

# MOBILITÉ DES SESSIONS DANS LES COMMUNICATIONS MULTIMÉDIAS EN MODE-CONFÉRENCE BASÉES SUR LE PROTOCOLE SIP

## Mémoire de Doctorat

Wajdi ELLEUCH

Jury :           Alain C. HOULE           (directeur)  
                  Philippe MABILLEAU   (codirecteur)  
                  Soumaya CHERKAOUI   (rapporteur).  
                  Samuel PIERRE           (évaluateur externe)  
                  Roch LEFEBVRE         (évaluateur interne)



Library and Archives  
Canada

Published Heritage  
Branch

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

Bibliothèque et  
Archives Canada

Direction du  
Patrimoine de l'édition

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file Votre référence*

*ISBN: 978-0-494-89632-7*

*Our file Notre référence*

*ISBN: 978-0-494-89632-7*

#### NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

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

#### AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

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

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

## **RÉSUMÉ**

Ce mémoire traite la problématique de la mobilité des sessions pour le transfert des communications multimédias basées sur le protocole SIP. Plusieurs aspects sont étudiés et des mécanismes proposés afin de permettre la mobilité des sessions avant, durant et après leur établissement. En plus d'une communication impliquant deux intervenants, il a été possible d'étendre l'utilisation de la mobilité des sessions pour l'appliquer aux scénarios de communications en mode conférence regroupant plusieurs intervenants. Les mécanismes de mobilité de session développés au cours de cette thèse sont par la suite déployés pour (1) permettre des transformations entre différentes topologies de conférences et (2) construire un modèle de conférence adapté pour l'échange de la voix au sein des groupes de communication à large échelle.

**Mots-clés :** mobilité session, protocole SIP, Voix sur IP, MVoIP, modèles de conférence, conférence multimédia, conférence large échelle.

## **REMERCIEMENTS**

Je tiens avant tout à exprimer toute ma gratitude et ma reconnaissance au Professeur Alain C. Houle, mon Directeur de thèse, pour tout son soutien, sa confiance et ses encouragements sans cesse. J'ai eu le privilège de travailler sous la direction d'Alain en tant que Professionnel de Recherche avant d'entamer mes études doctorales. Toute cette période fut pour moi une expérience exceptionnelle, très enrichissante autant sur le plan scientifique que personnel. Alain m'a fait découvrir un modèle et une vision professionnels plus humaniste, plus humble et plus modeste dont je m'en sers à tous les jours et que j'essaierais de transmettre !

Je tiens également à remercier chaleureusement le Professeur Philippe Mabilieu, mon Codirecteur, d'avoir accepté sans hésitation de m'encadrer pour finaliser ma thèse. J'ai déjà travaillé sous la direction de Mr Mabilieu durant ma maîtrise et je garde d'excellents souvenirs.

Je remercie le Professeur Samuel Pierre, le Professeur Roch Lefebvre et le Professeur Soumaya Cherkaoui d'avoir accepté d'évaluer mon travail et de faire partie du Jury malgré un emploi de temps très chargé.

Je remercie toutes les personnes que j'ai pu côtoyées pendant ces huit années passées à Sherbrooke. La liste est tellement longue, cependant, il m'est impossible de ne pas remercier ma gang d'amis : Jean, Alexis, Audrey, Ramzi, Guillaume, Ramin, Baligh, Mohamed, , Asma et Moez.

Une pensée toute particulière à mes chers parents, mon frère Ali et ma sœur Olfa pour leurs encouragements et leur soutien même à distance. Eh oui, après ces années passées loin de vous, nous revoilà tous réunis !

## TABLE DES MATIÈRES

<b>1</b>	<b>INTRODUCTION GÉNÉRALE .....</b>	<b>1</b>
1.1	INTRODUCTION .....	1
1.2	OBJECTIFS DE LA THÈSE.....	3
1.3	APERÇU DES CONTRIBUTIONS.....	5
1.4	PLAN DE LA THÈSE .....	6
<b>2</b>	<b>FOURNITURE DE SERVICES DANS LES ENVIRONNEMENTS MOBILES.....</b>	<b>10</b>
2.1	ACTEURS DE LA FOURNITURE DE SERVICE ET CONCEPT DU VHE .....	10
2.2	LES ASPECTS DE LA FOURNITURE DE SERVICES .....	11
2.2.1	<i>La mobilité .....</i>	12
2.2.2	<i>Choix du paradigme de communication .....</i>	13
2.2.3	<i>Déploiement et gestion des services.....</i>	14
2.2.4	<i>Personnalisation de service.....</i>	14
2.2.5	<i>Sensibilité au contexte .....</i>	15
2.2.6	<i>Conclusion.....</i>	16
2.3	LES TRAVAUX DE NORMALISATION .....	16
2.4	LES TRAVAUX DE RECHERCHE .....	18
2.5	UTILISATION DE SIP POUR LA FOURNITURE DE SERVICES .....	19
2.6	CONCLUSION.....	21
<b>3</b>	<b>SIP : PROTOCOLE, MÉCANISMES ET SERVICES.....</b>	<b>23</b>
3.1	INTRODUCTION .....	23
3.1.1	<i>Syntaxe des messages .....</i>	23
3.1.2	<i>Mécanisme d'envoi de messages.....</i>	24
3.2	LE DIALOGUE SIP .....	25
3.2.1	<i>Introduction .....</i>	25
3.2.2	<i>Création du dialogue SIP.....</i>	25
3.2.3	<i>Modification du dialogue.....</i>	27
3.2.4	<i>Terminaison du dialogue .....</i>	28
3.3	LE PRINCIPE DE LA NOTIFICATION D'ÉVÈNEMENT DE SIP .....	28
3.4	LA DESCRIPTION DE LA SESSION .....	29
3.5	LE CONTRÔLE DE COMMUNICATION AVEC SIP .....	31
3.5.1	<i>Introduction .....</i>	31
3.5.2	<i>Fonction d'enregistrement.....</i>	31
3.5.3	<i>Fonction de redirection .....</i>	32
3.5.4	<i>Fonction du mandataire .....</i>	32
3.6	EXEMPLES DE SERVICES DE COMMUNICATION BASÉS SUR SIP.....	33
3.6.1	<i>Les services basés sur un échange de flux de données direct .....</i>	33
3.6.2	<i>Les services basés sur l'échange de messages textuels .....</i>	33
3.6.3	<i>Les services combinés .....</i>	33
3.7	CONCLUSION.....	34
<b>4</b>	<b>LA MOBILITÉ SELON SIP .....</b>	<b>35</b>
4.1	INTRODUCTION .....	35
4.2	MOBILITÉ DU TERMINAL SIP .....	35
4.2.1	<i>Mobilité en précommunication.....</i>	36
4.2.2	<i>Mobilité en mi-communication.....</i>	37
4.3	MOBILITÉ DE L'ENVIRONNEMENT DU SERVICE SIP .....	38
4.4	MOBILITÉ DE L'UTILISATEUR SIP.....	39
4.5	MOBILITÉ DE LA SESSION SIP .....	40
4.5.1	<i>Mobilité d'une session en cours d'établissement .....</i>	41
4.5.2	<i>Mobilité d'une session déjà établie.....</i>	44

4.6	DISCUSSION ET CONCLUSION .....	44
<b>5</b>	<b>LA MOBILITÉ DES SESSIONS SIP EN COURS D'ÉTABLISSEMENT AVEC CONTRÔLE.....</b>	<b>46</b>
5.1	INTRODUCTION GÉNÉRALE.....	46
5.2	PERMETTRE LA MOBILITÉ DE SESSION EN COURS D'ÉTABLISSEMENT.....	47
5.2.1	<i>Présentation de la problématique</i> .....	47
5.2.2	<i>Le modèle de communication</i> .....	48
5.2.3	<i>Analyse des données échangées</i> .....	49
5.2.4	<i>Analyse du protocole de transport des données</i> .....	50
5.3	CONCEPTION DU PROTOCOLE.....	50
5.3.1	<i>Protocole d'établissement de la session</i> .....	50
5.3.2	<i>Le modèle du protocole</i> .....	51
5.3.3	<i>Définition de la structure des messages</i> .....	52
5.4	EXEMPLE DE SERVICE.....	53
5.4.1	<i>Le service « Caller-ID »</i> .....	53
5.4.2	<i>Le service « Make-Call »</i> .....	54
5.5	CONCLUSION.....	55
<b>6</b>	<b>LA MOBILITÉ DES SESSIONS SIP DÉJÀ ÉTABLIES.....</b>	<b>57</b>
6.1	INTRODUCTION : VERS UN DISPOSITIF VIRTUEL.....	57
6.2	LES CONTRAINTES LIÉES À LA MOBILITÉ DE SESSION.....	58
6.3	LES OPTIONS DE LA MOBILITÉ DE SESSION .....	59
6.3.1	<i>Transfert et récupération</i> .....	59
6.3.2	<i>Transfert total ou partiel</i> .....	59
6.3.3	<i>Modes de transfert</i> .....	59
6.3.4	<i>Types de média à transférer</i> .....	60
6.4	TRANSFERT DE SESSION AVEC CONTRÔLE .....	61
6.4.1	<i>Introduction</i> .....	61
6.4.2	<i>Transfert vers un seul nœud</i> .....	62
6.4.3	<i>Transfert vers plusieurs nœuds</i> .....	63
6.5	TRANSFERT DE SESSION AVEC PERTE DE CONTRÔLE .....	64
6.6	RÉCUPÉRATION DE SESSION .....	65
6.7	DISCUSSIONS ET CONCLUSIONS .....	67
<b>7</b>	<b>LA MOBILITÉ DES SESSIONS DANS UN MODÈLE DE CONFÉRENCE MULTIMÉDIA COMPLÈTEMENT MAILLÉ.....</b>	<b>72</b>
7.1	INTRODUCTION .....	72
7.2	LE MODÈLES DES CONFÉRENCES COMPLÈTEMENT MAILLÉES.....	73
7.2.1	<i>Structure générale</i> .....	73
7.2.2	<i>Les messages du protocole</i> .....	74
7.2.3	<i>Mappage du protocole vers SIP</i> .....	75
7.3	PERMETTRE LA MOBILITÉ DE SESSION.....	76
7.3.1	<i>Extension des messages du protocole</i> .....	76
7.3.2	<i>Analyse du flux de messages</i> .....	77
7.3.3	<i>Problème de l'échec lors du transfert de session</i> .....	81
7.4	MAPPAGE VERS SIP DES MESSAGES ÉTENDUS DU PROTOCOLE .....	82
7.5	COMPOSANTES DE LA PILE ARCHITECTURALE .....	84
7.6	ÉVALUATION DU TRAFIC ADDITIONNEL GÉNÉRÉ .....	86
7.7	ANALYSE DE COMPORTEMENT .....	90
7.7.1	<i>Interruption du média lors de transfert</i> .....	90
7.7.2	<i>Latence du transfert total</i> .....	91
7.8	CONCLUSION.....	92
<b>8</b>	<b>MOBILITÉ DES SESSIONS POUR UNE TRANSITION DU MODÈLE HERMÉTIQUEMENT COUPLÉ VERS LE MODÈLE COMPLÈTEMENT COUPLÉ.....</b>	<b>94</b>

8.1	INTRODUCTION .....	94
8.2	HERMÉTIQUEMENT COUPLÉ VS. COMPLÈTEMENT COUPLÉ .....	95
8.2.1	<i>L'approche hermétiquement couplée</i> .....	96
8.2.2	<i>L'approche complètement couplée</i> .....	97
8.3	LA TRANSITION VERS LE MODÈLE COMPLÈTEMENT COUPLÉ .....	98
8.3.1	<i>Les approches de transitions</i> .....	98
8.3.2	<i>Les étapes de la transition</i> .....	99
8.3.3	<i>Le protocole des messages</i> .....	100
8.4	ÉVALUATION DU COÛT DE LA TRANSITION .....	101
8.5	CONCLUSION .....	103
9	<b>MOBILITÉ DES SESSIONS POUR LE SUPPORT DES CONFÉRENCES P2P À LARGE ÉCHELLE</b> .....	104
9.1	INTRODUCTION .....	104
9.2	APPROCHES DE COUPLAGE MÉDIA .....	107
9.3	LE MODÈLE DU SYSTÈME .....	109
9.4	LES COMPOSANTES DU SYSTÈME .....	111
9.4.1	<i>Gestionnaire de la conférence</i> .....	111
9.4.2	<i>Gestionnaire d'adhésion</i> .....	112
9.4.3	<i>Gestionnaire de l'espace de dialogue</i> .....	112
9.4.4	<i>Protocole de l'application</i> .....	113
9.5	LE RÉSEAU DE DISTRIBUTION DU FLUX MÉDIA .....	113
9.6	MÉCANISMES DE SUPPORT DE L'ADHÉSION .....	116
9.6.1	<i>Introduction</i> .....	116
9.6.2	<i>Ajout d'un nouvel utilisateur</i> .....	116
9.6.3	<i>Départ ou échec du NF</i> .....	117
9.6.4	<i>Départ ou échec du NDM</i> .....	118
9.6.5	<i>Départ ou échec du NA</i> .....	119
9.7	IMPLÉMENTATION DU SYSTÈME EN SIP .....	120
9.7.1	<i>Mappage des opérations du protocole</i> .....	120
9.7.2	<i>Implémentation de l'offre/réponse média</i> .....	121
9.7.3	<i>Implémentation de l'assignation de la charge média</i> .....	121
9.7.4	<i>Implémentation du contrôle de l'espace de dialogue</i> .....	122
9.7.5	<i>Implémentation de la gestion de la conférence</i> .....	122
9.8	ÉVALUATION DES PERFORMANCES DU SYSTÈME .....	123
9.8.1	<i>Environnement de simulation</i> .....	124
9.8.2	<i>Paramètres de la simulation</i> .....	126
9.8.3	<i>Résultats de la Simulation</i> .....	127
9.8.4	<i>Conclusions des simulations</i> .....	131
9.9	CONCLUSION .....	132
10	<b>CONCLUSIONS ET PERSPECTIVES</b> .....	134
10.1	CONCLUSION GÉNÉRALE ET CONTRIBUTIONS .....	134
10.2	LES LIMITATIONS DU TRAVAIL RÉALISÉ .....	136
10.3	PERSPECTIVES DE RECHERCHES .....	137
10.3.1	<i>Étendre la mobilité des sessions en cours d'établissement</i> .....	137
10.3.2	<i>Permettre la récupération des sessions transférées</i> .....	137
10.3.3	<i>Adaptabilité du service selon les capacités du nouveau terminal lors de la mobilité des sessions</i> .....	138
10.3.4	<i>Personnalisation de l'environnement du service lors de la mobilité des sessions</i> .....	139
11	<b>BIBLIOGRAPHIE</b> .....	140

## LISTE DES FIGURES

Figure 1 : Le concept VHE.....	11
Figure 2 : Les mécanismes de création d'un dialogue SIP.....	27
Figure 3 : Diagramme de séquence d'un exemple de notification d'évènement.....	29
Figure 4 : Modèle de couches d'une application SIP mettant en évidence la mobilité.....	35
Figure 5 : Scénario d'échange de messages lors de la mobilité en pré-communication.....	37
Figure 6 : Une seule adresse SIP pour plusieurs terminaux.....	40
Figure 7 : Plusieurs adresses SIP pour une seule adresse IP.....	40
Figure 8 : Illustration de la mobilité de session en cours d'établissement lors d'un appel entrant.....	43
Figure 9 : Illustration de la mobilité de session en cours d'établissement lors d'un appel sortant.....	43
Figure 10 : Vue générale du système.....	48
Figure 11 : Modèle d'échange de messages par le mécanisme <i>SUBSCRIBE/NOTIFY</i> .....	51
Figure 12 : le modèle en couches du protocole.....	52
Figure 13 : Exemple de structure de message échangé.....	53
Figure 14 : Exemple de diagramme de séquence lors d'un appel entrant.....	54
Figure 15 : Exemple de diagramme de séquence lors d'un appel sortant.....	55
Figure 16 : Diagramme de séquence pour un transfert avec contrôle vers un seul nœud.....	62
Figure 17 : Diagramme de séquence pour un transfert avec contrôle vers plusieurs nœuds.....	63
Figure 18 : Diagramme de séquence pour un transfert en mode avec perte de contrôle.....	64
Figure 19 : Diagramme de récupération de session dans le mode avec perte de contrôle.....	66
Figure 20 : Vue sur les interconnexions dans un modèle complètement maillé.....	73
Figure 21 : Exemple d'échange de message pour l'ajout d'un nouveau participant.....	74
Figure 22 : Exemple d'échange de messages dans le mode de transfert avec contrôle.....	79
Figure 23 : Exemple d'échange de messages dans le mode avec perte de contrôle.....	80
Figure 24 : Organigramme de la procédure de transfert de session.....	82
Figure 25 : Interaction entre les composantes de l'architecture en couche.....	85
Figure 26 : Trafic de messages additionnel pour le premier scénario.....	89
Figure 27 : Trafic de messages additionnel pour le deuxième scénario.....	89
Figure 28 : Trafic de messages additionnel pour le troisième scénario.....	89
Figure 29 : Présentation de quelques modèles de conférence.....	96
Figure 30: Inter-modèles transition overview.....	99
Figure 31 : La topologie de la conférence après (n-1) étapes.....	99
Figure 32 : Les étapes de la transition vers le modèle complètement couplé (N = 4).....	101
Figure 33 : Évaluation de la quantité de messages échangés.....	103
Figure 34 : Quelques approches pour le service MVoIP à large échelle.....	107
Figure 35 : Topologie générale du réseau de contrôle et du réseau media.....	110



Figure 36 : Composantes du système.....	111
Figure 37 : Aperçu de la gestion du droit à la parole dans l'espace de dialogue.....	113
Figure 38 : Cas d'utilisation du traitement audio.....	116
Figure 39 : Mécanisme d'ajout d'un nouvel utilisateur .....	117
Figure 40 : Mécanisme pour le départ du NF.....	118
Figure 41 : Mécanisme pour le départ du NMD .....	119
Figure 42 : Mécanisme pour le départ du NA .....	120
Figure 43 : Variation du PESQ selon le taux de codage et la détection de silence.....	127
Figure 44 : Variation de la bande passante consommée selon le taux de codage et la détection de silence.....	128
Figure 45 : Exemple d'un arbre de distribution média à différents niveaux .....	128
Figure 46 : Variation de la valeur du PESQ pour un taux de codage à 40 000 bit/s selon le détecteur de silence et le niveau dans l'arbre.....	129
Figure 47 : Variation de la valeur du PESQ pour un taux de codage à 32 000 bit/s selon le détecteur de silence et le niveau dans l'arbre.....	130
Figure 48 : Variation de la valeur du PESQ pour un taux de codage à 24 000 bit/s selon le détecteur de silence et le niveau dans l'arbre.....	130
Figure 49 : Variation de la valeur du PESQ pour un taux de codage à 16 000 bit/s selon le détecteur de silence et le niveau dans l'arbre.....	131

## **LISTE DES TABLEAUX**

<b>Tableau 1 : Mappage entre les messages abstraits et les messages SIP</b>	<b>76</b>
<b>Tableau 2 : Mappage entre les messages étendus abstraits et les messages SIP</b>	<b>84</b>
<b>Tableau 3 : les valeurs du MOS selon l'ITU-T</b>	<b>125</b>

## LISTE DES ACRONYMES

<b>Acronyme</b>	<b>Définition</b>
3GPP	<i>Third Generation Partnership Project</i>
ALM	<i>Application Layer Multicast</i>
CDP	<i>Coupled Distributed Processing</i>
CPL	<i>Call Processing Language</i>
DDP	<i>Decoupled Distributed Processing</i>
IMS	<i>Internet Multimedia Subsystem</i>
MMoIP	<i>MultiMedia over IP</i>
MOS	<i>Mean Opinion Score</i>
MVoIP	<i>Multiparty Voice over IP</i>
NA	Nœud Administrateur
NAV	Nœud Audio Vidéo
NC	Nœud Correspondant
ND	Nœud Distributeur
NF	Nœud Feuille
NL	Nœud Local
NM	Nœud Mobile
NMD	Nœud Mixeur Distributeur
NRA	Nœud Replica Administrateur
NU	Nouvel Utilisateur
NV	Nœud Vidéo
PESQ	<i>Perceptual Evaluation of Speech Quality</i>
PoC	<i>Push to talk over cellular</i>
PTT	<i>Push to Talk</i>
P2P	<i>peer to peer</i>
QoS	<i>Quality of Service</i>
RTP	<i>Real Time Protocol</i>
RTCP	<i>Real Time Control Protocol</i>
SDP	<i>Session Description Protocol</i>
SIP	<i>Session Initiation Protocol</i>
VHE	<i>Virtual Home Environment</i>
VoIP	<i>Voice over IP</i>

# 1 INTRODUCTION GÉNÉRALE

## 1.1 INTRODUCTION

Avec le développement des réseaux IP sans fil et la prolifération des terminaux mobiles, de nouveaux besoins de mobilité se sont développés. En effet, les utilisateurs, avec ou sans leurs terminaux, sont de plus en plus mobiles et ont tendance à vouloir accéder à leurs outils de travail habituels de n'importe quel endroit et à n'importe quel moment. Il est possible de définir deux types de mobilité : la mobilité du terminal et la mobilité du service. La problématique de la mobilité du terminal concerne principalement le cas d'un terminal mobile qui devrait continuer à profiter du service même lors d'un changement de réseaux d'accès. D'un autre côté, la problématique de la mobilité du service est plutôt centrée sur l'utilisateur et associe donc le service à l'utilisateur et non au terminal. Ainsi, un utilisateur devrait pouvoir continuer à utiliser ses services même lorsqu'il change d'un terminal à un autre.

Il existe présentement une grande variété de terminaux capables de se connecter à Internet et d'exécuter des services de communication basés sur IP. Ces terminaux peuvent être stationnaires ou mobiles et peuvent avoir des caractéristiques et des performances très variées. Généralement, les terminaux stationnaires fournissent une meilleure performance et un confort d'utilisation supérieurs aux terminaux mobiles. Cependant, les terminaux mobiles sont facilement transportables et permettent une certaine mobilité géographique de l'utilisateur. Il existe donc un besoin de pouvoir utiliser tous les terminaux d'une façon interchangeable afin de profiter des avantages de chacun. Cette utilisation interchangeable s'inscrit dans la problématique de la mobilité de session, lorsqu'un utilisateur en cours de communication décide de changer de terminal sans interrompre sa communication.

Pour répondre à un tel besoin, plusieurs solutions ont été proposées. Ces solutions sont basées, en grande partie, sur l'utilisation des agents mobiles [46] [22] [86], sur l'introduction des intergiciels (*middlewares*) propriétaires [93] [88] [4] ou même sur l'utilisation de serveur mandataire (*Proxy*) personnel [54]. L'objectif de ces solutions et

## Introduction Générale

de permettre le transfert de l'environnement d'exécution du service d'un terminal à un autre d'une façon transparente à l'utilisateur. Malgré leur apport théorique, ces solutions manquent de réalisme puisque leur implémentation dans la vie courante reste très difficile. Plus encore, ces solutions ne sont pas adaptées pour des services de communication multimédia à flux continue où le changement de session ne doit pas interrompre la communication.

Depuis quelques années, le 3GPP (*Third Generation Partnership Project*) a choisi le protocole SIP [74] (Session Initiation Protocol) pour être le standard d'établissement de sessions dans les IMS [1] (*Internet Multimedia Subsystem*). L'ensemble des IMS inclut les services de communication basés sur IP, en particulier, la Voix sur IP (*VoIP*), le Multimédia sur IP (*MMoIP*), la vidéoconférence, la messagerie instantanée, les systèmes de présence, le Push to talk over cellular (*PoC*), les jeux en réseau, les services communautaires, le partage de contenu, etc. Avec l'utilisation de la technologie SIP, la notion de « Session » et de « Dialogue » dans une communication sur IP devient plus concrète. Ainsi, SIP permet d'initier et de créer un dialogue SIP, de mener une négociation de la session média à établir, d'identifier chaque utilisateur avec une adresse unique SIP, de faire la correspondance entre la personne et la machine utilisée, et même de transférer une communication d'un terminal à un autre. Ce transfert de communication lors d'un service basé sur SIP est appelé dans ce travail « mobilité de la session SIP ». Même s'il existe des mécanismes de base sous le protocole SIP qui permettent d'effectuer un transfert de session, l'étendue de ces mécanismes ne permet pas de couvrir les cas de mobilités lorsque la session est en cours d'établissement. Plus encore, ces mécanismes existants ne couvrent que le cas d'une communication entre deux intervenants et ne sont donc pas adaptés pour les cas de communications en mode conférence. L'investigation de ces deux aspects a permis de proposer quatre axes de recherche traités dans ce rapport. Les deux premiers visent à étendre l'utilisation de la mobilité de la session pour des cas spécifiques alors que les deux autres axes utilisent la mobilité de session comme outil pour soit effectuer des transitions dans les

topologies de communication soit pour supporter des systèmes de conférence à large échelle.

## 1.2 OBJECTIFS DE LA THÈSE

Nous proposons dans ce rapport de thèse d'étudier la mobilité des sessions à la fois en cours de leur établissement mais aussi lorsque la communication est déjà établie. Ainsi, nous visons à explorer et à atteindre les objectifs suivants :

1. Il existe des solutions qui assurent la mobilité des sessions multimédias avant et après leur établissement. Malheureusement, ces solutions ne sont plus valables lorsqu'il s'agit d'assurer la mobilité d'une session en cours d'établissement. L'objectif est de permettre aux deux intervenants de contrôler ensemble la mobilité d'une session lors de son établissement de façon à pouvoir l'accepter, la rejeter, l'annuler, le transférer, etc. Comment concevoir un tel protocole ? Comment l'intégrer dans les systèmes de communications existant basés sur SIP ? Est-ce que l'utilisation du mécanisme *Subscribe/Notify* de SIP couplée avec une approche en couche permet une telle coopération entre les intervenants pour offrir cette mobilité de session?
2. Les mécanismes existants se sont focalisés sur le cas de la mobilité de session dans des scénarios de communication entre deux intervenants. Pour les scénarios d'une communication multimédia en mode-conférence, l'utilisation de l'option de mobilité de session par un des participants peut certainement avoir des répercussions sur les autres participants. De nouveaux mécanismes doivent donc être prévus pour supporter le service de communication multimédia au sein de la conférence lors de la mobilité de session tout en gardant la cohérence du groupe d'intervenant. Deux modes de transfert de session peuvent s'appliquer dépendamment si la machine qui initie le transfert garde le contrôle (mode avec contrôle) ou quitte complètement la conférence (mode sans contrôle ou appelé aussi avec perte de contrôle). La problématique proposée permet de répondre aux interrogations suivantes : Comment concevoir ces mécanismes de mobilité de

session au sein d'une communication en mode conférence ? Comment continuer à supporter les deux modes : avec et sans contrôle ? Quel mode est le plus avantageux en termes de volume des messages échangés et temps d'interruption de la communication ?

3. Le modèle de conférence centralisé (appelé aussi hermétiquement couplé) permet des communications multimédias basées sur SIP entre plusieurs intervenants. Cette topologie, même si elle se distingue par la facilité de son déploiement, peut présenter quelques limitations lorsque des évènements bien particuliers surviennent. En effet, le départ du participant central ou l'augmentation du nombre de participants peuvent mettre en péril la continuation de la conférence ou la qualité de service lors du traitement média. D'autres modèles de conférence (i.e. le modèle complètement couplé) même s'il est à la fois plus coûteux et plus complexe à établir et à maintenir, il permet en contre partie de résoudre les limitations du modèle hermétiquement couplé. Est-il possible d'envisager une transition, à la demande, du modèle hermétiquement couplé vers le modèle complètement maillé ? Comment développer une telle transformation ? Quelle serait l'approche la plus convenable pour que l'interruption et l'échange de messages soient optimisés ?
4. Plusieurs modèles et architectures sont proposés pour supporter le service de communication de voix sur IP en mode-conférence. Malheureusement, une très grande partie des ces modèles est conçue pour s'adapter à des conférences à petite échelle incluant un nombre très limité de participants. Même si le modèle qui se base sur un serveur de conférence dédié permet de supporter un plus grand nombre de participant, cette solution s'avère coûteuse à mettre en place puisqu'elle nécessite, du côté serveur, une large bande passante et une puissance de calcul dont les frais ne peuvent garantir la gratuité et le large déploiement de la solution. Même si le modèle décentralisé permet d'éviter les limitations du modèle centralisé, son utilisation dans le cadre d'une conférence multimédia à large échelle s'avère complexe puisqu'il faut garantir le contrôle (et l'administration) de la conférence. La mobilité de session dans le mode « avec contrôle » peut être utilisé afin de créer à la fois un réseau média étendu et décentralisé alors que le réseau de

contrôle et d'administration reste centralisé autour de l'administrateur. Comment intégrer ces mécanismes de mobilité de session ? comment gérer les fonctionnalités nécessaires au bon fonctionnement de la conférence (création/destruction, ajout/suppression de participants, gestion du droit à la parole, distribution et affectation de la charge média, etc.) ?

### 1.3 APERÇU DES CONTRIBUTIONS

À travers ce travail de thèse, il a été permis de proposer différentes contributions dans le domaine de la mobilité des sessions et des communications multimédias en mode conférence. Il est possible d'énumérer ces contributions de la façon suivante :

1. Analyse, conception et implémentation d'un nouveau protocole qui permet la prise en charge de la mobilité des sessions en cours d'établissement avec contrôle. Il s'agit d'un mécanisme qui se base sur le protocole SIP couplé à une approche en couches pour prendre en charge les différentes opérations à travers un codage en XML. Ainsi, il a été possible de supporter plusieurs opérations et garantir le contrôle de la session pour les cas d'appels entrants et d'appels sortants. Ce même protocole que nous proposons a été aussi conçu pour permettre une grande ouverture pour de nouveaux services généralement utilisés dans le domaine de la voix sur IP. Cette première contribution a été présentée dans [23].
2. Dans le cadre de l'élargissement du domaine d'application de la mobilité de session, le cas spécifique d'une communication en mode conférence complètement maillée a été étudié. Et il a été possible par la suite de proposer de nouveaux mécanismes basés sur SIP pour permettre à un participant d'une conférence multimédia de type complètement maillée d'utiliser l'option de mobilité de session lorsqu'il change d'un terminal à un autre. Pour cela, le nouveau protocole que nous avons mis en place supporte des opérations de transfert et de récupération selon deux modes : avec contrôle et avec perte de contrôle. Cette deuxième contribution a été présentée dans [24].



3. Exploitation et déploiement des mécanismes de mobilité de sessions pour réaliser une transition entre le modèle hermétiquement maillé vers le modèle complètement maillé selon une approche de type implicite et une autre de type explicite. Ce travail a permis d'assurer la continuité de la conférence même si le *focus* doit partir ou lorsque la charge de traitement média devient très élevée pour être traitée par un seul participant. Dans cette contribution présentée dans [25], nous avons proposé une solution à cette limitation en effectuant la transition vers le modèle complètement maillé.
  
4. Création d'une nouvelle architecture qui supporte les conférences Voix sur IP à travers l'utilisation de mécanismes de mobilité des sessions en cours de communication. L'approche que nous proposons permet de gérer le réseau de distribution et le réseau de mixage à travers un réseau séparé de contrôle et d'administration de la conférence. Le protocole proposé supporte toutes les opérations d'ajout/départ des participants et permet aussi l'optimisation et la maintenance du réseau média dépendamment de l'algorithme à utiliser à travers des opération de migration, de séparation et de fusionnement des participants qui jouent le rôle de mixeurs. Cette contribution a été présentée dans [26][27]. Nous avons aussi proposé une étude d'évaluation des performances du réseau média à travers la simulation d'une dizaine de niveaux arborescents de codage/décodage pour montrer que la qualité de parole perçue reste acceptable lors de l'utilisation du codec G .726 à 40 000 ou 32 000 bit/s par exemple. Cette dernière contribution a été présentée dans [28].

## 1.4 PLAN DE LA THÈSE

Pour répondre aux différents objectifs présentés dans la section précédente, le plan de rapport s'organise selon deux grandes parties : (1) la première partie vise à positionner la problématique de la mobilité des services dans son contexte. Elle évoque les travaux effectués dans les environnements mobiles ayant comme objectif de garantir la mobilité de la communication lors d'un changement de terminal. La section suivante

## **Introduction Générale**

---

définit la mobilité des sessions dans les communications SIP par rapport aux autres niveaux de mobilité (mobilité du terminal, de l'utilisateur et de l'environnement du service). Au cours de cette même partie qui inclut la première contribution citée précédemment, une étude sur deux cas de mobilité des sessions est proposée : lorsque la session est en cours d'établissement et lorsqu'elle est déjà établie. (2) la deuxième partie de ce rapport se consacre à l'étude et au déploiement de la mobilité des sessions SIP dans les communications multimédias en mode-conférence. Trois grands thèmes sont abordés dans cette partie pour décrire les trois dernières contributions citées dans la section précédente : (a) l'utilisation de la mobilité de session dans les conférences de type complètement maillée (b) l'utilisation de la mobilité de session comme outil de transition entre modèles de conférence et (c) l'utilisation de la mobilité de session pour permettre un système de communication à large échelle.

# **Partie 1**

**La mobilité des sessions de communications  
multimédias basées sur le protocole SIP**

Cette première partie permet de mettre en contexte la problématique de la mobilité par rapport aux autres besoins rencontrés lors de la fourniture de service dans les environnements mobiles. L'adoption du protocole SIP pour être le standard lors de la signalisation dans les IMS a justifié l'intérêt d'orienter cette étude de la mobilité selon ce protocole. Ainsi, il a été jugé utile de consacrer la section 3 à la présentation des grandes caractéristiques du protocole SIP. Cette même section introduit les entités et les mécanismes qu'offre SIP pour gérer les dialogues et les sessions média entre intervenants. Il a été possible par la suite d'exposer les différents niveaux de mobilité selon SIP. En effet, la section 4 permet de mettre la mobilité des sessions dans son contexte par rapport à d'autres aspects de mobilité comme celle de l'utilisateur, de l'environnement d'exécution ou du terminal. La problématique de la mobilité des sessions SIP est présentée selon deux niveaux distincts : lorsque la session est en cours d'établissement et lorsqu'elle est déjà établie. Chacune de ces deux problématiques est traitée selon la présence ou l'absence de contrôle durant ou après le transfert de session. La section 5 traite le cas de la mobilité de session en cours d'établissement alors que la section 6 étudie le cas des sessions déjà établies.

## **2 FOURNITURE DE SERVICES DANS LES ENVIRONNEMENTS MOBILES**

### **2.1 ACTEURS DE LA FOURNITURE DE SERVICE ET CONCEPT DU VHE**

Un environnement mobile peut inclure tout environnement faisant référence à la mobilité de l'utilisateur et/ou la mobilité du terminal. Il s'agit d'un système composé de sites mobiles qui permet à ses utilisateurs d'utiliser les services proposés indépendamment de leur position physique.

D'une façon très générale, et indépendamment du type de service à offrir, la fourniture de service dans un environnement mobile concerne principalement trois acteurs :

- ✓ **L'utilisateur final** : Cela désigne le terminal ou l'utilisateur du terminal qui utilise le service.
- ✓ **L'opérateur réseau** : L'opérateur réseau gère l'abonnement de l'utilisateur. Il fournit généralement les informations d'accès à chaque abonné ainsi que l'infrastructure d'exécution et les applicatifs nécessaires à l'exécution du service. Il peut également proposer ses propres services aux utilisateurs mobiles.
- ✓ **Le fournisseur de service** : L'opérateur réseau fournit également des services proposés par des fournisseurs de services indépendants. L'utilisateur n'a pas nécessairement connaissance de l'existence de ces fournisseurs de services puisqu'ils peuvent être masqués par l'opérateur réseau.

Le VHE (*Virtual Home Environment*) est un concept d'environnement de service personnalisé défini par le 3GPP dans le cadre de la normalisation de l'UMTS [90]. Le VHE définit un système qui permet à des utilisateurs nomades d'accéder et d'utiliser, d'une manière personnalisée, l'ensemble des services offerts sans se soucier de leur localité, du type de réseau d'accès et du type de terminal (dans les limites des capacités du

réseau et du terminal). La Figure 1 présente le cas d'un utilisateur mobile qui utilise différents équipements dans des réseaux d'accès hétérogènes.

Même s'il est adopté et normalisé par l'UMTS, le concept du VHE n'a rien de spécifique à la technologie cellulaire (GSM, CDMA). Il est donc possible de transposer le concept du VHE et de l'appliquer aux réseaux basés sur le protocole IP (Internet Protocol).

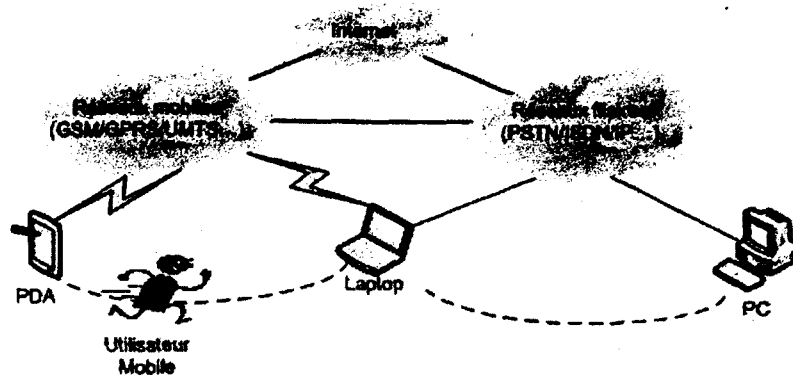


Figure 1 : Le concept VHE

Les objectifs du VHE sont principalement :

- ✓ La personnalisation des services pour les utilisateurs ayant des préférences et des besoins différents,
- ✓ La portabilité de l'environnement personnalisé à travers divers réseaux d'accès et sur des terminaux de nature et de capacités différentes.

## 2.2 LES ASPECTS DE LA FOURNITURE DE SERVICES

Il existe différents aspects à considérer lors de la mise en place d'une solution de fourniture de service. Parmi ces aspects, il est possible de citer :

- ✓ La mobilité
- ✓ Le choix du paradigme de communication
- ✓ Le déploiement et la gestion des services

- ✓ La personnalisation de service
- ✓ La sensibilité au contexte

### 2.2.1 La mobilité

Le concept de la mobilité dans les futurs réseaux de communication renvoie à la mobilité globale qui regroupe la mobilité personnelle, la mobilité du terminal et la mobilité de service.

- ✓ La **mobilité personnelle** correspond à la capacité de l'utilisateur d'accéder à des services personnalisés à partir de n'importe quel terminal (fixe ou mobile) et la capacité du réseau de fournir ces services selon les préférences de l'utilisateur. La mobilité personnelle permet aux utilisateurs d'utiliser leurs services personnalisés indépendamment du terminal utilisé.
- ✓ La **mobilité du terminal** correspond à l'aptitude du terminal à accéder aux services, quel que soit l'endroit où il se trouve et la vitesse de son déplacement.
- ✓ La **mobilité de service**, aussi appelée portabilité du service, fait référence à la capacité du réseau à fournir les services souscrits à l'endroit où se trouvent le terminal et l'utilisateur. Les services exacts que l'utilisateur peut demander sur son terminal dépendent des propriétés du terminal (que l'on appelle les capacités du terminal) et du réseau qui sert ce terminal.

Il existe des situations où les contraintes de la mobilité sont associées aux contraintes de continuité du service. Dans ce cas, il s'agit d'assurer à la fois la mobilité personnelle, du service ou du terminal de manière à ce que les transitions soient transparentes (*seamless*) et non-interventionnistes (*Handoff*).

