : A Network-Centric Approach". IEEE Multimedia , Volume 5, Issue 2 , April-June 1998. pp. 64 –73.
- [18] K.H. Liu, V.T. Tsaoussidis. "Efficient Network for Collaborative Services and Application Development". International Conference on Multimedia and Expo, IEEE, Volume 1, 2000. pp. 53 –56.
- [19] S. Sitarama, A. Dong, A.M. Agonino. "Speech Acts and Peer Learning in Product Development Case-Study Discussions". International Conference on Engineering Education, 2001. pp. 18-23.
- [20] M. Furini, D.T. Towsley. "Real-Time Traffic Transmissions Over the Internet". Transactions on Multimedia, IEEE, Volume 3, Issue 1 , March 2001. pp. 33 –40.
- [21] F.B. King, H.J. Mayall. "Asynchronous Distributed Problem-based Learning". Advanced Learning Technologies, 2001. International Conference on Proceedings. IEEE, 2001. pp. 157 –159.
- [22] G. Licea, J. Favala, J.A. Garcia, J.A. Aguilar. "A Pattern System Supporting QoS for Synchronous Collaborative Systems". Conference on Protocols for Multimedia Systems - Multimedia Networking, 1997. IEEE, 1997. pp. 223 –228.

- [23] D. Harkey, R. Orfali. "Client-Sever Programming with JAVA and CORBA". Wiley Computer Publishing, New York, Second Edition, 1998.
- [24] S.Pierre, M. Kassouf. "Towards a Telecommunication Platform for Supporting Distributed Virtual Laboratories". International JI. Of Educational Telecommunications, 2001. pp. 157-194.
- [25] F. Garcia, G. Auriol, C. Chassot, A. Lozes, E. Lochin, P. Anelli. "Conception, implémentation et mesure des performances d'une architecture de communication a QoS garantie dans un domaine IPv 6 a services différenciés". Colloque sur l'ingénierie des protocoles, 2002. pp. 16-30.
- [26] S. Blake, D. Black. "An Architecture for Differentiated Services". RFC 2475, 1998.
- [27] O. Medina, J.M Bonnin, L. Toutain. "Service DiffServ pour les flux audio et vidéo". ENST Bretagne, France, 2000.
- [28] B. Gaidioz, P. Primet. "The Equivalent Differentiated Services Model". INRIA, Rapport de recherche 4387, 2002.
- [29] Z. Choukair, D. Retailleau, M. Hellstrom. "Environment for Performing Collaborative Distributed Virtual Environments with QoS". Seventh International Conference on Parallel and Distributed Systems, 2000. Proceedings., 2000. pp. 111-118.
- [30] K.Nichols, S. Blake. "Definition of the Differentiated Services Field (DS Field)". RFC 2474, 1998.

- [31] C. Chassagne. Qualité de Service dans l'Internet. CNRS-UREC, 1998.
- [32] OPNET. Tutorial. OPNET Technologies, 1999.
- [33] OPNET. "Representing Network Traffic". OPNET Technologies, 2001.
- [34] OPNET. "Configuring Applications and Profiles". OPNET Technologies, 2001.
- [35] W.E. Johnston, D. Agarwal, E.O. Luce. "The Virtual Laboratory : Using Networks to Enable Widely Distributed Collaboratory Science". NSF & BNS and Networking and Application Researchers Workshop, LBNL report number 37466. 1997.
- [36] OPNET. "Simulation Analysis for the Analysis of QoS". OPNET Technologies, 2001.