

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOULOU D MAMMERI, TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE



Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

Master Académique en Electronique
Option : Télécommunication et Réseaux

Thème

Conception d'une application VoIP
PC vers PC

Proposé et Dirigé par :

Mr. M. LAHDIR

Présenté par :

Mr. IBOUCHICHENE Mourad
Mr. DAF Karim

Soutenu le : / /2010

Promotion 2010



Nous tenons à remercier nos parents, notre encadreur Mr.LAHDIR Mourad pour ses conseils, son orientation et son aide le long de notre projet de fin d'études.

Nous remercions également, Mr.TALEB Hakim pour son aide précieuse.

Nos remerciements à tous nos enseignants et à toute personne ayant contribué de près ou de loin à la réalisation de ce projet.



Tab. II.1 Comparaison entre protocoles (H.323 ET SIP)	35
Tab. II.2 Classe numéro Délai par sens de commentaires	46
Tab. III.1 Débit des codeurs	52
Tab. III.2 Cout CPU des codeurs (ms)	52

Fig. I.1 Le modèle TCP/IP et le modèle OSI.....	3
Fig. I.2 La couche Application	4
Fig. I.3 La couche Transport.....	4
Fig. I.4 La couche Internet	4
Fig. I.5 La couche Réseau	5
Fig. I.6 Entête UDP	8
Fig. I.7 Entête TCP	9
Fig. I.8 Entête IP	10
Fig. I.9 Entête ARP	11
Fig. II.1 Le réseau RTC publique	14
Fig. II.2 Schéma globale du RTC	15
Fig. II.3 Principe de la VoIP.....	16
Fig. II.4 Mise en paquet de l'information.....	17
Fig. II.5 Le scénario PC 2 PC.....	19
Fig. II.6 Le scénario PC 2 Phone	20
Fig. II.7 Le scénario Phone 2 Phone	20
Fig. II.8 La voix sur IP dans l'entreprise.....	21
Fig. II.9 Architecture d'une passerelle VoIP.....	22
Fig. II.10 Bloc traitement de la voix	23
Fig. II.11 Pile du protocole H.323	28
Fig. II.13 SIP dans le modèle OSI	29
Fig. II.14 Exemple de communication SIP.....	30
Fig. II.15 Entête RTP	38
Fig. II.16 Entête RTCP.....	40
Fig. III.1 Processus d'échantillonnage	46
Fig. III.2 Système de compression élémentaire	48
Fig. III.3 Principe du DPCM	49
Fig. III.4 Principe de LPC	50
Fig. IV.1 Architecture générale de l'application	58
Fig. IV.2 Modèle du dialogue	60
Fig. IV.3 Diagramme de cas d'utilisation d'un client	62
Fig. IV.4 Paquetage de haut niveau de cas d'utilisation vue Système.....	65
Fig. IV.5 Diagramme de cas d'utilisation du paquetage Session RTP	66

Table des figures

Fig. IV.6 Diagramme de cas d'utilisation du paquetage Connexion	69
Fig. IV.7 Diagramme d'états UAC lors de la connexion au Serveur	73
Fig. IV.8 Diagramme d'états de UAC et UAS lors de l'établissement d'un appel	74
Fig. IV.9 Diagramme de classes	79

Sommaire

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

CHAPITRE I : GENERALITES SUR LES RESEAUX IP

I.1 Préambule	3
I.2 Description du modèle	3
I.2.1 Modèle TCP/IP	3
I.2.2 Les protocoles de la couche application.....	5
I.2.3 Les protocoles de la couche transport	6
I.2.4 Les protocoles de la couche internet.....	9
I.3 Discussion.....	11

CHAPITRE II : ETATS DE L'ART DE LA VOIP

II.1Préambule.....	12
II.2 Définition de la voix sur IP (VoIP)	13
II.2.1 Le réseau classique RTC	14
II.2.1.1 Structure générale de réseau RTC.....	15
II.2.1.2 Etablissement d'une liaison sur le réseau RTC	16
II.3 Principe de la VoIP	16
II.3.1 L'architecture VoIP	18
II.3.1.1 Ordinateur à ordinateur (PC 2 PC)	18
II.3.1.2 Ordinateur à téléphone (PC2 Phone)	19
II.3.1.3 Téléphone à téléphone (Phone 2 Phone)	20
II.3.2 Principe architectural d'une passerelle VoIP	22
II.4 Standards VoIP.....	24
II.4.1 Protocole H.323	24
II.4.1.1 Présentation	24
II.4.1.2 Fonctionnement	25
II.4.2 Protocole SIP	28
II.4.2.1 Présentation	28
II.4.2.2 Fonctionnement	30
II.4.2.3 Sécurité et authentification	33
II.5 comparaison entre les Standards H.323 et SIP.....	34
II.6 Transport RTP et RTCP.....	35
II.6.1 Fonction de RTP	36
II.4.2 Fonction de RTCP	38
II.7 Avantages de la VoIP.....	40
II.8 Problème et qualité de service (QoS)	41
II.8.1 Latence	41
II.8.2 Perte de paquets	42
II.8.3 Gigue	43

Sommaire

II.9 Etat du marché	44
II.10 Discussion	44

CHAPITRE III : INTRODUCTION AU CODAGE DE LA PAROLE

III.1 Préambule	45
III.2 Généralité sur le codage de la parole	46
III.2.1 Système de compression élémentaire	46
III.2.2 classification des codeurs	48
1. Codeurs différentiels	49
2. Codeurs par synthèses	49
3. Codeurs entropiques	51
III.2.3 Evaluation des codeurs	51
III.3 Qualité des codeurs	53
III.4 Présentation du codeur G.711	54
III.5 Présentation du codeur ILBC	54
III.6 Présentation du codeur SPEEX	55
III.7 Discussion	56

CHAPITRE IV : CONCEPTION

IV.1 Préambule	57
IV.2 Choix de la méthode de conception	57
IV.3 Architecture générale de l'application	57
IV.4 Etude conceptuelle	61
IV.4.1 Spécification des besoins par des cas d'utilisations	61
IV.4.2 Spécification des besoins par des états transitions	72
IV.4.3 Identification des classes	75
IV.4.4 Identification et documentation des relations entre classes	76
IV.4.5 Diagramme de classes	78
IV.4.6 Déroulement de l'application	80
IV.5 Discussion	81
Conclusion	82
Bibliographie	83

L'évolution de la téléphonie ne se limite pas au téléphone fixe ou au système GSM. Cependant, elle s'intéresse également à l'informatique. Cette évolution intervient grâce aux nouvelles technologies permettant des débits de plus en plus élevés.

De nos jours, la voix sur IP (VoIP) occupe une place privilégiée dans le monde des télécommunications. L'avantage incontesté de cette technologie est sa possibilité d'intégrer la voix, la vidéo et les données sur une même infrastructure Internet existante. Grâce à cette technologie, les coûts des communications interurbaines ont chuté de manière considérable ce qui laisse croire qu'elle a encore de beaux jours devant elle.

Les nouvelles technologies qui se développent pour les différents aspects pratiques permettent aussi de multiples applications à la voix et à l'image en temps réels. Dans ce contexte, notre travail consiste à concevoir une application VoIP ordinateur à ordinateur (PC vers PC).

Différents standards et protocoles de communications ont été élaborés pour rendre possible cette communication numérique. Parmi ces protocoles, ceux de signalisation SIP et H.323 ainsi que les protocoles de transport en temps réels RTP et RTCP. Que nous avons étudiés et comparés dans le présent travail. Nous avons également modélisé, en utilisant l'approche par objet, le protocole SIP. Sa mise en œuvre a été réalisée selon une architecture client-serveur à l'aide des bibliothèques multimédias du langage de programmation Java pour transmettre et recevoir la voix et la vidéo via Internet. De plus, nous avons introduit le formalisme UML (Unified Modeling Language) qui est une méthode de conception orientée objet très utilisée dans le domaine de l'informatique.

Afin de réaliser notre travail, le manuscrit est organisé comme suit :

Le chapitre I est consacré à des notions générales sur les réseaux IP, où nous donnerons un aperçu sur le modèle TCP/ IP dont nous avons identifié les différentes couches et protocoles les constituant.

Le chapitre II présente l'état de l'art de la VoIP, en rappelant le principe du RTC (Réseau Téléphonique Commuté), définir le principe de la voix sur IP, ses avantages et ses

problèmes puis la définition des protocoles de transport en temps réels et standards de signalisations associés à la voix IP (SIP, H.323).

Le chapitre III met l'accent sur un aspect important dans le domaine de la VoIP qui est le codage de la parole, où nous présenterons les codeurs utilisés dans la conception.

Le chapitre IV abordera la partie la plus importante dans notre travail, qui est la conception d'une application VoIP avec la signalisation de session SIP et les protocoles de transport en temps réels RTP /RTCP, utilisant le formalisme de conception informatique UML.

Enfin, nous terminons notre mémoire par une conclusion et des perspectives de continuité.

I.1 Préambule

Dans les années 1970, le département de la Défense américain, ou DOD (Department Of Defense), décide, devant le foisonnement de machines utilisant des protocoles de communication différents et incompatibles, de définir sa propre architecture. Cette architecture, dite TCP/IP, est à la source du réseau Internet. Elle est aussi adoptée par de nombreux réseaux privés, appelés intranet.

Les deux principaux protocoles définis dans cette architecture sont les suivants :

- IP (Internet Protocol), de niveau réseau, qui assure un service sans connexion.
- TCP (Transmission Control Protocol), de niveau transport, qui fournit un service fiable avec connexion.

I.2 Description du modèle [1]

I.2.1 Le modèle TCP/IP

Le modèle TCP/IP peut en effet être décrit comme une architecture réseau à 4 couches :

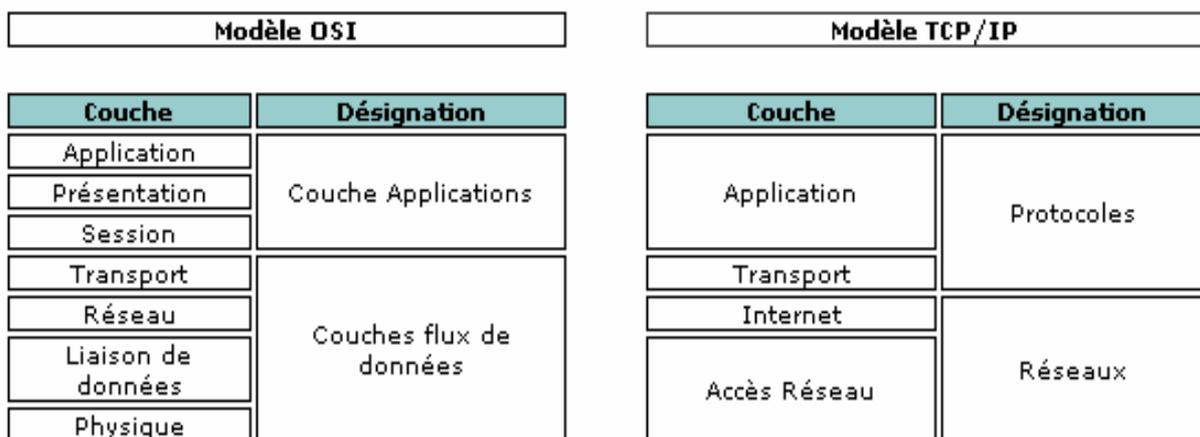


Figure II.1 Le modèle TCP/IP et le modèle OSI

Le modèle OSI (Open Systems Interconnexion) a été mis à côté pour faciliter la comparaison entre les deux modèles. Il y a 4 couches principales dans l'environnement TCP/IP :

Ø **La couche application** les applications interagissent avec les protocoles de la couche transport pour envoyer ou recevoir des données.

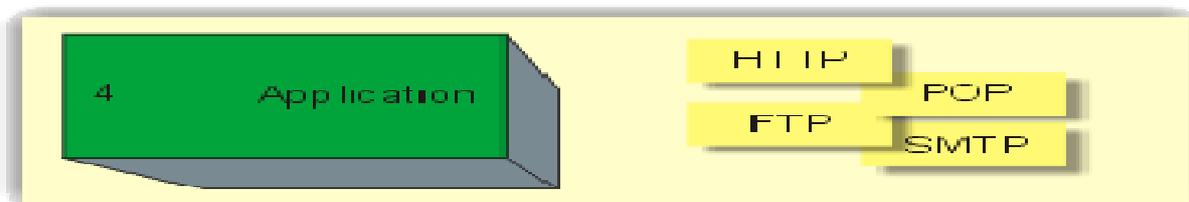


Figure I.2 La couche application

Ø **La couche transport** chargée de fournir un moyen de communication de bout en bout entre deux programmes d'application. Agit en mode connecté et en mode non connecté. Elle divise le flux de données venant des applications en paquets, transmis avec l'adresse de destination IP au niveau IP.



Figure I.3 La couche transport

Ø **La couche Internet** : encapsule les paquets reçus de la couche transport dans des datagrammes IP. Agit en mode non connecté et non fiable.