### 2.2.2 Choix du paradigme de communication

Un des premiers aspects à considérer dans la conception d'une architecture distribuée est le choix du paradigme de communication à utiliser entre les différentes entités constituant l'environnement. Ce choix doit en premier lieu prendre en compte les besoins de l'environnement et de la plateforme à développer.

Les environnements mobiles constituent un des domaines les plus adaptés à l'utilisation des agents mobiles [84][91]. En effet, les possibilités offertes par les agents mobiles à savoir, le déplacement du traitement à l'endroit où se trouvent les données, et le fonctionnement asynchrone (le client peut se déconnecter pendant le traitement) sont compatibles avec les besoins des terminaux utilisant des liaisons lentes et coûteuses.

Dans le cadre de la fourniture de services, les agents mobiles peuvent être utilisés pour la fourniture instantanée de services, la recherche du service le plus adapté aux préférences de l'utilisateur, la personnalisation, la répartition de charge et la gestion des déconnexions. Cependant, les solutions basées sur les agents mobiles sont en perte de vitesse et ne sont plus d'actualité. Ce fait est principalement lié aux limitations de déploiement des environnements d'exécution des agents mobiles dans des terminaux mobiles légers. D'autres paradigmes plus classiques tels que le passage de messages, l'appel de procédure à distance, l'invocation d'objet à distance ou même l'évaluation à distance peuvent permettre une communication inter-systèmes dans un environnement réparti.

Durant les dernières années, plusieurs protocoles de communication ont été standardisés afin de permettre une interconnexion et une interopérabilité entre systèmes répartis. Ces protocoles peuvent se diviser en deux catégories : des protocoles basés sur le modèle de communication client/serveur (*HTTP, FTP, etc.*) et les protocoles basés sur le modèle de communication point à point P2P comme l'exemple de SIP. Ce faisant, les informations habituellement contenues dans un agent mobile ou transmises entre agents peuvent désormais être codées dans des formats textuels, comme le format de codage XML [32]. Ces informations textuelles codées sont facilement



transportables par les protocoles du Web [48]. Il existe même des protocoles comme le XML-RPC [99] destinés à la l'invocation distante du service. Des technologies plus récentes, comme SOAP [82] assurent même l'annonce du service, sa découverte et même l'encapsulation des données dans le corps du message.

### **2.2.3 Déploiement et gestion des services**

Les fournisseurs de service ont intérêt à ce que leurs services soient utilisables depuis tous les terminaux d'accès et quel que soit le réseau utilisé. Ceci pose le problème de la portabilité des services dans des environnements différents. Dans le cas d'une utilisation de solution basée sur les agents mobiles, il est important que les entités qui entrent en communication directe aient les mêmes types de plateforme d'exécution. Autrement, il est possible de faire appel à des plateformes de gestion de services qui jouent le rôle de médiateur entre les différentes entités impliquées dans le processus de fourniture de service. Il est aussi important de garantir la gestion des services et assurer leur mise à jour en cas de besoin (enregistrement de nouveaux services, mise à jour des informations associées aux services proposés, annonce de la disponibilité des services aux utilisateurs mobiles, suppression de services, etc.). La gestion, la maintenance et la mise à jour des services peuvent être plus complexes lorsque le service n'est pas fourni par des serveurs centralisés, mais plutôt exécuté directement entre clients selon le modèle P2P.

### **2.2.4 Personnalisation de service**

La personnalisation des services offre aux utilisateurs le moyen de définir et de modifier la manière dont les services leur sont livrés afin de satisfaire leurs besoins. La personnalisation est un des éléments clés d'un système de fourniture de services pour les futurs réseaux. Un tel système doit offrir à l'utilisateur un environnement de travail personnalisé construit sur la base de la notion de profil. Un profil peut être vu comme une représentation structurée des besoins de l'utilisateur. Il comprend un ensemble de paramètres que l'utilisateur aura spécifiés sous forme de préférences, soit dans son

contrat de souscription, soit au cours de son utilisation du système. L'utilisateur doit pouvoir :

- ✓ Créer un nombre quelconque de profils,
- ✓ Modifier ces profils,
- ✓ Sélectionner le profil qu'il veut appliquer à son environnement de travail, appelé profil actif. Un profil est dit actif si l'environnement courant de l'utilisateur respecte les préférences contenues dans ce profil.

La personnalisation des services constitue un des besoins à offrir lors de la mobilité personnelle de l'utilisateur.

### 2.2.5 Sensibilité au contexte

Plusieurs définitions ont été proposées pour décrire ce qu'est le contexte dans le domaine de l'informatique. Schilit [11], inventeur du terme *context-aware computing*, définit le contexte par rapport à la localisation, les identités des objets et des personnes environnantes. Brown [10] définit le contexte comme étant les éléments de l'environnement de l'utilisateur que son ordinateur connaît. Il rajoute à la définition de Schilit la notion de temps, de saison, de température, etc. Ce n'est qu'avec Dey [5] qu'une définition précise du contexte a été définie :

« Le contexte est toute information qui peut être employée pour caractériser la situation d'une entité. Une entité est une personne, un endroit ou un objet qui est en rapport avec l'interaction entre un utilisateur et application, y compris l'utilisateur et l'application eux-mêmes ».

Le contexte est composé d'un ensemble d'informations qui sont disponibles au moment de l'interaction. Ces informations peuvent être de différentes catégories. Il est possible de citer : [55]

- ✓ Information sur l'identité de l'utilisateur,
- ✓ Information spatiale : localisation, orientation, vitesse, accélération, etc.,
- ✓ Information temporelle : l'heure, la date, la saison,

- ✓ Information sur l'environnement : température, qualité de l'air, le niveau de la lumière ou du bruit,
- ✓ Situation sociale : les personnes qui sont à proximité,
- ✓ Ressources environnantes : les terminaux accessibles,
- ✓ Disponibilité des ressources : les piles, l'affichage, le réseau, la bande passante,
- ✓ Mesures physiologiques : tension artérielle, battement du cœur, vitesse de respiration, activité des muscles, timbre de la voix,
- ✓ Activité : en cours de discussion, d'écriture, de marcher, de courir, etc.,
- ✓ Les données de l'emploi du temps.

### **2.2.6 Conclusion**

La problématique de la fourniture de services incorpore plusieurs aspects. Cette section s'est limitée à présenter les aspects de mobilité, du choix du paradigme de communication, du déploiement et de la gestion des services, de la personnalisation du service et de la sensibilité au contexte. D'autres contraintes comme la qualité du service et sa localisation peuvent aussi s'ajouter.

Ces aspects sont encore plus complexes à considérer lorsque l'environnement du service est caractérisé par une grande hétérogénéité (les terminaux, le réseau d'accès et le fournisseur de service sont tous hétérogènes). Pour résoudre ce problème, plusieurs instances de normalisation (3GPP, UMTS, ITU, IETF, W3C, etc.) ont trouvé dans cette hétérogénéité une raison d'être afin de fournir des standards pour la fourniture de service dans les environnements mobiles. Cette même problématique a aussi été le sujet de plusieurs travaux de recherche qui seront présentés dans les sections suivantes.

## **2.3 LES TRAVAUX DE NORMALISATION**

La fourniture de services dans les environnements mobiles regroupe plusieurs acteurs et partenaires dont les compétences sont diverses et nécessite donc le savoir-faire de plusieurs domaines. De ce fait, elle pose plusieurs défis techniques. La gestion

de la mobilité, la localisation, la personnalisation, la découverte de service, l'adaptabilité et la sensibilité au contexte sont autant de besoins nécessaires pour la création et la fourniture de services interactifs et adaptables.

Ces besoins nécessitent de nouvelles architectures capables de déployer, gérer, découvrir et adapter dynamiquement et de manière transparente des services multi réseaux, et multi terminaux. Ces architectures doivent, contrairement aux architectures existantes, supporter les propriétés non traditionnelles telles que la sensibilité au contexte et l'adaptabilité et doivent également offrir un accès à la fois aux services traditionnels mais aussi aux services à valeur ajoutée.

La problématique de la fourniture de services dans les futurs réseaux mobiles a été relevée par différentes instances de normalisation. Plusieurs travaux de standardisation ont traité cette problématique selon différents points de vue : terminal, réseaux de l'opérateur ou fournisseurs de services.

Les solutions orientées terminal (*MExE* [56], *WAP* [94] et *iMode* [42], *CC/PP*, *UAProf*, etc.), s'intéressent à l'adaptabilité des services aux différents terminaux mobiles. Les solutions et les architectures orientées réseau (telles que *CAMEL* [14], *SPIRITS* [39] et *PINT* [62]) s'intéressent aux problèmes liés à la fourniture et au contrôle des services dans des réseaux hétérogènes. Enfin, les solutions orientées fournisseur de services (telles qu'*OSA* [58] et les *Web Services* [48]) fournissent des interfaces réseau permettant aux fournisseurs de services d'accéder aux fonctionnalités et aux capacités du réseau quelle que soit sa nature.

Le constat relevé actuellement est l'absence d'une vision globale de cette problématique. En effet, les standards et les travaux de normalisation existants traitent partiellement la fourniture de service selon le point de vue terminal mobile, opérateur réseau ou fournisseur de services. Ainsi, il n'existe pas de solutions complètes qui répondent aux besoins de tous les acteurs impliqués dans le processus de fourniture de services.

## 2.4 LES TRAVAUX DE RECHERCHE

Plusieurs travaux issus de la recherche se sont intéressés aux problématiques de fourniture de service dans les environnements mobiles. Nous nous limitons dans cette section aux solutions apportées pour satisfaire la mobilité de l'utilisateur. Ces recherches semblent se concentrer sur deux thèmes : la mobilité du terminal et la mobilité personnelle [22]. Les recherches dans le domaine de la mobilité du terminal se sont intéressées à la fourniture de solutions se rapportant à quatre couches parmi les sept couches du modèle OSI (la couche liaison de données, la couche réseau, la couche transport ou même la couche session). Dans les dernières années, les recherches dans la couche réseau ont apporté de nouveaux standards comme le Mobile IP [60]. Les travaux sur la couche transport [53] ou sur la couche session [17] apportent plus de flexibilité, mais nécessitent des modifications dans le protocole ou l'ajout de code additionnel dans chaque terminal. Cela rend le déploiement de telles solutions peu évident. Dans le cas de la mobilité personnelle, les travaux de recherche restent plus limités. Cette problématique de mobilité a été traitée selon deux points de vue : la contactabilité [4][46][54] et la personnalisation [22][87]

À titre d'exemple, le *Mobile People Architecture* (MPA) [54] propose une solution basée sur un *Personal Online ID* (POID) afin de permettre aux utilisateurs d'être contactés utilisant des moyens de communication tels que le courriel, les messages ICQ ou téléphone). La solution proposée utilise également un *personal proxy* situé dans le réseau local de chaque utilisateur. Le POID permet une identification unique de l'utilisateur alors que le *personal proxy* permet de suivre les mouvements de l'utilisateur, ses préférences et même d'effectuer la conversion de contenu et de permettre le mappage entre les messages des protocoles de communication. Étant donné que tous les appels traversent le *personal proxy*, l'emplacement de l'utilisateur ainsi que son terminal peut être « cachés » par rapport aux autres utilisateurs. MPA garantit la contactabilité mais pas la personnalisation.

ICEBERG [93] fournit des fonctionnalités similaires au MPA, cependant, il a été conçu pour prendre en charge différents types de réseaux dans Internet. Cela permet,

par exemple, à un utilisateur avec un cellulaire d'être contacté par d'autres utilisateurs d'Internet et vice versa. Dans tous les cas, le contenu doit être adapté selon les caractéristiques du réseau. Cette adaptation de la communication est effectuée par l'introduction d'un « point de présence » dans ICEBERG appelée (iPOP). Cette solution est basée sur la plateforme « Ninja clustering » [88] pour fournir l'environnement d'exécution où les bibliothèques d'adaptation peuvent être téléchargées et configurées à la demande. ICEBERG, tout comme MPA, traite seulement la problématique de la contactabilité lors de la mobilité personnelle. TOPS [4], de son côté est un framework conçu pour fournir des services de la téléphonie sur IP. Il présente un répertoire de services utilisé pour localiser chaque utilisateur ainsi qu'une couche applicative pour le protocole de signalisation des communications téléphoniques. Ce framework utilise aussi un canal logique d'encapsulation et de décapsulation des messages pour séparer les applications téléphoniques du reste des couches de transport inférieures.

Il existe aussi d'autres solutions qui traitent la mobilité personnelle en utilisant la technologie des agents mobiles [46] [22] [86]. Ces solutions proposent l'utilisation d'un agent local et d'un agent distant, situé sur le site visité. Ces deux agents peuvent s'échanger des informations sur l'environnement d'exécution du service et de suivre ainsi le déplacement de l'utilisateur. D'autres solutions proposent une migration des agents d'un site à un autre et de transférer avec eux toutes informations utiles pour l'utilisation du service. Malheureusement, malgré leur apport scientifique, ces solutions sont en perte de vitesse puisque leur réalisation reste très coûteuse, voire impossible. Aussi, la migration des agents d'un site à un autre peut causer des délais d'attente qui ne permettent pas de garantir la continuité dans l'exécution des services ayant des contraintes de temps réel.

## **2.5 UTILISATION DE SIP POUR LA FOURNITURE DE SERVICES**

Depuis quelques années, le 3GPP a choisi le protocole SIP [74] (Session Initiation Protocol) pour être le standard d'établissement de sessions dans les IMS [1] (Internet Multimedia Subsystem). En utilisant SIP, plusieurs besoins liés à la fourniture de service

peuvent être résolus. Cette section propose une vue générale des besoins pour des services basés sur le protocole SIP :

- ✓ **Choix du paradigme de communication** : Le protocole SIP est basé sur un modèle de communication point-à-point (P2P) où chaque utilisateur peut à la fois agir comme client pour initier la communication ou comme serveur pour répondre à une demande.
- ✓ **Déploiement et gestion du service** : il existe sur le marché une grande variété de piles logicielles qui permettent la création et le déploiement de services basés sur SIP. Ces piles SIP sont généralement conformes aux dernières spécifications du protocole et permettent donc à plusieurs utilisateurs de piles SIP de fournisseurs différents de communiquer en utilisant le même service.
- ✓ **Personnalisation du service** : Le protocole SIP permet une communication personne-à-personne et non pas machine-à-machine. Le protocole SIP permet à chaque personne d'avoir une adresse unique qu'il pourra utiliser pour s'enregistrer et bénéficier du service. Ce concept permet de résoudre en grande partie les problèmes liés à la personnalisation du service puisqu'un utilisateur est capable de créer plusieurs adresses SIP et de définir ses préférences au sein de l'application.
- ✓ **Adaptabilité du service** : Lors de la signalisation d'une communication, le protocole SIP permet une négociation entre les terminaux afin de s'échanger leurs préférences. Des protocoles sous-jacents à SIP, comme SDP [40], permettent de décrire les capacités et les préférences de chaque terminal. Cela permet de résoudre une bonne partie de la problématique liée à l'adaptabilité du service.
- ✓ **Qualité de service** : contrairement aux réseaux à commutation de circuits, tels que les réseaux téléphoniques commutés où un circuit de communication est dédié pendant toute la durée de la communication, il

est possible dans les services basés sur IP comme les services SIP, que des paquets se perdent ou qu'ils ne soient pas acheminés à leur destinataire au bon moment. Ce problème est d'autant plus prononcé lorsqu'il s'agit de fournir une communication temps réel permettant un échange audio ou vidéo par exemple. Un service SIP peut faire appel à des protocoles adjacents comme le RTCP [76], qui fonctionne conjointement avec le RTP, afin de garantir la transmission des flux de données avec des informations relatives à la qualité du service (QoS). RTCP permet au récepteur de renvoyer vers l'émetteur un rapport incluant le nombre de paquets perdus, la gigue et le délai aller-retour.

- ✓ **Enregistrement et localisation de service** : SIP fournit des composantes logiques qui permettent à chaque abonné de s'enregistrer et de préciser ainsi l'adresse de la machine qui sera utilisée pour recevoir son service. SIP permet aussi la localisation des abonnés, mais, aussi, offre des mécanismes de redirection des appels.

Étant donné qu'il existe en ce moment un besoin de plus en plus croissant pour assurer la mobilité personnelle, celle du terminal et celle du service lors de la fourniture de service avec SIP, ce document propose, à la section 4, une étude de la mobilité selon SIP.

## 2.6 CONCLUSION

La première partie de ce document a été consacrée à l'étude de la problématique de la fourniture de services dans les environnements mobiles. Cette étude montre qu'il est important pour le concepteur d'une solution de fourniture de services de prendre en compte plusieurs besoins comme la mobilité, la personnalisation, l'adaptabilité, la qualité de service, la découverte de service, le déploiement, la gestion de la solution, etc. Cette étude se voulait générique et concerne différents types de fournisseur de service. Dans la réalité du marché de la téléphonie par exemple, il existe une certaine compétition entre fournisseurs de la téléphonie classique PSTN, de la téléphonie



cellulaire et de la téléphonie IP. Tous ces fournisseurs sont contraints à proposer des solutions satisfaisantes en terme de besoins de fourniture de service. La compétition entre ces différentes technologies incite chaque type de fournisseur à offrir les mêmes services que ses concurrents, mais aussi d'introduire de nouveaux services à valeur ajoutée. Il est possible de citer ici des travaux en cours à l'IETF à fin d'offrir à la téléphonie IP des services additionnels comme le service « Push To Talk » déjà offert dans la téléphonie cellulaire [63]. Aussi, il existe en ce moment un effort d'implémentation afin de doter la téléphonie IP de services habituellement fournis par la téléphonie classique, comme le cas du service Automatic Callback qui permet d'informer l'appelant lorsque la ligne de son correspondant se libère.

L'étude des besoins de fourniture de services en utilisant le protocole SIP montre bien que ce protocole permet de répondre à une bonne partie des besoins généralement requis lors de la mise en place d'une solution de fourniture de service. Il a été jugé utile de consacrer la section suivante (section 3) à l'étude du protocole SIP, des mécanismes qu'il offre, des composantes architecturales qu'il propose et de l'étendue des services qu'il est en mesure de fournir.

## **3 SIP : PROTOCOLE, MÉCANISMES ET SERVICES**

### **3.1 INTRODUCTION**

Le protocole d'établissement de session SIP a été défini pour la première fois dans l'IETF-RFC 2543 [41]. Un des objectifs initiaux de SIP était de rester aussi simple que possible, et de respecter les principes « classique » d'ingénierie de protocoles réseau, comme l'isolation entre les couches protocolaires.

SIP est un protocole de contrôle de la signalisation qui fait partie de la couche application. Le protocole SIP, inspiré largement du protocole HTTP, vise principalement l'établissement, la modification et la terminaison des sessions entre deux ou plusieurs participants pour l'échange de données. SIP se distingue par rapport à d'autres protocoles de signalisation, comme le H.323, par sa flexibilité et son extensibilité. Il a été implémenté par un grand nombre de vendeurs pour initialement offrir des services VoIP. Les extensions de SIP lui ont permis de devenir un candidat potentiel pour la mise en place de systèmes de signalisation pour l'établissement de conférence, pour la mise en place de service de présence, de service de messagerie instantanée et de notification d'évènements d'une façon plus large.

#### **3.1.1 Syntaxe des messages**

Les concepteurs de SIP, ayant souhaité retrouver la simplicité de HTTP, se sont orientés vers un protocole basé sur des échanges de messages "plain text", réutilisant en grande partie la même nomenclature que HTTP. SIP définit à la base six méthodes qui peuvent être classées en deux catégories : Les méthodes REGISTER et OPTIONS, utilisées hors du contexte d'un dialogue SIP, et les méthodes INVITE, ACK, CANCEL et BYE directement liés à la gestion du dialogue. De nouvelles méthodes comme SUBSCRIBE, NOTIFY, PRACK, INFO ont été aussi définies par l'IETF pour étendre les méthodes SIP de base.

Un message SIP est composé d'une partie d'entête et d'une partie du corps de message (Payload). Les entêtes SIP peuvent être obligatoires ou optionnels

dépendamment de la méthode utilisée, de l'état d'avancement du dialogue SIP et des besoins du service. Il faut aussi noter que SIP permet de créer des en-têtes additionnels (extra-headers) en cas de besoin. Il est possible de citer les en-têtes suivants :

- ✓ **Via** : indique le protocole de transport utilisé ainsi que la route de la requête. La requête SIP peut transiter par plusieurs serveurs mandataires et chaque serveur peut ajouter une ligne à ce champ,
- ✓ **From** : indique l'adresse de l'appelant,
- ✓ **To** : indique l'adresse de l'appelé,
- ✓ **Call-Id** : unique pour chaque dialogue, elle contient aussi l'adresse hôte. La valeur de cet entête doit être la même pour tous les messages d'une même transaction au sein d'un dialogue SIP,
- ✓ **Cseq** : commence par un nombre aléatoire et identifie chaque message SIP d'une façon séquentielle. (s'incrémente pour chaque message d'une même transaction SIP),
- ✓ **Contact** : indique une ou plusieurs adresses qui peuvent être utilisées pour contacter un utilisateur.

### **3.1.2 Mécanisme d'envoi de messages**

Lorsque SIP est utilisé sur des protocoles de transport non fiables, la fiabilité des transactions autres qu'INVITE se fonde sur la retransmission de message. L'émetteur transmettra de nouveau la requête s'il ne reçoit pas de réponse provisoire ou définitive en moins de 500 ms (ou une meilleure estimation du temps d'aller-retour, si l'agent utilisateur la calcule). Il continuera à le retransmettre jusqu'à réception d'une réponse, en doublant l'intervalle de retransmission à chaque essai, jusqu'à atteindre 4 s. Le délai d'attente peut atteindre les 32 s si aucune réponse n'est reçue. Après ce délai, la transaction échoue.

SIP est un protocole qui peut être implémenté en utilisant un protocole de transport en mode connecté (TCP) ou en mode non connecté (UDP). Pour pallier au problème de perte de message SIP, le mécanisme requête/réponse est utilisé par défaut

pour toutes les requêtes SIP mise à part la méthode INVITE qui est basée sur le mécanisme requête/réponse/acquittement.

## 3.2 LE DIALOGUE SIP

### 3.2.1 Introduction

SIP propose deux différents mécanismes pour la création, la modification, le rafraîchissement et la terminaison d'un dialogue SIP. Même s'il existe des cas où l'utilisation de l'une ou de l'autre de ces deux méthodes n'altère pas le fonctionnement du service, il est important de présenter la spécification de chaque méthode et les différences dans le fonctionnement de leur mécanisme respectif. Il est important de souligner ici qu'il existe un certain abus de langage dans la littérature qui fait que les termes « dialogue » et « session » sont utilisés d'une façon parfois peu appropriée. Le terme « dialogue » devrait être utilisé dans un contexte de signalisation SIP, alors que le terme « session » devrait être évoqué dans le contexte média.

Un dialogue peut être vu comme étant un concept qui permet de représenter une relation point à point entre deux utilisateurs SIP. En plus d'organiser l'ordre d'envoi et de réception des messages, le dialogue représente un contexte qui permet d'interpréter les messages SIP. Un dialogue est identifié dans chaque agent utilisateur par un identifiant unique qui est composé de la valeur du *Call-ID*, la valeur du *tag* local et du *tag* du destinataire. Un *tag* est une variable aléatoire qui s'ajoute à l'URI dans les champs entête *To* et *From*.

### 3.2.2 Création du dialogue SIP

SIP propose deux mécanismes différents pour la création d'un dialogue : la méthode *INVITE* et la méthode *SUBSCRIBE*. L'utilisation de chaque méthode est adaptée à un besoin particulier. La Figure 2 montre l'établissement d'un dialogue en utilisant ces deux méthodes :

- ✓ **Le mécanisme INVITE :** À l'origine, le mécanisme de la méthode INVITE a été conçu pour des services de la téléphonie sur IP. Le mécanisme de création d'un dialogue avec la méthode INVITE respecte le modèle requête/réponse/acquittement. Avant la réception d'une réponse finale 2xx, 3xx, 4xx, 5xx ou 6xx, l'initiateur du dialogue avec INVITE peut recevoir des réponses provisionnelles du type 1xx (100 TRYING ou 180 RINGING par exemple). Cela permet à SIP de s'approcher au maximum de la logique de fonctionnement des systèmes de téléphonie traditionnelle (PSTN). Lors de la réception d'une réponse, l'initiateur de la requête INVITE envoie un acquittement (ACK) pour confirmer la réception de la réponse. La requête ACK est utilisée exclusivement pour le mécanisme INVITE.
  
- ✓ **Le mécanisme SUBSCRIBE/NOTIFY :** Depuis la publication du RFC 3265 [65], la création d'un dialogue SIP est devenue possible à l'aide de la méthode SUBSCRIBE. Contrairement à la méthode INVITE, SUBSCRIBE utilise le modèle requête/réponse. La confirmation de l'établissement du dialogue SIP se fait lors de la réception de la classe 2xx OK ou du message NOTIFY. Cela rend la méthode SUBSCRIBE assez liée à la méthode NOTIFY, d'où l'appellation « mécanisme SUBSCRIBE/NOTIFY ». Ce mécanisme est utilisé pour demander une souscription/abonnement (*subscription*) à des ressources ou à des états pour recevoir des notifications lorsque des événements relatifs à ses ressources ou à ses états se produisent.

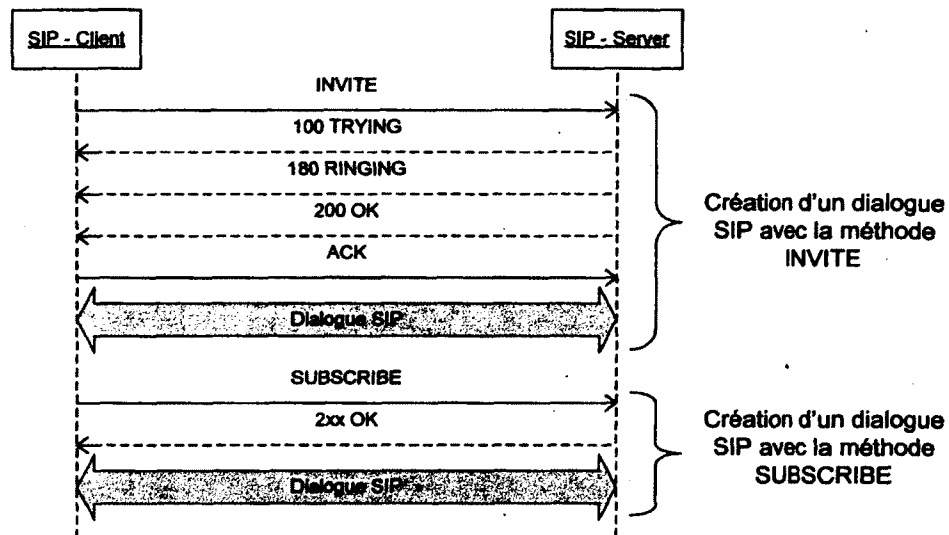


Figure 2 : Les mécanismes de création d'un dialogue SIP

### 3.2.3 Modification du dialogue

Il est possible d'apporter des modifications à un dialogue SIP déjà établi. Pour cela, il suffit d'envoyer une méthode SIP (INVITE ou SUBSCRIBE) avec les nouveaux paramètres. Cette requête devrait porter les mêmes valeurs des champs d'entête du *Call-Id*, du *From Tag* et du *To Tag* pour être traitée au sein du dialogue déjà établi.

Chaque dialogue SIP est caractérisé par une durée d'expiration. Pour pouvoir modifier cette durée, il est important de renvoyer la même requête initiale i.e INVITE ou SUBSCRIBE, avec la même valeur du *Call-Id* et en utilisant les mêmes *tags* des entêtes *From* et du *To* tout en incrémentant le numéro de séquence *CSeq* et en précisant la nouvelle valeur d'expiration. Le rafraîchissement du dialogue devra se faire avant son expiration. Pour le cas du mécanisme SUBSCRIBE/NOTIFY, il est possible de modifier l'état du dialogue en ajoutant de nouvelles souscriptions.

### 3.2.4 Terminaison du dialogue

Un dialogue initié par la méthode INVITE peut être annulé avec la méthode CANCEL si aucune réponse finale n'est encore reçue. Autrement, le dialogue peut être terminé à l'aide de la méthode BYE. Pour le cas du mécanisme SUBSCRIBE/NOTIFY, il est possible de mettre fin à la souscription en envoyant un message de rafraîchissement de dialogue (voir la section précédente) en précisant la valeur « 0 » dans le champ d'entête « expires » de la requête. La fin de la souscription est confirmée par la réception d'un message NOTIFY final avec la valeur « terminated » dans le champ d'entête « Subscription-State ». Le dialogue SIP se termine lorsque toutes les souscriptions qu'il supporte sont terminées. Si dans un même dialogue, les mécanismes INVITE et SUBSCRIBE/NOTIFY sont conjointement utilisés, la terminaison du dialogue ne peut se faire que lorsque toutes les souscriptions soient terminées et qu'un message BYE est utilisé.

## 3.3 LE PRINCIPE DE LA NOTIFICATION D'ÉVÈNEMENT DE SIP

Le principe de la notification d'évènement, présenté brièvement dans la section précédente, a été initialement introduit par le RFC 3265 (SIP – Specific Event Notification) [65], fournit un *framework* extensible qui permet aux agents utilisateurs SIP de recevoir des notifications à partir d'autres agents SIP lorsqu'un évènement particulier se produit. Utilisé habituellement dans les systèmes de présence et dans la messagerie instantanée [21] [20] [67], le mécanisme SUBSCRIBE/NOTIFY peut être généralisé à tout système de communication qui nécessite une synchronisation d'évènements.

Une souscription peut avoir trois états différents : actif (active), en attente (pending) ou terminé (terminated). Ces états peuvent être contrôlés par l'introduction d'un observateur d'état (watcher) [69] [68]. La Figure 3 présente un exemple de notification d'évènement utilisant le mécanisme SUBSCRIBE/NOTIFY.

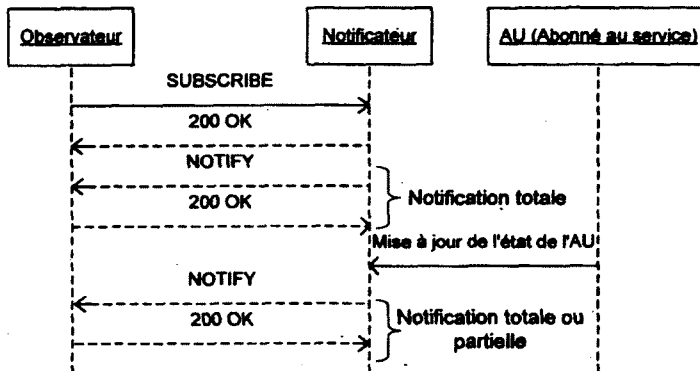


Figure 3 : Diagramme de séquence d'un exemple de notification d'évènement

### 3.4 LA DESCRIPTION DE LA SESSION

Le SDP (Session Description Protocol) est un protocole normalisé par l'IETF et introduit initialement dans le RFC 2327 [40]. Le protocole SDP peut être utilisé pour la description d'une session multimédia lors de son annonce, de l'envoi de l'invitation ou de toute autre forme d'initiation de session multimédia. Le contenu SDP est textuel et il est donc possible de l'ajouter au corps (*payload*) des messages du protocole de transport utilisé (SIP, SAP, RTSP, HTTP, etc.).

Le protocole SDP est bien populaire dans les services de communication basés sur le protocole SIP pour la signalisation des sessions et sur le protocole RTP pour la transmission et la gestion des flux multimédias (voix, vidéo, etc.). Ce protocole SDP utilise un système de codage basé sur des lignes de texte de la forme <type> = <value>. Les informations incluses dans SDP peuvent être obligatoires ou optionnelles et peuvent se classer en trois catégories :

- ✓ Description de la session : contient obligatoirement la version du protocole, l'identifiant de la session, la description du propriétaire de la session et le nom de la session. Aussi, la description de la session peut contenir des données optionnelles comme des informations sur la bande



passante, l'adresse courriel du propriétaire de la session ou son numéro de téléphone.

- ✓ Description du temps : contient obligatoirement la période pendant laquelle la session est active. Optionnellement, lorsque la session a été active sur plusieurs intervalles de temps, il est possible d'ajouter ce type d'information.
- ✓ Description du média : contient obligatoirement le nom de chaque média utilisé ainsi que les adresses de transport (l'adresse IP et le port destinataire). Optionnellement, il est possible d'ajouter des informations sur le titre du média, sur la bande passante, sur la clé de cryptage, mais aussi il est possible de définir des attributs supplémentaires pour définir le mode d'envoi/réception (*sendrecv*, *sendonly*, *recvonly*), la qualité du média, le nombre d'images par seconde, etc..

Il est important aussi de préciser que la description de la session et des médias utilisés devrait contenir l'adresse locale ainsi que le numéro de port disponible pour recevoir le flux. Cette information pourrait être incluse au début de la description de la session pour s'appliquer à tous les médias utilisés et peut aussi être spécifique à chaque type de média inclus dans la session. Il faut aussi noter que l'adresse de destination du flux média est indépendante de l'adresse de destination du dialogue établie ou à établir.

Par ailleurs, il existe en ce moment des travaux au sein de l'IETF pour normaliser un nouveau protocole baptisé MSRP (Message Sessions Relay Protocol) [44]. MSRP est un format inspiré de SDP et qui permet d'ouvrir une session média entre les hôtes qui s'échangent des messages textuels. Le contenu MSRP pourrait être échangé en début de session à l'aide de protocole de signalisation comme SIP. L'utilisation du protocole MSRP est intéressante pour des services basés sur l'échange de messages textuels comme l'exemple de la messagerie instantanée.

## 3.5 LE CONTRÔLE DE COMMUNICATION AVEC SIP

### 3.5.1 Introduction

Contrairement à d'autres protocoles web comme HTTP ou FTP, SIP est un protocole qui repose sur un modèle de communication point à point (P2P). Ainsi, chaque utilisateur SIP a la possibilité d'agir comme client et d'initier le dialogue. Un Agent Utilisateur SIP (A.U.S) devrait pouvoir fonctionner aussi bien en tant que client qu'en tant que serveur.

SIP utilise un vocabulaire spécifique pour désigner les différentes fonctions impliquées dans le contrôle des terminaux et des passerelles. Il définit à la base quatre entités logiques de contrôle de communications :

- ✓ Le serveur d'enregistrement (*registrar*) pour effectuer l'enregistrement des terminaux;
- ✓ Le serveur de localisation (*location server*) pour la localisation courante (sous forme d'adresse URI SIP ou d'adresse IP) des utilisateurs;
- ✓ Un serveur de redirection (*redirection server*) qui peut être utilisé en conjonction avec un *registrar* pour rediriger les appels vers la localisation courante de l'utilisateur;
- ✓ Un serveur mandataire (*proxy server*) qui peut être de trois types différents : *stateless* (en n'intervenant pas dans l'état de la communication), *stateful* (en contrôlant l'état de la communication SIP), ou *forking* (en étant capable dupliquer un message INVITE ou SUBSCRIBE vers plusieurs destinataires).

### 3.5.2 Fonction d'enregistrement

Un *registrar* (ou dispositif d'enregistrement) est une composante logique de l'architecture SIP responsable des traitements des messages REGISTER et sert à mémoriser la localisation courante d'un user agent. L'adresse IP d'un user agent peut changer dans de nombreuses circonstances : connexion à Internet à travers un

fournisseur d'accès qui alloue des adresses dynamiques, connexion sur un réseau local utilisant le protocole de DHCP (Dynamic Host Configuration Protocol) ou déplacement physique du terminal et changement du réseau d'accès. Pour pouvoir atteindre le terminal à partir de son adresse IP, il est nécessaire de mémoriser l'association entre une adresse IP éventuellement dynamique et l'URI SIP de l'utilisateur.

L'information d'enregistrement peut être expirée et disparaître du *registrar* si elle n'est pas rafraîchie continuellement. Par défaut, l'expiration de l'enregistrement est fixée à une heure. Cette durée peut être modifiée à travers l'entête « expires » de la méthode REGISTRAR.

### 3.5.3 Fonction de redirection

Un serveur de redirection peut être utilisé conjointement avec un registrar pour rediriger les appels vers la localisation courante de l'utilisateur appelé. Il peut également être utilisé comme un moyen de base de distribution d'appels. Dans tous les cas, il répond à une requête INVITE par un message de type 3xx ou bien rejette l'appel.

### 3.5.4 Fonction du mandataire

Le mandataire (proxy) est un dispositif qui agit comme un serveur d'un côté (il reçoit des requêtes), et comme un client de l'autre (il retransmet des requêtes). Rigoureusement, un mandataire devrait être presque entièrement transparent vis-à-vis des messages des A.U, se contentant de les relayer et d'y apporter des modifications mineures. Un Mandataire peut par exemple relayer la requête sans aucun changement vers sa destination finale, décider d'en vérifier la validité, authentifier des utilisateurs, dupliquer des requêtes, résoudre des adresses, annuler des appels en cours d'établissement, etc.

## **3.6 EXEMPLES DE SERVICES DE COMMUNICATION BASÉS SUR SIP**

### **3.6.1 Les services basés sur un échange de flux de données direct**

Initialement, le protocole SIP a été conçu pour l'établissement de session média à travers des mécanismes de création et de gestion d'un dialogue qui permettaient l'échange de données relatives à la description de session. À sa création, SIP était principalement destiné aux applications de téléphonie sur IP offrant des services basés sur la Voix sur IP (VoIP), du Multimédia sur IP (MMoIP) et des communications en mode-conférence. L'utilisation de protocole de description de session comme SDP combiné à l'utilisation de protocoles de transport de contenu multimédia en temps réel comme RTP, permet l'échange de différents contenus audio et vidéo.

### **3.6.2 Les services basés sur l'échange de messages textuels**

Basé sur le protocole HTTP, le protocole SIP présente plusieurs souplesses pour l'acheminement des données textuelles entre utilisateurs. Des méthodes comme MESSAGE ou NOTIFY peuvent être utilisées au sein d'un dialogue SIP pour permettre un tel échange. Des services comme la messagerie instantanée peuvent bénéficier de ces solutions. D'un autre côté, il est possible d'utiliser le principe de notification d'évènement de SIP (voir section 3.3) pour fournir des services basés sur la présence.

### **3.6.3 Les services combinés**

Il est possible de concevoir de nouveaux services basés sur SIP par la combinaison de plusieurs services. Un service de base, à l'exemple de la messagerie instantanée, peut bénéficier des données d'un service de présence des utilisateurs, et s'assurer ainsi de la disponibilité de la personne avant même d'envoyer le message. Cette information est fournie à travers le transfert de l'état de l'utilisateur (disponible, absent, non connecté, etc.).

D'autres services comme le PTT (Push to Talk) peuvent aussi être créés par la combinaison du service de conférence avec celui de la présence. Un tel service permettra à un utilisateur de s'assurer que les personnes de sa liste de contacts sont en mesure d'accepter la conférence ou pas avant même de les inviter.

### **3.7 CONCLUSION**

L'objectif de cette étude du protocole SIP était de présenter les mécanismes offerts, l'extensibilité de la syntaxe des messages utilisés, la diversité des fonctions des composantes intégrées (fonction d'enregistrement, fonction de redirection, fonction du mandataire) et les capacités de création de services. Le déploiement de SIP dans les communications en mode-conférence prouve l'extensibilité, la souplesse et la richesse de ce protocole. Cependant, nous nous posons des questions sur les limites de ce protocole et sur l'étendue de ses capacités à supporter les différents aspects de mobilité généralement demandés pour la fourniture de service. La section suivante apporte des éléments de réponse à ces questions.

## 4 LA MOBILITÉ SELON SIP

### 4.1 INTRODUCTION

L'apparition et la prolifération des terminaux mobiles légers supportant les technologies de communication sans fil basées sur IP ont largement participé à la création d'un besoin en termes de mobilité dans les réseaux IP. Cette mobilité touche différents aspects de la communication. Dans le contexte d'une communication basée sur le protocole SIP, il est possible de distinguer quatre types de mobilité : la mobilité du terminal, la mobilité de la session, la mobilité de l'environnement du service et la mobilité personnelle de l'utilisateur (voir Figure 4) :

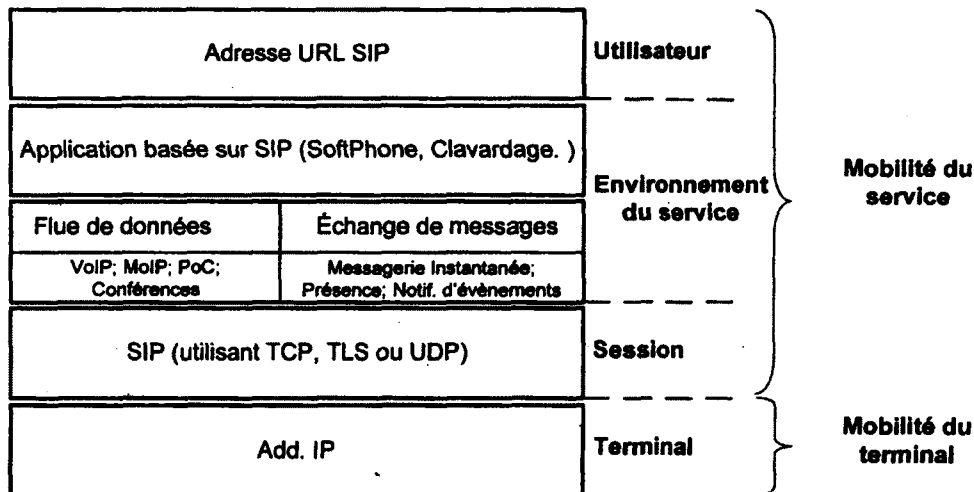


Figure 4 : Modèle de couches d'une application SIP mettant en évidence la mobilité

### 4.2 MOBILITÉ DU TERMINAL SIP

La mobilité du terminal permet à la machine de pouvoir continuer à utiliser le service même lorsqu'elle change de réseau. Il existe principalement deux cas : la mobilité avant l'établissement de la communication (précommunication) et la mobilité après l'établissement de la communication (mi-communication)

### 4.2.1 Mobilité en précommunication

La mobilité du terminal durant la période de préappel est relativement moins complexe à mettre en œuvre. Lorsque le NM (noeud mobile) obtient une nouvelle adresse avant de recevoir ou d'initier un dialogue SIP, il peut toujours se réenregistrer. L'étape de réenregistrement consiste à envoyer une requête REGISTER au *registraire* pour indiquer la nouvelle adresse IP du terminal. Il est cependant important que la couche applicative puisse détecter les changements de l'adresse IP sur la machine. Pour cela, le client devrait interroger d'une façon continue (poll) le système d'exploitation. Une solution améliorée serait de permettre à l'application de s'inscrire auprès du système d'exploitation afin qu'elle puisse être notifiée lorsque l'adresse IP de la machine change.

Dans d'autres situations, l'utilisateur change d'adresse IP, mais aussi d'adresse SIP lorsqu'il visite un nouveau réseau. Dans ce cas, il devrait indiquer au serveur de redirection ses nouvelles adresses. La Figure 6 illustre un tel scénario. L'appelant (NC) sera ainsi notifié de la nouvelle adresse en recevant le message (2) en réponse à son message INVITE (1). Ce message (2) contient la nouvelle adresse que le NC doit composer. Il est important de noter ici qu'il existe d'autres implémentations SIP qui permettent de rediriger l'appel à la nouvelle adresse de façon plus transparente à l'appelant. Dans une telle implémentation, le serveur de redirection pourra communiquer directement avec son Proxy et envoyer l'INVITE initiale à la bonne adresse.

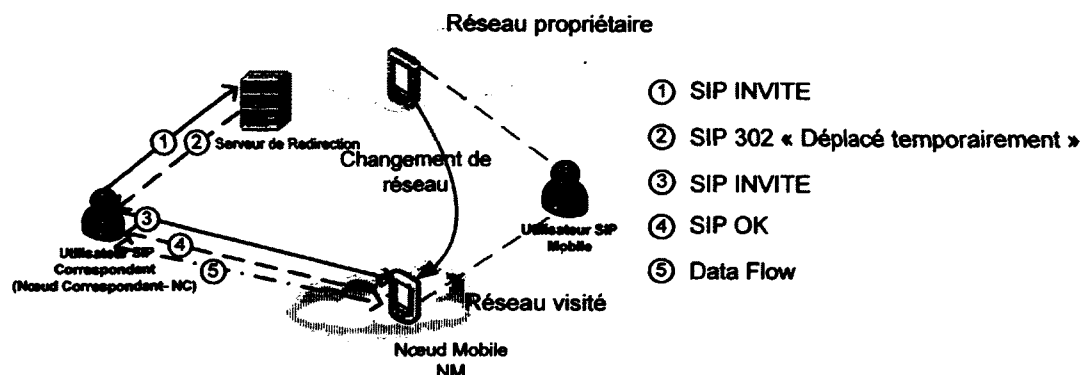


Figure 5 : Scénario d'échange de messages lors de la mobilité en pré-communication

#### 4.2.2 Mobilité en mi-communication

Pour la mobilité du terminal durant l'appel (mi-communication), le NM envoie une requête INVITE au NC (Nœud Correspondant) sans passer par un *Mandataire*. (Le mandataire peut être utilisé uniquement si, dans la requête initiale d'appel, l'entête *Record-Route* est utilisé). Cette requête INVITE devrait contenir une mise à jour de la description de la session et donc indiquer la nouvelle adresse IP utilisée. Ainsi, la mise à jour de la nouvelle localité prendra un délai à sens unique dès que la couche applicative détecte la nouvelle adresse. Pour des accès à large bande, le délai sera probablement égal au délai de propagation en plus de quelques millisecondes. Cependant, les systèmes à bande passante limitée peuvent imposer des délais de plusieurs dizaines de millisecondes [77]. Dans le cas où le nouveau réseau attribue une nouvelle adresse IP au NM, cela causera l'interruption de la communication en cours et l'établissement d'une nouvelle session. Avec le protocole IP version 6, le NM peut garder son adresse IP, dans ce cas, il est possible de ne pas interrompre la communication avec le NC. Il existe des solutions qui traitent le problème de la mobilité en mi-communication à partir des couches réseau inférieures. Il est possible de citer la solution apportée par le protocole *Mobile-IP* pour permettre aux messages envoyés à l'ancienne adresse d'être routés vers la nouvelle adresse. Le déploiement de cette technologie semble être coûteux puisqu'il faut que le routeur du site visité communique avec le routeur du site d'origine. La technologie *Mobile-IP* [61] devrait garantir un délai de transition moins important que



celui venant de la couche applicative, basé sur le protocole SIP. Cette problématique ne sera pas traitée dans ce travail.

### 4.3 MOBILITÉ DE L'ENVIRONNEMENT DU SERVICE SIP

La mobilité du service permet aux utilisateurs de maintenir un accès à leurs services même en cas de déplacement, de changement de terminal ou de réseau de fournisseur de service. Dans un environnement de VoIP par exemple, il est important pour l'utilisateur de pouvoir maintenir des informations utiles à l'utilisation de son service i.e. la liste des contacts, l'historique des appels, les préférences du média, les règles à appliquer aux appels reçus, etc.

Pour assurer la mobilité de telles informations, il est important de pouvoir les coder pour faciliter leur transfert entre terminaux. Différents formats peuvent être utilisés. Comme exemple, le format *CPL* (Call Processing Language) permet de décrire les instructions de manipulation des appels (règles d'appel) [71][50][51]. D'un autre côté, le format *XCAP* [70] peut-être utilisé pour la description, la récupération et la mise à jour d'une liste de contacts pour chaque utilisateur. D'une façon générale, il est possible de décrire les informations relatives à l'utilisation d'un service dans un format portable et indépendant du système. L'utilisation de solutions centralisée pour stocker les données relatives à l'utilisation du service peut aider à la réalisation de cette mobilité de l'environnement du service. Le modèle de codage selon XML peut être une solution très satisfaisante et même très efficace du moment que l'information présente une hiérarchisation qui facilite le décodage des données. Cependant, il existe une grande variété de formats XML prédéfinis, normalisés, dans la majorité du temps, par l'organisme W3C. Il est important de s'assurer que le format à créer n'a pas été défini préalablement. Il est aussi possible d'utiliser des formats existants et de proposer des extensions pour mieux les adapter aux besoins du problème à traiter.

Par ailleurs, il devrait être possible de mettre à jour les données de définition du service de n'importe quel terminal. Une des solutions pour assurer la mobilité du service serait de permettre à l'utilisateur de transporter ces informations avec lui, sous forme

d'une puce mémoire, une clé USB par exemple ou dans la carte SIM pour les cellulaires. Cette solution peut présenter certaines limitations puisque les changements ne peuvent se faire sur des terminaux distants. Une solution plus efficace serait de faire le stockage des informations relatives au service dans un serveur et de les rendre accessibles pour tous les terminaux que possède l'utilisateur.

Les formats de plusieurs éléments de données, comme la liste de contacts, ou la configuration de l'interface utilisateur pourrait être standardisée. SIP offre pour cela des services de synchronisation de l'information pour différents clients à partir d'un serveur central. L'architecture générale d'un tel système peut être en partie prédite par l'installation d'un serveur local associé aux adresses des utilisateurs. Par exemple, un utilisateur identifié par une adresse SIP : « *alice@USherbrooke.ca* » utiliserait un serveur SIP donc le nom de domaine serait USherbrooke.ca afin d'y stocker les informations du service.

Il est possible qu'une même personne possède plusieurs adresses SIP. Il faut donc prévoir des solutions qui permettraient à l'utilisateur de bénéficier des mêmes informations reliées au service indépendamment de l'adresse SIP qu'il utilise. Cette solution devrait permettre à différents domaines de s'échanger des informations reliées au service.

#### 4.4 MOBILITÉ DE L'UTILISATEUR SIP

La mobilité personnelle d'un utilisateur SIP permet de s'adresser à un seul utilisateur situé sur différentes machines en utilisant la même adresse logique [78]. L'opération de mappage est possible à la fois de 1 à n (une adresse pour plusieurs terminaux potentiels) ou de m à 1 (plusieurs adresses pour rejoindre un seul terminal).

Dans la pratique, il peut exister une certaine complication lors de la mise en place de ce genre de solution. Deux cas se présentent :

- ✓ Une seule adresse SIP peut s'enregistrer simultanément en utilisant l'adresse IP de plusieurs terminaux. Dans ce cas, il est possible de recevoir

le même appel sur différents terminaux. Cela présente de nouvelles problématiques pour permettre à plusieurs applications situées sur plusieurs terminaux de coopérer pour traiter un même appel. (voir Figure 6)

- ✓ Faire pointer plusieurs adresses SIP sur une même adresse IP. Ainsi, la même personne peut faire enregistrer toutes ses adresses SIP en utilisant une seule application installée sur un seul terminal. Par analogie au système de téléphonie cellulaire, cela revient à équiper un même téléphone de plusieurs cartes SIM. (Voir Figure 7)

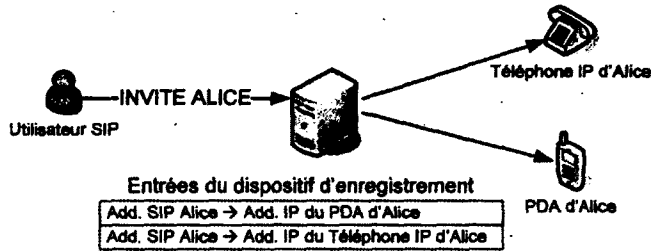


Figure 6 : Une seule adresse SIP pour plusieurs terminaux

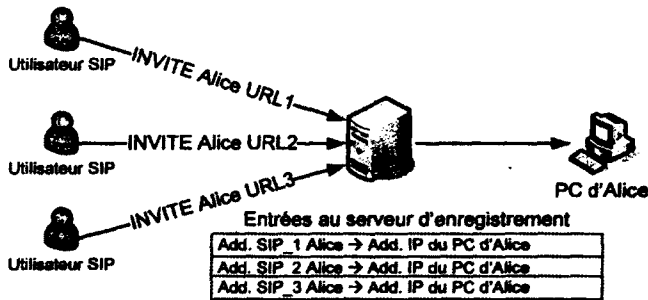


Figure 7 : Plusieurs adresses SIP pour une seule adresse IP

## 4.5 MOBILITÉ DE LA SESSION SIP

La mobilité de la session permet à une session d'être transférée d'un terminal à un autre et donc d'une adresse SIP à une autre. Les deux terminaux peuvent appartenir à la même personne qui possède plusieurs adresses SIP. La problématique de la mobilité de

session peut être traitée selon deux contextes : avant l'établissement final de l'appel (communication en cours d'établissement) et en cours d'appel (communication établie).

#### 4.5.1 Mobilité d'une session en cours d'établissement

La Mobilité de la session en précommunication consiste à transférer la session lorsqu'elle est en cours d'établissement d'un NM vers un NC. Il est possible de distinguer deux scénarios pour ce type de mobilité : mobilité lors d'un appel entrant (*dial-in*) et mobilité lors d'un appel sortant (*dial-out*).

- **Appel entrant** : le NM reçoit un appel et désire le transférer vers un NC. Dans ce cas, NC peut traiter (accepter, refuser, modifier ou terminer), d'une manière distante, la demande de communication destinée initialement au NM. Il est possible de distinguer deux modes de mobilité de session en cours d'établissement lors d'un appel entrant:

- ✓ **Sans contrôle** : Dans le premier mode, NM répond au NAV par une requête de redirection d'appel vers NC (message 2 de la Figure 8.a). Ainsi, le NAV abandonne la tentative de joindre le NM et régénère une deuxième requête d'initiation d'appel vers le NC. Cette méthode est appelée « sans contrôle » puisque le NM ne peut plus suivre l'évolution de la communication.
- ✓ **Avec contrôle** : L'intégration d'une telle option nécessite la coopération entre le NM et le NC pour traiter l'appel reçu, comme illustré dans la Figure 8.b. D'après les études effectuées, il n'a pas été possible de trouver des mécanismes disponibles pour répondre à ce genre de situation. L'introduction d'un nouveau système de communication spécifique à cette situation semble nécessaire pour assurer la prise de décision en mode coopératif (avec contrôle) lors de la réception d'un appel. La section suivante (section 5) se consacrera à l'étude de ce cas de mobilité.

- **Appel sortant** : Ce deuxième scénario présente le cas où le NC devient l'appelant et initie donc la procédure d'appel vers un terminal NAV, mais en utilisant le NM. En d'autres termes, le NC devrait commander le NM pour qu'il compose l'appel comme illustré dans la Figure 9.

- ✓ **Sans contrôle** : Dans ce premier mode, le NC initie l'appel, mais ne sera pas en mesure de suivre son évolution ni même de le contrôler. C'est le NM qui se charge d'assurer le suivi de la création de la session. Pour ce faire, le NC envoie une requête de type INVITE (Figure 9.a) où il indique l'adresse du NM en utilisant l'entête « *Contact* » alors que l'adresse du NAV est indiquée dans l'entête *To*. Le NM devrait agir comme un Proxy SIP pour pouvoir interpréter cette requête et la réacheminer au NAV. Cette méthode ne permet pas au NC d'établir une communication avec le NAV puisqu'aucune négociation des capacités médias n'est possible.
- ✓ **Avec contrôle** : L'intégration d'une telle option nécessite la coopération entre le NM et le NC pour traiter l'appel à établir, comme illustré dans la Figure 9.b. La fourniture d'un tel service ne semble pas évidente puisque SIP ne fournit pas de mécanisme qui répond exactement au besoin du problème. L'introduction d'un nouveau système de communication, spécifique à cette situation, semble nécessaire pour assurer la prise de décision en mode coopératif lors de la composition d'un appel. La section suivante (section 5) est consacrée à l'étude de ce cas de mobilité.

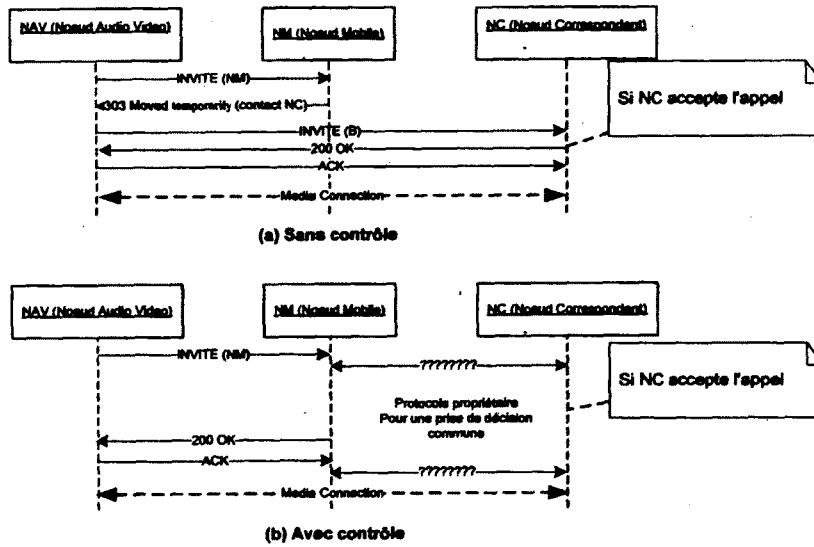


Figure 8 : Illustration de la mobilité de session en cours d'établissement lors d'un appel entrant

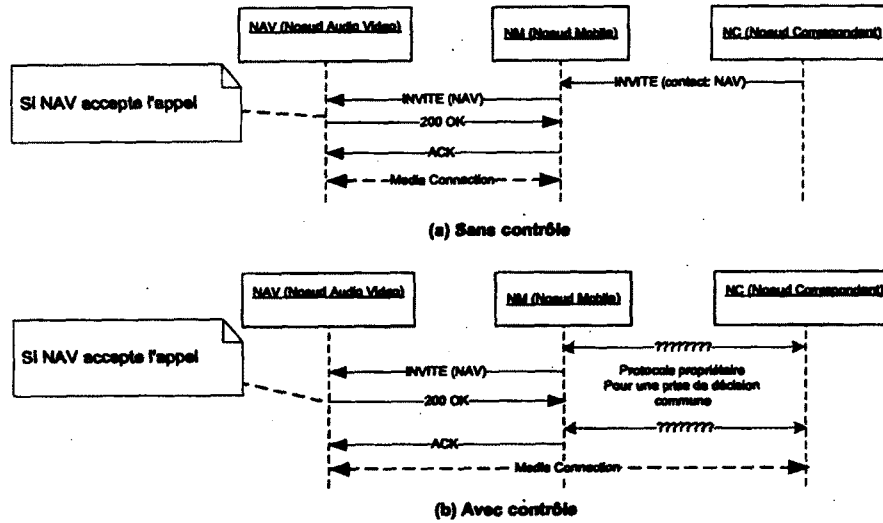


Figure 9 : Illustration de la mobilité de session en cours d'établissement lors d'un appel sortant

#### **4.5.2 Mobilité d'une session déjà établie**

Il s'agit ici d'assurer le transfert, mais aussi la récupération des sessions d'un premier terminal vers un deuxième lorsque le premier est en cours de communication. Pour illustrer un tel scénario, il est possible de considérer le cas d'un utilisateur en cours de communication multimédia sur IP (VoIP, MMoIP, Messagerie Instantanée, etc.) en utilisant son ordinateur de bureau. En déployant l'option de mobilité de session, l'utilisateur peut quitter son bureau en transférant sa communication en cours vers un autre terminal portatif (PDA par exemple). Ce même utilisateur, en sortant de son bureau, entre dans une salle équipée d'un vidéoprojecteur. Dans ce cas, il pourra transférer la partie vidéo de sa session vers ce projecteur et profiter ainsi d'une meilleure qualité d'image tout en continuant sa conversation.

Pour réaliser un tel scénario, plusieurs mécanismes offerts par SIP peuvent être déployés. Deux modes peuvent être utilisés pour implémenter un tel transfert. Le premier permet de garder le contrôle après avoir fait le transfert alors que le deuxième mode cause une perte de contrôle sur la session transférée. Le transfert peut aussi être total ou partiel et doit supporter l'option de récupération. Ces mécanismes sont étudiés dans la section 6 de ce document (page 57).

#### **4.6 DISCUSSION ET CONCLUSION**

Dans cette étude de la mobilité selon SIP, l'objectif était de trouver des solutions basées sur SIP pour répondre aux besoins de mobilité des services SIP. Cela revient à donner à ces services les moyens d'assurer eux-mêmes leur mobilité sur les différents niveaux déjà présentés (terminal, session, environnement du service et utilisateur SIP).

Les solutions apportées au niveau de la mobilité du terminal et lors des mi-communications peuvent être améliorées par l'utilisation de solutions traitant les couches inférieures du modèle OSI (couche liaison de donnée et couche réseau).

Les solutions apportées au niveau de la mobilité de l'utilisateur SIP peuvent causer des complications lors de leur mise en place : pour le cas de plusieurs adresses SIP

enregistrées pour une seule adresse IP, il est important que l'application puisse traiter les appels entrants destinés à plusieurs adresses SIP de la même manière que lorsqu'il s'agit d'une seule adresse. Dans le deuxième cas, lorsque plusieurs terminaux utilisent la même adresse SIP, il est possible que des problèmes de cohérences se produisent lorsque l'utilisateur manipule la même communication en utilisant plusieurs terminaux d'une façon interchangeable. Par exemple, accepter l'appel avec un premier terminal et terminer l'appel avec le second. Dans ce cas, il est important que tous les terminaux collaborent pour garder un état cohérent de l'appel. Cette problématique ne sera pas traitée dans le cadre de ce travail.

Les solutions apportées au niveau de la mobilité de l'environnement donnent une certaine portabilité des données attachées au service pour permettre à l'utilisateur de bénéficier du même environnement lorsque d'autres niveaux de mobilité sont exécutés (mobilité de la session ou du terminal). Il est possible d'utiliser le protocole SIP pour transporter les informations relatives à l'environnement du service vers un élément central qui sera utilisé pour diffuser l'information aux autres terminaux du même utilisateur.

L'étude de la mobilité des sessions montre que cette problématique peut être traitée selon deux contextes : Mobilité lorsque la session en cours d'établissement et mobilité lorsque la session est déjà établie. Pour le cas d'une session en cours d'établissement, il est possible de réutiliser des mécanismes du protocole SIP pour rediriger l'appel. Cependant, ces mécanismes ne permettent pas de garantir le contrôle de la communication. La problématique d'offrir un service de mobilité de session en cours d'établissement avec contrôle, nécessite le développement de nouveaux mécanismes. Cette problématique sera traitée dans la section suivante de ce rapport. Pour le cas de la mobilité des sessions déjà établies, il est important de développer des mécanismes pour assurer le transfert des sessions avec et sans contrôle, mais aussi d'assurer leurs récupérations. Plusieurs contraintes et options liées au transfert de session doivent être prises en charge par la solution. Cette problématique est traitée dans la section 6 de ce rapport.



## **5 LA MOBILITÉ DES SESSIONS SIP EN COURS D'ÉTABLISSEMENT AVEC CONTRÔLE**

### **5.1 INTRODUCTION GÉNÉRALE**

Le coût actuel des téléphones IP peut être considéré comme un problème majeur pour les compagnies qui projettent de migrer vers la technologie de la téléphonie IP pour leurs besoins en communications. En effet, la migration vers une telle solution nécessite le remplacement de tous les équipements existants. Actuellement, un grand nombre de téléphones IP sont disponibles sur le marché. Une bonne partie de ces téléphones proposent des capacités logicielles et matérielles avancées (grand écran LCD, main libre, caméra, etc.), mais leur prix reste relativement dispendieux. Cela a pour effet de freiner leur expansion sur le marché de masse. Pour pallier à ce problème, plusieurs fournisseurs proposent des téléphones IP de base avec des capacités matérielles et logicielles limitées. Ces limitations posent de réels problèmes lors du déploiement de services avancés ou même lors de l'implémentation de nouveaux services de téléphonie IP. Par exemple, un utilisateur muni d'un téléphone IP sans afficheur ne sera pas capable de voir le numéro de l'appelant avant de prendre l'appel. Pour résoudre une telle limitation, nous proposons dans ce chapitre un nouveau protocole qui permet, entre autres, de « connecter » le téléphone IP de base à un ordinateur de bureau. Ainsi, lorsque l'appel est reçu sur le téléphone IP, un mécanisme de mobilité de session en cours d'établissement permet de transférer les données de l'appel reçu sur une application implémentée sur l'ordinateur de bureau. Une amélioration logique de ce service serait de créer une collaboration complète entre l'ordinateur de bureau et le téléphone IP afin de permettre de traiter l'appel entrant à partir d'un des deux terminaux disponibles à l'utilisateur. Les différents mécanismes, qui seront implémentés sur chacune de ces deux machines, devront permettre la collaboration par une notification mutuelle des séquences d'évènements concernant la session échangée. Cette collaboration totale entre les deux terminaux peut se baser sur l'utilisation d'un nouveau protocole qui prendra en charge d'échange des données

nécessaires au développement de nouveaux services à valeur ajoutée qui ne peuvent pas être supportés par un téléphone IP de base.

Nous proposons donc dans ce chapitre dCe chapitre se compose en trois grandes sections. La première section introduit le principe de fonctionnement de la collaboration ainsi qu'une étude des principaux choix technologiques. La section suivante présente la conception du protocole et son implémentation. Les deux avant dernières sections explorent en détail l'implémentation de quelques exemples de services qui utilisent la mobilité de session en cours d'établissement pour mettre en œuvre le cas des appels entrants et le cas des appels sortants. La dernière section conclut la section et propose des suggestions pour les travaux futurs.

## **5.2 PERMETTRE LA MOBILITÉ DE SESSION EN COURS D'ÉTABLISSEMENT**

### **5.2.1 Présentation de la problématique**

Le système se compose en trois entités comme indiqué dans Figure 10: un agent utilisateur SIP générique désigné par l'abréviation NAV, un téléphone IP désigné par NM et un ordinateur de bureau désigné par NC (Nœud Correspondant). Dans ce système, le MN joue le rôle d'élément central puisqu'il divise le système en deux sous-systèmes. Le premier sous-système est basé sur un protocole SIP de base, qui respecte les RFC en vigueur et donc qui permet l'établissement, la modification et la terminaison des communications multimédias basée sur IP. Dans le deuxième sous-système, la collaboration entre le NC et le NM est assurée par un protocole « maison » qui devrait garantir la mobilité de la session en cours d'établissement. L'objectif ici est de permettre aux deux intervenants (NC et NM) d'avoir le contrôle sur le dialogue à établir et de choisir dans quel nœud la session média aura lieu. L'étude sera faite pour le cas d'un appel entrant reçu par le MN ou bien par un appel sortant composé par le NC. Cette même collaboration pourrait aussi supporter d'autres services offerts par le MN au NC. Il est possible de citer l'exemple d'un service d'échange de l'historique des appels, l'écoute de la boîte de message, etc.

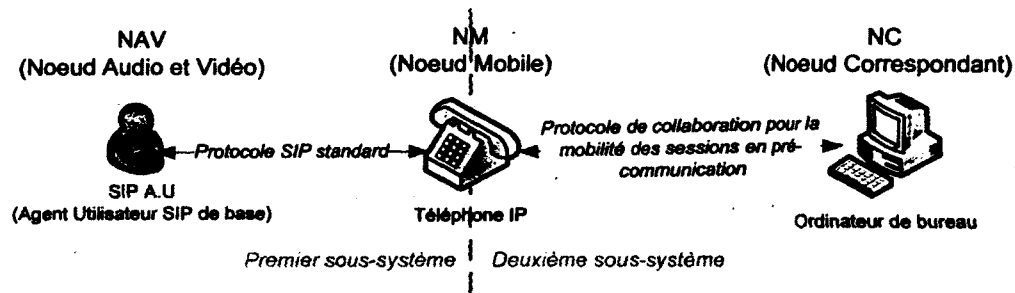


Figure 10 : Vue générale du système

### 5.2.2 Le modèle de communication

Lors du choix du modèle de communication, plusieurs critères doivent être considérés. Dans le cas du système proposé, le protocole doit permettre une communication « orientée-session » (*stateful*) avec des capacités de garantir un échange séquentiel des données entre les terminaux à l'intérieur de la session établie. La session de communication aura une certaine durée de vie et pourra persister à travers les différentes opérations (transactions) du protocole. Ainsi, les échanges, qui feront partie d'une session, pourront être traités dans une logique bien définie selon les besoins du problème. Ceci permettra à chacun des deux terminaux d'apporter les changements à l'état de la session durant la durée de vie de la communication. Par exemple, les informations d'authentification propre au dialogue à établir restent valides jusqu'à la terminaison de la communication.

Plusieurs technologies ont été étudiées pour implémenter ce protocole de collaboration. Parmi ces technologies, il est possible de citer les solutions basées sur les objets distribués comme *CORBA*, *JAVA/RMI* ou *COM/DCOM*. Ces solutions se basent principalement sur des modèles de communication de types client/serveur [9][30]. Dans ce modèle de communication, les services sont définis et implémentés du côté du serveur, et il devient possible au client de les utiliser en formulant des requêtes de demande de service. Le client ne pourra pas automatiquement être notifié lorsqu'il y a des changements de l'état d'un des services du côté du serveur. Ainsi, ce modèle se repose sur une relation « maître-esclave » entre les deux terminaux et réduit donc les capacités de collaborations du modèle de communication. D'un autre côté, le modèle

P2P [38] peut garantir une communication symétrique où tous les terminaux sont égaux c.-à-d. le terminal peut à la fois envoyer des requêtes, mais aussi offrir des services. Chaque terminal peut donc agir comme client, mais aussi comme serveur à la place d'agir comme client seulement ou serveur seulement. Ce modèle de communication semble donc plus adapté pour une collaboration entre agents utilisateurs, et donc entre le NM (téléphone IP) et le NC (l'ordinateur de bureau) dans l'exemple spécifique présenté dans l'introduction de ce chapitre.

### 5.2.3 Analyse des données échangées

Le protocole de collaboration à mettre en place doit principalement supporter un échange de données relatif à la session à établir. Il existe des cas où l'information échangée contient plusieurs sessions (faire des appels successifs, téléchargement de l'historique des appels, etc.). Chaque appel peut être décrit par un ensemble d'informations concernant l'identification de l'appelé, de l'appelant, le temps de début, etc. Si ces informations sont traitées sous forme d'objets, dans ce cas, il faut développer plusieurs méthodes pour gérer chaque niveau hiérarchique de l'information. Une solution plus simple serait de garder les données dans leur état textuel. Coder les données échangées dans un format lisible peut aussi faciliter le processus du débogage lors de l'implémentation. XML est une lingua franca pour l'échange d'information qui utilise un mécanisme d'encodage bien spécifié, mais flexible, adapté pour un contenu organisé hiérarchiquement [31]. XML peut donc être la solution pour encoder les données échangées entre le téléphone IP et l'ordinateur de bureau. Il existe des systèmes RPC (*Remote Procedure Call*) basés sur le modèle d'encodage XML-RPC [2] [99] ou SOAP-XML [98]. Alors que le XML-RPC offre un ensemble très limité d'espace de nom (*name-space*), le SOAP-XML, à l'inverse, présente un ensemble très exhaustif, mais relativement coûteux. En effet, le SOAP-XML nécessite l'ajout d'un contenu textuel additionnel pour envelopper le message à transmettre. De ce fait, il serait intéressant de définir et déployer un environnement de codage basé sur des espaces de nom propriétaires et adaptés au problème.

### 5.2.4 Analyse du protocole de transport des données

Il existe au moins deux grands modèles pour le transport des données entre les deux terminaux collaborateurs dans un environnement de communication basé sur le protocole IP. Alors que le protocole *HTTP* [7][37] est un choix populaire dans les plateformes de communication Web de type client/serveur, son introduction dans le monde des communications P2P semble causer quelques complexités lors de la gestion de la session. En effet, lors de l'utilisation de *HTTP*, il devient difficile de doter chaque terminal de capacité de communiquer en générant à la fois des requêtes et des réponses [6]. D'un autre côté, l'agent utilisateur localisé par défaut dans les téléphones IP est généralement compatible avec le protocole SIP et donc offre des composants existants qui pourraient être réutilisés pour implémenter le protocole de collaboration. Le choix de SIP comme protocole de transport se justifie encore plus par la facilité d'intégration du protocole, de sa grande flexibilité et de son support du contenu textuel échangé.

## 5.3 CONCEPTION DU PROTOCOLE

### 5.3.1 Protocole d'établissement de la session

L'objectif est de fournir un environnement spécifique à SIP pour la notification d'évènement qui sera simple et efficace afin de supporter les options de base de la mobilité de session tout en restant assez flexible pour fournir des services plus complexes. La création de dialogues SIP peut être initialisée par les méthodes des mécanismes INVITE [74] ou SUBSCRIBE [65]. D'autres méthodes SIP comme INFO [19], OPTION [74], MESSAGE [13] ou NOTIFY [65] pourraient être utilisées à l'intérieur des dialogues SIP afin d'acheminer les informations textuelles entre les nœuds collaborateurs.

Les derniers RFCs de l'IETF montrent qu'il existe une grande complémentarité entre les méthodes SUBSCRIBE et NOTIFY. Le mécanisme du SUBSCRIBE/NOTIFY, initialement introduit par IETF RFC 3265 (*SIP – Specific Event Notification*), propose un

cadre logiciel extensible à travers lequel les nœuds utilisateurs de SIP peuvent solliciter des notifications, de la part d'un nœud distant, lorsqu'un évènement spécial, auquel ils sont souscrits, vient d'avoir lieu [65]. Le mécanisme SUBSCRIBE/MOTIFY, déjà utilisé dans les systèmes de présence [67] et de messagerie instantanée, peut être généralisé à d'autres systèmes qui désirent synchroniser leurs états respectifs en s'échangeant les évènements appropriés. À travers la Figure 11, il est possible de voir un exemple d'échange de message par le mécanisme SUBSCRIBE/NOTIF.

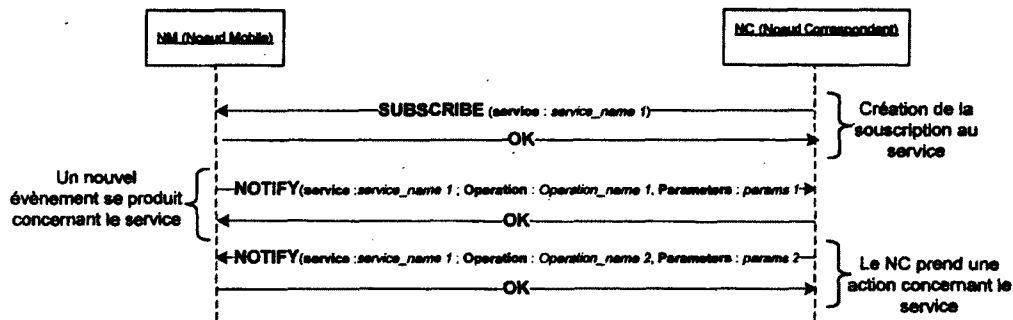


Figure 11 : Modèle d'échange de messages par le mécanisme SUBSCRIBE/NOTIFY

### 5.3.2 Le modèle du protocole

Il est possible de proposer un modèle en couches pour représenter le protocole de communication qui répond aux spécifications du problème. Dans ce modèle, illustré par la Figure 12, seule la couche application du modèle OSI est représentée et détaillée. La première couche de ce modèle (couche session) est représentée par le mécanisme SUBSCRIBE/NOTIFY du protocole SIP. La deuxième couche (couche service) définit le type de service pour lequel les deux agents comptent collaborer. La troisième couche (couche opération) décrit l'évènement à transmettre alors que la dernière couche (couche paramètres) contient les données de configuration propre à chaque évènement.

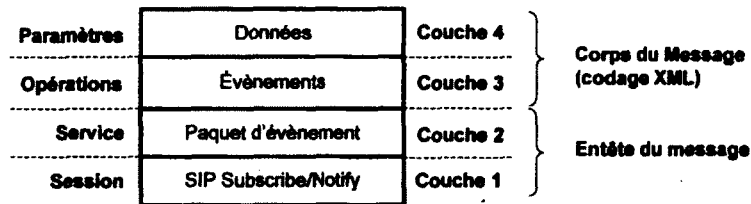


Figure 12 : le modèle en couches du protocole

### 5.3.3 Définition de la structure des messages

Lors de l'envoi, le message SIP transite par le modèle en couche de la Figure 12 en le parcourant du haut vers le bas. À chaque passage, la couche concernée encapsule ses données dans le message. Et inversement, à la réception, le modèle en couche est parcouru du bas vers le haut tout en décapsulant les informations relatives à chaque couche.

Les entêtes des messages SIP échangés par le protocole de communication doivent se conformer aux RFC de L'IETF en vigueur. L'échange de messages entre le NM et le NC se fait à travers des messages SIP NOTIFY à l'intérieur d'un dialogue SIP existant, crée par la méthode SUBSCRIBE. Chaque requête SIP doit inclure le nom de la méthode utilisée et indiquée dans l'entête obligatoire « method ». Au moins, deux services peuvent être définis pour répondre aux besoins de la mobilité de la session en cours d'établissement lors des scénarios d'un appel entrant et d'un appel sortant. Chaque service est indiqué dans l'entête « event » inclus dans la demande de souscription véhiculée par la méthode SUBSCRIBE. Tous les échanges qui succèdent la souscription se font avec la méthode NOTIFY et doivent inclure le même « événement ». Les termes « Caller\_ID » et « Make\_Call » peuvent être utilisés et standardisés pour définir respectivement le service dans le cas d'un appel rentrant ou d'un appel sortant. D'autres termes pourraient être utilisés pour définir d'autres services additionnels basés sur le même protocole. Le protocole proposé utilise le corps des messages SIP pour inclure l'opération et les paramètres qui s'y attachent. Ces informations sont structurées dans un format XML encapsulé dans des messages SIP NOTIFY. La Figure 13 illustre un exemple de structure de message envoyé par la méthode NOTIFY dans le cadre du

service d'appel entrant « caller-id ». Les opérations relatives aux services d'appel entrant et appel sortant seront détaillées dans les deux prochaines sous-sections.

```
Request line: NOTIFY sip : PC_SIP_URI SIP/2.0
Method: NOTIFY
Message Header: // Entête du message //
  Max-Forwards: 70
  Content-Length: 267
  Via: SIP/2.0/UDP Local_IP_Addr;branch=z9hG4bKe89100756
  Call-ID: a0ca01198b4afcl
  From: NM_SIP_URI;tag=9f82847c9fbb5ee
  To: NC_SIP_URI
  CSeq: 736513480 NOTIFY
  Expires: 600
  Event: Caller-ID
  Content-Type: Pre-communication_Mobility_protocol_Operation
  User-Agent: IP-Phone UA Version
Body Message // Corps du message //
  <rpc>
    <operation-name>caller-id-request</operation-name>
    <call-description>
      <from>SenderUri</from>
      <to>ReceiverUri</to>
      <call-id>...</call-id>
      <cseq>...</cseq>
      <contact>Proxy_URI</contact>
    </call-description>
  </rpc>
```

Figure 13 : Exemple de structure de message échangé

## 5.4 EXEMPLE DE SERVICE

Le protocole développé dans le cadre de ce travail permet d'implémenter des services à valeur ajoutée pour un téléphone IP de base. Deux services qui présentent le cas d'un appel entrant (caller-ID) et d'un appel sortant (Make-Call) sont présentés. Ces exemples sont illustrés par des diagrammes d'échange de messages qui se basent sur le modèle proposé dans la Figure 11.