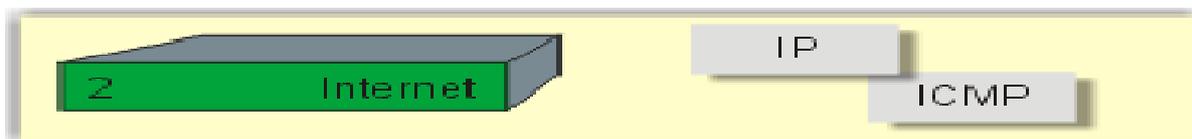


Figure I.4 la couche internet

Ø **La couche hôte réseau** assure la transmission d'un datagramme venant de la couche IP en l'encapsulant dans une trame physique et en transmettant cette dernière sur un réseau physique.

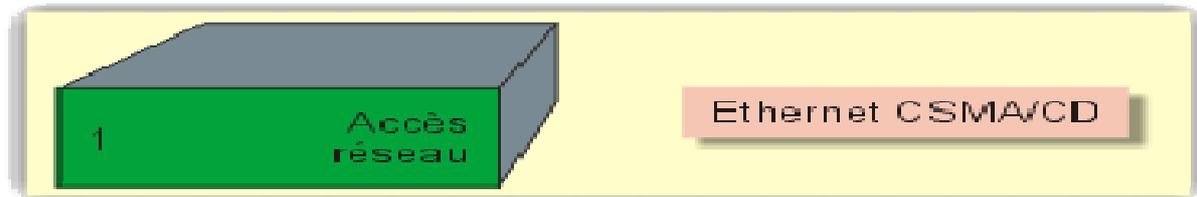


Figure I.5 La couche réseau

I.2.2 Les protocoles de la couche application

Les protocoles d'application sont des protocoles de haut niveau, adaptés aux besoins d'applications spécifiques. Ils s'appuient sur UDP et TCP pour permettre le transfert d'informations entre une application serveur et ses applications clients.

Ø Le protocole HTTP (Hyper Text Transfert Protocol)

Ce protocole est utilisé pour la navigation web entre un serveur HTTP et un butineur. Le protocole assure qu'un client comme : Internet Explorer ou Netscape peut envoyer des requêtes et recevoir les réponses de serveurs HTTP sans problèmes particuliers.

Ø Le protocole FTP (File Transfert Protocol)

Protocole qui permet d'assurer le transfert de fichiers de façon indépendante des spécificités des OS (Operating System). Ainsi, un client FTP sous Windows peut télécharger un fichier depuis un serveur UNIX.

Ø Le protocole SMTP (Simple Mail Transfert Protocol)

Le protocole qui permet d'acheminer le courrier depuis le serveur SMTP de l'émetteur, jusqu'au serveur SMTP du destinataire, qui le classe dans les boîtes aux lettres de ces clients.

Ø Le protocole POP3 (Post Office Protocol version 3)

Le protocole qui permet au client de relever à distance le courrier classé dans sa boîte aux lettres.

ØLe protocole TELNET (Tele Network)

C'est le "couteau suisse " du travail à distance. En fait, un client TELNET est une console en mode texte, capable de se connecter sur la plupart des serveurs, comme POP3 ou SMTP. Il devient alors possible d'envoyer et de lire des messages, si l'on connaît les commandes inhérentes aux protocoles SMTP et POP3.

Un serveur TELNET permet cependant des choses bien plus puissantes puisqu'il devient possible de prendre à distance le contrôle d'un hôte. C'est un outil qui permet l'administration à distance d'une machine, du moment que l'on est capable d'ouvrir une session et d'acquérir les droits de "super utilisateur".

I.2.3 Les protocoles de la couche transport [2], [3]

ØLe protocole UDP (User Datagram Protocol)

Le protocole UDP est basé en couche 4 de modèle OSI. Il n'ouvre pas de session et n'effectue pas de contrôle d'erreur. Il est alors appelé "mode non connecté". Il est donc peu fiable, cependant, il permet aux applications d'accéder directement à un service de transmission de datagrammes rapides

UDP est utilisé pour transmettre de faibles quantités de données où le coût de la création de connexions et du maintien de transmissions fiables s'avèrent supérieurs aux données à émettre. UDP peut également être utilisé pour les applications satisfaisant à un modèle de type

"interrogation réponse". La réponse étant utilisée comme un accusé de réception à l'interrogation. On y trouve classiquement SNMP et DNS. UDP est aussi utilisé dans un second cas, tel que la voix sur IP.

L'envoi en temps réel est primordial, donc si une trame n'arrivait pas, la retransmission serait inutile. Chaque machine contient un ensemble de points de destination abstraits appelés protocole ports, identifiés par un entier positif codé sur deux octets. Une application qui souhaite communiquer sur le réseau avec une autre application doit se raccorder à un port. Une application est donc identifiée sur le réseau par :

- L'adresse IP de la station sur laquelle elle se trouve.
- Le protocole TCP ou UDP.
- Le port number auquel elle s'est raccordée.

Cette connexion logique entre deux ports est appelée : **Socket**. UDP est un protocole de transport utilisant directement IP ce qui entraîne qu'il offre un service de transport :

- Non fiable (sans acquittement).
- Sans connexion.
- Sans contrôle de flux.

C'est aux applications de prendre en charge l'acquittement, la connexion et la remise dans l'ordre des messages. Voici la structure de l'entête UDP basé sur 8 octets.

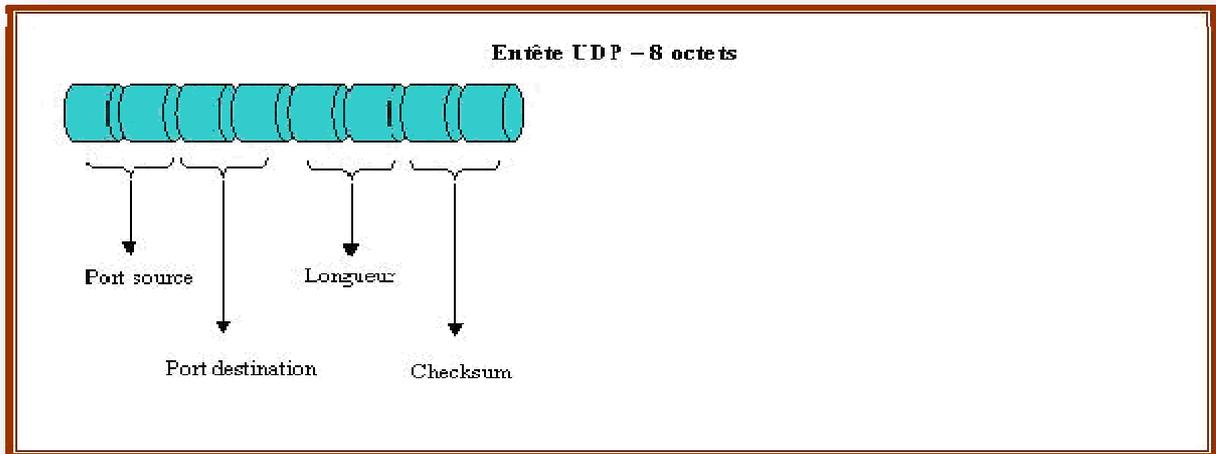


Figure I.6 L'entête UDP

§ Le protocole TCP (Transfert Control Protocol)

Le protocole TCP est basé en couche 4 de modèle OSI. Il ouvre une session et effectue lui-même le contrôle d'erreur. Il est alors appelé "mode connecté".

TCP fournit un service :

- Fiable (canal sans erreurs).
- Avec contrôle de flux.
- Ordonné.
- En mode full duplex.
- En mode connecté.

TCP tout comme UDP utilise la notion de port excepté que TCP utilise la connexion comme abstraction de port. Une connexion est identifiée par une paire de « End points » : Host (@IP d'une station) et Port (port TCP). Voici la structure de l'entête TCP basé sur 20 octets.

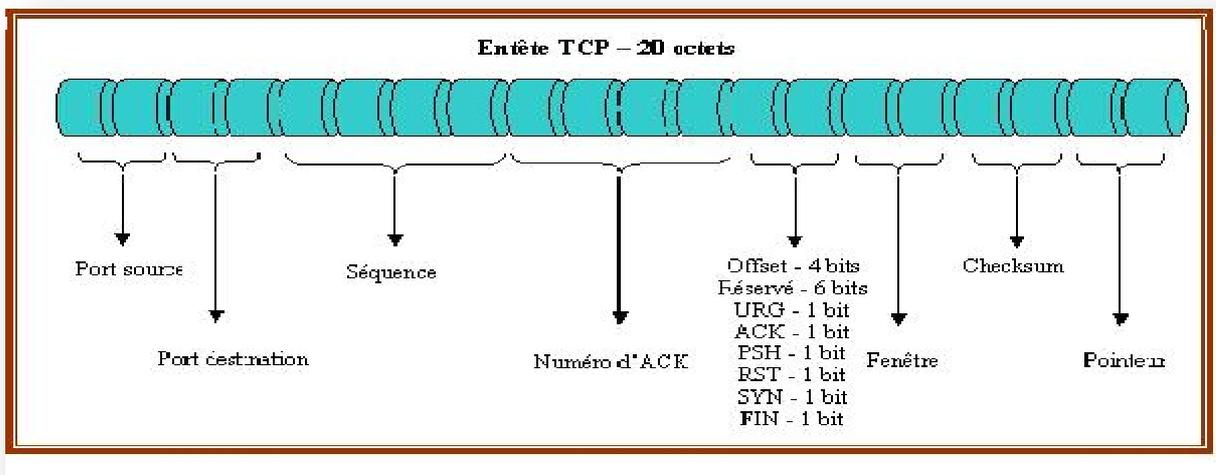


Figure I.7 L'entête TCP

I.2.4 Les protocoles de la couche internet

- **Le protocole IP (Internet Protocol)**

IP signifie "Internet Protocol", protocole Internet. Il représente le protocole réseau le plus répandu. Il permet de découper l'information à transmettre en paquets, de les adresser, de les transporter indépendamment les uns des autres et de recomposer le message initial à l'arrivée. Ce protocole utilise ainsi une technique dite de commutation de paquets.

Au niveau IP, les données des utilisateurs ou des applications sont encapsulées à l'intérieur d'unités de transfert appelées datagrammes IP. Le protocole IP fournit un service d'acheminement des datagrammes IP sans connexion et non fiable.

Un datagramme se compose d'un entête et des données. Avant transmission sur un réseau physique, le datagramme IP est encapsulé dans une trame physique. Voici la structure de l'entête IP basé sur 20 octets.

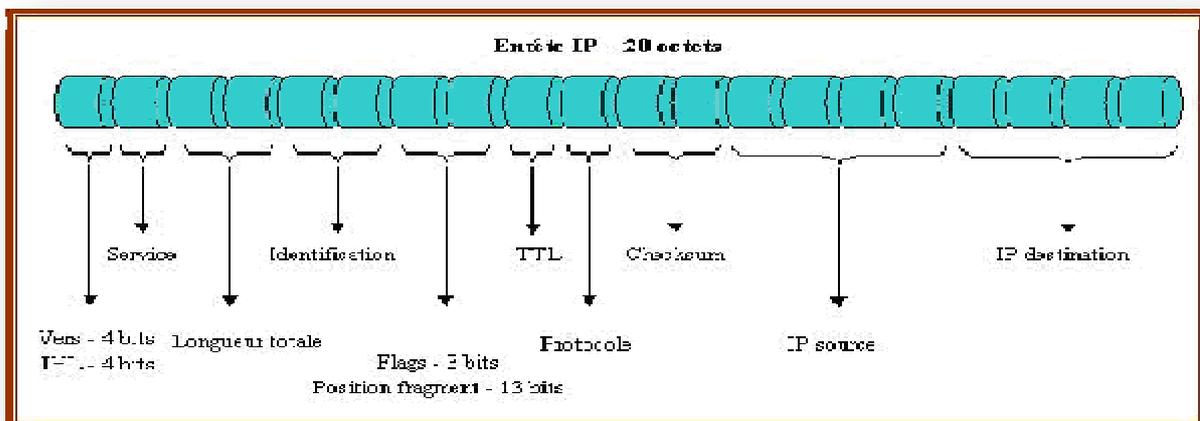


Figure I.8 l'entête IP

- **Le protocole ARP (Address Resolution Protocol)**

Le protocole ARP (Address Resolution Protocol) fonctionne en couche Internet du modèle TCP/IP correspondant à la couche 3 du modèle OSI. L'objectif d'un ARP est de permettre de résoudre une adresse physique par l'intermédiaire de l'adresse IP correspondante d'un invité distant.

Le protocole ARP apporte un mécanisme de « translation » pour résoudre ce besoin. Il permet d'obtenir l'adresse physique (MAC, niveau 2) d'une machine connaissant son adresse IP (logique, niveau 3). Voici l'entête du protocole ARP dans le cadre spécifique d'IP sur Ethernet.