### 5.4.1 Le service « Caller-ID »

Dans cet exemple de service, le NM peut être un téléphone IP qui n'est pas équipé d'afficheur. L'utilisateur peut donc utiliser son ordinateur, appelé NC dans notre exemple, pour recevoir une fenêtre de notification sur son écran lorsque le NM reçoit un appel entrant. Il est possible d'utiliser soit l'ordinateur soit le téléphone IP pour



donner une réponse à l'appel. La Figure 14 contient un exemple d'échange de messages lors d'un appel entrant dans le cadre du service « Caller-ID ».

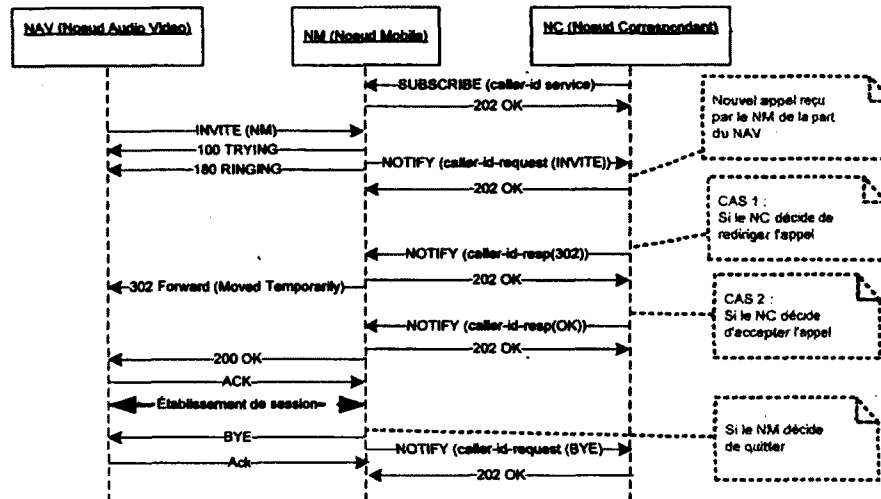


Figure 14 : Exemple de diagramme de séquence lors d'un appel entrant

### 5.4.2 Le service « Make-Call »

Le service « make-Call » permet à un NC (un ordinateur de bureau) de commander un téléphone IP pour effectuer un appel. Dans cet exemple de service, le NM peut être un téléphone IP qui n'est pas équipé de clavier. L'utilisateur peut donc utiliser la liste de contact de l'ordinateur pour effectuer les appels. Cette fonctionnalité est encore plus utilisable dans un centre d'appel où les numéros sont enregistrés dans une base de données. Lorsque l'appel est établi, un signal sonore peut être enclenché pour que l'opérateur prenne l'appel. La Figure 15 présente un exemple d'échange de messages lors d'un appel entrant dans le cadre du service « Make-Call ».

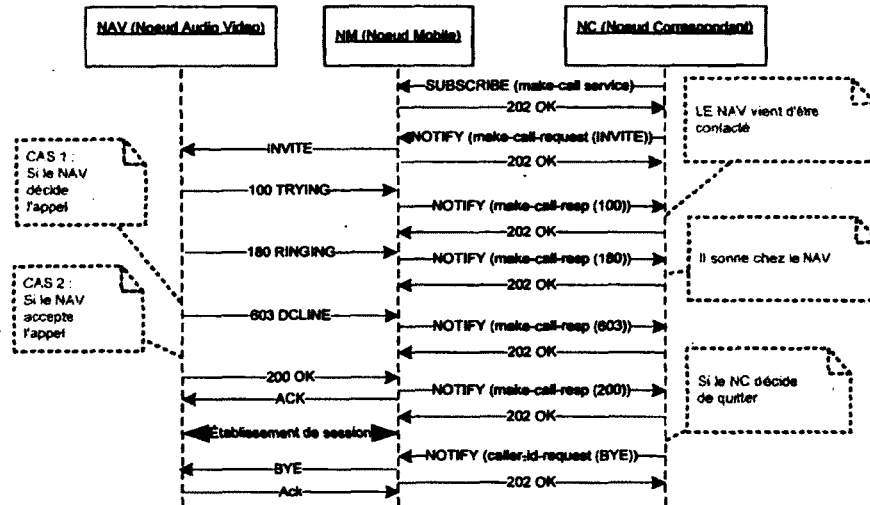


Figure 15 : Exemple de diagramme de séquence lors d'un appel sortant

## 5.5 CONCLUSION

Ce travail a permis d'apporter une solution au problème de la mobilité des sessions lorsqu'elles sont en cours d'établissement tout en permettant leur contrôle et leur suivi. La solution proposée se présente sous forme d'un protocole basé sur SIP couplé à un modèle en couches afin de supporter différents services contenant différentes opérations selon différents paramètres. L'utilisation du format XML a aussi permis une meilleure prise en charge des différentes opérations et de leurs paramètres. L'objectif est de permettre une collaboration totale entre les nœuds qui s'échangent la session. Ce protocole a été implémenté réellement sur des ordinateurs de bureau interconnectés dans un réseau local. Les tests ont été effectués lors d'un appel entrant et lors d'un appel sortant. Malgré la complexité d'implémentation d'un tel système, les résultats sont très positifs et aucun problème n'a pu être rapporté. Cette complexité réside du fait qu'il faut respecter l'interopérabilité avec les équipements existants tout en offrant une solution compatible avec le fonctionnement existant du NM (téléphone IP). Cependant, il est possible que plusieurs NC soient connectés sur le NM. Dans ce cas, le protocole devrait être capable de garantir la synchronisation entre les différents

**La mobilité des sessions SIP en cours d'établissement avec contrôle**

---

intervenant. Ce cas n'a pas été pris en compte, mais il pourrait être développé dans un projet futur.

## 6 LA MOBILITÉ DES SESSIONS SIP DÉJÀ ÉTABLIES

### 6.1 INTRODUCTION : VERS UN DISPOSITIF VIRTUEL

Depuis les dernières années, les terminaux mobiles n'ont pas cessé de se développer et de s'équiper de larges capacités leur permettant de supporter des communications multimédias basées sur IP. Cependant, ces terminaux sont encore limités en termes de bande passante, de taille d'affichage, de puissance de calcul et d'autonomie. Par ailleurs, les terminaux fixes tels que les téléphones IP, les unités de vidéoconférence, les systèmes embarqués/logiciel de téléphonie, malgré leur mobilité réduite, continuent à présenter des plateformes commodes pour le support des communications multimédias basées sur IP. Il est donc intéressant d'offrir la possibilité d'utiliser ces différents terminaux (fixes et mobiles) d'une façon interchangeable et continue (*seamless transition*) même en cours de communication. Cela permet de combiner les avantages de tous les dispositifs dans un seul, appelé « Dispositif virtuel ».

Le SIP a été sélectionné par le 3GPP pour être le standard d'établissement de session dans les IMS [1]. Plusieurs piles logicielles sont actuellement disponibles sur le marché pour supporter les applications basées sur SIP. Il devient donc naturel de spécifier des solutions basées sur ce protocole pour assurer la continuité et l'ubiquité des services de communication sur IP. Il existe en ce moment un *Draft* [79] de l'IETF intitulé « *Session Initiation Protocol (SIP) Session Mobility* » qui propose des solutions à la mobilité des sessions SIP entre deux utilisateurs. L'opération de transfert de session regroupe au minimum trois acteurs afin de réaliser un scénario qui peut être comme suit : le Nœud mobile (NM), en communication avec un Nœud Correspondant (NC), décide de transférer sa session vers un autre Nœud Local (NL). Le NL peut aussi être appelé Nœud Audio (NA) s'il supporte les sessions audio ou Nœud Vidéo (NV) s'il supporte les sessions vidéo. Ces abréviations seront donc utilisées dans le reste de ce document.

## 6.2 LES CONTRAINTES LIÉES À LA MOBILITÉ DE SESSION

Quatre contraintes devraient être respectées pour assurer la mobilité de la session :

- ✓ **Interopérabilité** : Aucune capacité spécifique ne devrait être requise au NC du moment où il utilise un terminal compatible avec SIP. Le NM devrait être capable de supporter le transfert en utilisant les recommandations décrites par les RFCs ainsi que par les Internet Drafts matures.
- ✓ **Compatibilité** : Un des objectifs est d'effectuer le minimum de changement du côté des terminaux SIP existants. La composante logicielle qui supporte la mobilité de session est appelée mobilité étendue. Les terminaux de base ainsi que ceux dotés d'une mobilité étendue devraient pouvoir recevoir les transferts d'appel de la part du NM. Naturellement, les téléphones IP commerciaux seront considérés comme étant de vase et ne pourront pas supporter les fonctionnalités avancées de cette mobilité étendue.
- ✓ **Flexibilité** : Les différences dans les capacités des terminaux devront être réconciliées. Le transfert devrait être possible à la fois vers les terminaux qui ne supportent pas le même codec d'origine et même vers les terminaux qui n'ont aucun codec en commun avec les autres participants de la session. Le transfert devrait aussi prendre en compte les caractéristiques d'affichage, la bande passante et s'adapter.
- ✓ **Continuité et transparence (*Seamlessness*)** : Le transfert de session devrait se faire de la façon la plus continue et la plus transparente possible. Il devrait provoquer une interruption minimale pour le flux de média et ne devrait pas paraître à l'utilisateur distant comme étant un nouvel appel.

## 6.3 LES OPTIONS DE LA MOBILITÉ DE SESSION

### 6.3.1 Transfert et récupération

La mobilité de la session devrait prendre en compte à la fois les opérations de transfert, mais aussi de récupération de la session active. L'opération de transfert permet de déplacer la session vers un ou plusieurs autres terminaux. L'opération de récupération permet de commander le transfert d'un terminal distant vers le terminal local. Cela permet donc à la personne qui a initialement transféré une session vers un autre terminal de la récupérer. Il est possible d'imaginer l'exemple d'une personne qui rentre dans une salle équipée d'un grand projecteur IP et qui veut l'utiliser pour afficher la vidéo de sa communication en cours. Avant de quitter la salle, l'utilisateur pourra récupérer la partie vidéo de sa communication et continuer sa session.

### 6.3.2 Transfert total ou partiel

Le média de la session peut être transféré soit complètement vers un autre terminal soit être séparé entre plusieurs terminaux. Une personne en communication audio et vidéo pourra utiliser le projecteur disponible pour afficher la vidéo et le système son de la salle pour écouter la communication. Cette option de pouvoir séparer le média sur différents terminaux permet de tirer un maximum de profit de chacun. Il est donc possible d'imaginer un système qui pourrait assister l'utilisateur en lui suggérant de transférer une des parties de la session dès qu'un terminal disponible, de meilleures capacités, est découvert.

### 6.3.3 Modes de transfert

Dépendamment du besoin de l'utilisateur, il existe deux modes de transfert de session :

- ✓ **Transfert de session avec contrôle** : dans ce mode, le NM se base sur les techniques de *third-party call control* [72] pour effectuer le transfert. Il établit une session SIP avec chacun des deux terminaux et met à jour sa

session avec le NC à travers les paramètres de SDP. Ainsi, le NC sera capable d'établir une session de média avec chaque terminal. Ce mode oblige le NM à rester actif pour maintenir et contrôler la session.

- ✓ **Transfert de session avec perte de contrôle:** il existe des cas, pour lesquels, l'utilisateur peut avoir besoin de procéder à un transfert complet de sa session vers un autre terminal. Ce scénario peut avoir lieu dans un exemple où les batteries d'un terminal mobile sont sur le point de se vider, ou bien lorsque l'utilisateur doit quitter son terminal fixe et se déplacer emportant avec lui un terminal mobile. Dans ces deux cas, l'utilisateur n'a pas besoin de laisser sa session active sur le terminal et voudrait bien transférer à la fois la partie signalisation et média à un autre terminal de substitution afin de continuer la communication.

Les sections 6.4 et 0 présentent ces deux modes avec plus de détails.

### 6.3.4 Types de média à transférer

En utilisant le protocole SIP, il est possible de créer des sessions pour transférer différents types de médias. L'utilisateur devrait être capable de transférer un ou plusieurs des médias de la session vers un ou plusieurs terminaux. Dans ce document, on se limitera aux médias audio, vidéo et messages textuels.

Les médias audio et vidéo peuvent être négociés avec SDP et transférés par le protocole RTP. Il existe différentes façons de transférer les messages textuels. Des méthodes comme MESSAGE ou NOTIFY peuvent être utilisées. La partie « Body message » peut être utilisée pour contenir des données textuelles. Il s'agit là du même principe utilisé par les requêtes INVITE pour véhiculer des données codées selon le protocole SDP. Les données contenues dans les messages texte peuvent être codées selon la norme MIME [36]. Cette norme permet d'ajouter des fichiers au message envoyé (image, son, vidéo, documents textuels, etc.). L'utilisation de cette norme pour les données échangées entre deux utilisateurs SIP permet d'offrir des services de messagerie comparables à ceux des systèmes de courriels.

Il existe aussi des cas où deux ou plusieurs agents utilisateur SIP auraient besoin de s'échanger des données pour des besoins du fonctionnement du service. Il est possible de citer des exemples où les agents utilisateurs doivent se synchroniser, transmettre et mettre à jour leurs états, se partager des tâches et collaborer ensemble. Pour ces besoins, il est important que l'information échangée entre systèmes soit exhaustive et facilement lisible par le receveur. Le format XML peut être un bon format de codage pour contenir ce type d'information. Pour faciliter les opérations de codage et de décodage, il est conseillé de faire appel à des applications de traitement du code XML, appelées parseurs.

## **6.4 TRANSFERT DE SESSION AVEC CONTRÔLE**

### **6.4.1 Introduction**

La problématique de transfert de session SIP avec contrôle entre terminaux a été initialement un sujet d'étude dans le RFC 3725 [72] pour la réalisation de nouveaux services basés sur SIP. Ces mécanismes de transfert reposent sur l'utilisation d'un élément central appelé « contrôleur » pour gérer l'échange de messages SIP entre, d'un coté, le ou les terminaux (nœuds) en cours de communication et, d'un autre coté, le nouveau terminal destinataire de la session à transférer.



## 6.4.2 Transfert vers un seul nœud

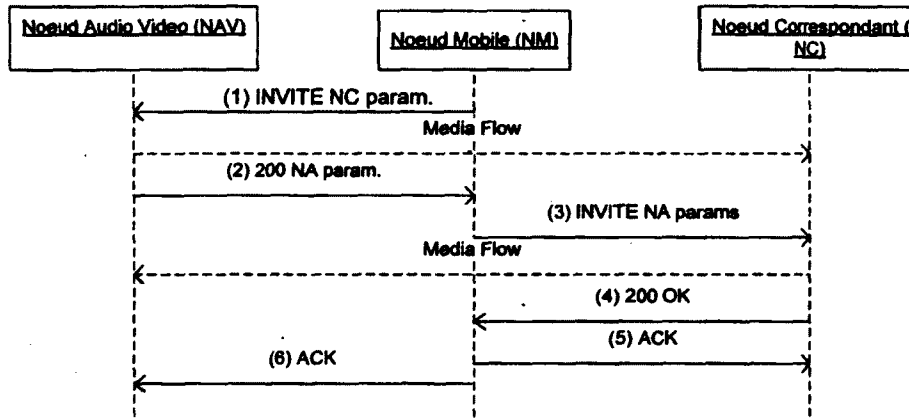


Figure 16 : Diagramme de séquence pour un transfert avec contrôle vers un seul nœud

L'échange de message illustré dans la Figure 16 permet à un NM, ayant une session avec le NC, de transférer la partie audio et vidéo de la session vers le nœud NAV (Nœud Audio et Vidéo). Pour réaliser un tel transfert, le NM commence par établir une session avec le NAV en donnant les paramètres SDP du NC. Le NAV répond en donnant ses propres paramètres, le NM les transmet au NC pour qu'il mette à jour sa session SDP. Voici un exemple d'un contenu SDP à transmettre au NC :

```
v=0
c= IN Add_NAV
m=audio 4400 RTP/AVP 0
m=video 5400 RTP/AVP 34
```

Le Dialogue SIP reste actif entre NM et NC. Cependant, les sessions audio et vidéo sont établies entre NC et le NAV.

Il existe différents mécanismes d'échange de messages pour établir une session média entre le NC et le NAV. Le mécanisme proposé ici correspond à la recommandation de l'IETF pour le cas où les terminaux répondent rapidement aux messages INVITE.

### 6.4.3 Transfert vers plusieurs nœuds

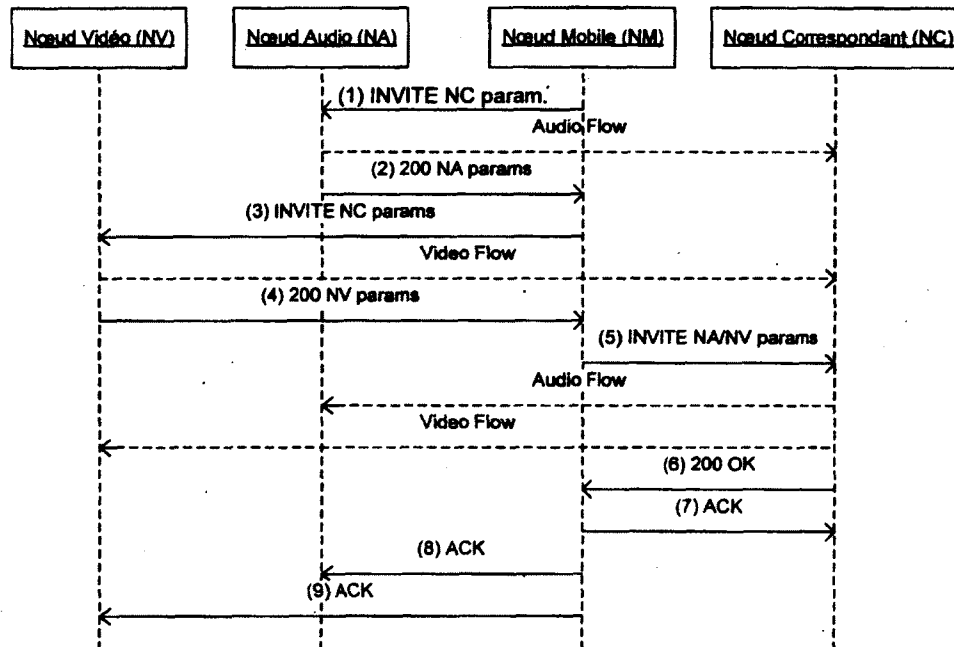


Figure 17 : Diagramme de séquence pour un transfert avec contrôle vers plusieurs nœuds

Pour diviser une session entre plusieurs terminaux, le NM utilise le même principe que celui utilisé pour le transfert vers un seul nœud (voir Figure 17). Il utilise les paramètres SDP du NC pour créer une session SIP avec chacun des deux nœuds. Dès la réception des réponses SDP de la part des deux nœuds, le NM procède à leur combinaison dans une seule offre SDP qu'il envoie au NC pour mettre à jour la session déjà établie. (Voir l'exemple suivant)

```
v=0
m=audio 48400 RTP/AVP 0
c= IN Add_NA
m=video 58400 RTP/AVP 34
c= IN Add_NV
```

## 6.5 TRANSFERT DE SESSION AVEC PERTE DE CONTRÔLE

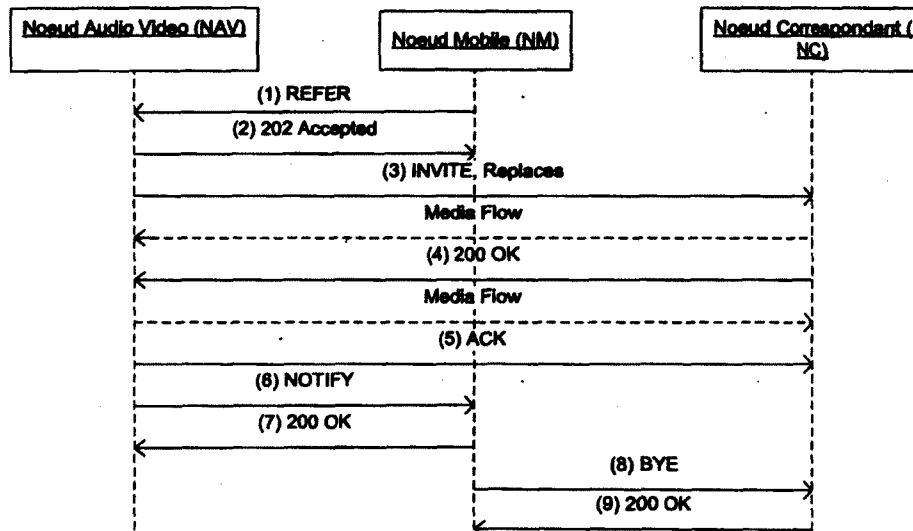


Figure 18 : Diagramme de séquence pour un transfert en mode avec perte de contrôle

Le mode transfert de session avec perte de contrôle implique que le NM cède son dialogue à un autre nœud. Ce dialogue sera donc transféré ainsi que toutes les sessions qui lui sont attachées. Ce mode de transfert est possible à l'aide de la méthode SIP REFER [80]. Ce message est envoyé par le *referer* au *referee* pour le référer à une autre adresse URI. Pour cela, le message de la méthode REFER fournit deux entêtes essentiels :

L'entête «Refer-To » pour désigner l'adresse URI à contacter. Cette adresse devrait être, dans un contexte de la mobilité de session, l'URI SIP du NC, mais qui peut être, dans d'autres contextes, un autre type d'adresse de ressources (une adresse URL HTTP par exemple). L'entête « Referred-By » [81] est utilisé pour spécifier l'identité du *referer*, le NM dans notre cas. Cela permet au *referee* d'utiliser l'adresse de *referer* pour

s'authentifier auprès de la ressource référencée. Les entêtes principaux du message REFER peuvent être cryptés et envoyés dans le corps du message dans le format S/MIME pour authentifier la requête. La Figure 18 montre un exemple d'échange de messages pour réaliser le transfert de session dans ce mode.

Le NM envoie la requête REFER (message 1) suivante au NA :

```
REFER sip:AN@local_device.example.com SIP/2.0
To: <sip:AN@example.com>
From: <sip:NM@example.com>
Refer-To:<sip:NC@domaine.com;capacités_audio;capacités_video?
Replaces="1@mob.example.com;
to-tag=bbb;from-tag=aaa">
Referred-By: <sip:NM@example.com>

----- [S/MIME authentication body] -----
```

Dans ce message, les capacités audio et vidéo du NC peuvent être indiquées dans l'entête « Refer-To ». Les informations relatives à la session en cours sont incluses dans l'entête « Replaces » et doivent être incluses dans la requête INVITE (message 2). Selon [52], le NC ne peut qu'accepter la requête INVITE en remplaçant la session en cours lorsque ce transfert de session est référencé par un des participants à la session en cours.

Lorsque le dialogue est établi entre le NA et le NC, le NA informe le NM en lui envoyant, conformément à [81], une requête NOTIFY. Cette requête devrait contenir les champs d'entête « To » (incluant le tag), « From » (incluant tag) et « Call-ID » de la session établie afin de permettre au NM de récupérer la session plus tard. Le mécanisme de récupération est présenté dans la section suivante.

## 6.6 RÉCUPÉRATION DE SESSION

Pour le cas d'une session transférée selon le mode avec contrôle, la récupération devrait être facile à supporter. Il suffit pour cela au NM d'envoyer une requête INVITE au NC et de mettre à jour la description de la session avec les paramètres d'origine.

Pour le cas d'une session transférée en mode avec perte de contrôle, antérieurement par le NM, la récupération devient plus complexe. La Figure 19 montre un exemple de diagramme de séquence pour effectuer cette récupération.

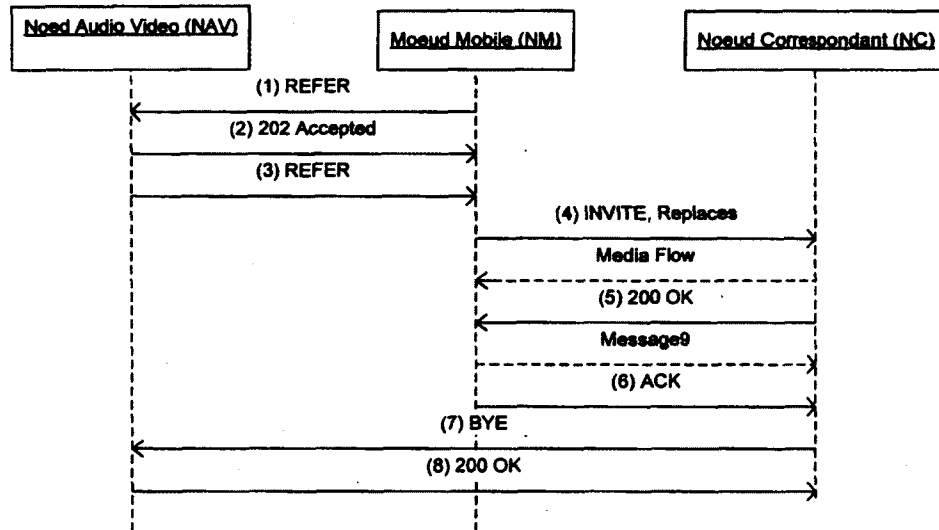


Figure 19 : Diagramme de récupération de session dans le mode avec perte de contrôle

Pour effectuer cette récupération, en mode avec perte de contrôle, le NM envoie une requête du type INVITE (message 4) avec les entêtes « Replaces » et « referred By ».

Voici un exemple d'un tel message :

```
INVITE sip:NC@domaine.com SIP/2.0
To: <sip:NC@domaine.com>
From: <sip:NM@example.com>
Replaces: 1@local_device.example.com;to-tag=aa;from-tag=bbb
Referred-By: <sip:NA@local_device.example.com>
----- [S/MIME authentication body] -----
```

Le NM a besoin d'être référé par le NA et d'inclure son URI dans l'entête « Referred-By », en plus d'inclure dans le corps du message une partie S/MIME provenant du NA.

Dans une logique de fonctionnement normale, le NM devrait initier le processus de récupération et envoyer un REFER au NM. Mais dans la pratique, il est possible que le

NA ne possède pas d'interface utilisateur, ou que le NM veuille activer la récupération à distance et à sa demande. Pour cela, la solution proposée par [79], permet au NM de commencer par envoyer un « nested REFER » [52]. Il s'agit d'une requête REFER (message1) qui permet au NA de répondre par une requête REFER (message 3). Pour envoyer cette requête initiale, le NM devrait utiliser les paramètres qui lui ont été transmis dans la requête NOTIFY lorsqu'il a transféré sa session (voir section précédente). La deuxième requête REFER devrait spécifier le champ de l'entête à inclure dans le message INVITE. Cette requête peut-être comme suit :

```
REFER sip:NM@example.com SIP/2.0
To: <sip:NM@example.com>
From: <sip:NA@local_device.example.com>
Refer-To: <sip:NC@domaine.com;capacités_audio;capacités_video?
          Replaces="1@local_device.example.com;
          to-tag=aaa;from-tag=bbb">
Referred-By: <sip:NA@local_device.example.com>
----- [S/MIME authentication body]
```

Pour obtenir le message 2, le NM envoie une requête REFER comme suit :

```
REFER sip:NA@local_device.example.com SIP/2.0
To: <sip:NA@local_device.example.com>
From: <sip:mn@example.com>
Refer-To: <sip:mn@domaine.com;
Refer-To="<sip:NC@domaine.com;audio;video?
          Replaces=1@local_device.example.com;
          to-tag=aaa;from-tag=bbb">
```

Dans le cas où le NM essaie de récupérer une session qu'il n'a jamais transférée et qui ne lui appartient pas, il lui devient impossible d'envoyer la requête REFER puisqu'il ne connaît pas les paramètres de la session d'origine. Cependant, ces paramètres peuvent lui être envoyés en se souscrivant au *Dialog Event Package* [75] du NA. Cette souscription nécessite la gestion d'autorisations. Elle n'est garantie pour le moment que pour le propriétaire initial de la session.

## 6.7 DISCUSSIONS ET CONCLUSIONS

L'étude de la mobilité des sessions SIP montre qu'il existe en ce moment des propositions de solutions concrètes et réalisables pour permettre à une session SIP

d'être transférée d'un terminal vers un ou plusieurs autres terminaux. Ces transferts peuvent se faire selon le mode avec contrôle ou avec perte de contrôle.

D'un autre côté, la solution apportée au problème de la récupération des sessions transférées reste non complète. En effet, il existe des cas où le NL transfère à son tour la session à un autre terminal et dans ce cas, il devient impossible au NM de retrouver et de récupérer sa session. Cette éventualité n'a pas été prise en compte. Une solution qui permet le suivi des sessions transférées peut être développée. Cependant, cette problématique ne sera pas traitée dans le cadre de ce travail.

Les mécanismes proposés pour assurer la mobilité de session n'ont traité que les cas où le NM est en communication exclusivement avec un autre nœud. Il serait intéressant d'étudier les cas où le NM est en conférence avec d'autres personnes et de pouvoir traiter les différents scénarios selon la topologie de la conférence. Plus encore, la mobilité des sessions avec contrôle si elle est effectuée par un seul nœud peut permettre une gestion centralisée de la signalisation tout en assurant une distribution du réseau média. Une telle architecture sera étudiée dans la section 9 (page 104).

La deuxième partie de ce rapport se consacre à l'étude de la mobilité des sessions SIP dans les communications multimédias en mode-conférence.

---

## **Partie 2**

**La mobilité des sessions SIP dans les  
communications multimédias en mode  
conférence**



La première partie de ce rapport a permis d'aborder la problématique de mobilité de sessions lorsque la communication est en cours d'établissement et lorsqu'elle est déjà établie. Ces études ont été effectuées pour des cas de communication entre deux intervenants. Si on prend le cas d'une communication multimédia en mode-conférence, l'utilisation de l'option de mobilité de session par un des participants cause des répercussions sur le reste des participants. De nouveaux mécanismes doivent donc être prévus pour permettre le support de la mobilité de session. La section suivante traite ce problème spécifique selon une mobilité de sessions avec et avec perte de contrôle.

Comme on est capable de voir l'option de mobilité des sessions comme étant un service additionnel à offrir aux utilisateurs de services de communication multimédia, il est possible de voir cette mobilité de session comme un outil qui permet de résoudre d'autres problématiques. Un premier exemple de problématique concerne le cas d'une communication en mode-conférence de type hermétiquement couplée. Une telle conférence présente une certaine limitation lors de son utilisation lorsque l'utilisateur central désire quitter. Dans ce cas, les autres participants ne pourront pas continuer à bénéficier de cette conférence. À l'aide de l'option de mobilité de session, nous proposons d'effectuer une transition vers un autre modèle de conférence qui permet au nœud central de quitter sans affecter les autres participants. Cette problématique est traitée dans la section 8.

Un deuxième exemple d'application de la mobilité des sessions traite les modèles de conférences à large échelle. Plus particulièrement, nous abordons les conférences de voix sur IP dont le traitement média se base sur les terminaux des participants (qu'on appellera « nœuds ») de façon à créer un réseau média interconnecté. La complication de cette approche s'explique par le fait que les nœuds peuvent s'ajouter et partir de la conférence à n'importe quel moment. Il faut donc que l'arbre-média soit géré et contrôlé par une solution qui permet d'assurer sa maintenance et de supporter toutes les opérations qui optimisent le réseau média. La section 9 aborde donc ce cas

**d'utilisation de la mobilité des sessions (plus particulièrement en mode avec contrôle)  
pour permettre la gestion des conférences à large échelle.**

## **7 LA MOBILITÉ DES SESSIONS DANS UN MODÈLE DE CONFÉRENCE MULTIMÉDIA COMPLÈTEMENT MAILLÉ**

### **7.1 INTRODUCTION**

Lors des sections précédentes, il a été possible de présenter des options et des cas d'utilisations relatifs à la mobilité de session. Les options de mobilité avec et sans perte de contrôle ont été abordées pour des cas de session en cours d'établissement ou déjà établies. Ces études ont été effectuées pour des cas de communication entre deux intervenants. Si on prend le cas d'une communication multimédia en mode-conférence, l'utilisation de l'option de mobilité de session par un des participants peut avoir des répercussions sur le reste des participants. De nouveaux mécanismes doivent donc être prévus pour continuer à supporter le service de communication au sein de la conférence même en cas de mobilité de session. La première partie de ce travail consiste à présenter le modèle de conférence complètement maillé et à expliquer les différents messages de protocole ainsi que leur mappage vers SIP. Par la suite, l'ajout de la fonction de mobilité des sessions dans ce type de conférence est traité ainsi que les différentes extensions et les nouveaux mécanismes à apporter. Une étude de l'implémentation de la nouvelle solution est incluse dans ce rapport et traite les principaux composants à ajouter à la pile architecturale des nœuds participants. À la fin, nous évaluons le trafic additionnel généré par l'ajout de l'option de mobilité de session pour le mode de transfert avec et sans perte de contrôle. Avant de conclure ce travail, nous analysons les performances du système pour évaluer l'impact de l'interruption du média et la latence durant le transfert.

## 7.2 LE MODÈLES DES CONFÉRENCES COMPLÈTEMENT MAILLÉES

### 7.2.1 Structure générale

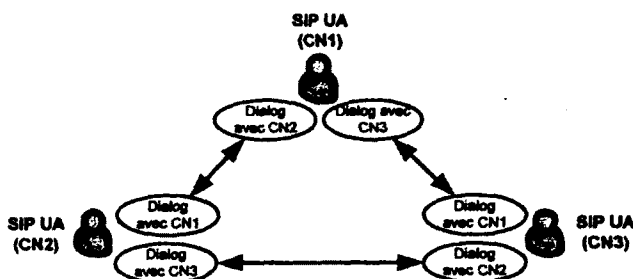


Figure 20 : Vue sur les interconnexions dans un modèle complètement maillé.

Dans le modèle de conférence complètement maillée, chaque terminal doit créer et gérer un dialogue de signalisation avec chacun des participants à la conférence. De ce fait, dans une conférence à  $N$  utilisateurs, chaque participant gère  $(N-1)$  dialogues comme illustrés dans la Figure 20. Ce modèle associe, à chaque dialogue de signalisation, un lien média qui permet l'échange multimédia entre les participants. Le rôle de chaque dialogue est donc de gérer la session média qui lui est associée. Les dialogues de la conférence sont initiés, créés, mis à jour et terminés à travers l'échange de messages au sein du protocole de signalisation présenté plus loin dans cette section.

Le modèle complètement maillé est plus coûteux en termes de volumes d'échange des messages du protocole par rapport à une topologie centralisée. Cependant, ce même modèle offre une robustesse et une souplesse qui permettent respectivement aux participants de pouvoir quitter sans affecter la conférence et aux nouveaux utilisateurs de se joindre facilement à la conférence librement, en mode ad-hoc, sans passer par un participant en particulier ou par une machine de tierce partie (serveur de conférence). Cependant, ce modèle nécessite, de la part de chaque participant, d'avoir les capacités logicielles et surtout matérielles pour pouvoir recevoir et envoyer un flux média établi avec chacun des  $(N-1)$  autres participants. En effet, les  $(N-1)$  flux média reçus doivent être mixés et exécutés (joués/affichés) localement par chaque membre

alors que le flux sortant est dupliqué puis envoyé vers les autres membres. Ceci a pour effet de limiter le nombre de participants, bien évidemment, du moment où la bande passante ou les capacités d'un des participants présente des limitations.

### 7.2.2 Les messages du protocole

Le protocole de base pour gérer une conférence de type complètement maillée se repose sur un ensemble de messages abstraits présentés dans [49]. Ces messages peuvent être classés en trois catégories : les requêtes (JOIN, CONNECT, LEAVE et UPDATE), les réponses (OK avec la possibilité d'utiliser la réponse REJECT pour JOIN et CONNECT) et les acquittements (ACK) pour les transactions initiées par les requêtes JOIN ou CONNECT. Lors de l'ajout d'un nouvel utilisateur, les messages JOIN et CONNECT sont utilisés. La requête JOIN est utilisée par les membres de la conférence pour inviter un nouvel utilisateur à rejoindre la conférence alors que la requête CONNECT est initiée par le nouvel utilisateur pour contacter les participants restants. Ces deux requêtes sont utilisées au sein d'un mécanisme transactionnel à trois étapes (requête/réponse/acquittement). Le message UPDATE peut être utilisé pour permettre aux participants de pouvoir échanger des informations concernant la conférence et donc de pouvoir se mettre à jour. La requête LEAVE est utilisée pour terminer un des dialogues de la conférence. La Figure 21 présente un exemple d'échange de message pour permettre au nœud D de se joindre à la conférence établie entre A, B et C.

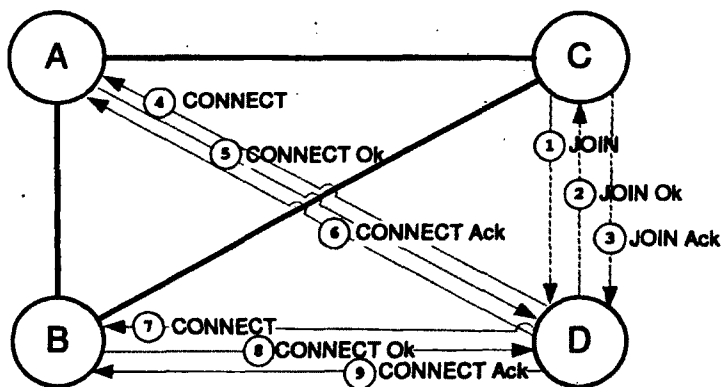


Figure 21 : Exemple d'échange de message pour l'ajout d'un nouveau participant

### 7.2.3 Mappage du protocole vers SIP

Pour implémenter le protocole abstrait, présenté dans la section précédente, il est possible de faire un mappage vers le protocole SIP. Puisque les messages JOIN et CONNECT permettent l'établissement d'un dialogue, il est possible de les mapper vers le message SIP INVITE. Le message LEAVE qui permet de terminer une session peut être mappé vers le message BYE ou CANCEL, dépendamment de l'état de l'avancement de l'état d'établissement du dialogue, alors que la méthode UPDATE peut être mappé soit en message SIP re-INVITE ou sinon en message SIP UPDATE. Les phases suivantes du mécanisme de la transaction permettent de mapper le message OK vers la classe 2xx (réponse SIP avec succès), REJECT peut être mappé vers les classes 4xx, 5xx, 6xx qui expriment des réponses d'erreurs et le ACK reste un SIP ACK. La transcription du protocole abstrait vers le protocole SIP nécessite l'utilisation de quelques entêtes supplémentaires. Chaque message nécessite d'inclure un identifiant unique de la conférence en ajoutant l'entête *Conference-ID*. Grâce à cet entête, chaque agent utilisateur peut traiter la requête comme faisant partie de la conférence et non pas comme faisant partie d'une communication de base. Chaque conférence est identifiée par un ID unique. Cet ID peut être généré par le créateur initial de la conférence, possiblement en utilisant la même procédure utilisée lors de la génération de la valeur du champ *call-ID*.

L'entête "*Invited-By*" devrait être inclus dans les messages CONNECT, utilisé par les nouveaux participants pour spécifier l'identité de l'utilisateur qui l'a invité. Finalement, la liste des participants devrait être échangée par les participants et pourrait être fournie à travers l'utilisation de l'entête *Conference-Member*. Le Tableau 1 présente le mappage des messages abstraits vers le protocole SIP et indique aussi les différents entêtes à utiliser. Les entêtes marqués par (\*) sont optionnels.

Message abstrait	Méthode SIP	Entête ajoutée
JOIN	INVITE	Conference-Id Conference-Member
CONNECT	INVITE	Conference-Id Invited-by
UPDATE	reINVITE UPDATE	Conference-Id Conference-Member
OK	2xx	Conference-Id Conference-Member*
ACK	ACK	Conference-Id Conference-Member*
LEAVE	BYE ou CANCEL	Conference-Id
REJECT	4xx, 5xx ou 6xx	Conference-Id

Tableau 1 : Mappage entre les messages abstraits et les messages SIP

## 7.3 PERMETTRE LA MOBILITÉ DE SESSION

### 7.3.1 Extension des messages du protocole

Les messages abstraits présentés pour créer et gérer les conférences complètement maillées ne peuvent pas être déployés pour supporter les options de la mobilité de session. De nouveaux mécanismes de messages, complémentaires à ceux présentés dans le Tableau 1, sont proposés dans le cadre de ce travail. Ces mécanismes sont les suivants : MEDIA-JOIN/OK/ACK, JOIN-REFER/OK, CONNECT-REPLACE/OK/ACK and CONNECT-NOTIFY/OK.

Le mécanisme MEDIA-JOIN est utilisé principalement dans le mode de transfert avec contrôle lorsque le NM désire transférer une partie ou la totalité de sa session média tout en gardant le contrôle sur la communication. Le mécanisme initié par MEDIA-JOIN, basé sur une transaction à trois niveaux, est utilisé conjointement avec le mécanisme CONNECT pour permettre la négociation et l'établissement média entre NL et chaque NC à l'intérieur de deux dialogues contrôlés par le NM (voir Figure 22). La

réponse à la requête MEDIA-JOIN peut-être soit positive (message OK), sinon négative (message REJECT).

Concernant le message JOIN-REFER, il est utilisé dans le cas où le MN réfère un NL pour reprendre la session établie entre le NM et chaque NC. Il s'agit ici d'une cession totale de la session média près laquelle, le NM perd tout contrôle. Les paramètres du dialogue existant établi entre le NM et le NC doivent être inclus dans le message JOIN-REFER est utilisé par le NC lors de l'échange de l'envoi de la requête CONNECT-REPLACE. Le résultat de cette demande de remplacement de nœud pourrait être reporté au NM en utilisant le message CONNECT\_NOTIFY. Lorsque la notification à l'issue de ce transfert est positive, le NM pourra mettre fin à son dialogue avec le NC spécifié en envoyant le message LEAVE. Le message JOIN-REFER pourrait aussi être utilisé par le NM comme un REFER emboîté (nested REFER) pour demander au NL d'envoyer une demande REFER. Ce cas d'utilisation pourrait être utilisable pour permettre au NM de récupérer ses sessions d'origine. Pour ce faire, la procédure devrait se répéter pour inclure tous les NL concernés.

### 7.3.2 Analyse du flux de messages

Cette partie explore les diagrammes de messages qui seront déployés lors de chaque événement qui aura lieu dans chaque des modes de transfert (avec ou sans perte de contrôle). Les événements qui seront considérés dans ce travail sont enclenchés par :

- ✓ Un participant décide de transférer sa session
- ✓ Un participant décide de récupérer sa session
- ✓ Un nouvel utilisateur s'ajoute à la conférence
- ✓ Un participant quitte la conférence

Le transfert des sessions dans une conférence complètement maillée devrait être considérée comme étant une transaction globale qui résulte d'un ensemble de transactions individuelles entre le NM et chacun des NC. Les Figure 22 et Figure 23



montre respectivement le diagramme de messages échangés entre les nœuds participants pour les deux modes de transfert : avec et sans perte de contrôle.

- ✓ **Le mode de transfert avec contrôle** : Le fonctionnement du mode de transfert avec contrôle s'accommode aux différents événements d'ajout et de suppression de participants lorsqu'elles surgissent après le transfert d'une session. Dans ce mode, le NM reste actif et agit comme une passerelle logicielle pour acheminer les différents événements qui se produisent entre le NL et chacun des NC. Par exemple, les requêtes CONNECT ou LEAVE envoyées par un NC se traduisent par un changement des paramètres medias fixés entre le NM et le NL. Ainsi, si un nouvel utilisateur s'ajoute à la conférence, le NM devrait informer le NL afin que le NL fournisse une offre média pour desservir le nouveau NC comme le montre la séquence de messages (de 16 à 21) de la Figure 22.
- ✓ **Le mode de transfert avec perte de contrôle** : Le mode de transfert avec perte de contrôle permet au NM de procéder au transfert de session vers NL et de quitter la conférence par la suite. Le NM initie la procédure en référant la listes des NC au NL en utilisant le mécanisme JOIN-REFER comme le montre les messages 3 et 4 de la Figure 23. Le mécanisme de CONNECT-REPLACE est utilisé par le NL pour contacter chaque NC. Lorsqu'un nouveau dialogue est établi entre le NC et le NL, le NM est notifié à travers le message CONNECT-NOTIFY. Dans ce cas le NM envoie un message LEAVE au NC concerné. La procédure de récupération peut être initiée par le NM, et dans ce cas, il envoie un message JOIN-REFER pour demander l'envoi d'un message JOIN-REFER en utilisant le même principe expliqué dans l'IETF RFC 3515. Le NL peut aussi initier la procédure de récupération et dans ce cas, aucun message JOINT-REFER emboité n'est nécessaire de la part du NM. Le NL ne met fin à sa communication avec un NCi que lorsque la nouvelle session s'établit entre le NM et le NCi correspondant.

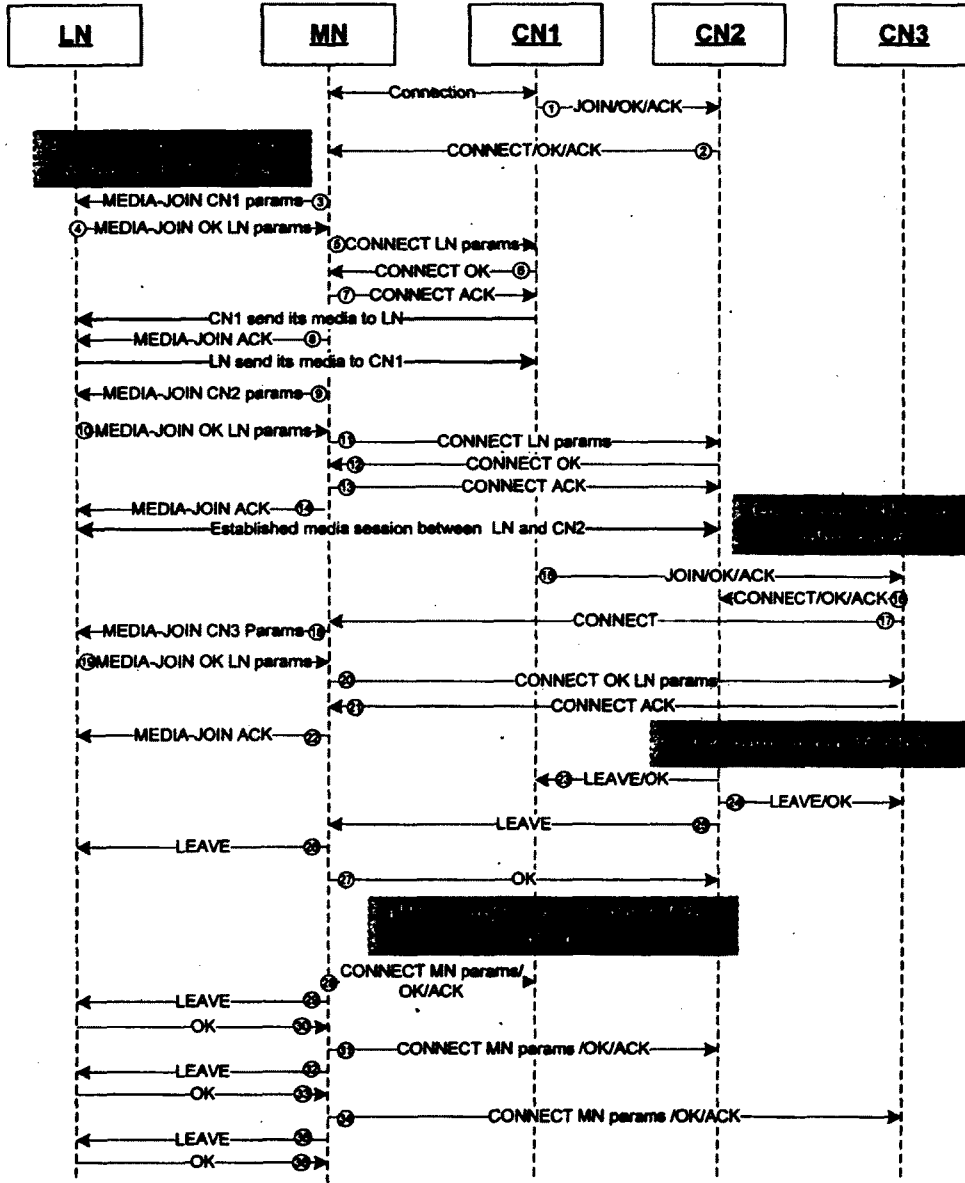


Figure 22 : Exemple d'échange de messages dans le mode de transfert avec contrôle

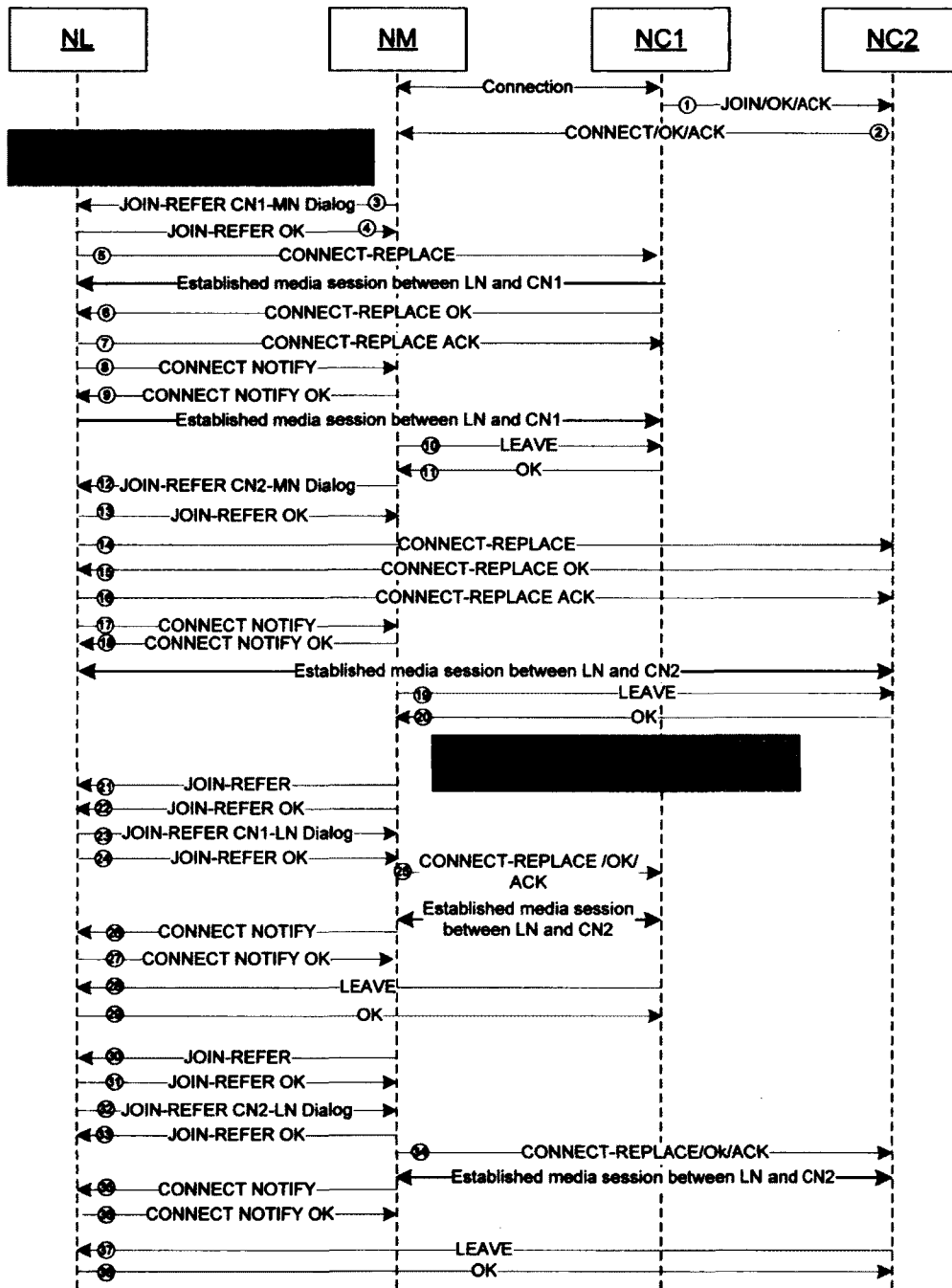


Figure 23 : Exemple d'échange de messages dans le mode avec perte de contrôle

### 7.3.3 Problème de l'échec lors du transfert de session

La procédure de transfert de session dans les deux modes implique le transfert de tous les dialogues établis entre les NC et le NM. Durant cette procédure, le NM procède par un transfert étape par étape où chaque étape correspond à une session établie entre le NM et les (N-1) participants. Il est possible que le transfert d'une des sessions échoue pour une des raisons liées à des problèmes de réseau, à un échec lors de la négociation media, ou autres. Dans ce cas, NM peut effectuer de nouvelles tentatives de transfert jusqu'à atteindre un nombre maximal défini selon les préférences de l'utilisateur. Lorsque le transfert échoue alors que le nombre de tentatives est atteint, la procédure de transfert de session doit être annulée et toutes les sessions précédemment transférées seront récupérées.

La Figure 24 montre un organigramme qui présente les principales étapes de la procédure de transfert pour les deux modes.

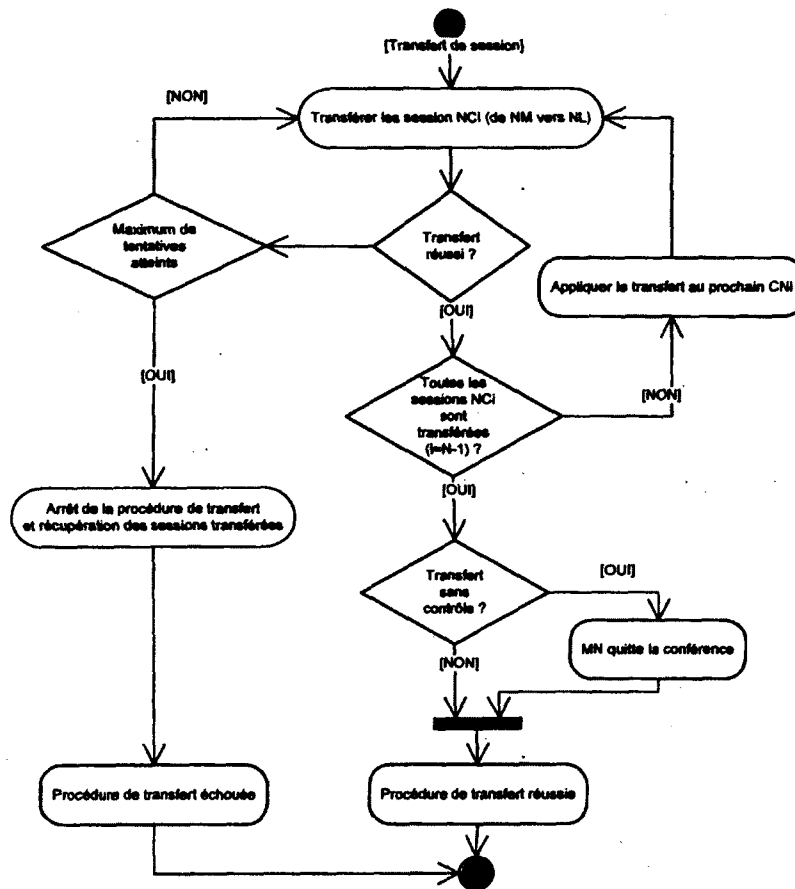


Figure 24 : Organigramme de la procédure de transfert de session

## 7.4 MAPPAGE VERS SIP DES MESSAGES ÉTENDUS DU PROTOCOLE

Au cours des sous-sections précédentes, un nouveau protocole basé sur des messages de bases ainsi que des messages étendus a été défini pour permettre la mobilité de la session au sein d'un modèle de conférence complètement maillé. Cette section propose de faire le mappage des messages étendus vers des messages conformes au protocole SIP.

Le mode de transfert avec contrôle peut être assuré en suivant la technique du Third party call control Flow I spécifié dans [72]. Ce type de modèle de flux correspond au cas où le NC répondrait immédiatement aux requêtes. Le message MEDIA-JOIN peut

donc être mappé vers le message SIP INVITE. Le message OK est mappé vers la classe de réponse avec succès de type 2xx alors que la réponse négative REJECT est mappée vers les classes de réponse 4xx, 5xx ou 6xx selon de type d'erreur exprimé. Le message d'acquiescement ACK reste inchangé.

Dans le mode de transfert avec perte de contrôle, le message JOIN-REFER utilisé pour initier le transfert peut être mappé vers le message SIP REFER [80]. Les entêtes *Refer-To* et *Referred-By* [81] inclus dans les messages REFER peuvent être utilisée par le NL lors de l'envoi de la requête CONNECT-REPLACE au NC. La requêtes CONNECT-REPLACE peut être mappé vers le message SIP INVITE qui contient l'entête *Replaces* comme indiqué dans [52]. Cet entête *Replaces* est utilisé pour contenir des informations sur le dialogue existant entre NC -NM à remplacer. Finalement, le message CONNECT-NOTIFY peut être mappé vers le message SIP NOTIFY utilisé pour rapporter l'état de l'opération de transfert effectué par la requête JOIN-REFER.

Pour garantir l'intégrabilité des messages étendus avec les messages de base du protocole utilisé pour établir une conférence complètement maillée, il est nécessaire d'ajouter quelques entêtes supplémentaires aux messages mappés. Le message MEDIA-JOIN mappé vers SIP INVITE devrait inclure un entête de *Conference-ID*. De cette façon, le NL sera capable d'interpréter le message SIP comme faisant partie de la conférence en cours. Cela permet d'effectuer un traitement média adapté qui consisterait par exemple à mixer les flux audio entrants et de les jouer localement. Dans le mode de transfert avec perte de contrôle, il est important que le message *CONNECT-REPLACES* envoyé par NM au NC inclue l'entête *Conference-ID* pour identifier la conférence à laquelle le NL demande de participer. D'autres entêtes comme le *Conference-Members* et le *Invited-By* peuvent être utilisés pour maintenir la cohérence au sein de la conférence. La valeur de ses entêtes peut être acheminée dès le début de l'opération de transfert lors de l'envoi de la requête JOIN-REFER.

Le Tableau 2 présente brièvement le mappage des messages ainsi que les entêtes additionnels à inclure. Les entêtes marqués par (\*) sont considérés optionnels.

Message abstrait	Méthode SIP	Entêtes ajoutées
MEDIA-JOIN	INVITE	Conference-ID Conference-Member*
JOIN-REFER	REFER	Conference-ID Conference-Members*
CONNECT-REPLACE	INVITE	Replaces Conference-ID Invited-by* Conference-Members*
CONNECT-NOTIFY	NOTIFY	Conference-ID

Tableau 2 : Mappage entre les messages étendus abstraits et les messages SIP

## 7.5 COMPOSANTES DE LA PILE ARCHITECTURALE

Le service de communication de base entre deux agents peut être représenté en mettant en couche trois niveaux de composants : la couche de transport TCP/IP, la pile logicielle de SIP et couche de la logique de fonctionnement de l'application comme indiqué dans la Figure 25.

Il existe des mécanismes inclus dans les piles logicielles de SIP pour traiter efficacement les dialogues et pour fournir les évènements et les opérations appropriés à la couche supérieure c.-à-d. la couche de la logique de fonctionnement. Ces mécanismes tendent à réduire la complexité de la tâche de gestion des sessions SIP. En général, lors de la création d'un nouveau dialogue de communication SIP, un nouveau gestionnaire de traitement (handling manager) est instancié. Chaque message SIP qui fait partie d'un dialogue existant est automatiquement redirigé vers le gestionnaire de traitement approprié. Dans le cas d'une communication de type complètement maillée, chaque conférence se doit de maintenir un ensemble de dialogues. Un gestionnaire de traitement de la conférence se charge ainsi d'assurer le fonctionnement global des différentes communications qui à leur tour sont traitées individuellement au sein d'un gestionnaire de traitement de dialogue.

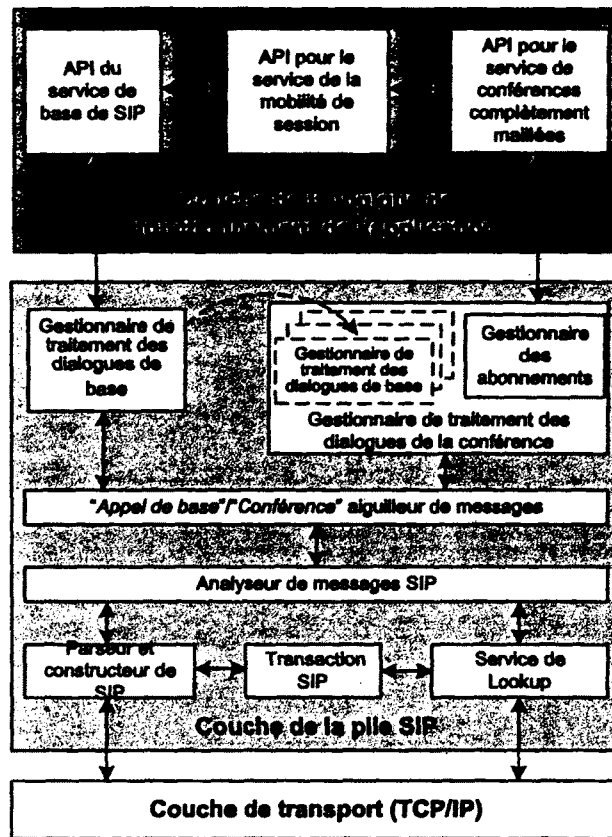


Figure 25 : Interaction entre les composantes de l'architecture en couches

Pour supporter une conférence complètement maillée, il est possible d'étendre la logique de fonctionnement de l'application SIP en ajoutant un traitement supplémentaire pour les dialogues qui appartiennent à chaque conférence. Cependant, une telle solution réduit la performance globale du système en introduisant un traitement supplémentaire puisque chaque message SIP devra remonter à la dernière couche avant d'être identifié comme faisant partie des dialogues de la conférence.

La solution proposée introduit un nouveau composant au niveau de la pile du protocole SIP pour rediriger les messages directement au gestionnaire de dialogue approprié (gestionnaire de dialogue de base ou gestionnaire de dialogues de la conférence). Ce composant, appelé « *Appel de base/Conférence* » aiguilleur de messages » se charge de vérifier l'existence de l'entête conférence-ID dans le message



reçu avant de le rediriger. Pour chaque nouvelle valeur du *Conference-ID*, une nouvelle instance du gestionnaire de traitement des dialogues de la conférence est créée. L'entête *Conference-Members* est analysé par un gestionnaire des abonnements (*Membership manager*) intégré dans le gestionnaire de traitement des dialogues de la conférence. Ce gestionnaire crée une nouvelle instance du gestionnaire de traitement des dialogues lorsqu'un nouveau membre est ajouté à la conférence.

Les deux modes de transfert peuvent être implantés directement dans la couche de la logique de fonctionnement de l'application du moment où la couche de la pile SIP est conforme aux derniers RFCs de l'IETF concernant le protocole SIP. Par exemple, il est impératif que le gestionnaire de dialogue utilisé puisse supporter les entêtes *Replaces* inclus dans le message SIP INVITE ou bien les entêtes *Refer-To* et *Referred-By* inclus dans le message SIP REFER.

Il est possible d'ajouter un API pour le service de la mobilité de session sous forme d'un composant séparé qui permet l'utilisation simultanée du service de base de SIP et du service de la conférence complètement maillée.

## 7.6 ÉVALUATION DU TRAFIC ADDITIONNEL GÉNÉRÉ

Cette section évalue le coût en termes de nombre de messages additionnels associés avec chaque mode de transfert. En considérant le coût unitaire d'un message ( $C_m$ ), deux stratégies différentes peuvent être adoptées :

- ✓ La première stratégie définit un ensemble d'évènements dans un scénario au sein de la conférence. Ce scénario inclut un nombre de sessions transférées ( $N_{st}$ ), un nombre de nœuds ajoutés ( $N_{na}$ ), un nombre de nœuds partis ( $N_{np}$ ) et un nombre de sessions récupérées ( $N_{sr}$ ). Pour un nombre initial ( $N$ ) de participants, il est possible de calculer le cout associé à chaque scénario en fonction du  $C_m$ .
- ✓ Dans la seconde stratégie, on peut utiliser un scénario prédéfini où le nombre de participants  $N$  de la conférence varie.

Pour simplifier l'évaluation, les mécanismes de retransmission des messages du protocole SIP ne sont pas considérés. Ainsi, chaque message envoyé est censé être reçu avec succès. Aussi, les réponses provisionnelles de la classe 1xx pour le cas de SIP ne sont pas comptabilisées dans cette évaluation puisque leur n'affecte pas le retard de l'établissement de la communication. La négociation média qui pourrait avoir lieu lorsqu'un nœud participant change de préférences (changement de type de codec par exemple) n'est pas considérée puisqu'il est difficile d'évaluer la fréquence à laquelle ces changements peuvent avoir lieu. Cette étude évaluative se base sur les diagrammes de flux présentés dans la Figure 22 et la Figure 23.

- Coût d'une session transférée ( $C_{\alpha}$ )
  - Mode avec contrôle :  $C_{\alpha} = (N-1) * 6 * C_m$  (1)
  - Mode avec perte de contrôle :  $C_{\alpha} = (N-1) * 9 * C_m$  (1')
- Coût d'une session récupérée ( $C_{\alpha r}$ ) :
  - Mode avec contrôle :  $C_{\alpha r} = (N-1) * 5 * C_m$  (2)
  - Mode avec perte de contrôle :  $C_{\alpha r} = (N-1) * 11 * C_m$  (2')
- Coût de l'ajout d'un nœud ( $C_{na}$ ) :
  - Mode avec contrôle :  $C_{na} = N_{st} * 3 * C_m$  (3)
  - Mode avec perte de contrôle :  $C_{na} = 0$  (3')
- Coût du départ d'un nœud ( $C_{\alpha}$ ) :
  - Mode avec contrôle :  $C_{np} = N_{st} * 2 * C_m$  (4)
  - Mode avec perte de contrôle :  $C_{np} = 0$  (4')
- Coût d'un scénario dans les deux modes de transfert ( $C_{scenario}$ ) :
  - $C_{scenario} = N_{st} * C_{st} + N_{na} * C_{na} + N_{np} * C_{np} + N_{sr} * C_{sr}$  (5)

Pour simplifier le calcul, il est possible de considérer que le nombre des sessions transférées sera égal au nombre de sessions récupérées ( $N_{st} = N_{sr}$ ) et que le nombre de noeuds qui s'ajoutent à la conférence est égal au nombre de noeuds qui la quittent ( $N_{na} = N_{np}$ ). Cela implique que l'équivalence (5) peut s'écrire :

$$C_{scenario} = N_{st} * (C_{st} + C_{sr}) + N_{na} * (C_{na} + C_{np}) \quad (6)$$

Il est possible d'évaluer le cout associé à chaque scénario. Le ( $C_{sena1}$ ) et ( $C_{sena2}$ ) représentent le cout associé à un scénario dans le mode de transfert, respectivement, avec contrôle et avec perte de contrôle. En utilisant l'équivalence (6) et en remplaçant les couts par leurs valeurs associées évalués dans (1) (2) (3) et (4), il est possible d'évaluer chaque scénario :

$$C_{sena1} = N_{st} * C_m * ((N-1) * 11 + N_{ua} * 5) \quad (7)$$

$$C_{sena2} = N_{st} * C_m * (N-1) * 20 \quad (7')$$

En remplaçant le coût unitaire associé à l'envoi d'un message ( $C_m$ ) par la valeur "1", le coût du scénario calculé dans (7) et (7') pourra refléter le nombre de messages échangés pour chaque scénario. D'un autre coté (7) et (7') montrent que le rapport entre les deux modes de transfert ne dépend pas du nombre de sessions transférées ( $N_{st}$ ). Les Figure 26, Figure 27 et Figure 28 montrent le trafic de messages additionnel généré par chaque mode pour trois scénarios différents lorsque le nombre des utilisateurs qui s'ajoutent et qui se retirent varie.

À partir de la Figure 26, il est possible de constater que le mode de transfert avec contrôle est plus avantageux que le mode avec perte de contrôle en terme de trafic de message généré lorsqu'il n'y pas de noeuds qui s'ajoutent ou qui partent de la conférence. Dès qu'il y a des noeuds qui s'ajoutent et qui partent, le mode de transfert avec perte de contrôle affiche de meilleurs résultats pour un nombre très réduit de participants.

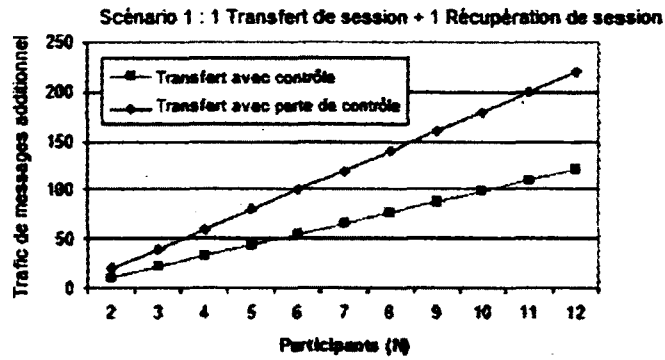


Figure 26 : Trafic de messages additionnel pour le premier scénario

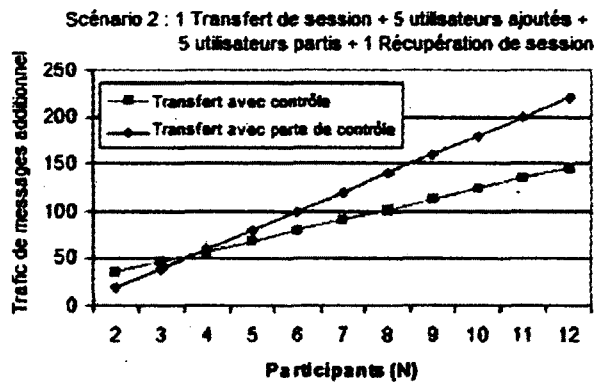


Figure 27 : Trafic de messages additionnel pour le deuxième scénario

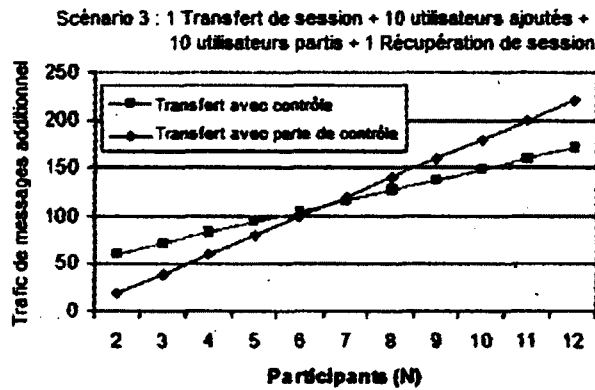


Figure 28 : Trafic de messages additionnel pour le troisième scénario

## 7.7 ANALYSE DE COMPORTEMENT

Pour pouvoir maximiser le confort d'utilisation de l'option de la mobilité de session au sein d'une conférence complètement maillée, il est important que le transfert soit effectué avec un temps d'interruption minimal. Cette section analyse les performances attendues basées sur les critères suivants :

### 7.7.1 Interruption du média lors de transfert

Lors d'une conférence multimédia, il est important de garantir la continuité du flux média audio, vidéo ou n'importe quel autre flux provenant d'autres applications qui se basent sur un échange de flux direct (live streaming). Ainsi, le temps d'interruption, qu'on appellera « temps mort », constitue une priorité lors de l'utilisation d'un service de transfert de session au sein de la conférence. Ce temps mort représente le délai entre le moment où le flux média s'arrête et un autre commence. Durant le transfert, il est possible qu'un segment de média ne soit pas joué et créant ainsi ce temps mort. Ceci pourrait avoir lieu lorsqu'un des nœuds commence à envoyer le flux média vers un autre nœud qui n'est pas encore prêt à le recevoir puisque la session nécessaire n'a pas encore été établie. Ce même temps mort peut aussi avoir lieu lorsqu'un nœud participant arrête d'envoyer le flux alors que le nouveau nœud, qui reçoit le transfert de session, n'est pas encore prêt à commencer l'envoi du média. Dans ce cas, le nœud destinataire n'est pas apte à recevoir pendant cette période. Les diagrammes de messages déjà présentés dans la Figure 22 et la Figure 23 montrent que ce délai ne dépasse pas le temps que prend un paquet pour être acheminé d'un nœud vers un autre. Le temps mort est donc inexistant ou infiniment petit. Ainsi, le NM n'a pas besoin de communiquer avec chaque NC en lui disant : « Veuillez attendre que le transfert soit terminé! ».

Dans le mode de transfert avec contrôle présenté par le diagramme de messages de la Figure 22, lors de la réception du message (11), le NC ne commence à envoyer vers le NL que lors de la réception de l'acquittement dans le message (13). Presque au même moment, le NL reçoit l'acquittement dans le message (14) et commence lui aussi à son

tour à envoyer vers le NC. Pour minimiser le temps d'interruption du côté de chaque NC, le NM peut continuer à envoyer son flux sortant vers les NC, même une seconde de plus, le temps que le NL commence à envoyer son flux vers les NCs. Ainsi, aucune interruption du flux média n'aura lieu, mis à part le temps pris par un paquet RTP pour être envoyé entre les deux terminaux. Il s'agit d'un temps tellement négligeable qui peut ne pas se faire remarquer par les intervenants.

Dans le mode avec perte de contrôle illustré par l'échange de message de la Figure 23, il est possible d'affirmer qu'il n'y aura pas d'interruption puisque le NC1 par exemple commence à envoyer son flux média dès la réception du message (5) CONNECT-REPLACE. Le NL envoie le flux média qui remplace le flux envoyé par NM dès qu'une réponse positive est reçue dans le message (6). Ainsi, du côté de chaque NC aucune interruption n'aura lieu puisque le flux envoyé par NM sera remplacé par NL. L'utilisateur (ayant à proximité le NM et le NL) qui enclenche l'opération de transfert verra une transition instantanée du flux reçu de son NM vers le NL.

### 7.7.2 Latence du transfert total

La latence du transfert total concerne le temps d'attente entre le moment où l'utilisateur initie l'opération de transfert (ou de récupération) et le temps où cette opération est achevée. Même si généralement le NM et le NL se trouvent géographiquement à portée de main ou dans le champ visuel de l'utilisateur, il est important que la latence du transfert soit minimisée pour garantir la continuité (seamless) du service. Une grande durée de cette latence peut laisser croire à l'utilisateur que l'option de transfert de session a échoué.

Les messages de signalisation échangés lors du transfert de session dans les deux modes, présentés dans la Figure 22 et la Figure 23, peuvent traverser de longues distances dépendamment de la localité du NC et du NM/NL. Le délai de la création d'un dialogue avec le NL dépend de la route empruntée. Si les dialogues sont établis à travers des serveurs mandataires, et si les serveurs mandataires du NM ou du NC sont géographiquement très lointains, les messages de l'établissement des connexions sont

routés triangulairement et en nécessitant plus de temps. Si, d'un autre côté, les serveurs mandataires sont situés à proximité de leurs nœuds correspondants ou qu'une connexion sans mandataire (en mode P2P) est utilisée. Le délai d'établissement des dialogues devient très réduit au point de le négliger. D'après [33], la durée d'établissement d'un dialogue SIP, même lorsqu'il s'agit d'une communication transcontinentale, ne dépasse pas une seconde.

Pour les deux modes de transfert, avec contrôle ou avec perte de contrôle, le diagramme d'établissement des dialogues utilise les mêmes mécanismes qu'un appel élémentaire de base. Il est donc possible d'affirmer que le transfert entre le NL et chacun des NC ne dépasse pas une seconde. Pour une conférence à N participants, la latence sera inférieure à N secondes. Puisque le modèle complètement maillé est adapté à une conférence à petite échelle d'une dizaine de participants [49], il est possible d'estimer la latence du transfert à une valeur qui ne dépassera pas les 10 secondes. Ce cas extrême s'il est atteint, peut-être problématique puisque l'utilisateur doit attendre 10 secondes pour effectuer la procédure de transfert. Il est possible dans ce cas d'ajouter une barre de progression ou un affichage textuel qui permet de suivre l'évolution du transfert.

L'utilisation du protocole SIP sur UDP peut aider à réduire le délai de connexion par rapport à l'utilisation de TCP. En effet, l'utilisation de TCP peut rajouter un délai supplémentaire dû aux requêtes d'établissement de la connexion sur la couche transport et l'envoi des requêtes d'acquiescement pour chaque message reçu. Le protocole SIP n'a pas forcément besoin de ces messages d'acquiescement puisqu'il a déjà son propre mécanisme de contrôle qui le fait.

## 7.8 CONCLUSION

Cette partie du rapport a permis de présenter un nouveau protocole pour implémentation des opérations pour offrir le service de mobilité de session au sein d'une conférence de type complètement maillée. Les mécanismes du protocole ainsi que les messages utilisés ont été détaillés ainsi que leur mappage vers le protocole SIP.

### Mobilité des sessions dans un modèle de conférence complètement maillé

Les deux modes de transfert (avec contrôle et avec perte de contrôle) ont été inclus dans cette étude et leur performance en termes de volume de messages a été présentée. Il a été possible de conclure que le mode avec perte de contrôle présente de meilleurs résultats lorsqu'il y a un important trafic d'ajout et de suppression de participants au sein de la conférence. Aussi, l'évaluation de l'interruption média dans le système affiche des résultats très convenables, qui ne devraient pas dépasser la seconde ce qui évite tout désagrément lors de l'utilisation du service.



## **8 MOBILITÉ DES SESSIONS POUR UNE TRANSITION DU MODÈLE HERMÉTIQUEMENT COUPLÉ VERS LE MODÈLE COMPLÈTEMENT COUPLÉ**

### **8.1 INTRODUCTION**

Plusieurs modèles et topologies ont été proposés pour implémenter des conférences entre plusieurs utilisateurs à travers les réseaux IP. Le modèle « end-system mixing » permet un couplage, entre participants, à travers une architecture centralisée autour d'un focus. Un tel couplage est identifié dans la littérature par le terme « *couplage hermétique* » pour traduire les termes anglais *tightly coupled*. Ce modèle hermétique se distingue par la simplicité de son déploiement et la facilité de son établissement et de sa gestion. Ces atouts justifient la popularité du déploiement du modèle à travers les réseaux intranet et internet. Le système Skype par exemple utilise un tel modèle pour établir les communications en mode-conférence. Cependant, l'architecture centralisée sur laquelle repose ce modèle présente quelques défaillances qui limitent son déploiement. Ces limitations apparaissent principalement dans deux scénarios : (1) lorsque le participant mixeur (the focus user) doit quitter la conférence sans affecter les autres participants qui désirent continuer la conférence. (2) lorsque le nombre de participants devient si élevé au point qu'il devient impossible au focus de pouvoir assurer convenablement la tâche de traitement média (mixage audio par exemple).