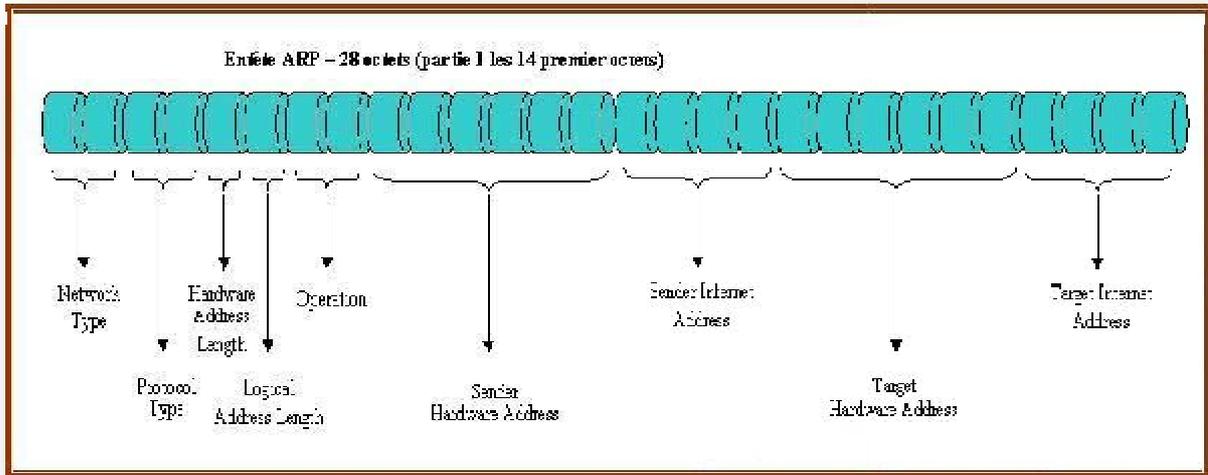


Figure I.9 L'entête ARP

I.3 Discussion

Dans ce chapitre, on a présenté, de manière générale, la description du modèle TCP/IP dont on a expliqué les différentes couches le constituant, en donnant l'aperçu sur les différents protocoles. Il vise simplement de faire comprendre les fondements des réseaux IP.

II.1 Préambule

La voix sur IP (Voice over IP) est une technologie de communication vocale en pleine émergence. En effet, la convergence du triple play (voix, données et vidéo) fait partie des enjeux principaux des acteurs de la télécommunication aujourd'hui.

Plus récemment l'Internet s'est étendu partiellement dans l'Intranet de chaque organisation, voyant le trafic total basé sur un transport réseau de paquets IP surpasser le trafic traditionnel du réseau voix (réseau à commutation de circuits). Il devenait clair que dans le sillage de cette avancée technologique, les opérateurs, entreprises ou organisations et fournisseurs devaient, pour bénéficier de l'avantage du transport unique IP, introduire de nouveaux services voix et vidéo.

Ce fût en 1996 que la première version voix sur IP appelée H323 a été mise en œuvre. Issu de l'organisation de standardisation européenne ITU-T sur la base de la signalisation voix RNIS (Q931), ce standard a maintenant donné suite à de nombreuses évolutions, quelques nouveaux standards prenant d'autres orientations technologiques.

Au lieu de disposer à la fois d'un réseau informatique et d'un réseau téléphonique commuté (RTC), l'entreprise peut donc, grâce à la VoIP, tous fusionner sur un même réseau.

Les nouvelles capacités des réseaux à haut débit devraient permettre de transférer de manière fiable des données en temps réel. Ainsi, les applications de vidéo ou audioconférence ou de téléphonie vont envahir le monde IP qui, jusqu'alors, ne pouvait raisonnablement pas supporter ce genre d'applications (temps de réponse important, gigue - jitter, Cos - Qos...). Jusque vers le milieu des années 90, les organismes de normalisation ont tenté de transmettre les données de manière toujours plus efficace sur des réseaux conçus pour la téléphonie.

Il semblerait que les entreprises après avoir émis un certain nombre de doutes sur la qualité de services soient désormais convaincues de la plus grande maturité technologique des solutions proposées sur le marché. Qu'il s'agisse d'entreprises mono -site ou multi- sites, les sondages montrent que le phénomène de migration vers les systèmes de téléphonie sur IP en entreprise est actuellement engagé.

Les premières technologies de VoIP imaginées étaient propriétaires et donc très différentes les unes des autres. Pourtant, un système qui est censé mettre des gens et des systèmes en relation exige une certaine dose de standardisation. C'est pourquoi sont apparus des protocoles standards, comme le H323 ou le SIP.

II.2 Définition de la voix sur IP (VoIP)

Voice Over Internet Protocol (VoIP) est le principe consistant à faire passer des communications téléphoniques numérisées dans des paquets IP, circulant dans un réseau IP à moindre coût et avec une qualité de transmission et une fiabilité comparable à celles du réseau téléphonique classique RTC. C'est le cas idéal d'une transmission sur un réseau IP unique ou une série de réseaux IP reliés entre eux. Cependant, dans la majorité des cas une conversation téléphonique devra passer également dans un ou plusieurs réseaux d'accès de technologies différentes. D'où la difficulté d'interconnecter des réseaux de technologies différentes tout en assurant une fiabilité et une qualité de transmission aussi bonne que dans le réseau RTC traditionnel.

Comme toute innovation technologique qui se respecte, la VoIP doit non seulement simplifier le travail, mais aussi faire économiser de l'argent. Les entreprises dépensent énormément en communication téléphonique ; or le prix des communications de la ToIP (Téléphonie sur IP) est dérisoire en comparaison. La ToIP exploite un réseau de données IP pour offrir des communications vocales à l'ensemble de l'entreprise sur un réseau unique voix et données.

La téléphonie sur IP est aujourd'hui basée sur les passerelles (Gateway) qui relient le monde de la téléphonie traditionnelle et le monde IP. Une passerelle est connectée d'un côté au réseau téléphonique classique, et de l'autre, au réseau Internet.

La passerelle se charge de la conversion du signal téléphonique en paquets IP et vice-versa. Ces deux opérations sont effectuées en même temps pour permettre une connexion duplex, mais avant d'étudier cette nouvelle manière de « téléphoner », il est important de comprendre l'infrastructure fondamentale qui est utilisée pour faire un appel téléphonique « traditionnel » sur le réseau RTC.

II.2.1 Le réseau classique RTC [4]

Le RTC est tout simplement le réseau téléphonique que nous utilisons dans notre vie de tous les jours et qui nous donne accès à de multiples fonctions. En effet, outre le fait de pouvoir téléphoner, le RTC nous permet d'utiliser de multiples services tels que la transmission et réception de fax, l'utilisation d'un minitel, accéder à Internet, etc. Il représente donc l'un des protocoles de discussion utilisé sur la paire de cuivre boucle locale.

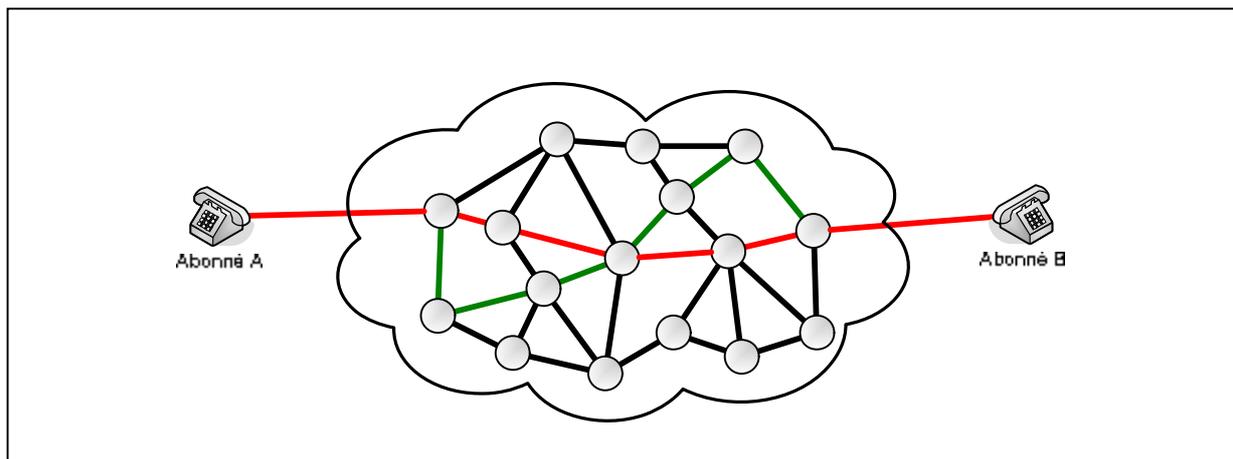


Figure II.1 Le réseau RTC Public

II.2.1.1 Structure générale du réseau RTC

La figure II.2 présente le schéma global du réseau RTC :

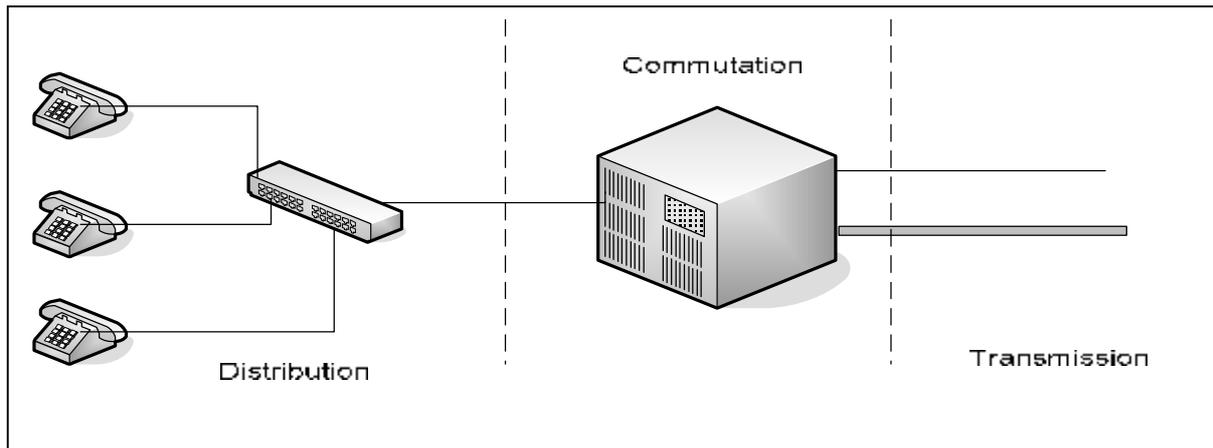


Figure II.2 Schéma global du RTC

- **La distribution** : celle-ci comprend essentiellement la liaison d'abonné ou boucle locale qui relie l'installation de l'abonné au commutateur le plus proche (le commutateur de rattachement) avec une paire de fils de cuivre. Cette ligne assure la transmission de la voix (fréquence vocale de 300 à 3400 Hz). [4]
- **La commutation** : c'est la fonction essentielle du réseau, elle consiste à mettre en relation deux abonnés, maintenir la liaison pendant tout l'échange et libérer les ressources à la fin de celui-ci. C'est le réseau qui détermine les paramètres de taxation et impute le coût de la communication à l'appelant.
- **La transmission** : la liaison de l'ensemble des commutateurs (réseau de transmission ou réseau de transport). Cette fonction est remplie soit par un système filaire cuivre (en voie de disparition), de la fibre optique ou des faisceaux hertziens. Aujourd'hui, le réseau est pratiquement intégralement numérisé, seule la liaison d'abonné reste analogique.

II.2.1.2 Etablissement d'une liaison sur le réseau RTC

Dès le décrochage du combiné téléphonique, c'est une chaîne de mécanisme qui se met en route pour établir une connexion, sans même qu'on s'en rende compte, à l'intérieur du système téléphonique. Ces activités sont déclenchées pour chaque action de l'utilisateur de la manière suivante :

- L'utilisateur décroche le combiné : Lorsque l'on décroche le combiné (off Hooke), le central RTC le détecte aussitôt, c'est la prise de ligne, et après vérification de la suffisance des ressources disponibles dans le commutateur de l'abonné, le central acquitte la demande en donnant la tonalité qui est une invitation à numéroté (dial tone).
- L'utilisateur forme un numéro : Après la numérotation complète, le commutateur de l'abonné émet une requête (demande de connexion) qui est transmise au moyen d'un réseau par paquet utilisé pour le contrôle des connexions. Ce processus est appelé signalisation, il utilise le protocole Common Channel Signaling System No.7 (SS7).
- L'utilisateur attend la réponse de l'usagé : Si des ressources point à point sont disponibles, l'utilisateur devrait entendre une tonalité indicative qui notifie l'activation de la sonnerie téléphonique du destinataire. S'il y a encombrement (congestion) dans le réseau, l'utilisateur entendra un signal rapide.