D'un autre côté, le modèle complètement maillé permet un étalement de la charge média entre les participants tout en leur permettant de quitter sans affecter le déroulement de la conférence. Permettre une transition vers le modèle complètement couplé, même en cours de communication, pourrait certainement aider à résoudre les limitations du modèle hermétiquement couplé. Il s'agit donc de développer une procédure de transformation intra-modèle qui assure cette transition et qui serait perçue par le focus du modèle hermétiquement couplé comme une option à enclencher pour prévenir les limitations du modèle en cours. Dans ce chapitre, présenté dans [25],

deux mécanismes qui supportent cette procédure de transformation sont investigués puis détaillés. Les performances en termes de volume de messages échangés sont évaluées et les approches utilisées sont comparées.

## 8.2 HERMÉTIQUEMENT COUPLÉ VS. COMPLÈTEMENT COUPLÉ

Plusieurs modèles ont été proposés pour permettre à plusieurs intervenants d'établir une communication au sein d'un groupe d'utilisateur. Il est possible de classifier ces modèles selon la topologie de la signalisation et de la distribution média. Le modèle « loosely-coupled » a été développé pour permettre la prise en charge d'un groupe de communication à large échelle. Ce modèle est basé sur la technique du « IP multicasting » pour que chaque participant actif puisse configurer une adresse de multicasting qui inclut la liste des participants à rejoindre. En plus de surcharger le trafic réseau, cette approche nécessite de la part des routeurs d'accepter la fonction du multicasting. Déployer cette solution à l'extérieur des réseaux locaux n'est pas garanti puisque la majorité des routeurs utilisés dans la toile sont configurés pour refuser la technique du multicasting. D'autres modèles de conférence utilisent des protocoles applicatifs de signalisation pour établir un couplage hermétique (tightly) en sélectionnant un ou plusieurs focus dont la responsabilité est de maintenir la communication au sein de la conférence. Un déploiement basique d'une telle approche crée une topologie d'arborescence unique. Cette arborescence peut être utilisée pour garantir un couplage des liens de signalisation et du flux média entre les différents nœuds participants (Figure 29.a). Ceci nécessite de la part du focus la prise en charge de la tâche du mixage et de distribution média. La signalisation nécessaire pour la gestion d'un tel modèle est analysée plus tard dans ce chapitre. D'autres modèles utilisent une topologie qui distribue la partie signalisation dans un couplage hermétique alors que le flux média suit une distribution de type complètement maillée qui vise à distribuer équitablement la charge de traitement média comme l'illustre la Figure 29.c. Lorsque la signalisation et le flux média sont complètement distribués, il devient possible d'obtenir une architecture semblable à celle de la Figure 29.b. Les modèles de la Figure 29.a et la

## Mobilité des sessions pour une transition du modèle hermétiquement couplé vers le modèle complètement couplé

Figure 29.c présentent une architecture dont la signalisation est centralisée autour du focus. Cela pose des limitations lorsque le focus veut quitter la conférence sans affecter les autres participants. Ainsi, il serait intéressant de focaliser l'étude sur une transformation du modèle de la Figure 29.a vers le modèle de la Figure 29.b.

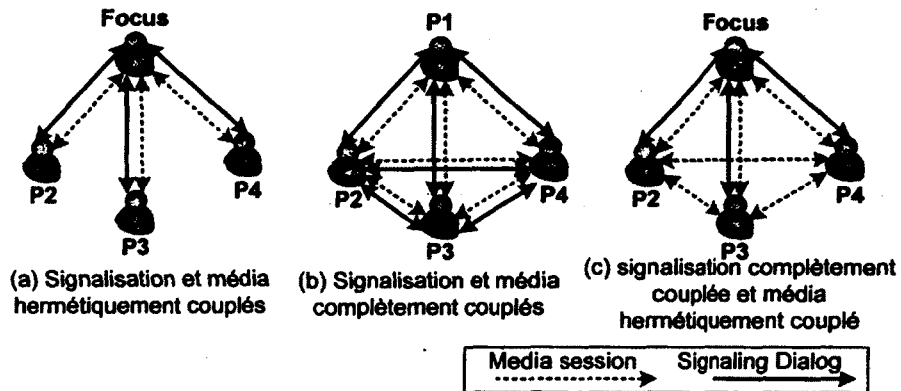


Figure 29 : Présentation de quelques modèles de conférence

### 8.2.1 L'approche hermétiquement couplée

Dans la topologie centralisée du modèle hermétiquement couplé, la gestion des dialogues de signalisation ainsi que le traitement média sont effectués par le focus. Ce focus récupère le flux média envoyé par chaque participant, le mixe avec les autres flux entrants et envoie un flux mixé approprié à chaque utilisateur séparément. Ainsi, chaque participant reçoit le mixage de tous les autres mis à part le sien. Deux modèles de conférences peuvent être envisagés en suivant ce principe de fonctionnement. Le focus peut être un des N participants (End-System Mixing Model) ou bien une machine dédiée appelée généralement serveur de conférence. (Server Mixing Model). La création et la gestion d'une conférence hermétiquement couplée selon les primitives du protocole de signalisation SIP ont été le sujet de plusieurs recherches antérieures. Alors que quelques-unes de ces recherches [45] [66] se sont intéressées à l'aspect de la signalisation, d'autres ont traité l'aspect de gestion média [83]. D'une manière générale, ces travaux traitent spécifiquement les problèmes liés à la création/destruction de la conférence et à l'ajout/suppression de participants. Par exemple, différents modes

d'ajout de participant (*Dial-in*, *Dial-out*, planifié ou ad-hoc) sont inclus dans ces travaux. Les mécanismes qui utilisent les primitives SIP INVITE, REFER et NOTIFY [74][80][65] sont largement déployés pour permettre à un nouvel utilisateur de s'intégrer dans une communication existante. Dans tous les modes, la conférence nécessite d'être uniquement identifiée par une adresse URI. Le participant qui joue le rôle de focus peut ajouter le paramètre « isfocus » dans le champ de l'entête contact de façon à ce que les autres participants puissent identifier la communication comme étant une conférence. Plus encore, dans [7], l'utilisateur peut s'inviter, par lui-même, en utilisant l'entête Join à l'intérieur de la requête INVITE lors de son envoi à l'URI de la conférence. Dans un tel champ, l'utilisateur devrait inclure des informations concernant le dialogue existant. Une telle information peut être obtenue à l'avance par différents moyens. Dépendamment de la nature de la conférence (publique ou privée), il est possible d'afficher ces informations sur le web, l'envoyer par courriel ou même par messagerie instantanée.

### **8.2.2 L'approche complètement couplée**

L'approche complètement couplée permet la réalisation d'une architecture de communication où chaque participant détient un lien direct de communication avec chacun des participants restants. Tous les participants sont donc égaux, aucun utilisateur n'est différent d'un point de vue topologique et aucun droit additionnel n'est accordée à part celui donné aux autres participants.

Chaque participant peut à n'importe quel moment ajouter les utilisateurs à la conférence. Si le nouvel utilisateur accepte, il établit les connexions nécessaires avec le restant des participants pour conserver le maillage complètement couplé. De la même manière, chaque participant peut à n'importe quel moment quitter la conférence, sans affecter les autres participants. Dans le modèle complètement couplé présenté dans [25], un ensemble de messages abstraits sont proposés pour permettre la gestion du fonctionnement de la conférence. L'implémentation du protocole, comme décrit dans [25] nécessite l'introduction de nouveaux entêtes. Par exemple, les messages échangés

au sein d'une même conférence incluent l'entête *Conference-ID* pour identifier d'une manière unique la conférence. Cet ID peut être généré par le créateur initial de la conférence et possiblement en utilisant la même procédure que celle utilisée pour générer la valeur du champ *Call-ID*. L'entête *Invited-By* doit aussi être utilisé lors de l'envoi du message INVITE par un nouveau participant. Cet entête est utilisé pour indiquer l'adresse du participant responsable de l'ajout du nouvel utilisateur. Finalement, la liste des membres de la conférence peut être incluse dans l'entête *Conference-members*. La méthode UPDATE peut aussi être utilisée pour assurer la cohérence au sein de la conférence, en permettant un échange de la liste des membres à chaque ajout/suppression d'utilisateur. Plus de détails sur ce modèle de conférence ont été présentés dans la section 7.2

## **8.3 LA TRANSITION VERS LE MODÈLE COMPLÈTEMENT COUPLÉ**

### **8.3.1 Les approches de transitions**

Pour permettre la transition d'un modèle de conférence vers un autre, il est nécessaire de développer des mécanismes qui permettent la création, la modification, le déplacement et même la suppression des dialogues de communication entre les utilisateurs. Puisque les topologies des conférences sources et destinations sont établies par des liens créés par le protocole SIP, il devient évident et naturel d'utiliser ce même protocole pour implémenter les mécanismes de transition. La transition du modèle hermétiquement couplé vers le modèle complètement couplé nécessite l'établissement de nouveaux dialogues entre les participants qui n'ont pas le rôle du focus, comme le montre la Figure 30. Pour le cas de la problématique présentée dans cette section, deux approches ont été explorées. La première approche, explicite, est basée sur le mécanisme de base de SIP REFER/NOTIFY alors que la deuxième approche est plutôt implicite s'inspire du protocole d'établissement des conférences complètement maillées détaillées dans la section 7.2 à la page 73 de ce rapport.

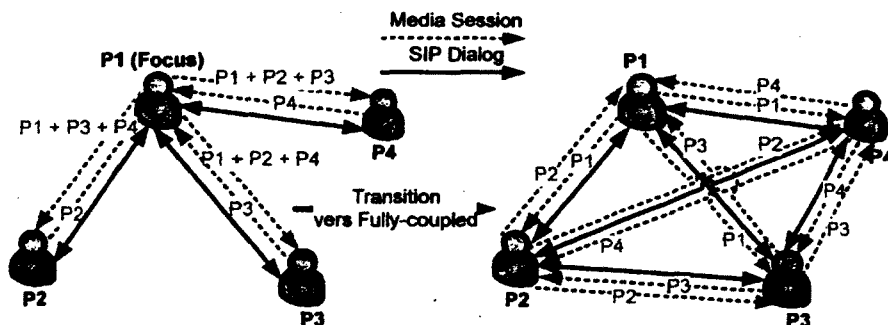


Figure 30: Aperçu de la transition entre modèles

### 8.3.2 Les étapes de la transition

Pour implémenter la transition vers le modèle complètement couplé, N-1 étapes sont définies pour une conférence initiale de N participants. À travers ces étapes, le participant focus (P1) doit procéder à une série d'invitations pour inclure, à chaque étape, un participant comme le montre la Figure 31. Après un nombre (n-1) d'étapes, P1 se retrouve dans une situation qui lui permet d'être un participant à la nouvelle conférence complètement maillée composée par {P2, P3, ...Pn} tout en continuant à assumer le rôle de Focus dans la conférence hermétiquement couplée composée par {Pn+1, Pn+2, ..., PN}.

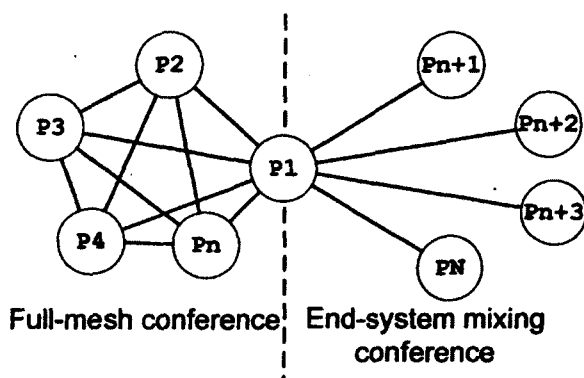


Figure 31 : La topologie de la conférence après (n-1) étapes

### 8.3.3 Le protocole des messages

Pour implémenter les (N-1) étapes, chaque approche utilise son propre mécanisme détaillé dans cette section. Dans l'approche explicite, le focus (P1) effectue la transition entre les deux modèles en ajoutant les participants du modèle hermétiquement couplé, un par un, vers le modèle complètement couplé. Lors de cet ajout, le focus, étant maître de la situation, réfère à chaque participant de contacter l'ensemble des participants de la nouvelle conférence complètement couplée, en cours de construction. Ainsi, pour une étape intermédiaire n, la transition du participant Pn+1 du premier groupe de conférencier vers le deuxième groupe nécessite du P1 l'envoi de (n-2) messages REFER. Lors de ces (n-2) messages, chaque message REFER doit correspondre à un des utilisateurs de la conférence complètement maillée. Lorsque P1 reçoit les (n-2) messages NOTIFY relatifs aux messages REFER, il peut considérer que la transition a réussi et il pourra dans ce cas passer à l'étape (n+1).

L'approche implicite est relativement différente de celle qui est explicite puisqu'elle se base sur le mécanisme d'établissement des conférences complètement maillées présenté dans [49]. Cette approche suppose que tous les nœuds supportent le protocole expliqué dans [49] et prennent donc en charge les entêtes supplémentaires de ce protocole. Pour exécuter l'étape n, le focus (P1) envoie à Pn+1 le message abstrait JOIN qui peut être mappé vers le message SIP INVITE. Ce message P1 inclut tous les participants existants de la conférence complètement couplée. Lors de la réception de ce message, Pn+1 contacte tous ces participants, un par un, en envoyant le message abstrait CONNECT qui peut lui aussi être mappé vers le message SIP INVITE. Ce message CONNECT inclut l'entête « Invited-By » qui spécifie l'adresse de P1 (l'invitant original de Pn+1). Lors de la réception du message CONNECT, chaque participant envoie le message UPDATE pour notifier le focus (P1) concernant la présence d'un nouveau participant. Ainsi, P1 sera au courant de la progression de l'établissement des dialogues dans le nouveau groupe de conférence et assurera l'enclenchement de l'étape n+1 dès que l'étape n est terminée.

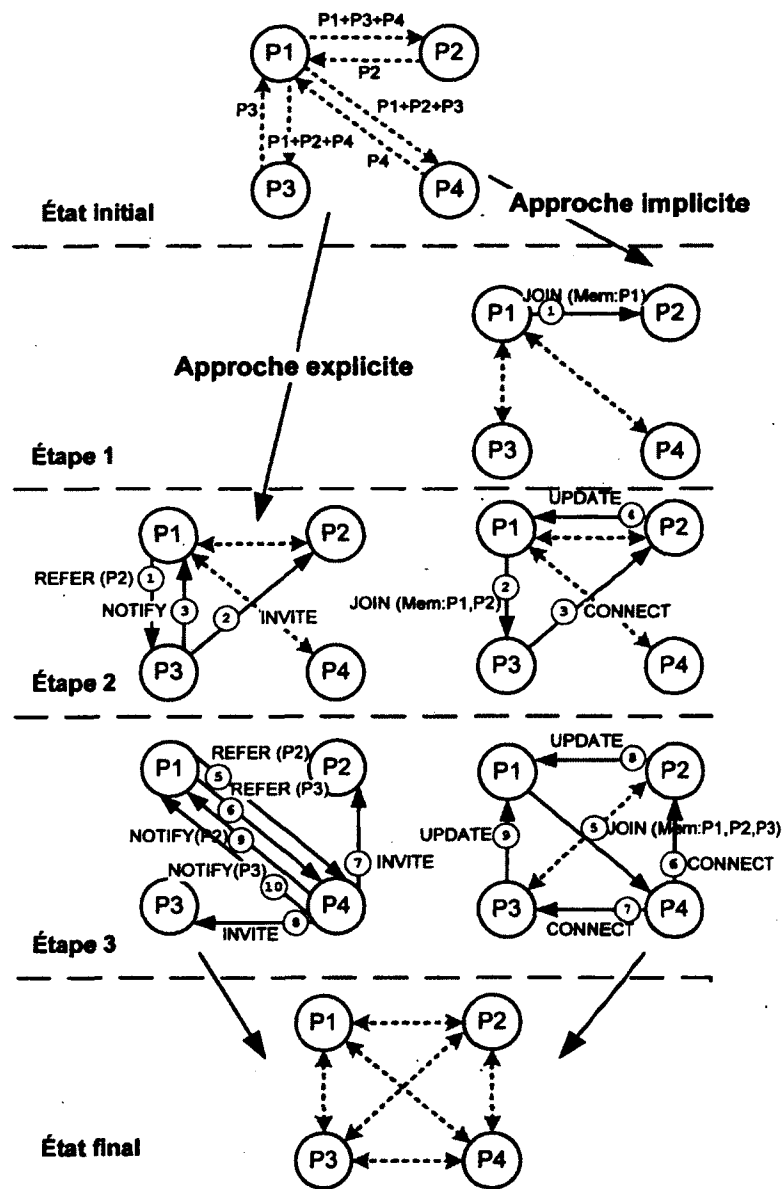


Figure 32 : Les étapes de la transition vers le modèle complètement couplé (N = 4)

## 8.4 ÉVALUATION DU COÛT DE LA TRANSITION

Pour évaluer le coût associé à la transition entre les deux modèles, nous considérons la quantité des messages échangés entre les N participants. Puisque les



Mobilité des sessions pour une transition du modèle hermétiquement couplé vers le modèle complètement couplé

approches implicites et explicites se basent sur des mécanismes différents, il a été jugé utile d'inclure une étude comparative pour évaluer les performances de chacune. Pour une transition avec N participants, le coût en terme de message ( $T_{cost}$ ) est décrit par la somme des (N-1) coûts de chaque étape ( $Step_{cost}$ ):

$$T_{cost} = \sum_{i=1}^{N-1} (Step_{cost})_i \quad (1)$$

Dans l'approche explicite, le coût de chaque étape ( $Step_{cost}$ ) est défini par :

$$(Step_{cost})_i = (ReferT_{cost} + InviteT_{cost} + NotifyT_{cost}) * (i - 1) \quad (2)$$

Dans l'approche implicite, le coût de chaque étape ( $Step_{cost}$ ) est défini par :

$$(Step_{cost})_i = JoinT_{cost} + (ConnectT_{cost} + UpdateT_{cost}) * (i - 1) \quad (3)$$

Dans (2) et (3), il est possible de remplacer chaque transaction par son coût associé en terme de messages échangés. Par exemple «  $ReferT_{cost}$  », représente le coût de la transaction initié par REFER. Puisque cette méthode REFER est basée sur le mécanisme « requête/réponse », le coût associé sera 2. Les transactions initiées par Notify et Update nécessitent aussi 2 messages chacune, alors que les transactions JOIN, CONNECT et INVITE nécessitent 3 messages. Dans cette évaluation, nous supposons que toutes les requêtes seront suivies par des réponses positives et que les transactions réussissent dès la première tentative.

À partir de la Figure 33, il est possible de remarquer que lorsque N est égal à 5, les deux approches ont un coût identique en termes de messages échangés. Pour des conférences avec 3 ou 4 participants, l'approche explicite est moins coûteuse. Cependant, pour un nombre de participants supérieur à 5, l'approche implicite nécessite moins de messages. La différence en termes de volume des messages augmente proportionnellement avec le nombre de participants.

## Mobilité des sessions pour une transition du modèle hermétiquement couplé vers le modèle complètement couplé

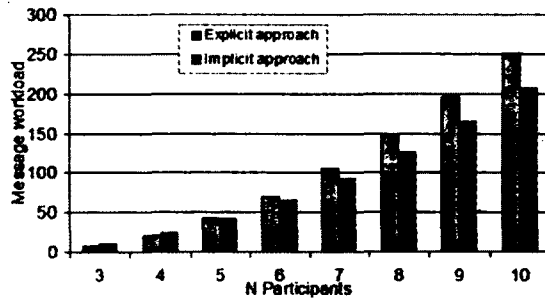


Figure 33 : Évaluation de la quantité de messages échangés

### 8.5 CONCLUSION

Dans ce travail, il a été possible de déployer les mécanismes de mobilité des sessions et des communications pour réaliser deux approches de transitions du modèle hermétiquement maillé vers le modèle complètement maillé. La transition permet de prévenir les limitations du modèle hermétiquement couplé. Ainsi, il devient possible d'assurer la continuité de la conférence même si le *focus* désire quitter ou lorsque la charge de traitement média devient très élevée pour être traitée par un seul participant. Dans ce travail, nous avons traité cette limitation en effectuant la transition vers le modèle complètement maillé, cependant, il est aussi possible d'envisager des solutions pour déléguer la fonction du *focus* à un autre participant, ou même garder la signalisation centralisée autour du *focus* et effectuer un balancement de la charge média. Cette deuxième alternative de balancement de la charge média semble très intéressante puisqu'elle permettrait la formation d'un groupe de conférence à large échelle. Plusieurs défis sont à résoudre lors de la construction d'un tel système. Nous consacrerons le chapitre prochain pour discuter d'une telle architecture qui combine le modèle distribué et le modèle centralisé pour construire un modèle composite probablement plus performant.

## 9 MOBILITÉ DES SESSIONS POUR LE SUPPORT DES CONFÉRENCES P2P À LARGE ÉCHELLE

### 9.1 INTRODUCTION

Plusieurs modèles et architectures sont proposés pour supporter le service de communication de voix sur IP en mode-conférence, généralement dénoté par l'abréviation MVoIP (*Multiparty Voice over IP*). Les premières solutions de MVoIP [57][95] se sont basées sur l'infrastructure réseaux en utilisant les routeurs pour rediriger les flux médias au sein de groupes de communication à large échelle. Ces routeurs utilisaient la technique du *multicasting* pour permettre une diffusion des données selon l'emplacement des utilisateurs dans le réseau. Puisque la fonction du *multicasting* n'est généralement pas supportée par les routeurs disponibles publiquement dans le réseau internet, cela a limité le déploiement de ces solutions à l'extérieur des réseaux locaux. Pour résoudre une telle limitation, les solutions actuelles de MVoIP implémentent le service de distribution des données directement du côté de l'utilisateur final à travers le service unicast de IP. Ainsi, cette distribution n'est plus tributaire de l'infrastructure réseau (routeur). L'objectif dans ces solutions est de créer une couverture (*overlay*) topologique basée sur des liens de communication et gérée par les applications à la place des liens de routages. Cette nouvelle génération de solution est généralement identifiée par l'abréviation ALM (*Application Layer Multicast*)[17][97]. La popularité des ALM se justifie par la facilité de leur déploiement puisqu'ils ne nécessitent que l'installation d'application logicielle de la part de l'utilisateur indépendamment de sa technologie d'accès et de son emplacement sur le réseau IP.

Dépendamment de l'assignation de la charge média, deux approches sont actuellement utilisées par les systèmes ALM pour supporter le service de MVoIP. La première approche implémente la charge de traitement média directement sur un des participants et distingue principalement deux modèles : Le premier se base sur un couplage topologique hermétique, généralement appelé ESM (*End system Mixing*) [18][97] qu'utilise par exemple le logiciel populaire Skype [89]. Le deuxième modèle

permet un couplage complet entre les participants dans une architecture complètement décentralisée [49] [64]. Même si ces deux modèles se distinguent par la facilité de leur déploiement, leur utilisation reste adaptée principalement pour les conférences à échelle réduite. En effet, le modèle ESM limite habituellement le nombre de participants à 4 ou à 5 dépendamment de la bande passante et de la puissance de traitement du focus, alors que le modèle complètement couplé limite le nombre à une dizaine de participants. La seconde approche fait intervenir une machine externe, dédiée, appelée serveur de conférence qui permet de contrôler la conférence et de traiter la tâche de mixage et de distribution média. Les solutions de MVoIP, qui permettent un déploiement à large échelle, utilisent cette approche. D'ailleurs plusieurs produits commerciaux comme IVIST [43] ou WEBEX [96] sont actuellement disponibles et peuvent supporter jusqu'à une centaine de participants simultanés à l'intérieur d'une même conférence. Déployer et maintenir un serveur de conférence dans un groupe de communication de telle envergure peut s'avérer très dispendieux puisque cela nécessite une très large bande passante de la part du serveur de conférence et des capacités de traitement non négligeables. Pour résoudre les problèmes relatifs à la surcharge du serveur central, à la congestion de la bande passante et au point central de défaillance, plusieurs travaux de recherches ont introduit une approche qui se base sur plusieurs serveurs de conférence afin d'étaler la tâche de traitement média sur l'ensemble des serveurs disponibles. Ainsi, les serveurs sont géographiquement dispersés et chaque participant sera attaché automatiquement au premier serveur disponible et géographiquement proche [83][47]. Une telle solution, même si elle évite le problème du point central de défaillance, introduit un coût additionnel lié au maintien des serveurs déployés. D'un autre côté, de plus en plus d'utilisateurs à travers internet disposent de terminaux de mieux en mieux équipés en termes de puissance de calcul et de bande passante. Ainsi, il sera possible de créer un système de MVoIP basé sur les terminaux finaux (*MVoIP based peers*) afin d'accomplir la charge de traitement média de manière « collective » et « coopérative » dépendamment de la disponibilité et des capacités de chaque terminal participant.

Contrairement aux systèmes de distribution de flux de données basé sur les infrastructures (réseaux ou serveurs de conférence), les solutions ALM basées sur les terminaux ont reçu récemment beaucoup d'attentions [37][12][15][16]. Ces solutions se fixent comme objectif principal de produire un système d'intercommunication efficient qui s'organise de lui-même (*self-organizing*) et capable de s'auto-évoluer (*self-improving*). Dans ces systèmes, un nœud participant peut décider du choix de son père, peut migrer dans l'arbre pour minimiser la transmission redondante de l'information sur les liens physiques. Malheureusement, nous avons trouvé que ces solutions se focalisent principalement sur les métriques réseaux pour former le groupe et pour évoluer. Ces systèmes ne prennent pas en compte l'activité au sein de la conférence. Par exemple, un participant qui contribue par une intervention audio (mode locuteur) dans la conférence devrait être traité d'une façon différente de celui qui ne fait qu'écouter (mode auditeur). Aussi, implémenter des fonctionnalités complexes de contrôle pour tous les utilisateurs finaux peut restreindre l'accès aux terminaux limités en termes de ressources. Plus encore, le concepteur d'un service de VoIP en mode-conférence à large échelle basé sur des nœuds participants, devrait considérer plusieurs aspects relatifs à la création de la conférence, la gestion de l'adhésion, le contrôle de la distribution média, et le contrôle du droit de la parole (*speech floor*) pour chaque participant. Implémenter toutes ces fonctionnalités pour tous les participants, de façon distribuée, devient une tâche d'une extrême complexité. En effet, une vision globale et unifiée de l'espace de dialogue (*floor*) ainsi qu'un contrôle unique des décisions relatives à l'ajout/suppression des utilisateurs et à la distribution média sont tous des contraintes requises pour le fonctionnement de la conférence. Maintenir une version mise à jour de ces informations et les partager entre les nœuds, pour une conférence à large échelle, est aussi une tâche dont la complexité limite sa faisabilité.

Pour résoudre ces défis, nous introduisons un système basé sur une approche qui sépare le réseau média du réseau de contrôle. Nous pensons qu'une telle séparation peut rendre possible la création d'un système de communication de Voix sur IP à large

échelle basé exclusivement sur le modèle P2P. La section suivante permet de détailler la solution proposée.

## 9.2 APPROCHES DE COUPLAGE MÉDIA

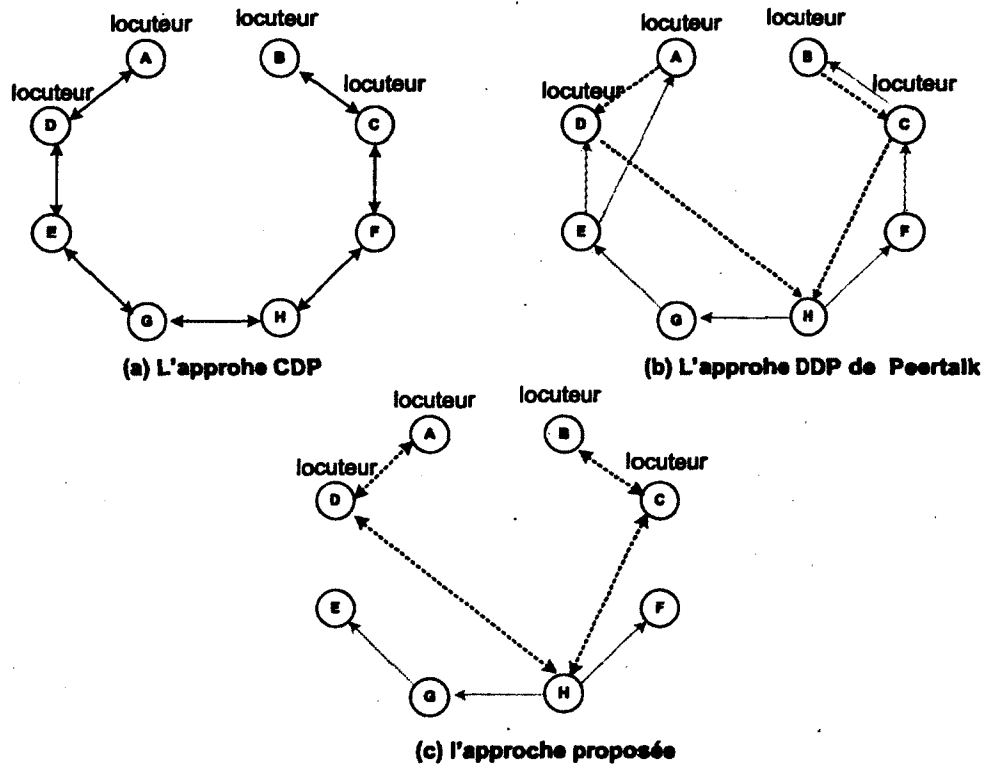


Figure 34 : Quelques approches proposées pour le service MVoIP à large échelle

Plusieurs travaux existants [8][64][49] ont proposé une approche de traitement distribué couplé (*Coupled Distributed Processing – CDP*) qui utilise le même arbre pour les flux audio mixés et le flux distribué comme illustré dans la

Figure 34.a. Cependant, il est possible de constater que le service de *MVoIP* présente des propriétés asymétriques : 1) le nombre de locuteurs (qui génèrent un flux audio) est généralement différent du nombre de nœuds auditeurs qui ne font qu'écouter la conversation, et 2) la bande passante entrante d'un nœud peut-être différente de la

bande passante sortante (exemple ADSL). Ainsi, l'approche CDP peut être suroptimale en utilisant le même réseau pour à la fois mixer et distribuer.

En début 2008, une première approche de traitement distribué découplé (*Decoupled Distributed Processing – DDP*) a été présentée par *PeerTalk* [37], illustré par la Figure 34.b. Ce modèle DDP partitionne l'acheminement des flux médias selon deux phases : 1) *la phase de mixage* qui permet de mixage de tous les nœuds locuteurs dans un seul flux à travers un arbre de mixage, et 2) *la phase de distribution* qui permet de distribuer le flux mixé à tous les nœuds participants. Ce modèle de traitement découplé est plus adapté aux applications MVoIP caractérisées par leur asymétrie. D'après [37], l'approche DDP de *PeerTalk* permet, par rapport à l'approche CDP, d'optimiser les délais de réception des flux audios. Aussi, elle permet d'optimiser l'utilisation de la bande passante. Même si le système *PeerTalk* décrit un algorithme qui permet d'optimiser la topologie de la conférence et de la rendre dynamique par rapport à l'activité des participants, nous remarquons qu'aucun mécanisme d'implémentation n'est proposé. Ce vide laisse croire que le système *PeerTalk*, à part sa contribution théorique, ne pourra pas être implémenté et déployé pour le grand public dans sa forme actuelle. Nous proposons dans ce travail des outils qui permettent l'implémentation des opérations de division, de fusionnement et de migration des mixeurs proposés dans les algorithmes de *PeerTalk*. Le protocole proposé se base sur SIP et utilise le service de la mobilité selon une approche unique [60] qui sépare le réseau média du réseau de contrôle.

Il a été aussi remarqué que le modèle de mixage de *PeerTalk* envoie le flux sortant du locuteur à tous les nœuds participant de façon à ce que ce flux revienne de nouveau à ce même locuteur. Cela a pour conséquence de produire un effet « *echo* ». Pour résoudre ce problème, l'utilisation d'une topologie qui se base à la fois le modèle CDP pour le réseau de mixage et sur le modèle DDP pour le réseau de distribution peut résoudre ce problème.

### 9.3 LE MODÈLE DU SYSTÈME

La solution proposée se base sur une approche qui sépare le réseau média du réseau de contrôle comme illustré dans la Figure 35. Ainsi, le réseau de traitement média utilise une topologie arborescente (multi centralisée) alors que le réseau de contrôle est basé sur une topologie totalement centralisée autour de l'administrateur de la conférence (contrôleur). Plus encore, chaque utilisateur qui rejoint la conférence peut offrir son service pour effectuer le traitement média à l'administrateur qui se charge du contrôle de la distribution média. Cet administrateur exploite principalement la technique du *Third Party Call Control* [72] pour établissant, mettre à jour et fermer d'une manière indirecte les liens médias qui relient les participants qui sont sous son contrôle. Le système permet le contrôle de l'espace de dialogue en attribuant le droit de parole aux participants qui le demandent. Les participants peuvent migrer au sein de l'arbre média dépendamment de leurs capacités et de leur activité. L'optimisation de l'arbre-média ne sera pas traitée dans ce travail qui se focalise plutôt sur la présentation des composantes du modèle et du protocole de communication à déployer. Le système proposé supporte le départ rapide ou la panne des participants et utilise des mécanismes qui permettent à d'autres nœuds de prendre la relève (handoff) afin de réparer l'arbre média dans de telles situations. Le système proposé implémente la gestion des sessions en utilisant des messages abstraits qui sont mappés vers le protocole SIP.

La solution introduit un modèle qui permet à plusieurs terminaux de supporter la charge de traitement média (mixage ou diffusion) alors que le contrôle de la conférence demeure centralisé et géré par un seul participant. Cet utilisateur occupe ainsi le rôle d'administrateur ou plus communément appelé modérateur. Pour élaborer une telle architecture, deux réseaux interconnectés sont construits pour permettre à la fois la distribution audio et le contrôle de la conférence comme indiqué dans la Figure 35. Le premier réseau média utilise trois types de nœuds ayant des rôles différents : Le Nœud Mixeur Distributeur (NMD), le Nœud Distributeur (ND) et le Nœud Feuille (NF). Le rôle du NMD est affecté aux nœuds qui permettent de mixer et de distribuer le flux média



pour les autres nœuds. Alors que le rôle du ND est attribué aux nœuds qui reçoivent le flux audio déjà mixé, le joue localement et le distribue à un ensemble de nœuds enfants. Ce rôle est adapté aux utilisateurs qui fonctionnent dans le mode locuteur seulement. Le dernier rôle concerne le NF qui est affecté aux nœuds qui détiennent une seule session média. Leur départ ne devrait pas causer de « dégâts » puisqu'il ne perturbe pas le fonctionnement de la conférence. Il ne nécessite généralement pas de modification dans la structure de l'arbre média à moins que ce soit pour des raisons d'optimisation. Un tel rôle est adapté pour les terminaux légers ayant des ressources limitées tels qu'un téléphone IP conventionnel, un terminal qui est connecté avec une bande passante très réduite ou appartenant à un participant qui ne désire pas partager la capacité de calcul ou sa bande passante avec les autres.

Le deuxième réseau établit une topologie centralisée qui permet de relier chaque participant au nœud de l'administrateur (NA). Pour chaque nœud utilisateur, un nouveau lien est créé. Ces liens sont utilisés pour délivrer l'information de contrôle entre le NA et tous les autres participants. Pour garantir la robustesse du système et pour ne pas se fier sur un point central unique de contrôle, le NA devra élire un nœud répliqua, appelé ici NRA (Nœud Répliqua Administrateur), parmi les participants afin de le remplacer en cas de départ ou d'échec.

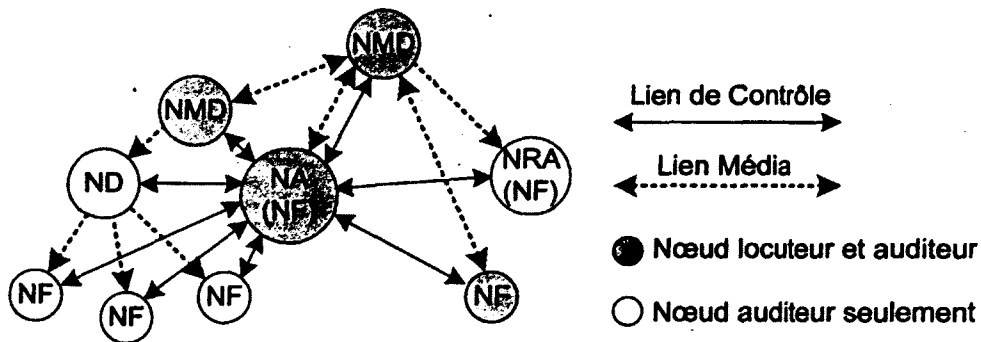


Figure 35 : Topologie générale du réseau de contrôle et du réseau média

## 9.4 LES COMPOSANTES DU SYSTÈME

Le système de conférence défini par le modèle permet à un large nombre de participants dispersé géographiquement d'établir une communication audio de groupe en temps réel. Supporter un nombre important de participant implique de nouveaux challenges pour la distribution média et le contrôle de la conférence. Le système proposé inclut les composantes suivantes pour répondre à ces challenges comme le montre la Figure 36:

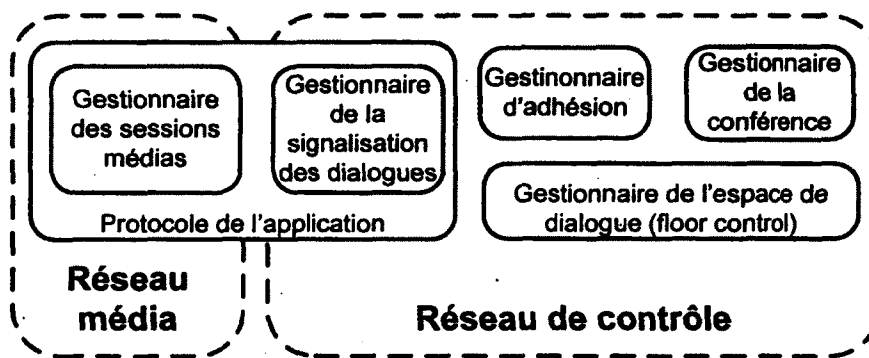


Figure 36 : Composantes du système

### 9.4.1 Gestionnaire de la conférence

Ce composant est responsable principalement de l'annonce de la conférence, de la modification de ses propriétés et de sa destruction. Chaque conférence est identifiée d'une façon unique par une adresse URI créée par le NA. Cette adresse est communiquée publiquement ou d'une manière privée aux autres utilisateurs du réseau internet/intranet en l'affichant dans une page web dédiée, par courriel, clavardage ou forum de discussions. Plusieurs paramètres d'ordre général sont définis par ce composant : titre/sujet de la conférence, nombre maximal de participant supporté, nombre d'intervenants qui peuvent parler simultanément, etc.

### 9.4.2 Gestionnaire d'adhésion

Ce gestionnaire est responsable des opérations d'ajout et de suppression des utilisateurs. Les requêtes d'ajout ou de suppression peuvent être initiés par les le nouvel utilisateur lui-même (le mode dial-in) ou par le NA (le mode dial-out). Avant d'accepter la demande d'ajout ou avant d'envoyer l'invitation, le gestionnaire d'adhésion devra consulter les restrictions imposées par le gestionnaire de la conférence pour vérifier les règles et les limitations imposées. (Utilisateur préautorisé, utilisateur faisant partie de la liste noire, nombre limite de participants atteints, etc.).

### 9.4.3 Gestionnaire de l'espace de dialogue

Ce gestionnaire gère les autorisations d'utilisation des ressources partagées au sein de la conférence en limitant le nombre d'accès simultanée à chaque ressource pour respecter sa capacité maximale et garantir ainsi une qualité de service acceptable. Les conférences basées sur les réseaux IP peuvent avoir plusieurs types d'espaces de dialogues c.-à-d. audio, vidéo, tableau blanc, pointeur de souris, jeux partagés, etc.. Puisque ce travail se limite au support d'un système de MVoIP, il est naturel de focaliser la gestion de l'espace de dialogue sur deux aspects principaux : le partage de la bande passante (assignation des flux médias) et l'attribution du droit à la parole (gestion des droits d'interventions). L'espace de dialogue utilise différents états pour gérer l'accès au droit à la parole comme le montre la Figure 37. Le participant qui se joint à la conférence est automatiquement mis dans la file d'attente « *ask to listen* ». Dès qu'un NDM distributeur est disponible, le participant pourra entrer dans le mode « *listening* » pour pouvoir suivre l'évolution de la communication. Dès que le participant formule une requête pour avoir le droit d'intervenir, il est placé dans la file « *ask to talk* » et il sera affecté au premier NMD mixeur disponible. La gestion des files d'attente et l'attribution des droits de parole sont à la responsabilité du NA.

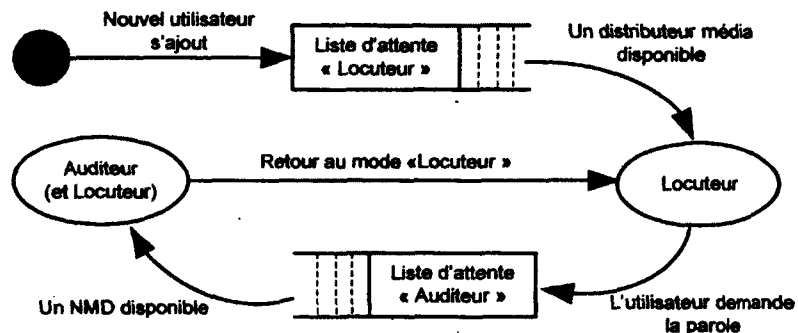


Figure 37 : Aperçu de la gestion du droit à la parole dans l'espace de dialogue

#### 9.4.4 Protocole de l'application

Ce composant implémente le protocole de communication qui permet à la fois l'établissement du réseau de contrôle/administration et l'établissement du réseau de distribution média entre les participants. L'établissement des communications nécessite que le protocole puisse supporter des opérations d'initialisation, de modification et de terminaison. D'un autre côté, l'échange des données supplémentaires entre participants peut se faire par un mécanisme souscription/notification pour reporter les événements nécessaires dès qu'elles se produisent. De telles fonctionnalités peuvent être supportées par le protocole SIP. En effet, SIP permet d'initier la conférence, permettre aux utilisateurs de la joindre ou de la quitter, de gérer les flux médias, généralement basés sur le protocole RTP, en utilisant une syntaxe de description adéquate à travers le protocole SDP. La technique du *Third Party Call Control* (3PCC) [72] définie par SIP est utilisée pour permettre l'établissement d'une session média entre deux utilisateurs qui possèdent un lien de contrôle avec le NA. Puisque le protocole SIP permet aussi de supporter un contenu textuel dans le corps de ses messages, il est possible d'encapsuler des données codées en format XML dans les messages échangés.

#### 9.5 LE RÉSEAU DE DISTRIBUTION DU FLUX MÉDIA

La solution se base sur une approche de traitement média en mode P2P pour acheminer le flux audio entre participants. Cette approche prend comme acquis qu'il y a

suffisamment de nœuds participants qui désirent partager leurs ressources lorsqu'ils s'ajoutent au système. Chaque nœud  $N$  participant peut limiter le nombre des nœuds qu'il veut supporter. Par exemple, on dénote  $(D)$  le débit du flux audio constant ( $D_{in}$ ) le débit entrant maximal supporté de la connexion IP et ( $D_{out}$ ) le débit sortant maximal. Le nombre maximal de flux audio entrants est  $(D_{in}/D)$  et le nombre maximal de flux sortants est  $(B_{out}/B)$ . Ses valeurs appelées *degré maximal entrant* et *degré maximal sortant* définissent respectivement pour chaque nœud le nombre maximal d'enfants à mixer et le nombre maximal d'enfants auxquels il est possible de distribuer le flux. Ces valeurs peuvent être suggérées par l'application installée sur chaque nœud et approuvées par l'utilisateur. Elles peuvent aussi être simplement configurées directement par l'utilisateur. Dans un souci de garantir une qualité de service appropriée, l'application doit prévenir l'utilisateur pour l'empêcher de choisir une valeur qui dépasse le degré maximal supporté. La valeur du degré maximal entrant et sortant peut être modifiée au cours de la communication pour s'adapter aux préférences de l'utilisateur ou aux différents changements dans l'état du réseau.

Lorsqu'un nouvel utilisateur se joint à la conférence, il peut notifier le NA de son offre de service en fournissant la valeur de son degré entrant et sortant. Le NA peut utiliser ses valeurs pour organiser l'attribution des charges médias sur chaque nœud. Par exemple, les nœuds qui affichent un degré entrant supérieur à 1 sont disponibles pour mixer l'audio pour les autres (voir Figure 38a) alors que les nœuds qui affichent une valeur élevée du degré entrant sont généralement disposés pour distribuer le flux aux autres (voir Figure 38b). Les nœuds qui proposent un degré sortant et entrant élevé sont généralement disposés pour mixer et distribuer (voir Figure 38c). Réellement, il est presque difficile d'utiliser un modèle de nœuds de la Figure 38a puisque dans ce cas, il faut supposer que les nœuds A, B et C ne reçoivent pas de flux entrant, et donc, ils ne sont pas en mesure de suivre l'évolution de la communication. Par exemple. Une telle architecture ne permettra pas au nœud A de recevoir le flux de B. Les nœuds qui ne désirent pas fournir leur offre de service devront fixer leur degré entrant et sortant à la valeur 1.

Les échantillons audio qui arrivent de toutes les entrées d'un nœud sont périodiquement accumulés et transformés en un seul échantillon par le mixeur. Cet échantillon résultant est normalisé puis envoyé dans un paquet à chaque destinataire du réseau de distribution. L'algorithme de mixage se base sur la séquence décodage-mixage-encodage. Généralement, lorsqu'un paquet audio arrive dans le mixeur, il est décodé vers un échantillon linéaire de 16 bits et mis par la suite dans un tampon d'attente relatif à chaque participant [83]. Chaque tampon est étiqueté par l'estampille temporelle (timestamp) correspondante. Le vacillement (jitter) qui s'introduit lors de l'arrivée des paquets IP est absorbé par le délai « play-out » de l'algorithme. À des intervalles réguliers, une routine de l'algorithme mixe un ensemble d'échantillons, provenant de différents tampons, en effectuant une addition des valeurs de chaque échantillon. D'un autre côté, la fonction de distribution permet à chaque nœud de dupliquer le nouvel échantillon mixé vers différentes copies qui sont envoyées à travers les ports sortants.

En traversant plusieurs nœuds-mélangeurs, le temps de mise en paquet (packetization) et le délai de play-out s'ajoute au délai total entre la source audio originale et celle reçue par les participants. Ce délai a encore plus d'impact lorsque le participant qui le reçoit est en mode « active speaker ». Ainsi, il est important d'optimiser l'organisation des NMD de façon à déployer un nombre minimale entre les participants ayant le mode « active speaker ». Le choix du premier NMD doit se porter sur celui qui offre des degrés entrants et sortants élevés. Durant la conférence, il est possible que des participants changent de mode (du mode écoute vers le mode « active speaker » ou vice versa par exemple). Il est donc important que le NA puisse réorganiser l'arbre de distribution du flux média pour prendre en compte ces changements. Le protocole utilisé devrait par conséquent offrir au NA de tels outils d'intervention pour changer la topologie en cours de communication.

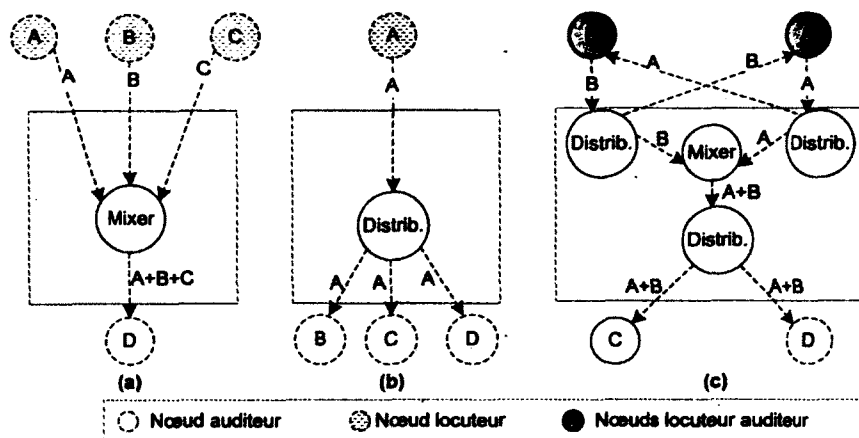


Figure 38 : Cas d'utilisation du traitement audio

## 9.6 MÉCANISMES DE SUPPORT DE L'ADHÉSION

### 9.6.1 Introduction

Cette section présente un protocole abstrait qui permet d'implémenter la signalisation du système de contrôle et le maintien des liens média entre participants. Dans cette approche, nous assumons qu'il y a assez de participants disponibles pour mixer et distribuer le média pour les autres. Du moment que tous les participants fournissent leurs paramètres pour définir leur degré entrant et sortant, il devient possible au NA d'être au courant des ressources disponibles pour faire le traitement média.

### 9.6.2 Ajout d'un nouvel utilisateur

Deux modes différents sont offerts pour ajouter un nouvel utilisateur à la conférence. Dans le mode Dial-in, le nouvel utilisateur initie l'appel en proposant une offre média à travers le message JOIN. Le NA redirige l'offre du nouvel utilisateur (NU) vers un NMD disponible et approprié en utilisant les messages (2) (voir Figure 39.a). Une réponse média est donc fournie par le NMD à travers le message (3). À ce moment-là, le NMD ouvre un port média pour envoyer le flux mixé vers l'adresse et le numéro de port fourni par le NU dans son offre média initiale. Cette réponse est acheminée vers le NU.

Le dialogue entre le NA et le NU est établi lors de la réception du message (5). L'ajout d'un nouvel utilisateur peut aussi se faire dans le mode «Dial-out » qui permet au NA de proposer au NU de faire partie de la conférence. Le mécanisme d'échange de message est presque semblable au premier mode, à la différence que le NA n'obtient l'offre média du NU que lors de la deuxième requête (2), comme le montre la Figure 39.b. pour obtenir cette offre, le NA formule sa première requête (1) sans proposer d'offre (appelé offre Black Hold). Dans ces deux modes, l'élément central demeure le NA qui garde le contrôle des deux liens de signalisations, le premier établies avec le NU et le deuxième avec le NMD.

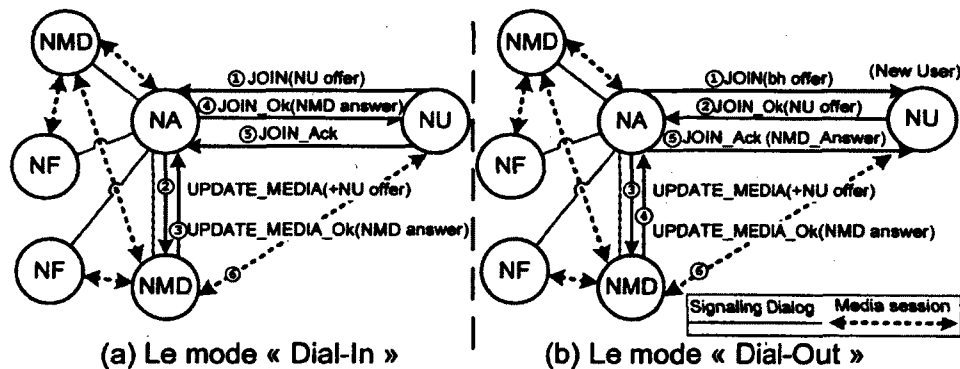


Figure 39 : Mécanisme d'ajout d'un nouvel utilisateur

### 9.6.3 Départ ou échec du NF

Le NF qui ne joue pas un rôle clé dans l'arbre de distribution média peut quitter la conférence, en mode « dial-in » en envoyant un message LEAVE (1) au NA comme illustré à la Figure 40. Le NA peut aussi décider de mettre fin à la participation du NF, en utilisant le mode « dial-out », en envoyant le message LEAVE en premier. Dans les deux modes, le NA devra envoyer le message (2) au père du NF. Ce père correspond au NMD qui assure la connexion du NF dans l'arbre-média.

Pour différentes raisons, liées à des problèmes matériels/logiciels ou même réseautiques, le NF peut quitter la conférence sans notifier le NA. Dans ce cas, le lien média entre le NF et son père est automatiquement déconnecté puisqu'un



évènement de type « media disconnected » sera normalement reporté par le protocole RTCP utilisé conjointement avec le protocole RTP qui assure la liaison média. Ce même départ brusque pourra être détecté par le NA qui détient un lien de contrôle avec le NF au sein d'un dialogue SIP. Un évènement de type « Dialogue disconnected » devrait être généré par la pile de protocole SIP du NA. Même si le NA et le NMD responsable du LF pourront détecter ce départ précipité du NF, il reste important d'utiliser les messages (2) et (3) échangés entre le NA et le NMD pour garantir la cohérence de la conférence.

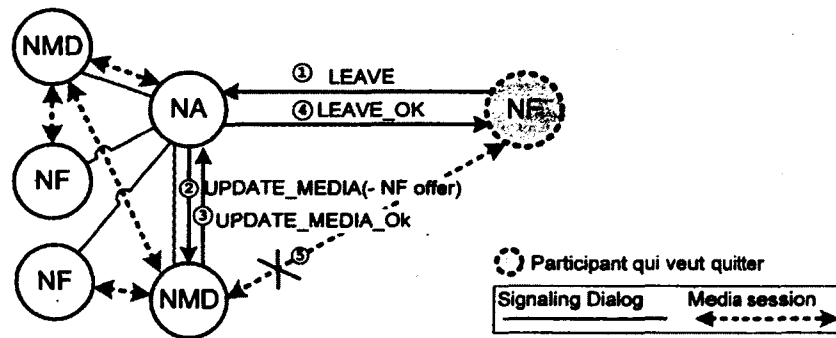


Figure 40 : Mécanisme pour le départ du NF

#### 9.6.4 Départ ou échec du NDM

Le départ où l'éventualité d'échec du NMD est moins évidente à supporter que celle du NF puisque ce départ peut affecter l'arbre de distribution média. En effet, si le NMD se retire, l'arbre-média se retrouvera partitionné en deux ou même plusieurs sous arbres qui ne seront plus connectés. Dans le cas où le départ s'annonce à l'avance (utilisation du mode dial-in ou dial-out), le NA peut agir d'une façon proactive pour prévenir l'impact de ce départ. Dans cette approche proactive, le NA devrait rediriger les enfants, qui peuvent être affectés par le départ du NMD, avant d'accepter la demande LEAVE. Ainsi, l'interruption média pourra être minimisée. Si le NMD quille sans prévenir, la reconstruction média sera exécutée par le NA, dans un mode réactif, dès que l'évènement « dialogue disconnected » est reçu. La Figure 41 illustre le cas où le NMD1 quitte la conférence. Dans ce cas, le NF2 sera séparé de l'arbre-média. Le NA vérifie s'il y a des nœuds disponibles (par rapport à leurs degrés entrant et sortant) pour leur

assigner le NF2. Dans ce cas, NA sélectionne NF3 pour être le nouveau père de NF2. Il procède donc avec un mécanisme d'échange de message (voir les messages de 2 à 7) pour établir un lien média entre NF2 et NF3. Après une telle opération, le NF3 se transforme en NMD.

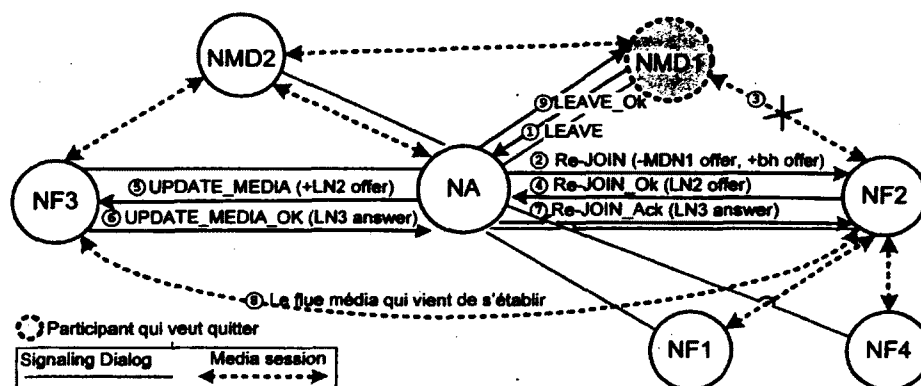


Figure 41 : Mécanisme pour le départ du NMD

### 9.6.5 Départ ou échec du NA

Puisque le système proposé se fixe comme objectif d'éviter le problème du point de défaillance central, le NA devra élire au moins un nœud répliqua, appelé ici NRA. Le NA devra fournir au RNA une information mise à jour qui permet de décrire la structure complète de la conférence. Cette information devra contenir une description des liens médias existants de l'arbre et des liens de contrôle établis. Ceci a pour objectif de permettre au RNA de pouvoir ré-établir les liens de la conférence. Ce changement de nœud administrateur peut se faire de manière proactive, si le NA notifie le RNA à l'avance ou bien de manière réactive dès que le RNA détecte que le NA est en défaillance. La procédure de reconstruction se base sur l'utilisation du message initial REPLACE\_AN qui se base sur un mécanisme de requête/réponse/acquittement comme le montre la Figure 42. En effectuant cette procédure, le RNA redevient le nouvel administrateur de la conférence en établissant des liens de contrôle avec chaque nœud participant. Ces nœuds notifient l'ancien NA de ce changement en envoyant un message

LEAVE. Dans un mode proactif, le NA pourra quitter la conférence théoriquement lorsqu'il aura reçu des messages LEAVE de tous les participants.

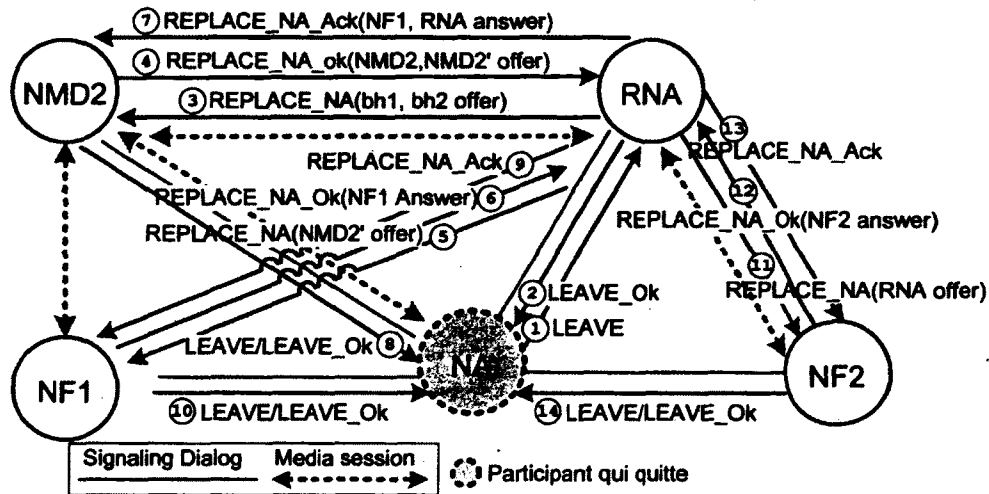


Figure 42 : Mécanisme pour le départ du NA

## 9.7 IMPLÉMENTATION DU SYSTÈME EN SIP

### 9.7.1 Mappage des opérations du protocole

Les messages du protocole défini et utilisé dans les diagrammes d'échange de message dans la section 9.6 peuvent être mappés vers SIP. Cette implémentation nécessite l'utilisation de quelque entête supplémentaire. Par exemple, chaque message devra indiquer l'identifiant de la conférence en utilisant l'entête Conf-ID. Le message JOIN utilisé pour ajouter de nouveaux utilisateurs à la conférence est passé vers le message SIP INVITE alors que le message RE-JOIN est passé vers reINVITE qui permet de mettre à jour un dialogue déjà établi. Les deux phases du processus d'établissement de la connexion qui suit JOIN et reJOIN se font mappé de façon à ce que JOIN\_OK et reJOIN\_OK devient un message SIP de la classe de réponse positive 2xx. Le JOIN\_ACK et le reJOIN\_ACK se traduisent par un simple SIP ACK pour finaliser chaque transaction.

Le message UPDATE\_MEDIA message est mappé vers le message SIP UPDATE basé sur un mécanisme de transaction à deux phases (requête/réponse). Le message

REPLACE\_AN mappé vers INVITE devra contenir l'entête « Replaces » pour identifier le dialogue existant à remplacer par le nouveau. En fin, le message LEAVE est remplacé par la requête SIP BYE

### 9.7.2 Implémentation de l'offre/réponse média

La solution proposée utilise le protocole SDP pour décrire le mécanisme d'échange média offre/réponse [73] qui établit le flux média entre deux participants. Le contenu SDP encapsulé dans les messages SIP fournit des informations sur la description générale de la session, le temps de création et les médias utilisés. Puisque le protocole de message qui implémente les opérations de création et de mise à jour des connexions est conforme à la procédure du *Third Party Call Control Flow* spécifié dans [72], nous utilisons le mécanisme offre/réponse pour négocier la session média dans chaque transaction. Chaque offre média est décrite dans le format suivant :

```
m=<media> <port> <transport> <fmt list>  
c= <IP address>
```

Par exemple, l'offre média du NU inclus dans le message UPDATE\_MEDIA présenté dans la Figure 39.a ajoute les lignes suivantes pour supporter chaque nouvel enfant dans l'arbre-média :

```
m=audio 49230 RTP/AVP 0  
c= IN IP4 224.5.6.7
```

Dans cet exemple, "49230" représente le port de réception média ouvert par le NU, le dernier chiffre "0" est le numéro prédéfini du type de payload du codec média et "224.5.6.7" est l'adresse IP du NU.

### 9.7.3 Implémentation de l'assignation de la charge média

L'utilisateur qui supporte des charges de distribution ou de mixage média peut publier ses capacités à l'intérieur du contenu SDP destiné habituellement pour la négociation des capacités médias. Pour supporter cette extension, nous ajoutons une nouvelle « lettre », par exemple « d= » dans la dernière ligne de l'offre SDP. Dans cette

ligne, il peut définir le degré entrant et sortant de chaque nœud. SDP supporte naturellement une telle extension. Par exemple, si le nœud supporte 3 flux entrants à mixer et peut fournir 6 flux sortants, il décrit ces capacités en ajoutant la ligne suivante :

```
d= IN 3 OUT 6
```

Lorsque l'utilisateur veut changer son offre de service, il peut contacter le NA en envoyant une requête de type UPDATE\_MEDIA contenant la nouvelle offre SDP. L'utilisateur qui ne supporte pas une telle extension, ne sera donc pas en mesure de proposer une offre de service. Dans ce cas, ce nœud devra être traité comme un NF.

#### 9.7.4 Implémentation du contrôle de l'espace de dialogue

La syntaxe SDP permet l'utilisation de paramètres supplémentaire pour spécifier la direction du flux média (*sendrecv*, *sendonly*, *recvonly*). Puisque tous les médias négociés sont contrôlés par le NA, il devient facile d'ajuster les offres et les réponses média de façon à définir le réseau média adéquat selon l'algorithme adopté. À titre d'exemple, si NF1 et NF2 sont connectés au même NMD et si le NF est un simple auditeur alors que le NF2 est un locuteur, l'offre média offerte au NMD sera de cette forme :

```
m=audio 49230 RTP/AVP 0
c= IN IP4 add_NF1
a=recvonly
m=audio 49231 RTP/AVP 0
c= IN IP4 add_NF2
a=sendrecv
```

#### 9.7.5 Implémentation de la gestion de la conférence

Chaque conférence est identifiée par une adresse SIP URI créée par le NA. Ce NA peut contacter un serveur d'enregistrement de SIP (registrar server) pour associer l'URI de la conférence à son adresse réseau (adresse IP). Il est possible d'utiliser des mécanismes externes pour publier l'URI de la conférence publiquement sur le Web. Lorsque le NA quitte la conférence, le NRA qui reprend le contrôle de la conférence devra recontacter le serveur d'enregistrement pour associer l'URI de la conférence vers la nouvelle adresse réseau. La gestion de la conférence se base sur un modèle XML qui permet de décrire la conférence et l'arbre-média qui associé. Chaque balise qui

contient un nœud de l'arbre-média inclut des informations qui permettent d'identifier les paramètres du dialogue établi avec le NA (call-id, to-tag et le from-tag). Un exemple de la structure XML créée par le NA est :

```
<Conf_description>
  <headers>
    <Conf_URI> ... </Conf_URI>
    <Conf_title> ... </Conf_title>
    <Max_participants> ... </Max_participants>
    <Creation_date> ... </Creation_date>
  </headers>
  <media_tree>
    <Node add=" Call_ID= to-tag= from-tag= >
      //children description
      <Node add=" Call_ID= to-tag= from-tag= media_param/>
      <Node add=" Call_ID= to-tag= from-tag= media_param/>
      //children description
      <Node add=" ... />
      <Node add=" ... />
    </Node>
  </media_tree>
</Conf_description>
```

À chaque nouvelle action d'ajout ou de suppression de participant, le NA doit modifier la structure XML et envoyer une nouvelle copie au NRA. Puisque le NA et le NRA ont un dialogue SIP qui les relie, le NA peut utiliser la requête SIP NOTIFY dans le même dialogue pour transférer le fichier XML. L'identification des dialogues SIP contenus dans le fichier XML peut être utilisée par la requête REPLACE\_AN pour reprendre les dialogues. Les paramètres "media\_param" sont utilisés pour reconstruire le flux original qui relie chaque nœud avec son père.

## 9.8 ÉVALUATION DES PERFORMANCES DU SYSTÈME

L'objectif de ce travail est de proposer un protocole qui permet d'implémenter un réseau média pour une conférence MVoIP à large échelle. Puisque le nombre de nœuds peut-être important dans un tel réseau média, et puisque chaque nœud peut effectuer un décodage/encodage des données audio reçues, la qualité audio reçue par les nœuds peut subir une dégradation. Il est donc important de pouvoir évaluer cette possible dégradation selon le niveau du nœud dans l'arbre et les paramètres du codec audio utilisé. Plus encore, dans une telle conférence, il est peut fréquent que deux intervenants parlent en même temps. Ceci a pour effet de créer des moments de silence non négligeables. L'utilisation d'un détecteur de silence, même si elle peut affecter la

qualité audio, permet d'économiser la bande passante dans les liens. Il serait donc intéressant de retrouver un bon compromis pour l'utilisation de l'option de détection de silence.

### 9.8.1 Environnement de simulation

L'évaluation du système proposée a été effectuée à travers l'environnement de simulation OMNet [59] sous Linux. Cet environnement permet de créer différentes topologies de nœuds réseau interconnectés à travers des liens de communication. Chaque lien peut être paramétré pour définir sa bande passante, son taux d'erreur et le délai de propagation qu'il occasionne. Quatre fonctionnalités principales ont été développées et implantées dans chaque nœud. Dépendamment de l'emplacement du nœud dans le réseau média, ces fonctionnalités s'activent automatiquement.

- ✓ **Fonction de locuteur** : Elle permet d'associer un fichier audio source (fichier .wav) au nœud. Ce fichier source est encodé selon les paramètres du codec utilisé et ensuite envoyé sous forme de paquets RTP vers les nœuds destinataires.
- ✓ **Fonction d'auditeur** : Cette fonction permet de recevoir les paquets RTP de chaque source et de les décoder selon le codec utilisé. Chaque contenu décodé est à la fois mis dans une mémoire temporaire (un tableau) et stocké dans un fichier audio de destination.
- ✓ **Fonction de mixeur-distributeur** : Cette fonction s'exécute en étroite collaboration avec la fonction d'auditeur qui permet de décoder le contenu des paquets RTP reçus. Lorsqu'il s'agit de plusieurs sources de flux entrant, cette fonction permet de les mixer de façon à offrir à chaque nœud destinataire le flux correspondant (voir Figure 38.c).
- ✓ **Fonction de distributeur** : Cette fonction permet au nœud de diffuser le même flux à tous les nœuds qui lui sont directement connectés et qui font partie de l'auditoire.

Pour supporter les opérations d'encodage et le décodage audio, la librairie *avcodec* du projet FFmpeg [34] est intégrée à l'environnement de simulation. Il s'agit d'une librairie en libre utilisation qui supporte un grand nombre de codec.

Pour faciliter et accélérer le processus d'évaluation de la qualité de la parole reçue dans chaque nœud, nous avons préféré utiliser le standard PESQ (Perceptual Evaluation of Speech Quality) [92] de l'ITU-T. L'utilisation d'une salle d'écoute aurait probablement donné un résultat plus proche de la réalité. Cependant, le grand nombre de tests à effectuer et surtout la durée prolongée des enregistrements audio à évaluer ont été des éléments déterminants pour orienter notre choix vers une méthode d'évaluation avec l'outil PESQ. Chaque test PESQ produit un résultat qui représente une note d'opinion moyenne connue sous l'abréviation MOS (*Mean Opinion Score*) qui donne une idée générale sur la qualité de la parole. Les valeurs du PESQ-MOS définies par la recommandation P.862 de l'ITU [85] varient de -0.5 (plus mauvais) jusqu'à 4.5 (meilleur). Ceci est différent de l'échelle MOS originale de l'ITU qui se classe dans un intervalle allant jusqu'à la valeur 5 (voir Tableau 3). Ceci peut se justifier par le fait que le PESQ produit une simulation des tests d'écoute. Il est donc optimisé pour reproduire le résultat moyen de tous les participants au test. Les statistiques ont montré que le meilleur résultat moyen auquel on peut s'attendre est de 4.5. Pour faire la différence entre le MOS original et le MOS du PESQ, nous utilisons dans le reste de ce document l'appellation « PESQ » pour signifier la valeur MOS du PESQ.

MOS	Qualité	détérioration
5	Excellent	imperceptible
4	Bon	Perceptible, mais pas dérangeant
3	Moyen	Légèrement dérangeant
2	pauvre	Dérangeant
1	mauvais	Très dérangeant

Tableau 3 : les valeurs du MOS selon l'ITU-T



### 9.8.2 Paramètres de la simulation

Le système proposé vise une conférence à large échelle avec un nombre important d'intervenants. Les nœuds participants s'échangent un contenu audio réel provenant de fichiers audio. Chaque fichier audio représente l'intervention d'un locuteur dans une conférence qui a duré 10 minutes. L'ensemble des locuteurs regroupe 4 intervenants (2 hommes et 2 femmes) situés dans des environnements avec un bruit environnant différent (silencieux, légèrement brouillant, très brouillant et extrêmement brouillant). En utilisant un logiciel de mixage audio, il a été possible de créer un fichier mixé qui permet de suivre toute la discussion qui s'effectue dans la conférence. Ce fichier mixé de 10 mn est utilisé comme source de référence pour mesurer la dégradation dans le réseau média.

La librairie *avcodec* utilisée supporte plusieurs types de codec. L'objectif de ce travail n'est pas de tester différents codecs ou de faire une étude comparative entre eux. Le critère principal était de tester la qualité de la parole lorsqu'elle traverse différentes opérations d'encodage/décodage. Le codec G.726 a été sélectionné pour faire l'évaluation. La simple et unique raison qui a motivé ce choix est qu'il supporte un codage en 16 000, 24 000, 32 000 et 40 000 bit/s et que son PESQ optimal est très satisfaisant (supérieur à 4).

Plusieurs autres paramètres ont été prédéfinis dans l'environnement de simulation, comme le nombre d'échantillons par paquet qui a été fixé à 64. Le G.726 fonctionne avec une source échantillonnée à 8 000 Hz, en insérant 64 échantillons dans chaque paquet, nous obtenons des paquets de 8 ms de parole. L'entête du paquet est fixé à 32 bits (4 octets), qui correspondent à la taille des entêtes d'un paquet RTP. Un détecteur de silence peut aussi être utilisé. Le principe de fonctionnement de ce détecteur de silence consiste à effectuer la vérification suivante avant d'envoyer le paquet : si la valeur absolue de l'amplitude du signal est inférieure à la valeur du détecteur de silence, un paquet avec un contenu vide est envoyé (32 bits d'entête seulement). Sinon, le paquet est envoyé sans modifier son contenu.

### 9.8.3 Résultats de la Simulation

Nous avons effectué une première série de simulation pour vérifier le PESQ pour différentes valeurs du taux de codage et des niveaux de détection de silence variant de 0 à 800. Durant ces simulations, nous considérons que les liens physiques qui relient les nœuds offrent un taux d'erreur faible qui n'a pas de répercussions significatives sur la dégradation du PESQ. D'après la Figure 43, il est possible de constater la dégradation du PESQ lorsque le taux de codage diminue et le niveau de détection de silence augmente. Nous constatons aussi que la diminution du PESQ n'est pas proportionnelle à la diminution du taux de codage. En effet, la différence de PESQ entre le codage à 40 000 bit et à 32 000 bits n'est pas très importante par rapport à celle entre 32 000 bit/s et 24 000 bit/s. Nous remarquons aussi que plus le niveau de silence augmente, plus la variation du PESQ convergent pour les différentes valeurs du taux de codage. Sans surprise, il est possible de voir que la désactivation de la fonction de détection de silence ou sa fixation à une valeur proche de 0 donnent le même PESQ. Cependant, à partir de la Figure 44 il est possible de constater l'effet de l'activation de la détection du silence par rapport à sa désactivation. Lors d'une détection proche de 0, la bande passante consommée diminue d'environ 20 %. Un gain remarquable qui valorise l'utilisation de l'option de détection du silence par les codecs.

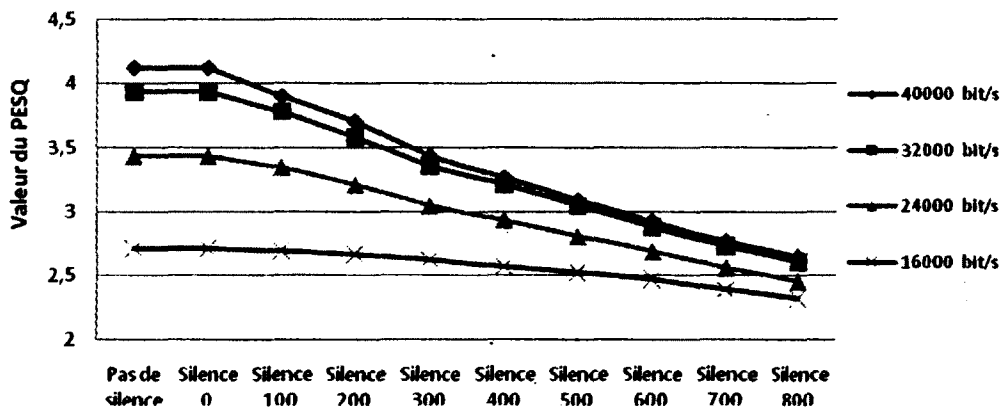


Figure 43 : Variation du PESQ selon le taux de codage et la détection de silence

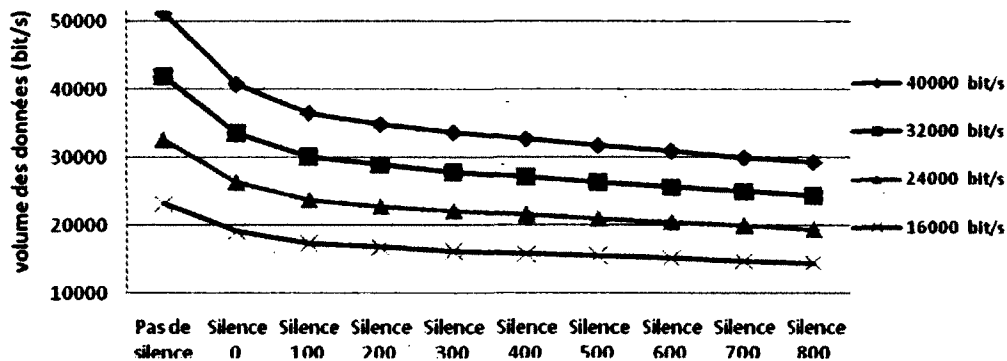


Figure 44 : Variation de la bande passante consommée selon le taux de codage et la détection de silence

La deuxième partie de simulation concerne l'évaluation de la dégradation de la qualité de la parole par rapport aux niveaux hiérarchiques dans l'arbre-média. Nous avons donc reproduit un arbre de distribution et de mixage audio et nous avons supposé que chaque nœud qui reçoit le flux le décode puis l'encode avant de l'envoyer aux nœuds du niveau suivant comme l'illustre la Figure 45.

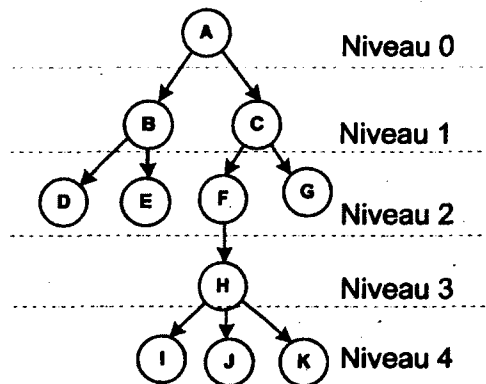


Figure 45 : Exemple d'un arbre de distribution média à différents niveaux

Quatre groupes de simulation ont été réalisés pour tester les résultats de chaque taux de codage. Dans chaque groupe, différents niveaux de détection de silence ont été utilisés. Ces taux de codage pour le codec G.726 affichent un comportement différent lorsque le niveau de l'arborescence dans l'arbre augmente. Par exemple, le PESQ du taux de codage à 40 000 se montre très sensible par rapport au niveau d'arborescence

lorsqu'il n'y a pas de suppression de silence. Le PESQ varie de 4.1 à 2.6 lorsque le niveau de l'arborescence varie de 0 à 9 (Figure 46). Contrairement, le taux de codage à 32 000 se stabilise à un PESQ de 3.9 indépendamment du niveau d'arborescence et sans l'utilisation de la détection de silence (Figure 47). Ce même constat est aussi valable pour les codeurs à 24 000 et 16 000 dont le PESQ se stabilise respectivement à 3.4 (Figure 48) et à 2.7 (Figure 49). Avec une augmentation du niveau de la détection du silence, le codeur à 40 000 affiche une diminution de la variation du PESQ par rapport aux différents niveaux d'arborescence. Alors que le codeur à 16 000 affiche une irrégularité dans les résultats du PESQ lorsque le niveau de détection de silence augmente. Cela peut nous amener à nous poser des questions sur l'efficacité du codec G.726 à ce taux de codage de 16 000 ou bien à l'objectivité du PESQ lorsque les valeurs sont au-dessous de 2.7.