II.3 Le principe de la VoIP [3]

La figure II.3 montre les différentes étapes d'une communication utilisant la technologie VoIP :

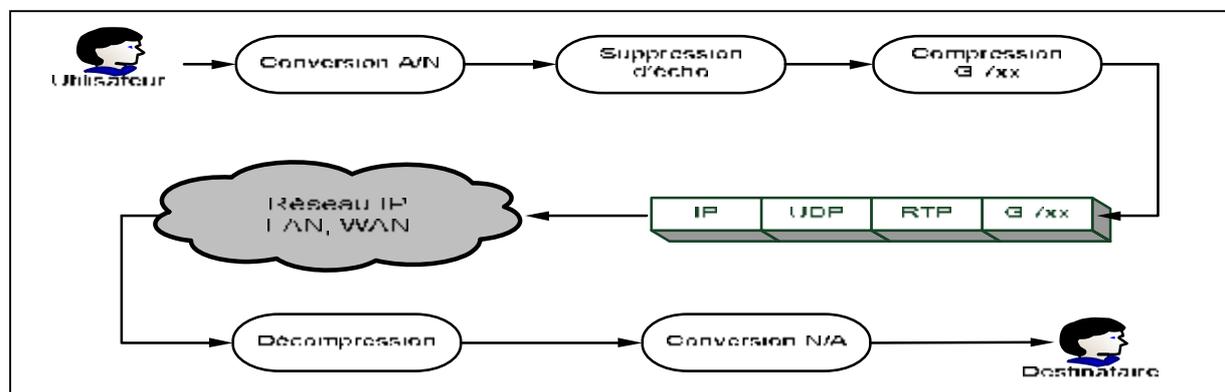


Figure II.3 Principe de la VoIP

- **Conversion A/N** : à l'aide d'un transducteur, la voix humaine génère un signal électrique analogique. Ce dernier est converti en une série de bits (trames) par un convertisseur Analogique/Numérique (A/N).
- **Suppression d'écho** : suppression de l'écho sur les signaux échantillonnés suivant le standard G.165 ou G.168 de l'ITU-T.
- **Compression G.7xx** : les trames de bits sont soumises à de multiples processus de compression. Il y a plusieurs techniques pour réaliser la compression les plus utilisées sont celles standardisées dans l'ITU-T G-séries. Les codecs les plus utilisés dans les systèmes VoIP sont : G.711 PCM, G.726 ADPCM, G.728 LD-CELP et G.729/G.729a CS-ACELP. Les codecs PCM et ADPCM appartiennent à une catégorie de codeurs audio nommés codeurs de la forme d'onde, pour notre étude nous allons utiliser le codec PCM G.711. Après compression, nous obtenons des trames avec un débit réduit.
- **Mise en paquet et transmission** : on encapsule ces trames dans des paquets d'abord avec le protocole RTP pour reconstruire la base de temps des flux à l'arrivée, ensuite grâce aux protocoles TCP ou UDP pour être acheminés vers le réseau du destinataire, via Internet grâce au protocole IP (figure II.4).

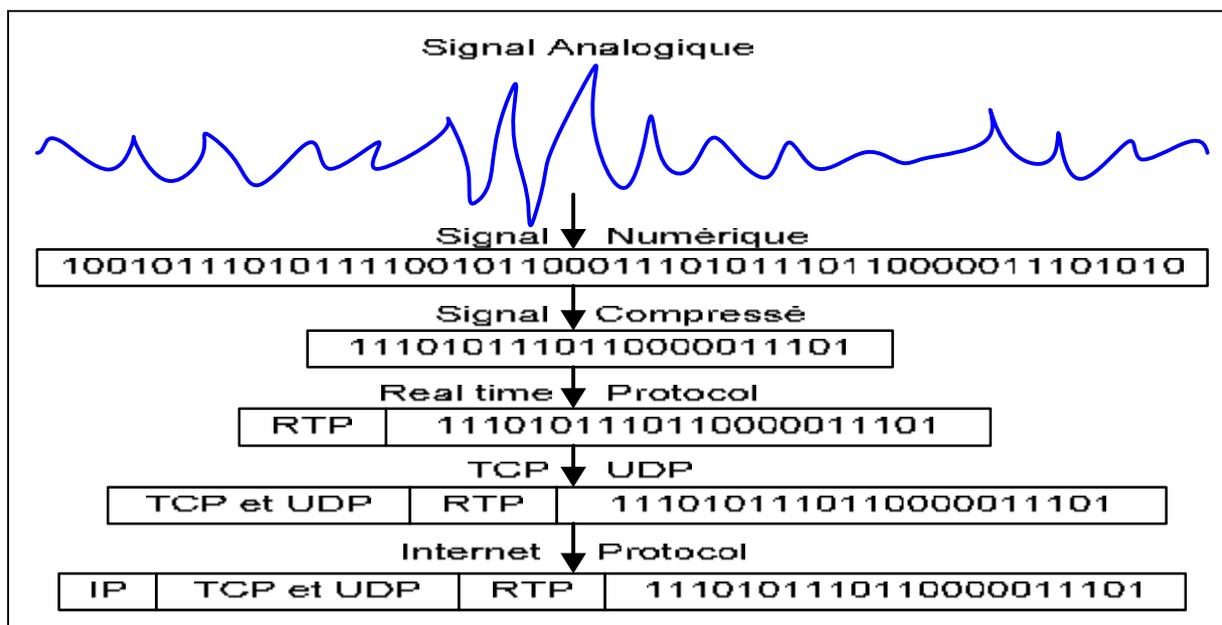


Figure II.4 Mise en paquet de l'information

- **Décompression** : après l'arrivée des trames compressées, elles sont d'abord synchronisées, ensuite décompressées pour essayer de restituer les trames d'origine soit par un logiciel VoIP ou un matériel VoIP spécial.
- **Conversion N/A** : un convertisseur N/A est utilisé pour rendre les trames de bits sous forme analogique afin de restituer le signal émis par l'utilisateur pour être écouté par le destinataire.

III.3.1 L'architecture VoIP [5]

Trois scénarios sont possibles pour établir une liaison téléphonique basée sur IP :

III.3.1.1 Ordinateur à ordinateur (PC 2 PC)

Dans ce scénario, les deux correspondants utilisent un PC rattaché au réseau Internet par l'intermédiaire d'un fournisseur d'accès Internet FAI. Cette technique nécessite des participants à la communication d'avoir un PC muni d'un modem, d'une carte réseau, d'un microphone, d'un haut-parleur et d'un logiciel de téléphonie IP compatible de chaque côté.

La voix est comprimée et décomprimée par un logiciel de compression. Ce mode de fonctionnement nécessitait auparavant que les correspondants se fixent un rendez-vous préalable sur Internet ou soient connectés en permanence.

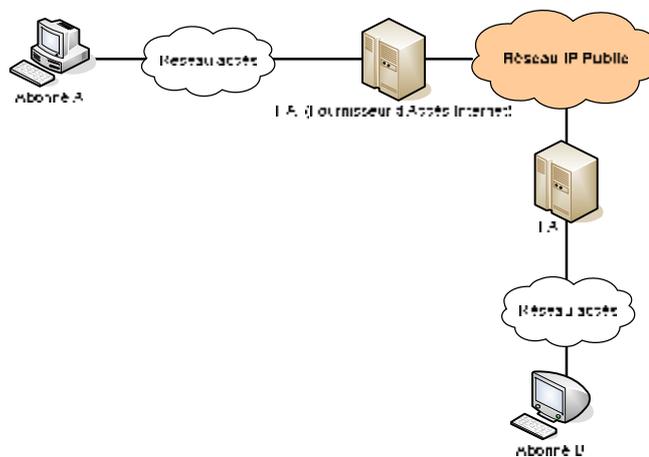


Figure II.5 Le scénario PC 2 PC

II.3.1.2 Ordinateur à téléphone (PC 2 Phone)

Dans ce scénario, l'un des correspondants utilise un PC rattaché au réseau Internet par un fournisseur d'accès Internet, l'autre correspondant utilise un téléphone rattaché au réseau téléphonique commuté.

Une passerelle est nécessaire entre les deux réseaux pour rendre possible cette technique et faire la conversion entre réseaux (dans ce cas, elle fait la conversion Internet - RTC et vice versa).

Elle se charge également de l'appel du correspondant et de l'ensemble de la signalisation relative à la communication téléphonique du côté du correspondant demandé.

Du côté PC, une signalisation d'appels est nécessaire pour établir une communication et négocier les paramètres de communication multimédia.

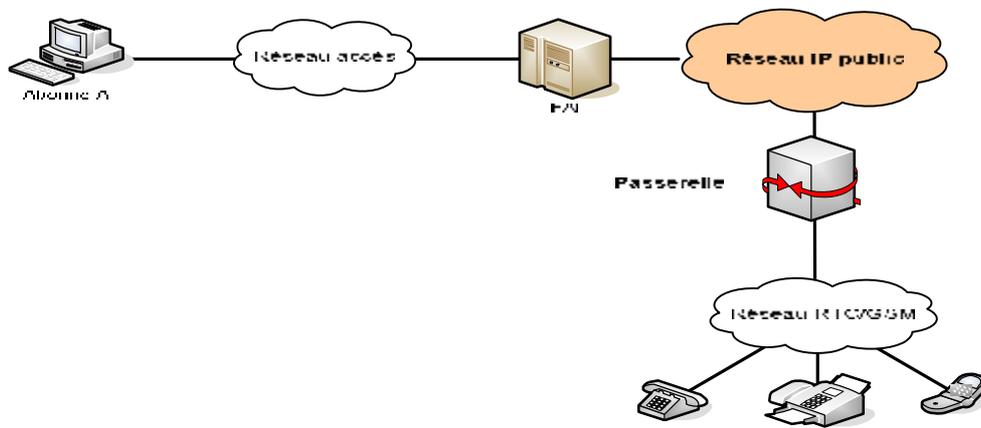


Figure II.6 Le scénario PC 2 phone

II.3.1.3 Téléphone à téléphone (Phone 2 Phone)

Chacun des deux téléphones doit être raccordé à une passerelle pour leur permettre de communiquer sur un réseau IP.

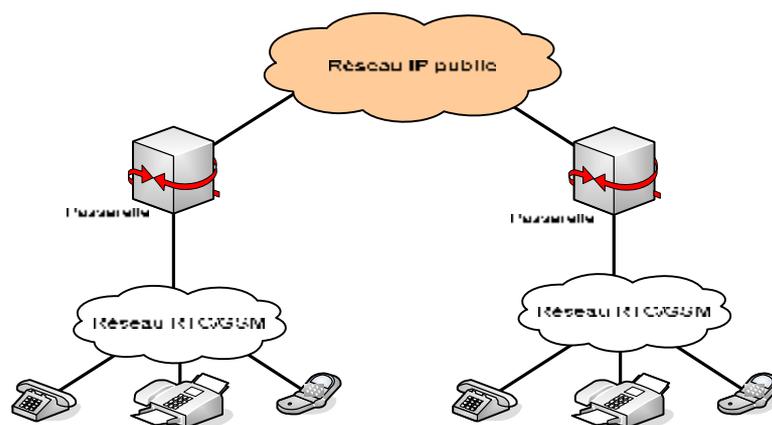


Figure II.7 Le scénario Phone 2 Phone

Pour que toutes les applications développées par différents fournisseurs d'équipement fonctionnent correctement, les organismes internationaux ont élaboré de nouvelles familles de

standards : la série H des recommandations de l'UIT, qui tient lieu de référence pour la communication multimédia et le transport de la voix sur les réseaux IP. La norme **H.323** semble s'être imposée.

D'autres standards concurrents à **H.323**, comme le **SIP "Session Initiation Protocol"** développé par l'IETF ou le **MGCP**, offrent une alternative acceptable. Voici le schéma général de l'utilisation de la Voix sur IP en entreprise :

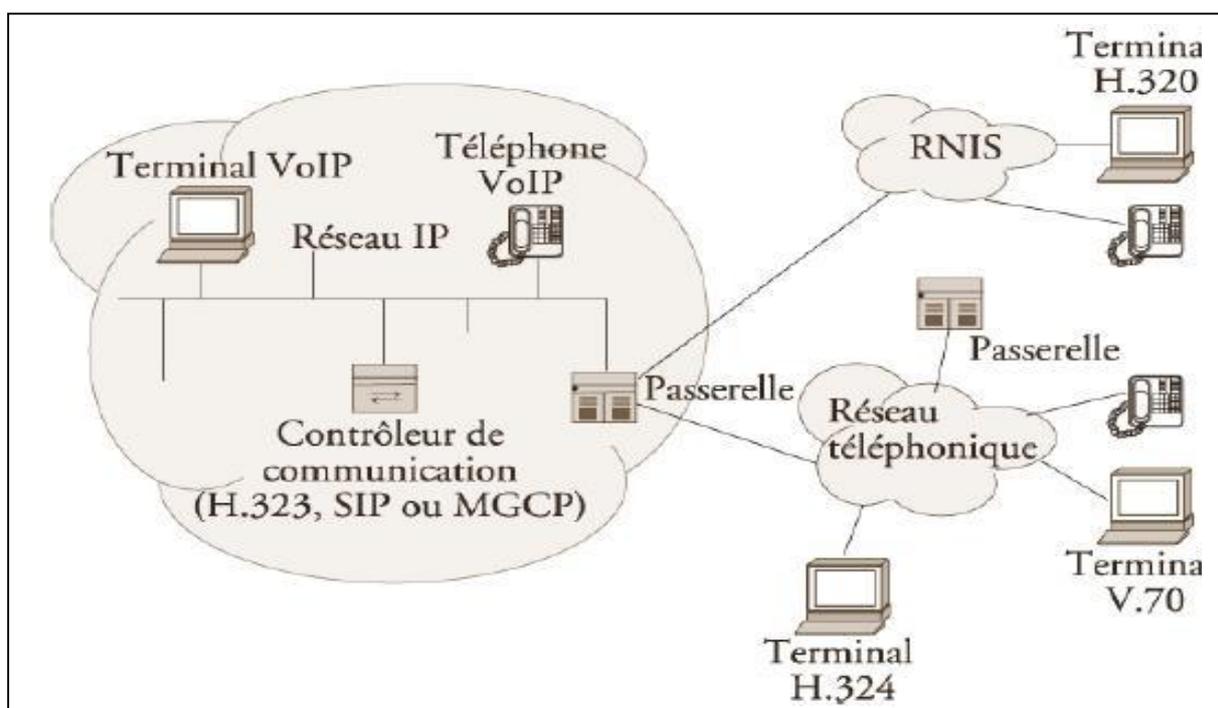


Figure II.8 La voix sur IP dans l'entreprise

Le schéma ci-dessus, décrit de façon générale la topologie d'un réseau de téléphonie IP. Elle comprend toujours des terminaux, un serveur de communication et une passerelle vers les autres réseaux. Chaque norme a ensuite ses propres caractéristiques pour garantir une plus ou moins grande qualité de service.

II.3.2 Principe architectural d'une passerelle VoIP [3]

Schéma bloc

La figure II.9 illustre les principales fonctions d'architecture pour une passerelle VoIP avec ses trois blocs fonctionnels majeurs :

- Traitement de la voix (Voice processing)
- Signalisation téléphonique (Telephony Signaling Gateway)
- Protocoles d'interface réseau (Network Interface Protocol)

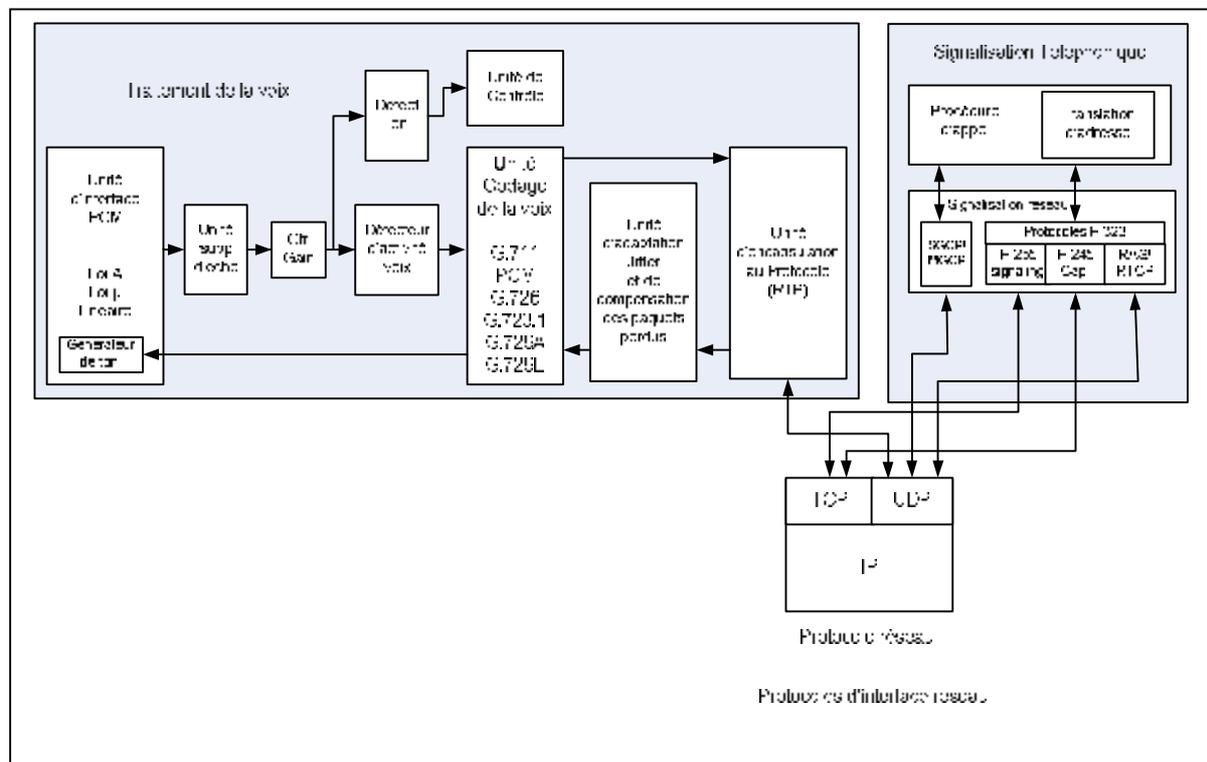


Figure II.9 Architecture d'une passerelle VoIP

- **Traitement de la voix** : Requiert toutes les fonctions requises pour le codage, mises en paquet et la transmission de la voix

- **Signalisation téléphonique** : Performe les fonctions pour l'établissement, le maintien et la fin d'un appel.
- **Protocoles d'interface réseau** : Le protocole TCP/IP fournit la structure pour la transmission de la VoIP à travers le système OSI.

Traitement de la voix

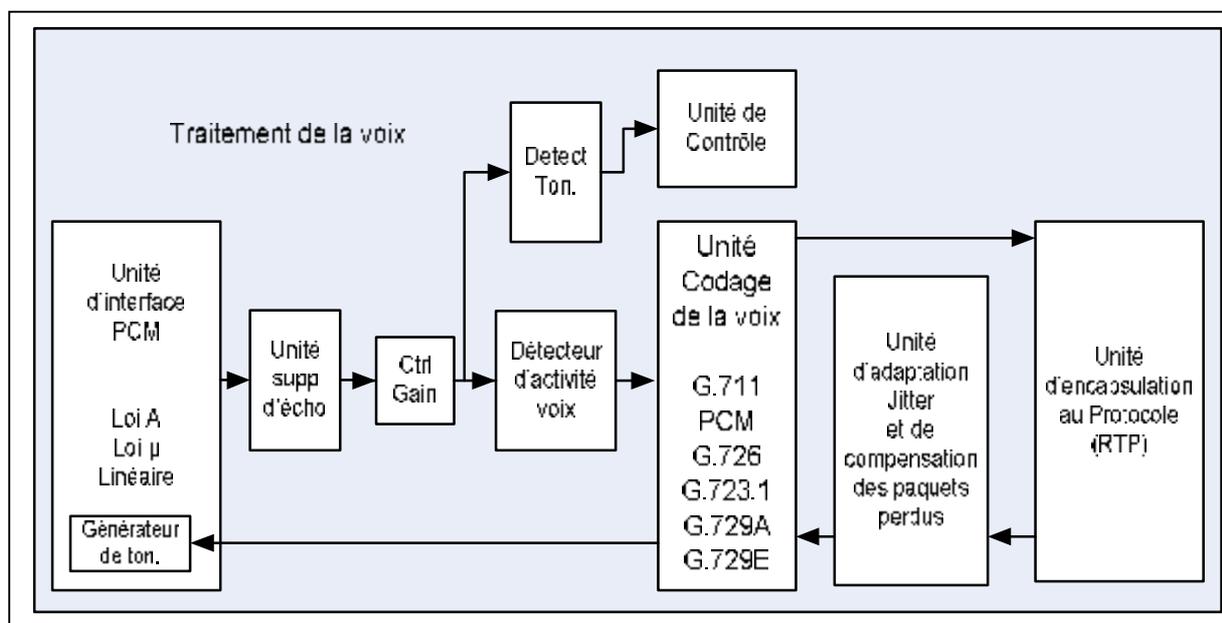


Figure II.10 Bloc Traitement de la voix

Les fonctions de traitement de la voix sont les suivantes :

- **Interface PCM** : adapte les données PCM, il inclut des fonctions comme la compression (loi A, loi μ) et l'échantillonnage, ce bloc contient aussi un Générateur de tonalité, qui génère des signaux DTMF et de progression d'appel.
- **Suppression d'écho** : l'écho est causé par les composants électroniques des parties analogiques du système, qui renvoient une partie du signal traité. Les passerelles doivent traiter l'écho électrique généré par le passage de 2 à 4 fils.

- **Détecteur d'activité voix (Voice Activity Detector)** : supprime la transmission des paquets quand aucun signal vocal n'est présent.
- **Détection Tonalité (Tone Detect)** : détecte les tonalités DTMF et différencie entre les signaux Voix et Fax.
- **Codage de la voix (Voice Coding)** : compresse les trames de bits du signal voix digitalisé pour une transmission, il y a différents codecs utilisés pour compresser la voix dans une application VoIP. Nous prenons comme exemple les standards ITU-T (G.711, G.723, G729, etc.), ce bloc sera traité dans le chapitre III.
- **Adaptation du Jitter** : bufférisse les paquets reçus avant d'être transmis au codec audio pour être restitué, ce bloc contient un mécanisme de mesure permettant l'adaptation du Jitter (qui est la variation de délai entre deux paquets consécutifs) aux performances du réseau (compensation de gigue).
- **Protocole d'encapsulation par paquet (Packet Protocol encapsulation)** : encapsule la voix compressée pour la transmission à travers le réseau IP, chaque paquet contient une séquence numérique qui permet de restituer son ordre à l'arrivée. Ce bloc permet aussi de détecter les paquets perdus et de reproduire correctement les périodes de silence.

II.4 Standards VoIP

II.4.1 Protocole H.323 [5]

II.4.1.1 Présentation

Avec le développement du multimédia sur les réseaux, il est devenu nécessaire de créer des protocoles qui supportent ces nouvelles fonctionnalités, telles que la visioconférence : l'envoi de son et de vidéo avec un souci de données temps réel. Le protocole H.323 est l'un d'eux. Il permet de faire de la visioconférence sur des réseaux IP.

H.323 est un protocole de communication englobant un ensemble de normes utilisées pour l'envoi de données audio et vidéo sur Internet. Il existe depuis 1996 et a été initié par l'ITU (International Communication Union), un groupe international de téléphonie qui

développe des standards de communication. Concrètement, il est utilisé dans des programmes tels que Microsoft Netmeeting, ou encore dans des équipements tels que les routeurs Cisco. Il existe un projet OpenH.323 qui développe un client H.323 en logiciel libre afin que les utilisateurs et les petites entreprises puissent avoir accès à ce protocole sans avoir à déboursier beaucoup d'argent.

II.4.1.2 Fonctionnement

Le protocole H.323 est utilisé pour l'interactivité en temps réel, notamment la visioconférence (signalisation, enregistrement, contrôle d'admission, transport et encodage). C'est le leader du marché pour la téléphonie IP. Il s'inspire du protocole H.320 qui proposait une solution pour la visioconférence sur un réseau numérique à intégration de service (RNIS ou ISDN en anglais). Le protocole H.323 est une adaptation de H.320 pour les réseaux IP. Actuellement, la visioconférence sur liaison RNIS est toujours la technique la plus déployée. Elle existe depuis 1990. Les réseaux utilisés sont à commutation de circuits, ils permettent ainsi de garantir une Qualité de Service (QoS) aux utilisateurs (pas de risque de coupure du son ou de l'image). Aujourd'hui, c'est encore un avantage indiscutable. Par contre, comme pour le téléphone, la facturation est fonction du débit utilisé, du temps de communication et de la distance entre les appels.

H.323 définit plusieurs éléments de réseaux :

- ✓ **Les terminaux :** dans un contexte de téléphonie sur IP, deux types de terminaux H.323 sont aujourd'hui disponibles. Un poste téléphonique IP raccordé directement au réseau Ethernet de l'entreprise. Un PC multimédia sur lequel est installée une application compatible H.323.
- ✓ **Les passerelles (GW: Gateway) :** elles assurent l'interconnexion entre un réseau IP et le réseau téléphonique, ce dernier pouvant être soit le réseau téléphonique public, soit un PABX d'entreprise. Elles assurent la correspondance de la signalisation et des signaux de contrôle et la cohésion entre les médias. Pour ce faire, elles implémentent les fonctions suivantes de transcodage audio (compression, décompression), de modulation, démodulation (pour les fax), de suppression d'échos, de suppression des silences et de contrôle d'appels. Les passerelles sont le plus souvent basées sur des

serveurs informatiques standards (Windows NT, Linux) équipés d'interfaces particulières pour la téléphonie (interfaces analogiques, accès de base ou accès primaire RNIS, etc.) et d'interfaces réseau, par exemple de type Ethernet. La fonctionnalité de passerelle peut toutefois être intégrée directement dans le routeur ainsi que dans les PBX eux-mêmes.

- ▼ **Les portiers (GK: Gatekeeper) :** Ils sont des éléments optionnels dans une solution H.323. Ils ont pour rôle de réaliser la traduction d'adresse (numéro de téléphone - adresse IP) et la gestion des autorisations. Cette dernière permet de donner ou non la permission d'effectuer un appel, de limiter la bande passante pour gérer le trafic sur le LAN. Les "gardes-barrières " ou GK alternatifs permettent également de gérer les téléphones classiques et la signalisation permettant de router les appels afin d'offrir des services supplémentaires. Ils peuvent enfin offrir des services d'annuaires.
- ▼ **Les unités de contrôle multipoint (MCU, Multipoint Control Unit) :** référence au protocole T.120 qui permet aux clients de se connecter aux sessions de conférence de données. Les unités de contrôle multipoint peuvent communiquer entre elles pour échanger des informations de conférence.