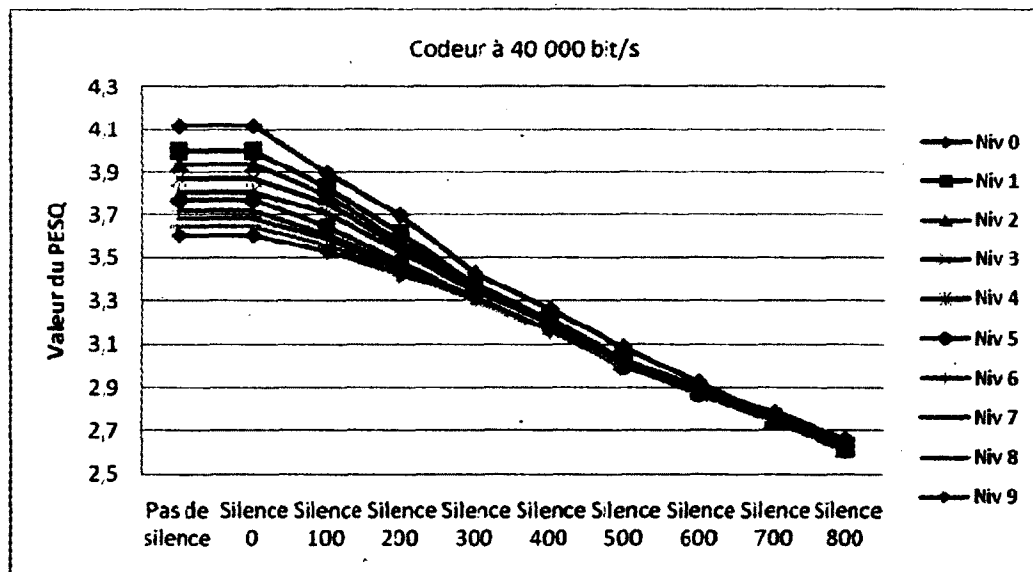


Figure 46 : Variation de la valeur du PESQ pour un taux de codage à 40 000 bit/s selon le détecteur de silence et le niveau dans l'arbre

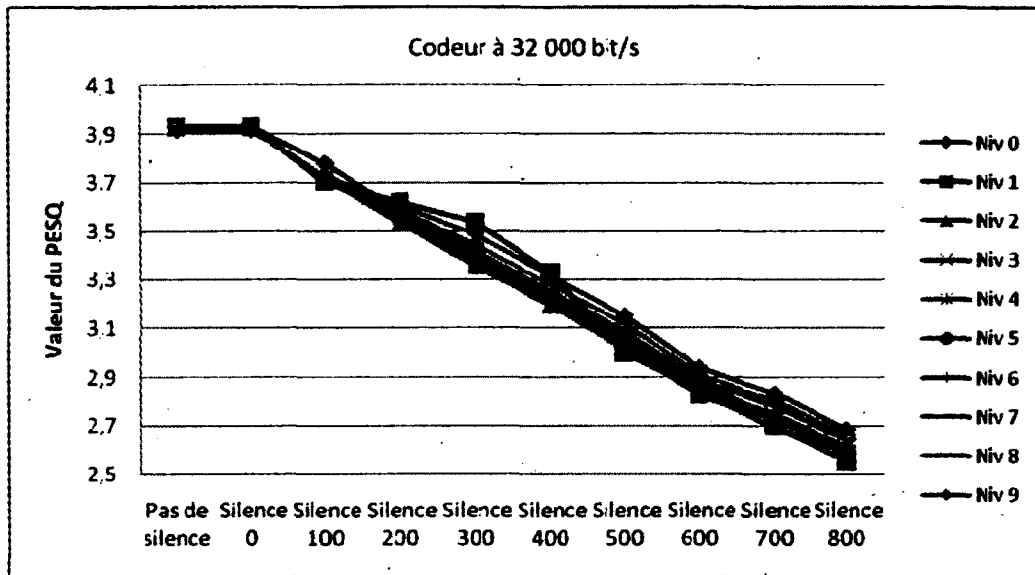


Figure 47 : Variation de la valeur du PESQ pour un taux de codage à 32 000 bit/s selon le détecteur de silence et le niveau dans l'arbre

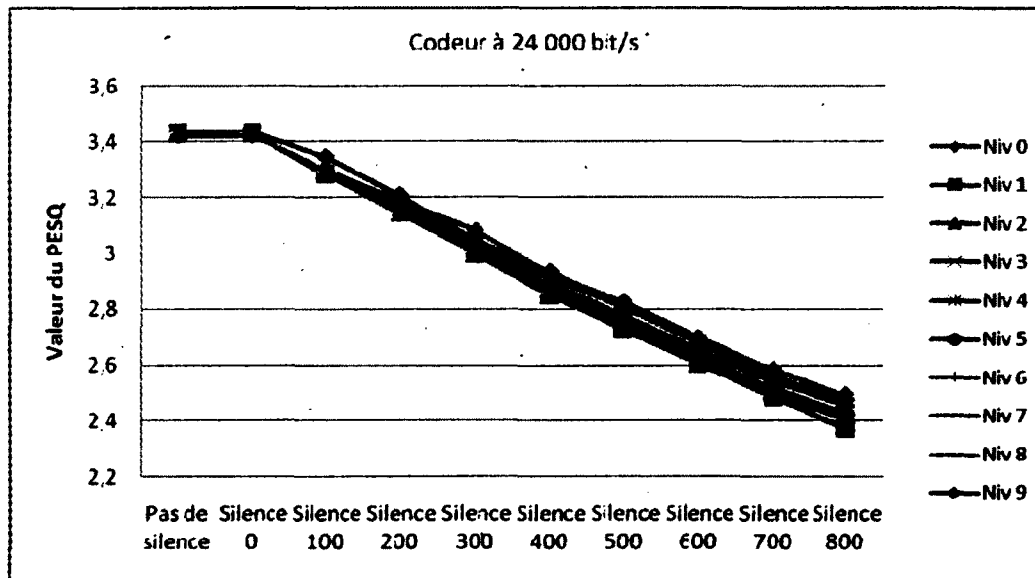


Figure 48 : Variation de la valeur du PESQ pour un taux de codage à 24 000 bit/s selon le détecteur de silence et le niveau dans l'arbre

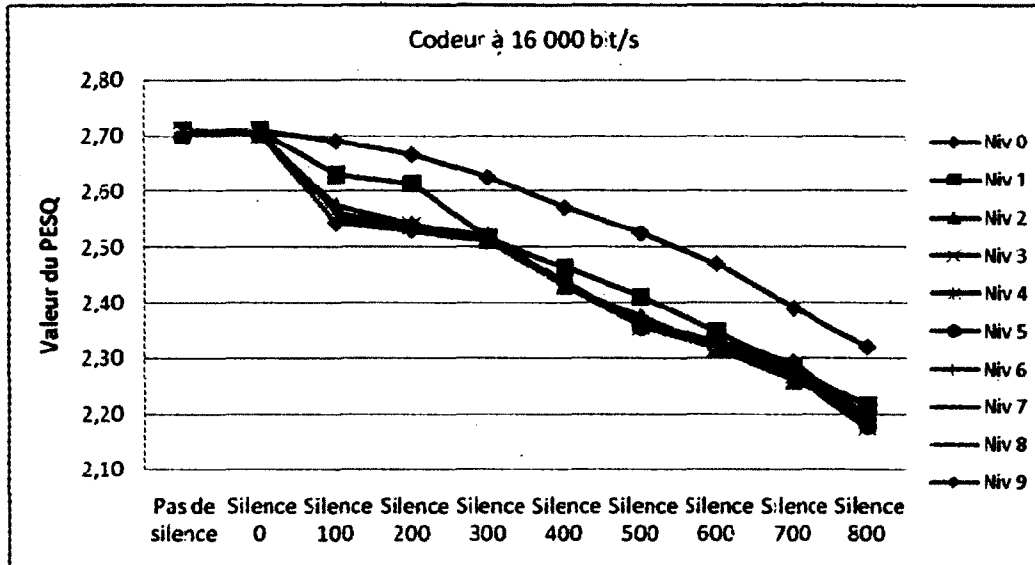


Figure 49 : Variation de la valeur du PESQ pour un taux de codage à 16 000 bit/s selon le détecteur de silence et le niveau dans l'arbre

#### 9.8.4 Conclusions des simulations

Les différents paramètres utilisés (niveau de silence et taux de codage) donnent différents résultats en termes de PESQ et de bande passante dépendamment du nombre de niveaux traversés. Si on se fixe comme objectif de garantir un PESQ supérieur à 3 pour tous les nœuds participants, dans ce cas, le codage à 16 000 ne devrait pas être utilisé puisqu'il donne un PESQ de 2.7 même dans des conditions idéales.

Si la topologie de la conférence est composée d'un nombre limité de niveaux d'arborescence (maximum de 3 niveaux), et si la bande passante n'est pas une contrainte, il est préférable d'opter pour un taux de codage à 40 000 bit/s tout en fixant le niveau de détection de silence à une valeur proche de 0. Cela permet d'obtenir un PESQ supérieur à 3.9 (Figure 46). Lorsqu'il y a plus que 3 niveaux d'arborescence, le codeur à 32 000 bit/s devient plus intéressant puisqu'il permet de conserver le PESQ dans une valeur proche de 3.9.

Si la contrainte principale de la conférence est de minimiser l'utilisation de la bande passante des participants, l'utilisation du taux de codage à 24 000 avec l'activation de la détection du silence pour des valeurs proches de 0 peut être une excellente solution. En effet, le codage à 24 000 avec un niveau de silence fixé à 100 fait gagner 35 % de la bande passante par rapport au codeur 40 000 et 21 % par rapport au codeur 32 000.

## 9.9 CONCLUSION

Ce travail a permis de présenter un protocole qui permet de supporter l'approche de distribution média découplée. Cette approche permet de séparer le réseau de mixage du réseau de distribution dans des modèles où la charge de traitement média est prise en charge par les nœuds pour permettre des conférences à large échelle. Durant ce travail, plusieurs besoins ont été identifiés en termes de gestion de la conférence et contrôle de l'adhésion des participants et de l'espace de dialogue. Le protocole proposé dans ce travail permet d'appliquer une approche centralisée pour construire un réseau média distribué entre les nœuds. Ce protocole peut être vu comme un outil qui permet à d'autres systèmes d'implémenter des conférences à large échelle où le réseau média est formé selon l'algorithme à appliquer. Ces algorithmes permettent une certaine optimisation en se basant sur des paramètres du réseau (bande passante, délais, gigue, taux d'erreur, etc.). Ils constituent un domaine de recherche distinct. Cependant, et puisque chaque opération d'encodage/décodage du signal audio peut occasionner une perte dans la qualité, nous avons évalué la dégradation de qualité audio en fonction des niveaux hiérarchiques de l'arbre. Cette évaluation, basée sur le codec G.726, montre que le PESQ peut rester au-dessus de 3.6 même après 10 opérations d'encodage/décodage. On peut rappeler ici que si chaque nœud supporte 3 nœuds-enfant dans l'étage suivant, il est possible de construire une conférence à près de 60 000 participants ( $3^{10}$ ). Le système proposé n'a pas abordé la problématique de l'optimisation du réseau média, cependant, une approche qui utilise un modèle couplé pour l'arbre de mixage et un modèle découplé pour l'arbre de distribution semble apporter de meilleurs résultats surtout qu'il permet de diminuer les

délais pour les locuteurs. En effet, cette contrainte de délais est nettement plus importante à considérer lorsqu'il s'agit d'un locuteur que pour le cas d'un auditeur. Un travail de recherche pourra être mené pour trouver une approche optimale dans le positionnement des participants dans l'arbre média.



# **10 CONCLUSIONS ET PERSPECTIVES**

## **10.1 CONCLUSION GÉNÉRALE ET CONTRIBUTIONS**

Ce travail a permis d'accroître notre connaissance et notre expertise au niveau des technologies de communication multimédia et principalement celles basées sur le protocole SIP. Les objectifs proposés en début de ce rapport sont focalisés sur deux aspects : La généralisation de l'utilisation de la mobilité de session et son déploiement dans les communications en mode-conférence. Deux cas ont été abordés dans la première partie de ce travail : lorsque la communication est en cours d'établissement et lorsqu'elle est déjà établie. Il a été possible de proposer un nouveau protocole qui permet la mobilité des sessions en cours d'établissement avec contrôle. Il s'agit d'un mécanisme qui se base sur le protocole SIP couplé à une approche en couche pour prendre en charge les différentes opérations à travers un codage en XML. Ainsi, il a été possible de supporter plusieurs opérations et garantir le contrôle de la session pour les cas d'appels entrants et d'appels sortants. En utilisant le protocole que nous proposons, il devient possible aux utilisateurs de contrôler à distance, en étant connectés à un réseau IP, tous les terminaux de la téléphonie IP qui sont à leur possession. Ce protocole a été aussi conçu pour permettre une grande ouverture pour de nouveaux services généralement proposés dans le domaine de la voix sur IP. Cette première contribution a été présentée dans [23]. Par la suite, et dans le cadre de l'élargissement du domaine d'application de la mobilité de session, le cas spécifique d'une communication en mode conférence complètement maillée a été étudié. Pour cela, nous avons introduit un nouveau protocole pour supporter des opérations de transfert et de récupération des sessions selon deux modes : avec et sans contrôle. Les nouveaux mécanismes proposés visent principalement à donner aux participants d'une conférence complètement maillée la possibilité d'utiliser les terminaux IP qui leur sont disponibles d'une manière complètement interchangeable. Ces deux modes de transfert proposés ont été évalués et leurs performances comparées. Il a été démontré que le mode avec perte de contrôle présente de meilleurs résultats lorsqu'il y a un

important trafic d'ajout et de suppression de participants au sein de la conférence. Ainsi, chaque participant peut choisir à l'avance le mode approprié avant de commencer le transfert dépendamment de la taille de la conférence et de l'activité au sein de cette conférence. Cette deuxième contribution a été présentée dans [24].

Les mécanismes de mobilité de sessions présentés et développés au cours de ce travail ont été exploités dans deux problématiques différentes. À travers la première problématique, il a été possible de réaliser deux approches de transition du modèle hermétiquement maillé vers le modèle complètement maillé par une approche de type implicite et une autre de type explicite. L'évaluation de ces deux approches montre que l'approche implicite devient plus intéressante en termes de volume des messages échangés lorsque le nombre de participants est supérieur à cinq. Les deux approches proposées visent à assurer la continuité de la conférence même si le focus doit partir ou lorsque la charge de traitement média devient très élevée pour être traitée par un seul participant. Dans cette contribution présentée dans [25], nous avons proposé une solution à cette limitation en effectuant la transition vers le modèle complètement maillé. Cependant, il aurait été possible d'envisager des solutions pour déléguer la fonction du *focus* à un autre participant, ou même garder la signalisation centralisée autour du *focus* et effectuer un balancement de la charge média. Cette deuxième alternative de balancement de la charge média semble très intéressante puisqu'elle permettrait la formation d'un groupe de conférence à large échelle. Ainsi, ce principe a été saisi et appliqué par l'utilisation de la mobilité des sessions dans des conférences à large échelle où le traitement média est basé exclusivement sur les nœuds participants. Dans cette contribution, nous avons identifié les composants de base pour contrôler une telle conférence multimédia et avons proposé une nouvelle approche qui permet de gérer le réseau de distribution et le réseau de mixage à travers un réseau séparé de contrôle et d'administration de la conférence. Nous avons aussi présenté un aperçu du principe d'implémentation de notre système dans un environnement SIP. Le protocole proposé supporte toutes les opérations d'ajout/départ des participants et permet aussi l'optimisation et la maintenance du réseau média dépendamment de l'algorithme à

utiliser. Cette contribution représente un ensemble de mécanismes pour implémenter les conférences Voix sur IP à très large échelle sans avoir recours à un serveur central de conférence. La solution proposée est présentée dans [26][27][28] peut être facilement utilisée par les systèmes existants et futurs pour réaliser leurs algorithmes d'optimisation de la distribution média. L'évaluation des performances du réseau média à travers la simulation d'une dizaine de niveaux arborescents de codage/décodage a montré que la qualité de parole perçue reste acceptable lors de l'utilisation du codec G.726 à 40 000 ou 32 000 bit/s par exemple.

## **10.2 LES LIMITATIONS DU TRAVAIL RÉALISÉ**

Ce travail de thèse nous a permis d'atteindre plusieurs défis. Cependant, il existe encore certains objectifs qui n'ont pas pu être satisfaits au complet dans le cadre de ce travail. Il est possible par exemple de citer les limitations suivantes :

1. L'application du protocole que nous proposons pour permettre la mobilité des sessions en cours d'établissement se limite au cas d'une communication qui met en jeu trois intervenants seulement (voir Figure 10). Les cas où le NM est en appel conférence nous semblent nettement plus complexe à réaliser. En effet, il est impératif d'enrichir le protocole proposé pour qu'il prenne en compte de nouveaux événements relatifs à l'activité au sein de la conférence par exemple.
2. Dans la section 6.6, nous proposons des mécanismes qui permettent la récupération des sessions transférées lors d'une mobilité en cours de communication avec perte de contrôle. Ces mécanismes ne sont valables que lorsque le nouveau terminal qui gère la session transférée garde cette session localement et ne la transfère pas à son tour sur un autre terminal. Pour les cas où la session est successivement transférée vers d'autres terminaux, il devient impossible avec le mécanisme que nous proposons d'assurer l'opération de récupération par le terminal qui a initialement décidé de transférer sa session.
3. Lors de la mise en place d'une conférence à large échelle basée sur le mode P2P, nous avons développé un simulateur sous l'environnement Omnet pour évaluer la

qualité audio lorsqu'elle traverse différents niveaux hiérarchiques de l'arbre de distribution média. Cependant, dans cette simulation, nous avons considéré que tous les liens réseaux sont équivalents et présentent des taux d'erreur et des débits identiques. Réellement, ces suppositions ne sont pas valables et plusieurs paramètres liés au réseau public d'Internet peuvent influencer la qualité audio. Il serait donc plus objectif d'implémenter le système et de le tester à travers internet lorsque les participants sont géographiquement dispersés.

### **10.3 PERSPECTIVES DE RECHERCHES**

#### **10.3.1 Étendre la mobilité des sessions en cours d'établissement**

Lors de l'étude de la problématique de la mobilité pour le cas spécifique où la communication est en cours d'établissement, il a été possible de proposer une solution de collaboration complète entre les nœuds lors du transfert de la session. Cependant, cette solution se limite pour le cas de deux intervenants. Il serait intéressant d'étendre la solution pour supporter la collaboration de plusieurs intervenants. Dans ce cas, plusieurs utilisateurs peuvent se connecter sur un même téléphone IP par exemple, et la décision pourra être prise par un des intervenants. Cette solution sera certainement utile si le téléphone IP de la compagnie est géré par plusieurs intervenants qui peuvent agir d'une façon concurrente sur le même appel. Il est important d'identifier les différents scénarios ainsi que les conflits possibles. Il faut aussi proposer des mécanismes pour prévenir et résoudre ces possibles conflits lors du transfert de session dans un tel environnement de collaboration multiutilisateur.

#### **10.3.2 Permettre la récupération des sessions transférées**

Les solutions apportées jusque-là pour la mobilité des sessions ne permettent pas toujours à un NM de suivre l'évolution des sessions qu'il a transférées dans le mode de transfert avec perte de contrôle. En effet, si on prend le cas d'un utilisateur muni d'un terminal A qui transfère sa session vers un terminal B qui transfère la session vers un

autre terminal C. Les solutions actuelles ne permettent pas au terminal A de pouvoir récupérer sa session. Il est important de permettre un meilleur contrôle des sessions transférées et de rendre possible leur récupération de n'importe quel terminal et à tout moment. Pour cela, nous pensons qu'il est possible de concevoir un système qui permet d'administrer et de suivre l'évolution de la session transférée. La solution préconisée se base sur l'intégration d'un système de présence, utilisé habituellement pour gérer la présence des personnes, et de l'adapter pour gérer la présence des sessions SIP. Ce même système de présence devrait pouvoir gérer les règles d'accès et permettre à l'utilisateur SIP, qui a initialement transféré sa session, de pouvoir la récupérer sans ambiguïté quels que soient les scénarios qui ont eu lieu après le transfert initial.

### **10.3.3 Adaptabilité du service selon les capacités du nouveau terminal lors de la mobilité des sessions**

Il est important lors du transfert de service d'adapter la communication aux caractéristiques du nouveau terminal et de son réseau. Même s'il existe une large gamme de services SIP qui utilisent des protocoles comme RTP et RTCP permettant de prendre en compte de telle adaptation, le service de la mobilité des sessions ainsi que certains autres services SIP qui fonctionnent selon le principe de la notification d'événements ont besoin d'adaptation. Dans ce cas, il est important, par exemple, de pouvoir adapter la quantité de données échangées et le taux des mises à jour selon les capacités du nouveau terminal. Il s'agit ici d'effectuer un certain filtrage de l'information dans le but de l'optimiser pour l'adapter aux capacités logicielles et matérielles du terminal receveur. Dans ce même souci d'optimisation de l'information échangée, il est possible d'introduire une notification partielle lorsqu'une notification complète crée une redondance de l'information chez le receveur. Ces informations échangées sont généralement présentées dans un format compatible avec la technologie XML et il est possible de proposer des solutions basées sur des modèles conceptuels comme celui du modèle MVC (Modèle/Vue/Contrôleur) pour pouvoir obtenir différentes vues d'un même modèle. Le contrôleur dans une telle architecture pourra être des scripts XSL, envoyés préalablement par le receveur, pour effectuer des transformations XSLT sur les

données XML sources du modèle (les informations à transférer). Les scripts XSL peuvent être paramétrables à partir des caractéristiques du terminal, mais il est possible de penser à construire des scripts XSL dynamiques à la demande du terminal selon leurs besoins. Pour cela, des techniques comme le *Policy-Based Management* peuvent être utilisées pour créer intuitivement des scripts de contrôle pour construire une information filtrée et adaptée. Une telle architecture, basée sur le modèle MVC, et qui met en œuvre des transformations XSLT pour produire différentes vues d'un même modèle de donnée, a été traitée dans [29]. Il existe actuellement des formats XML selon le modèle CC/PP (*Composite Capabilities/Preferences Profile*) qui décrit chaque type de terminal mobile. Ces informations pourraient être jumelées aux préférences de l'utilisateur afin de produire un format XSL applicable sur les données échangées.

#### **10.3.4 Personnalisation de l'environnement du service lors de la mobilité des sessions**

Le service de la mobilité de session devrait pouvoir garantir la personnalisation de l'environnement du service lorsqu'un même utilisateur procède à un transfert de session de son premier terminal vers son deuxième terminal. Même s'il s'agit de deux adresses SIP différentes, ces deux adresses représentent la même personne et la session transférée représente une seule communication. Il est donc important de pouvoir transférer l'environnement du service, selon les préférences de l'utilisateur, afin de garder un environnement de communication homogène. Pour le service de la messagerie instantanée, cela revient à garder par exemple la même couleur du texte envoyé et la même photo affichée aux destinataires. Pour réaliser un tel objectif, il est possible de suivre une approche « cross-layer » appliquée sur les couches supérieures du modèle OSI afin de rattacher, à la session transférée, les informations relatives à l'application. Il faut donc identifier et décrire l'environnement de la communication dans un format qui pourrait s'associer à chaque session transférée. Une extension du format SDP (*Session Description Protocol*) pourrait être suggérée afin d'encapsuler les informations à rattacher.

# 11 BIBLIOGRAPHIE

- [1] "3GPP: TS 23.228: IP Multimedia Subsystem (IMS) (Stage 2), Release 5", 2002.
- [2] Mark Allman, "AN EVALUATION OF XML-RPC," *ACM SIGMETRICS Performance Evaluation Review*, vol. 30, pp. 2-11, March, 2003
- [3] Allman M., "On the generation and use of TCP acknowledgments," *ACM Computer Communication Review*, Vol. 28, pp. 4-21, Oct. 1998.
- [4] Anerousis, N., "TOPS: An Architecture for Telephony over Packet Networks ", *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 1, 1999.
- [5] Anind K. Dey. "Providing Architectural support for Building Context-Aware Applications". Ph.D thesis. College of Computing, Georgia Institute of Technology, 2000.
- [6] Berger, S., Schulzrinne, H., Sidiroglou, S., Wu, X. "Ubiquitous Computing Using SIP," *International Workshop on Network and Operating System Support for Digital Audio and Video*, pp 82-89, 2003
- [7] Berners-Lee, T., Fielding, R., and Nielsen. H., "Hypertext Transfer Protocol - HTTP/1.0", IETF RFC 1945, 1996.
- [8] Bharambe, A., Padmanabhan, V., and Seshan, S., "Supporting Spectators in Online Multiplayer Games," *Proc. Third ACM Workshop Hot Topics in Networks (HotNets '04)*, 2004
- [9] Box. D., "Essential COM", Addison-Wesley Publishing Company, 1998.
- [10] Brown, P.J., Bovey, J.D., Xian, C., "Context-aware applications : from the laboratory of the marketplace", *IEEE Personal Communications*, 1997.
- [11] Bill N. Schilit. A., "Context-Aware System Architecture for Mobile Distributer Computing", Ph.D Thesis. Columbia University, 1995.
- [12] Byers, J.W., Considine, J., Mitzenmacher, M. and Rost, S., "Informed Content Delivery Across Adaptive Overlay Networks," *IEEE/ACM Trans. Networking*, vol. 12, no. 5, pp. 767-780, Oct. 2004.
- [13] Campbell, Ed. B., Rosenberg, J., Schulzrinne, H., Huitema, C., Gurle, D., "Session Initiation Protocol (SIP) Extension for Instant Messaging", IETF RFC 3428, 2002
- [14] CAMEL : 3GPP TS 22.078 v5.5.0. "Customised Applications for Mobile network Enhanced Logic (CAMEL)". Service description, Stage 1. December, 2001
- [15] Castro, M., P. Druschel, A.-M. Kermarrec, and A. Rowstron, "SCRIBE: A Large-Scale and Decentralized Application-Level Multicast Infrastructure" *Proc. ACM SIGCOMM '01*, Aug. 2001.
- [16] Castro, M., Druschel, P., Kermarrec, A.-M., Nandi, A., Rowstron, A. and Singh, A. "SplitStream: High-Bandwidth Multicast in Cooperative Environments," *Proc. 19th ACM Symp. Operating Systems Principles (SOSP '03)*, Oct. 2003.

- [17] Chan, J. Landfeldt, B., Liu, R., and Seneviratne, R., "A home-proxy based wireless Internet framework in supporting mobility and roaming of real-time services", *IEICE Transactions on Communications, Special Issue on Mobile Multimedia Communications E84-B(4)*, 2001.
- [18] Chu, Y., Rao, S. G., Seshan, S., and Zhang, H., "A Case for End System Multicast", *Measurement and Modeling of Computer Systems*, 2000.
- [19] Donovan, S., "The SIP INFO Method", *IETF RFC 2976*, 2000.
- [20] Day, M., Aggarwal, S., Mohr, G., Vincent, J., "Instant Messaging - Presence Protocol Requirements", *IETF RFC 2779*, 2000.
- [21] Day, M., Rosenberg, J., Sugano, H., "A Model for Presence and Instant Messaging", *IETF RFC 2778 - 2000*.
- [22] Di Stefano A., Santoro, C., "NetChaser: Agent support for personal mobility", *IEEE Internet Computing, Vol. 4, No. 2*, pp. 74-79, 2000.
- [23] Elleuch, W., Houle, A.C., Guénette, S., "Introducing a New XML-Based Protocol for Sip User-Agent Services Collaboration: Integration with IP-Phone and PC", *IEEE-Candienne, Ottawa*, 2006.
- [24] Elleuch, W., Houle, A.C., Guénette, S., "Enable Session Mobility in Full-Mesh Conferencing Model", *IEEE-WiMob, New-York*, 2007.
- [25] Elleuch, W., Houle, A.C., "Transition within SIP Based Multi-Party Communication: From Tightly-Coupled To Fully-Coupled Conference", *IEEE-Candienne, Niagara-Falls*, 2008.
- [26] Elleuch, W., Houle, A.C., "Multiparty Voice over IP (MVoIP) Peer-based System for Large-scale Conference Support", *IEEE-WiMob, Avignon*, 2008.
- [27] Elleuch, W., Houle, A.C., "SIP-based Protocol for P2P Large-scale Multiparty VoIP (MVoIP) Conference Support", *IEEE-CCNC, Las Vegas*, 2009.
- [28] Elleuch, W., Houle, A.C., "Speech Quality Evaluation for Decentralized Media Distribution within Large-Scale Multiparty VoIP Conference", *SETIT, Sousse*, 2011 (accepted).
- [29] Elleuch, W., Mabilieu, P., "Un outil d'aide à l'évaluation des connaissances à distance", *thèse de maîtrise, Université de Sherbrooke*, 2004.
- [30] Emmerich, W., "Engineering Distributed Objects". John Wiley & Sons, 2000.
- [31] Enns, R., "NETCONF Configuration protocol", *draftietf-netconf-prot-11*, 2006.
- [32] «eXtensible Markup Language (XML) », adresse web : [www.w3.org/XML](http://www.w3.org/XML)
- [33] Eysers, T., Schulzrinne, H.: Predicting Internet Telephony Call Setup Delay, *First IP Telephony Workshop, Berlin, Allemagne*, Apr. 2000
- [34] FFmpeg Project, "AVCODEC: the leading audio/videocodec library," Project web-page: <http://ffmpeg.mplayerhq.hu> [visité : 30/01/2010].



- [35] Fielding, R., Gettys, J., Mogul, J. C., Frystyk, H., and Berners-Lee, T., "Hypertext Transfer Protocol - HTTP/1.1" 1997. IETF RFC 2068.
- [36] Freed, N., Borenstein, N., "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", IETF RFC 2045, 1996.
- [37] Gu, X., Wen, Z., Yu, P. S., and Shae, Z. "peerTalk: A Peer-to-Peer Multiparty Voice-over-IP System". *IEEE Trans. Parallel Distrib. Syst.* 19, 4, 2008.
- [38] Gupta, M., Judge, P., Ammar, M. "A reputation system for peer-to-peer networks," *International Workshop on Network and Operating System Support for Digital Audio and Video*, pp. 144-152, 2004.
- [39] Gurbani, V. Faynberg, I., Lu, H.L., Brusilovsky, A. Unmehopa, M., Vemuri, K., and Gato, J., "The SPIRITS (Services in PSTN requesting Internet Services) Protocol". IETF Internet-Draft, Mars 2004.
- [40] Handley, M., Jacobson, V., "SDP - Session Description Protocol", IETF RFC 2327, 1998.
- [41] Handley, M., Schulzrinne, H., Schooler, E., Rosenberg, J., "SIP: Session Initiation Protocol", IETF RFC 2543, 1999.
- [42] Herman C.R., Di-Fa C., Yih-Farn C. and Ming-Feng C., "iMobile, a proxy-based platform for mobile services". In *Wireless Mobile Internet*, pages 3-10, 2001.
- [43] iVisit : Real Time Multi-party video conferencing and collaboration tools for PC and Mac, [www.ivisit.com](http://www.ivisit.com), 2008
- [44] Jennings, C., Mahy, R., Roach, A. B., "Relay Extensions for the Message Sessions Relay Protocol (MSRP)", IETF Draft - version 10, 2006.
- [45] Johnston, A., Levin, O., "Session Initiation Protocol (SIP) Call Control - Conferencing for User Agents", IETF RFC 4579, 2006
- [46] Jung, E., Park, Y.J., Park, C., "Mobile agent network for support personal mobility", *International Conference on Information Network (ICOIN)*, 1998.
- [47] Koskelainen, P., Schulzrinne, H., Wu, X., "A SIP-based Conference Control Framework", *International Workshop on Network and Operating System Support for Digital Audio and Video*, pp. 53-61, 2002.
- [48] Kreger, H., "Web Services Conceptual Architecture", Technical Report, IBM, WCSA 1.0, 2001.
- [49] Lennox, J., Schulzrinne, H., "A Protocol for Reliable Decentralized Conferencing", *International Workshop on Network and Operating System Support for Digital Audio and Video*, 2003.
- [50] Lennox, J., and Schulzrinne, H., "Call processing language framework and requirements," IETF Draft, 2000.
- [51] Lennox, J., and Schulzrinne, H. "CPL: a language for user control of internet telephony services", IETF Draft, 1999.

- [52] Mahy, R., Biggs, B., and R. Dean, "The Session Initiation Protocol (SIP) - Replaces Header", IETF RFC 3891, September 2004
- [53] Maltz, D.A. and Bhagwat, P. « MSOCKS: An architecture for transport layer mobility », INFOCOM, 1998.
- [54] Maniatis, P., Roussopoulos, M., Swierk, E., Lai, K., Ppenzeller, G., Zhao, X., Baker, M., "The mobile people architecture", Mobile Computing and Communications Review, Vol. 17, No. 1, p.p 36-42, 1999.
- [55] Mari Korkea-aho, "Context-Aware Applications Survey", Adresse Web : [users.tkk.fi/~mkorkeaa/doc/context-aware.html#chap2](http://users.tkk.fi/~mkorkeaa/doc/context-aware.html#chap2), 2000.
- [56] MExE : 3G TS 23.057, "3rd Generation Partnership Project ; Technical Specification Group Terminals; Mobile Station Application Execution Environment (MExE) ; Functional description ; Stage 2 (3G TS 23.057 version 3.0.0)".
- [57] Mike Macedonia and Don Bruzman entitled "MBONE, the Multicast Backbone" IEEE Computer, 1994
- [58] "OSA : Services and System Aspects, Service Aspects, Stage1 : Service Requirement for the Open Service Access (OSA)". Technical Report TS 22.127 v4.1.0, 3rd Generation Partnership Project, Mars 2001.
- [59] "Omnet++ 3.3," page web du projet : <http://www.omnetpp.org> [visité : 30/01/2010].
- [60] Papp, G. and GauthierDickey, C. "A location aware P2P voice communication protocol for networked virtual environments", In Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology, Japon, 2009.
- [61] Perkins, C.E., "Mobile networking through mobile IP", IEEE Internet Computing, Vol. 2, No. 1, pp. 58-69, 1998.
- [62] Petrack, S., Conroy, L., "The PINT Service Protocol, Extensions to SIP and SDP for IP Access to Telephone Call Services". IETF RFC 2848, 2000.
- [63] POC Collaborators, "OMA (Open Mobile Alliance) –Push to Talk Over Cellular", Technical section, 2007.
- [64] Radenkovic, M., GreenHalgh, C., "Multi-party Distributed Audio Service with TCP Fairness", Proc. of ACM Multimedia 2002, Juan-les-Pins, France, Décembre 2002.
- [65] Roach, A.B., "Session Initiation Protocol (SIP)-Specific Event Notification", IETF RFC 3265, June 2002.
- [66] Rosenberg, J., "A Framework for conferencing with the Session Initiation Protocol (SIP)", IETF RFC 4353, 2006
- [67] Rosenberg, J., "A Presence Event Package for the Session Initiation Protocol (SIP)", IETF RFC 3856, 2004

- [68] Rosenberg, J., "An Extensible Markup Language (XML) Based Format for Watcher Information", IETF RFC 3858, 2004
- [69] Rosenberg, J., "A Watcher Information Event Template-Package for the Session Initiation Protocol (SIP)", IETF RFC 3857, 2004.
- [70] Rosenberg, J., "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)", IETF RFC 4825, 2007
- [71] Rosenberg, J., Lennox, J., and Schulzrinne, H., "Programming internet telephony services", IEEE Network, Vol. 13, pp. 42-49, 1999.
- [72] Rosenberg, J., Peterson, J., Schulzrinne, H., Camarillo, G., "Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)", IETF RFC 3725, April 2004.
- [73] Rosenberg, J., Schulzrinne, H., "An Offer/Answer Model with the Session Description Protocol (SDP)", IETF RFC 2543, 2002.
- [74] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Sparks, R., Handley, A., and E. Schooler, "SIP: Session Initiation Protocol", IETF RFC 3261, 2002.
- [75] Rosenberg, J., Schulzrinne, H., Mahy, R., "An INVITE Initiated Dialog Event Package for the Session Initiation Protocol (SIP)", IETF RFC 4235, November 2005.
- [76] Schulzrinne, H., Casner, S., Frederl, R., Jacobson, V., "RTP: A Transport Protocol for Real-Time Applications", IETF RFC 3550, 2003.
- [77] Schulzrinne, H., Wedlund, E., "Application-Layer Mobility Using SIP", Mobile Computing and communications Review, Volume 1, Numéro 2, 2000.
- [78] Schulzrinne, H., "Personal mobility for multimedia services in the Internet", in European Workshop on Interactive Distributed Multimedia Systems and Services (IDMS), 1996.
- [79] Shacham, R., Schulzrinne, H., Thakolsri, S., Kellerer, W., "Session Initiation Protocol (SIP) Session Mobility", IETF Draft, February 2006.
- [80] Sparks, R., "The Session Initiation Protocol (SIP) Refer Method", IETF RFC 3515, 2003.
- [81] Sparks, R., "The Session Initiation Protocol (SIP) Referred-By Mechanism", IETF RFC 3892, 2004.
- [82] "Simple Object Access Protocol (SOAP)", adresse web [www.w3.org/TR/soap](http://www.w3.org/TR/soap).
- [83] Singh, K., Nair, G., Schulzrinne, H., "Centralized Conferencing using SIP", Dept. of Computer Science/Columbia University, New York, 2001
- [84] T. Finin, . Labrou, and Yun Peng. "Mobile Agents Can Benefit from Standards Efforts on Interagent Communication", IEEE Communication Magazine, 367 : 50-56, 1998

- [85] Telecommunication Standardization Sector of ITU (ITU-T), "Recommendation P.862: Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Feb. 2001.
- [86] Thai, B., Wan, R., and Seneviratne, A., Rakotoarivelo, T., "Integrated Personal Mobility Architecture: A Complete Personal Mobility Solution", Mobile Networks and Applications, 2003
- [87] Thanh, D., "Using mobile agent paradigm in mobile communications", in: Ericsson Conference Software Engineering (1999).
- [88] The Ninja Project, page web : [ninja.cs.berkeley.edu](http://ninja.cs.berkeley.edu) [visité : 01/05/2008]
- [89] The Skype Internet Telephony System, [www.skype.com](http://www.skype.com), 2009
- [90] TSG-SA Working Group, "Universal Mobile Telecommunications System (UMTS) - Service aspects, Virtual Home Environment (VHE)", Technical report UMTS 22.70 version 3.0.0, 1999
- [91] V. A. Pham and A. Karmouch. "Mobile software agents : an overview", IEEE Communication Magazine, 36 : 26-37, 1998
- [92] Voice Quality Testing of PESQ, site web : [opticom.de/technology/pesq.html](http://opticom.de/technology/pesq.html) [visité : 15/01/2010]
- [93] Wang, H.J., Raman, B., Biswas, R., Chuah, C.N., Gummadi, R., Hohlt, B., Hong, X., Kiciman, E., Mao, Z., Shih, J.S., Subramanian, L., Zhao, B. Y., Joseph, A.D., Katz, R.H., "ICEBERG: An Internet-core network architecture for integrated communications", IEEE Personal Communications, 2000
- [94] WAP Forum : "Wireless Application Protocol : Wireless Markup Language Specification", WAP Forum, 1998.
- [95] Watanabe K., "Distributed Multiparty Desktop Conferencing System: MERMAID", Proceedings of the Conference on Computer-Supported Cooperative Work, 1990
- [96] Webex : Cisco Web Meeting and Collaboration Solutions, [www.webex.com](http://www.webex.com), 2008
- [97] Wong, S. H. Y., Lui, J. C. S., "An Architectural Infrastructure and Topological Optimization for End System Multicast," mascots, p. 0481, 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'02), 2002
- [98] World Wide Web Consortium, "Simple object access protocol (soap) 1.1" site web : [www.w3.org/TR/SOAP/](http://www.w3.org/TR/SOAP/). [visité : 15/01/2006]
- [99] "XML-RPC Home page", adresse web : [www.xmlrpc.com](http://www.xmlrpc.com) [visité : 15/01/2006]