Dans un contexte de téléphonie sur IP, la signalisation a pour objectif de réaliser les fonctions suivantes :

- ▼ **Recherche et traduction d'adresses :** sur la base du numéro de téléphone du destinataire, il s'agit de trouver son adresse IP (appel téléphone-PC) ou l'adresse IP de la passerelle desservant le destinataire. Cette fonction est prise en charge par le Gatekeeper. Elle est effectuée soit localement soit par requête vers un annuaire centralisé.
- ▼ **Contrôle d'appel :** l'équipement terminal (« endpoint » = terminal H.323 ou passerelle) situé à l'origine de l'appel établit une connexion avec l'équipement de destination et échange avec lui les informations nécessaires à l'établissement de l'appel. Dans le cas d'une passerelle, cette fonction implique également de supporter la signalisation propre à l'équipement téléphonique à laquelle elle est raccordée (signalisation analogique, Q.931, etc.) et de traduire cette signalisation dans le format défini dans H.323. Le contrôle d'appel est pris en charge soit par les équipements terminaux soit par le Gatekeeper. Dans ce cas, tous les messages de signalisation sont

routés via le Gatekeeper, ce dernier jouant alors un rôle similaire à celui d'un PBX.
Services supplémentaires : déviation, transfert d'appel, conférence, etc.

Trois protocoles de signalisation sont spécifiés dans le cadre de H.323 à savoir :

- ✓ **RAS (Registration, Admission and Status)** : ce protocole est utilisé pour communiquer avec un Gatekeeper. Il sert notamment aux équipements terminaux pour découvrir l'existence d'un Gatekeeper et s'enregistrer auprès de ce dernier ainsi que pour les demandes de traduction d'adresses. La signalisation RAS utilise des messages H.225.0 6 transmis sur un protocole de transport non fiable (UDP, par exemple).
- ✓ **Q.931 - H.323** utilise une version simplifiée de la signalisation RNIS Q.931 pour l'établissement et le contrôle d'appels téléphoniques sur IP. Cette version simplifiée est également spécifiée dans la norme H.225.0
- ✓ **H.245** : ce protocole est utilisé pour l'échange de capacités entre deux équipements terminaux. Par exemple, il est utilisé par ces derniers pour s'accorder sur le type de codec à activer. Il peut également servir à mesurer le retard aller-retour (Round Trip Delay) d'une communication.

Une communication H.323 se déroule en cinq phases :

- ✓ Etablissement d'appel.
- ✓ Echange de capacité et réservation éventuelle de la bande passante à travers le protocole RSVP (Ressource reSerVation Protocol).
- ✓ Établissement de la communication audiovisuelle
- ✓ Invocation éventuelle de services en phase d'appel (par exemple, transfert d'appel, changement de bande passante, etc.).
- ✓ Libération de l'appel.

- H323 dans le modèle OSI [3]

Les différents protocoles sont représentés ci-dessous dans le modèle OSI :

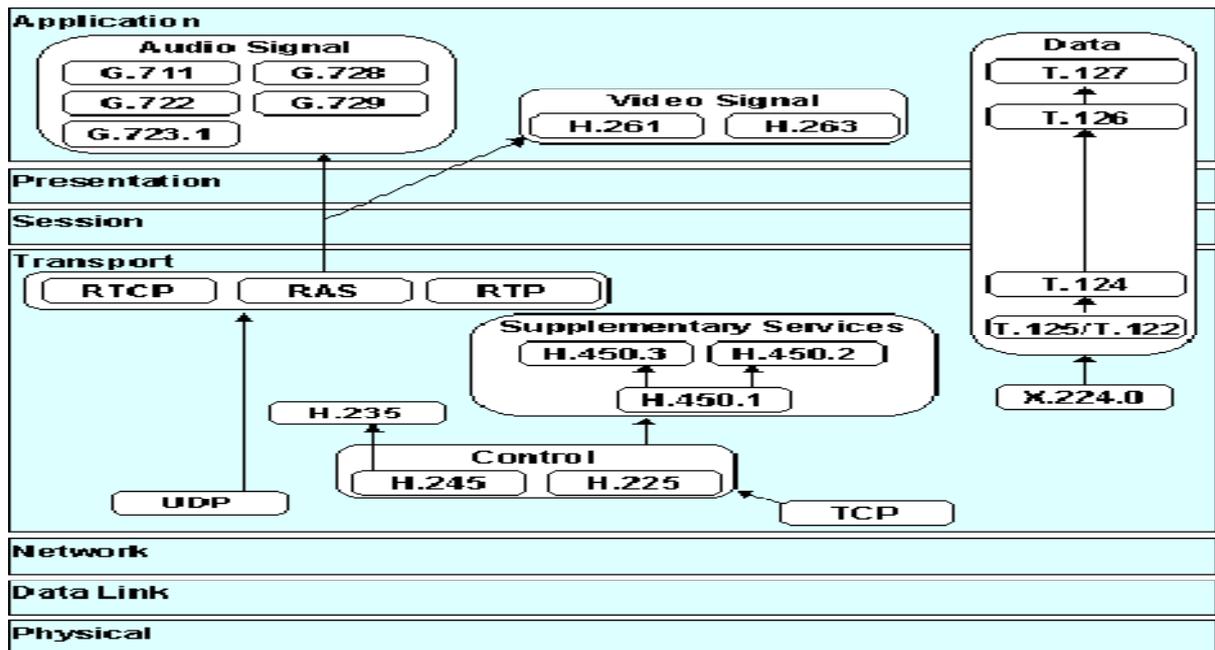


Figure II.11 Pile du protocole H.323

II.4.2 Protocole SIP [5]

II.4.2.1 Présentation

Le protocole SIP (Session Initiation Protocol) est un protocole normalisé et standardisé par l'IETF, qui a été conçu pour établir, modifier et terminer les sessions multimédias. Il se charge de l'authentification et localisation des multiples participants. Il se charge également sur les types de média utilisables par les différents participants en encapsulant des messages SDP (Session Description Protocol) .SIP ne transporte pas les données échangées durant la session comme la voix ou la vidéo. SIP étant indépendant de la transmission des données, tous types de données et de protocoles peuvent être utilisés pour cet échange. SIP est le standard ouvert de la VoIP inter opérable le plus étendu et vise à devenir le standard des télécommunications multimédia (son, image, etc.). SIP n'est pas donc seulement destiné à la

VoIP mais, pour de nombreuses autres applications telles que la visiophonie, la messagerie instantanée et la réalité virtuelle ou même les jeux vidéo.

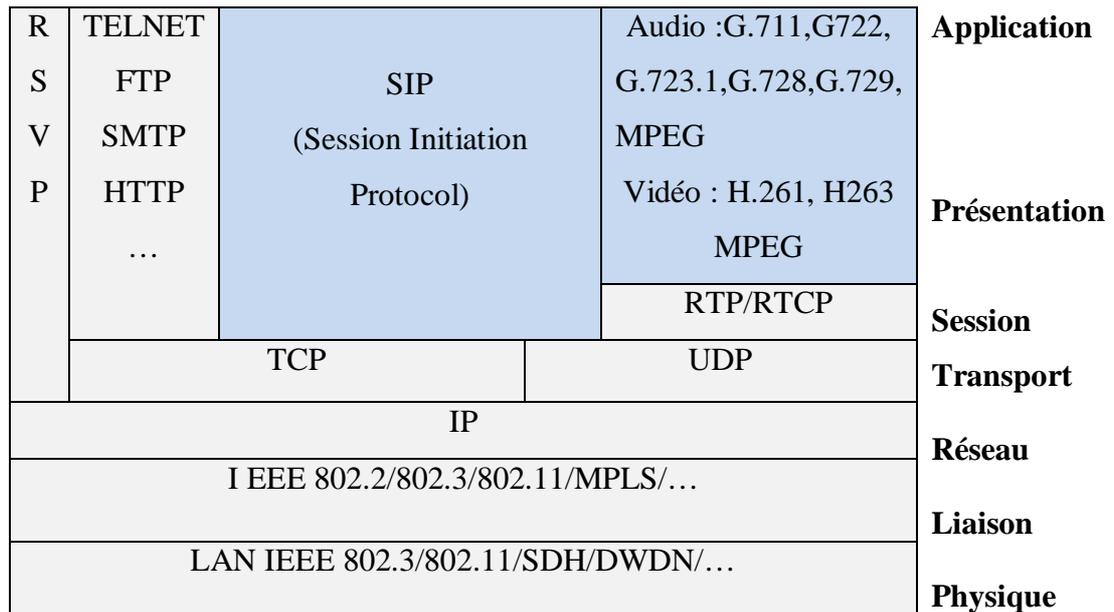


Figure II.12 SIP dans le modèle OSI

Comme on peut voir sur la figure II.12, SIP s'est superposé à 3 couches de modèle OSI : la couche application, la couche présentation et la couche session ; ceci afin de faciliter sa programmation, son intégration et son utilisation dans les logiciels de communication. SIP possède l'avantage de ne pas être attaché à un médium particulier et est censé être indépendant du protocole de transport des couches basses.

Voici un exemple de communication simple illustré par la figure II.13 ci-dessous [5]

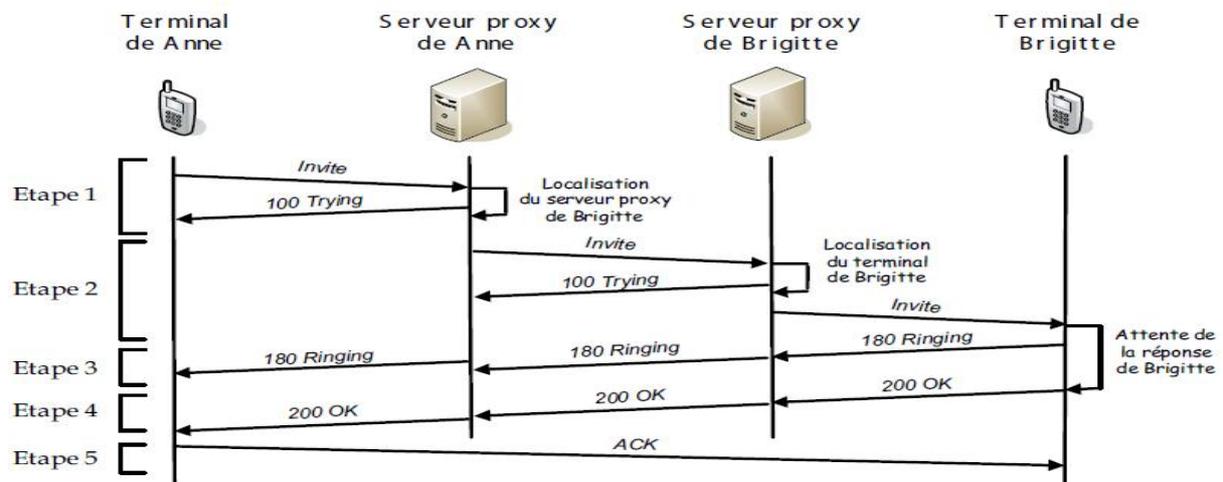


Figure II.13 Exemple de communication SIP

Ce schéma qui représente l'établissement d'une session SIP à travers un proxy entre deux utilisateurs montre les différentes étapes de cette communication ainsi que les requêtes d'établissement de session spécifiques au protocole.

Un utilisateur peut se connecter avec les utilisateurs d'une session déjà ouverte. Pour ouvrir une session, un utilisateur émet une invitation transportant un descripteur de session permettant aux utilisateurs souhaitant communiquer de s'accorder sur la compatibilité de leur média, SIP permet donc de relier des stations mobiles en transmettant ou redirigeant les requêtes vers la position courante de la station appelée.

II.4.2.2 Fonctionnement

SIP intervient aux différentes phases de l'appel :

- Localisation du terminal correspondant.
- Analyse du profil et des ressources du destinataire.
- Négociation du type de média (voix, vidéo, données...) et des paramètres de communication.

- Disponibilité du correspondant détermine si le poste appelé souhaite communiquer et autorise l'appelant à le contacter.
- Etablissement et suivi de l'appel avertissent les parties appelantes et appelées de la demande d'ouverture de session, gestion du transfert et de la fermeture des appels.
- Gestion de fonctions évoluées : cryptage, retour d'erreurs.

Avec SIP, les utilisateurs qui ouvrent une session peuvent communiquer en mode point à point, en mode diffusif ou dans un mode combinant ceux-ci. SIP permet donc l'ouverture de session en mode :

- **Unicast (point à point)** : communication entre deux machines.
- **Broadcast (diffusif)** : communication entre une machine et toutes les machines du réseau.
- **Multicast (combinatoire)** : communication entre une machine et quelques machines de réseau.

Voici les différents éléments intervenant dans l'ouverture de session :

- Suivant la nature des échanges, choix des protocoles les mieux adaptés (RSVP, RTP, RTCP, SAP, SDP).
- Détermination du nombre de session, par exemple, pour véhiculer de la vidéo, deux sessions doivent être ouvertes (l'une pour l'image et l'autre pour la voix).
- Chaque utilisateur et sa machine sont identifiés par une adresse que l'on nomme URL SIP et qui se présente comme une URL Mailto.
- Requête URL permettant de localiser le proxy server auquel est rattachée la machine de l'appelé.
- Requête SIP, une fois le client (machine appelante) connecté à un serveur SIP distant, il peut lui adresser une ou plusieurs requêtes SIP et recevoir une ou plusieurs réponses de ce serveur. Les réponses contiennent certains champs identiques à ceux des requêtes, tels que : Call-ID, Cseq.

Les échanges entre un terminal appelant et un terminal appelé se font par l'intermédiaire de requêtes :

- Ø **Invite** : cette requête indique que l'application (ou l'utilisateur) correspondante à l'URL SIP spécifiée est invitée à participer à une session. Le corps du message décrit cette session (par ex. : média supporté par l'appelant). En cas de réponse favorable, l'invité doit spécifier les médias qu'il supporte.
- Ø **Ack** : cette requête permet de confirmer que le terminal appelant a bien reçu une réponse définitive à une requête Invite.
- Ø **Options** : un proxy server en mesure de contacter l'UAS (terminal) appelé, doit répondre à une requête Options en précisant ses capacités à contacter le même terminal.
- Ø **Bye** : cette requête est utilisée par le terminal de l'appelé, afin de signaler qu'il souhaite mettre un terme à la session.
- Ø **Cancel** : cette requête est envoyée par un terminal ou un proxy server afin d'annuler une requête non validée par une réponse finale comme, par exemple, si une machine ayant été invitée à participer à une session, et ayant accepté l'invitation ne reçoit pas de requête Ack, alors elle émet une requête Cancel.
- Ø **Register** : cette méthode est utilisée par le client pour enregistrer l'adresse listée dans l'URL TO par le serveur auquel il est relié.

Une réponse à une requête est caractérisée, par un code et un motif, appelés code d'état et raison phrase respectivement. Un code d'état est un entier codé sur 3 bits indiquant un résultat à l'issue de la réception d'une requête.

Ce résultat est précisé par une phrase, textbased (UTF-8), expliquant le motif du refus ou de l'acceptation de la requête. Le code d'état est donc destiné à l'automate gérant l'établissement des sessions SIP et les motifs aux programmeurs. Il existe 6 classes de réponses et donc de codes d'état, représentées par le premier bit :

- **1xx** = Information - La requête a été reçue et continue à être traitée
- **2xx** = Succès - L'action a été reçue avec succès, comprise et acceptée
- **3xx** = Redirection - Une autre action doit être menée afin de valider la requête
- **4xx** = Erreur du client - La requête contient une syntaxe erronée ou ne peut pas être traitée par ce serveur.

- **5xx** = Erreur du serveur - Le serveur n'a pas réussi à traiter une requête apparemment correcte
- **6xx** = Echec général - La requête ne peut être traitée par aucun serveur

Dans un système SIP on trouve deux types de composantes, les Users Agents (UAS, UAC) et un réseau de serveurs :

- § **L'UAS (User Agent Server)** : il représente l'agent de la partie appelée. C'est une application de type serveur qui contacte l'utilisateur lorsqu'une requête SIP est reçue. Et elle renvoie une réponse au nom de l'utilisateur.
- § **L'UAC (User Agent Client)** : il représente l'agent de la partie appelante. C'est une application de type client qui initie les requêtes.
- § **Le relais mandataire ou PS (Proxy Server)** : auquel est relié un terminal fixe ou mobile, agit à la fois comme un client et comme un serveur. Un tel serveur peut interpréter et modifier les messages qu'il reçoit avant de les retransmettre
- § **Le RS (Redirect Server)** : il réalise simplement une association (mapping) d'adresses vers une ou plusieurs nouvelles adresses. (lorsqu'un client appelle un terminal mobile il fait la redirection vers le PS le plus proche ou en mode multicast le message émis est redirigé vers toutes les sorties auxquelles sont reliés les destinataires). Notons qu'un Redirect Server est consulté par l'UAC comme un simple serveur et ne peut émettre de requêtes contrairement au PS.
- § **Le LS (Location Server)** : il fournit la position courante des utilisateurs dont la communication traverse les RS et PS auxquels il est rattaché. Cette fonction est assurée par le service de localisation.
- § **Le RG (Registrar)** : c'est un serveur qui accepte les requêtes Register et offre également un service de localisation comme le LS. Chaque PS ou RS est généralement relié à un Registrar.

II.4.2.3 Sécurité et authentification [5]

Les messages SIP, peuvent contenir des données confidentielles, en effet le protocole SIP possède 3 mécanismes de cryptage :

- Cryptage de bout du corps du message SIP et de certains champs d'entête sensibles aux attaques.
- Cryptage au saut par saut (hop by hop) afin d'empêcher des pirates de savoir qui appelle qui.
- Cryptage au saut par saut du champ d'entête pour dissimuler la route qu'a empruntée la requête.

De plus afin d'empêcher tout intrus de modifier et retransmettre des réponses SIP, des mécanismes d'intégrité et d'authentification de messages sont mis en place. Et pour des messages SIP transmis de bout en bout, des clés publiques et signatures sont utilisées par SIP et stockées dans les champs d'entête autorisation.

Une autre attaque connue avec TCP ou UDP : «Deny of Service », lorsqu'un proxy server intrus envoie une réponse code 6xx au client (échec général). Le client peut ignorer cette réponse. S'il ne l'ignore pas et émet une requête vers le serveur « régulier » auquel il était relié avant la réponse de serveur « intrus », la requête aura de fortes chances d'atteindre le serveur intrus et non son vrai destinataire.

II.5 Comparaison entre les standards H.323 et SIP [5]

Voici les avantages du protocole H.323 :

- ✓ Il existe de nombreux produits (plus de 30) utilisant ce standard adopté par de grandes entreprises telles Cisco, IBM, Intel, Microsoft, Netscape, etc.
- ✓ Les cinq principaux logiciels de visioconférence Picturel 550, Proshare 500, Tricon 500, Smartstation et Cruiser 150 utilisent sur IP la norme H.323.
- ✓ Un niveau d'interopérabilité est très élevé, ce qui permet à plusieurs utilisateurs d'échanger des données audio et vidéo sans faire attention aux types de média qu'ils utilisent.

Voici les avantages du protocole SIP :

- ✓ SIP est un protocole plus rapide. La séparation entre ses champs d'entête et son corps du message facilite le traitement des messages et diminue leur temps de transition dans le réseau.
- ✓ Le nombre des entêtes est limité (36 au maximum et en pratique, moins d'une dizaine d'entêtes sont utilisées simultanément), ce qui allège l'écriture et la lecture des requêtes et réponses.
- ✓ SIP est un protocole indépendant de la couche transport. Il peut aussi bien s'utiliser avec TCP qu'UDP.

De plus, il sépare les flux de données de la signalisation, ce qui rend plus souple l'évolution "en direct" d'une communication (arrivée d'un nouveau participant, changement de paramètres...).

	SIP	H323
Nombre d'échanges pour établir la connexion	1,5 aller-retour	6 à 7 allers-retours
Maintenance du code protocolaire	Simple par sa nature textuelle à l'exemple de HTTP	Complexe et nécessitant un compilateur
Evolution du protocole	Protocole ouvert à de nouvelles fonctions	Ajout d'extensions propriétaires sans concertation entre vendeurs
Fonction de conférence	Distribuée	Centralisée par l'unité MC
Fonction de télé services	Oui, par défaut	H.323 v2 + H.450
Détection d'un appel en boucle	Oui	Inexistante sur la version 1 un appel routé sur l'appelant provoque une infinité de requêtes
Signalisation multicast	Oui, par défaut	Non

Tableau II.1 comparaison entre protocoles (H.323 et SIP)

II.6 Transport RTP et RTCP [2] [3]

RTP est un protocole qui a été développé par l'IETF, afin de faciliter le transport temps réel de bout en bout des flots donnés audio et vidéo sur les réseaux IP, c'est-à-dire sur les

réseaux de paquets. RTP est un protocole qui se situe au niveau de l'application et qui utilise les protocoles sous-jacents de transport TCP ou UDP. Mais l'utilisation de RTP se fait généralement au-dessus d'UDP ce qui permet d'atteindre plus facilement le temps réel. Les applications temps réelles comme la parole numérique ou la visioconférence constituent un véritable problème pour Internet. Qui dit application temps réel, dit présence d'une certaine qualité de service (QoS) que RTP ne garantit pas du fait qu'il fonctionne au niveau applicatif. De plus, RTP est un protocole qui se trouve dans un environnement multipoint, donc on peut dire que RTP possède à sa charge la gestion du temps réel, mais aussi l'administration de la session multipoint.

RTP et RTCP sont définis, depuis juillet 2003, par la RFC3550 rendant obsolète la version précédente RFC 1889.

II.6.1 Fonctions de RTP

Le protocole RTP, Real Time Transport Protocol, standardisé en 1996, a pour but d'organiser les paquets à l'entrée du réseau et de les contrôler à la sortie. Ceci de façon à reformer les flux avec ses caractéristiques de départ. RTP est géré au niveau de l'application donc ne nécessite pas l'implémentation d'un Kernel ou de bibliothèques. Comme nous l'avons dit dans l'introduction, RTP est un protocole de bout en bout. RTP est volontairement incomplet et malléable pour s'adapter aux besoins des applications. Il sera intégré dans le noyau de l'application. RTP laisse la responsabilité du contrôle aux équipements d'extrémité.

RTP est un protocole adapté aux applications présentant des propriétés temps réelles. Il permet ainsi de :

- ✓ Reconstituer la base de temps des flux (horodatage des paquets : possibilité de resynchronisation des flux par le récepteur).
- ✓ Mettre en place un séquençement des paquets par une numérotation, et ce, afin de permettre ainsi la détection des paquets perdus. Ceci est un point primordial dans la reconstitution des données. Mais il faut savoir que la perte d'un paquet n'est pas un gros problème si les paquets ne sont pas perdus en trop grand nombre. Cependant, il est très important de savoir quel est le paquet qui a été perdu afin de pouvoir pallier

cette perte. Et ce par le remplacement d'un paquet qui se compose d'une synthèse des paquets précédents et suivants.

- ✓ Identifier le contenu des données pour leurs associer un transport sécurisé.
- ✓ L'identification de la source c'est-à-dire l'identification de l'expéditeur du paquet. Dans un multicast, l'identité de la source doit être connue et déterminée.
- ✓ Transporter les applications audio et vidéo dans des trames (avec des dimensions qui sont dépendantes des codecs qui effectuent la numérisation). Ces trames sont incluses dans des paquets afin d'être transportées et doivent de ce fait être récupérées facilement au moment de la phase de dépaquetisation afin que l'application soit décodée correctement.

En revanche, ce n'est pas "la solution" qui permettrait d'obtenir des transmissions temps réel sur IP. En effet, il ne procure pas de :

- ✓ Réserve de ressources sur le réseau (pas d'action sur le réseau, cf. RSVP).
- ✓ Fiabilité des échanges (pas de retransmission automatique, pas de régulation automatique du débit).
- ✓ Garantie dans le délai de livraison (seules les couches de niveau inférieur le peuvent) et dans la continuité du flux temps réel.

- **Entête RTP**

L'entête d'un paquet RTP est obligatoirement constitué de 16 octets. Cette entête précède le "payload" qui représente les données utiles.

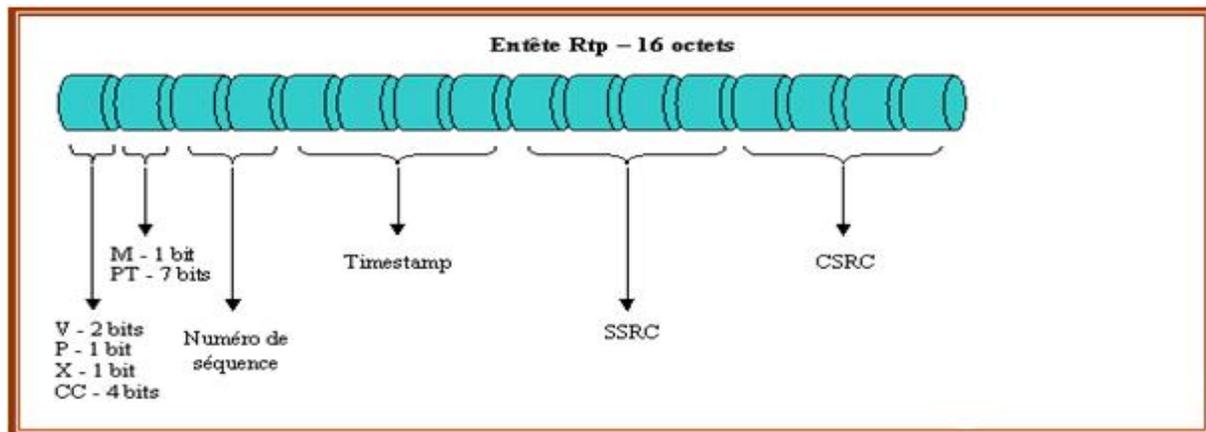


Figure II.14 Entête RTP

II.6.2 Fonctions de RTCP

Le protocole RTCP est fondé sur la transmission périodique de paquets de contrôle à tous les participants d'une session. C'est le protocole UDP (par exemple) qui permet le multiplexage des paquets de données RTP et des paquets de contrôle RTCP. Le protocole RTP utilise le protocole RTCP, Real time Transport Control Protocol, qui transporte les informations supplémentaires suivantes pour la gestion de la session :

- ✓ Les récepteurs utilisent RTCP pour renvoyer vers les émetteurs un rapport sur la QoS. Ces rapports comprennent le nombre de paquets perdus, le paramètre indiquant la variance d'une distribution (plus communément appelé la gigue : c'est-à-dire les paquets qui arrivent régulièrement ou irrégulièrement) et le délai aller-retour. Ces informations permettent à la source de s'adapter, par exemple, de modifier le niveau de compression pour maintenir une QoS.
- ✓ Une synchronisation supplémentaire entre les médias. Les applications multimédias sont souvent transportées par des flots distincts. Par exemple, la voix, l'image ou même des applications numérisées sur plusieurs niveaux hiérarchiques peuvent voir les flots gérés suivre des chemins différents.
- ✓ L'identification, en effet, les paquets RTCP contiennent des informations d'adresses, comme l'adresse d'un message électronique, un numéro de téléphone ou le nom d'un participant à une conférence téléphonique.

- ✓ Le contrôle de la session, car RTCP permet aux participants d'indiquer leur départ d'une conférence téléphonique (paquet Bye de RTCP) ou simplement de fournir une indication sur leur comportement.

Le protocole RTCP demande aux participants de la session d'envoyer périodiquement les informations citées ci-dessus. La périodicité est calculée en fonction du nombre de participants de l'application. On peut dire que les paquets RTP ne transportent que les données des utilisateurs. Tandis que les paquets RTCP ne transportent en temps réel, que de la supervision. On peut détailler les paquets de supervision en 5 types :

- **200** : rapport de l'émetteur
 - **201** : rapport du récepteur
 - **202** : description de la source
 - **203** : au revoir
 - **204** : application spécifique
-
- **Entête RTCP**

Ce protocole définit cinq paquets de contrôle :

- **200-SR (Sender Report)** : ce rapport regroupe des statistiques concernant la transmission (pourcentage de perte, nombre cumulé de paquets perdus, variation de délai (jiggle)). Ces rapports sont issus d'émetteurs actifs d'une session.
- **201 - RR (Receiver Report)** : ensemble de statistiques portant sur la communication entre les participants. Ces rapports sont issus des récepteurs d'une session.
- **202 - SDES (Source Description)** : carte de visite de la source (nom, e-mail, localisation).
- **203 - BYE** : message de fin de participation à une session.
- **204 - APP** : fonctions spécifiques à une application.

Voici l'entête commun à tous les paquets RTCP [3]

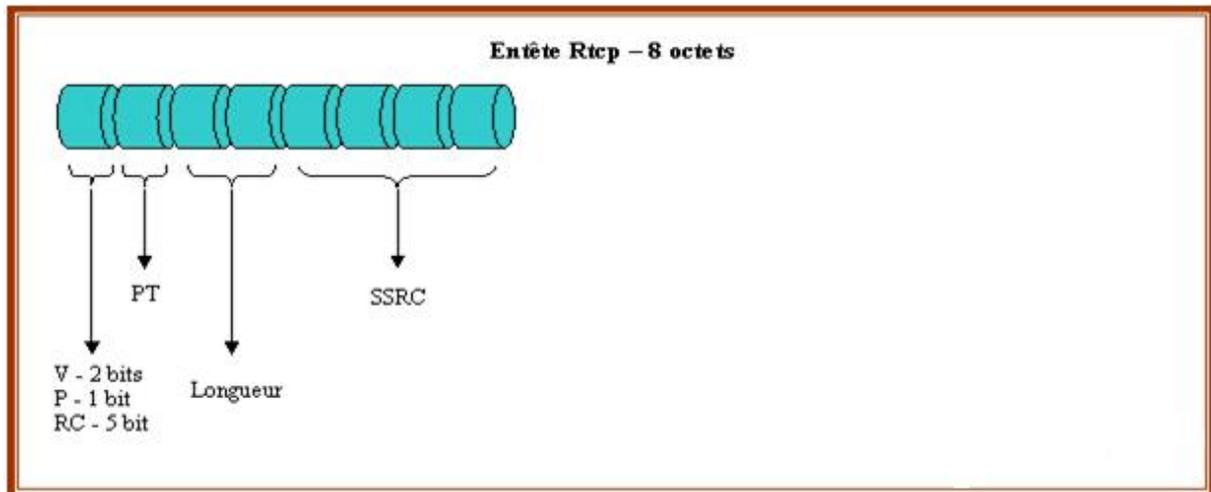


Figure II.15 Entête RTCP

II.7 Avantage de la VoIP [5]

La VoIP offre de nombreuses nouvelles possibilités aux opérateurs et utilisateurs bénéficiant d'un réseau basé sur IP. Les avantages les plus importants sont les suivants :

- Réduction des coûts : en migrant du trafic RTC vers le réseau IP, les entreprises peuvent réduire sensiblement les coûts de communications locales, et réduire considérablement le coût des appels à longues distances.
- Expansions aux niveaux utilisateurs et opérateurs plus rapides ; les évolutions de la VoIP deviennent moins dépendantes des choix technologiques antérieurs.
- La VoIP intègre une gestion de la voix mais également une gestion de la vidéo. Le réseau VoIP peut accueillir des applications vidéo de types vidéo conférence, vidéo surveillance, e-Learning, pour l'ensemble des utilisateurs à un coût d'infrastructure réseau supplémentaire minime.
- L'ensemble des services de voix peut être géré sur une plateforme centrale, avec un réseau capillaire IP ; il n'est donc pas utile de déployer des commutateurs dans chaque ville.

II.8 Problème et Qualité de service (QoS)

II.8.1 Latence

La maîtrise du délai de transmission est un élément essentiel pour bénéficier d'un véritable mode conversationnel et minimiser la perception d'écho (similaire aux désagréments causés par les conversations par satellites, désormais largement remplacés par les câbles pour ce type d'usage).

Or la durée de traversée d'un réseau IP dépend de nombreux facteurs :

- Le débit de transmission sur chaque lien.
- Le nombre d'éléments réseaux traversés.
- Le temps de traversée de chaque élément, qui est lui-même fonction de la puissance et la charge de ce dernier, du temps de mise en file d'attente des paquets, et du temps d'accès en sortie de l'élément.
- Le délai de propagation de l'information, qui est non négligeable si l' on communique à l'opposé de la terre. Une transmission par fibre optique, à l'opposé de la terre, dure environ 70 ms.

Noter que le temps de transport de l'information n'est pas le seul facteur responsable de la durée totale de traitement de la parole. Le temps de codage et la mise en paquet de la voix contribuent aussi de manière importante à ce délai.

Il est important de rappeler que sur les réseaux IP actuels (sans mécanismes de garantie de qualité de service), chaque paquet IP « fait son chemin » indépendamment des paquets qui le précèdent ou le suivent : c'est ce qu'on appelle grossièrement le « Best effort » pour signifier que le réseau ne contrôle rien. Ce fonctionnement est fondamentalement différent de celui du réseau téléphonique où un circuit est établi pendant toute la durée de la communication.

Les chiffres suivants (tirés de la recommandation UIT-T G114) sont donnés à titre indicatif pour préciser les classes de qualité et d'interactivité en fonction du retard de

transmission dans une conversation téléphonique. Ces chiffres concernent le délai total de traitement, et pas uniquement le temps de transmission de l'information sur le réseau.

Classe n° Délai par sens Commentaires [3]

Classe n°	Délai par sens	Commentaires
1	0 à 150 ms	Acceptable pour la plupart des conversations
2	150 à 300 ms	Acceptable pour des communications faiblement interactives
3	300 à 700 ms	Deviens pratiquement une communication half duplex
4	Au-delà de 700 ms	Inutilisable sans une bonne pratique de la conversation half duplex

Tableau II. 2 Classe n° Délai par sens Commentaires

En conclusion, on considère généralement que la limite supérieure "acceptable", pour une communication téléphonique, se situe entre 150 et 200 ms par sens de transmission (en considérant à la fois le traitement de la voix et le délai d'acheminement).

II.8.2 Perte de paquet

Lorsque les buffers des différents éléments réseaux IP sont congestionnés, ils libèrent automatiquement la bande passante en se débarrassant d'une certaine proportion des paquets entrant, en fonction de seuils prédéfinis. Cela permet également d'envoyer un signal implicite aux terminaux TCP qui diminuent d'autant leur débit au vu des acquittements négatifs émis par le destinataire qui ne reçoit plus les paquets. Malheureusement, pour les paquets de voix, qui sont véhiculés au-dessus de UDP, aucun mécanisme de contrôle de flux ou de retransmission des paquets perdus n'est offert au niveau du transport. D'où l'importance des protocoles RTP et RTCP qui permettent de déterminer le taux de perte de paquet, et d'agir en conséquence au niveau applicatif.

Si aucun mécanisme performant de récupération des paquets perdus n'est mis en place (cas le plus fréquent dans les équipements actuels), alors la perte de paquet IP se traduit par

des ruptures au niveau de la conversation et une impression de hachure de la parole. Cette dégradation est bien sûr accentuée si chaque paquet contient un long temps de parole (plusieurs trames de voix de paquet). Par ailleurs, les codeurs à très faible débit sont généralement plus sensibles à la perte d'information, et mettent plus de temps à reconstruire un codage fidèle.

Enfin, connaître le pourcentage de perte de paquets sur une liaison n'est pas suffisant pour déterminer la qualité de la voix que l'on peut espérer, mais cela donne une bonne approximation. En effet, un autre facteur essentiel intervient ; il s'agit du modèle de répartition de cette perte de paquets, qui peut être soit régulièrement répartie, soit répartie de manière corrélée, c'est-à-dire avec des pics de perte lors des phases de congestion, suivies de phases moins dégradées en terme de QoS.

II.8.3 Gigue

La gigue est la variance statistique du délai de transmission. En d'autres termes, elle mesure la variation temporelle entre le moment où deux paquets auraient dû arriver et le moment de leur arrivée effective. Cette irrégularité d'arrivée des paquets est due à de multiples raisons, dont l'encapsulation des paquets IP dans les protocoles supportés, la charge du réseau à un instant donné, la variation des chemins empruntés dans le réseau, etc.

Pour compenser la gigue, on utilise généralement des mémoires tampons (buffer de gigue) qui permettent de lisser l'irrégularité des paquets. Malheureusement, ces paquets présentent l'inconvénient de rallonger d'autant le temps de traversée globale du système. Leur taille doit donc être soigneusement définie, et si possible adaptée de manière dynamique aux conditions du réseau.

La dégradation de la qualité de service due à la présence de gigue se traduit en fait, par une combinaison des deux facteurs cités précédemment : le délai et la perte de paquets ; puisque d'une part on introduit un délai supplémentaire de traitement (buffer de gigue) lorsque l'on décide d'attendre les paquets qui arrivent en retard, et que d'autre part on finit tout de même par perdre certains paquets lorsque ceux-ci ont un retard qui dépasse le délai maximum autorisé par le buffer.

II.9 Etat du marché [3]

On compte une bonne vingtaine de firmes sur le marché. Les principaux sont Cisco, Clarent, Avaya, Alcatel, Nortel Network, Siemens, Ténovis, 3COM... Ce qu'il faut souligner, c'est le fait qu'il y ait peu de concurrents, car comme nous l'avons dit précédemment, la téléphonie sur IP est un marché très jeune et très novateur. D'ailleurs, le fait que la téléphonie sur IP soit un marché chevauchant deux secteurs qui se rapprochent et étaient complètement différents auparavant, la téléphonie et l'informatique nous assistent ici à une concurrence ayant des origines différentes. En effet, nous retrouvons le géant de l'équipement réseaux Cisco en concurrence avec des entreprises de téléphonies telles qu'Alcatel ou Siemens. Mais Cisco et Clarent arrivent largement en tête, sur un marché qui de 259 millions de dollars année 2005 et 2,89 milliards en 2006. La téléphonie sur IP propose 3 types de terminaux différents : Les hardphones qui sont des téléphones physiques IP, les softphones qui sont des logiciels permettant de téléphoner sur IP au travers d'un PC et les téléphones IP Wi-fi qui sont des téléphones sans-fil IP. Mais la plupart des concurrents proposent ces 3 produits qui sont plutôt homogènes. Un softphone Cisco et un Softphone Siemens sont quasi identiques. Seule l'interface graphique les distingue. Pour le client, le produit des 2 concurrents est identique dans la mesure où il apporte les mêmes services.

II.10 Discussion

Dans ce chapitre, on a traité l'état de l'art de la voix sur IP dont on a étudié les différents protocoles de transport en temps réels (RTP, RTCP) ainsi, ceux de signalisations (H.323, SIP). En effet, la conception d'un système VoIP s'appuie sur ces protocoles.

III.1 Préambule

Le codage de la parole est une discipline très utilisée de nos jours, on la retrouve dans presque tous les domaines où la voix prédomine, réseau de téléphone cellulaire, le fixe, etc., et bien entendu la VoIP où elle joue un rôle majeur et occupe une grande partie des efforts fournis lors des développements des plateformes VoIP.

Le codage de la parole utilise des redondances que l'on peut trouver dans le signal parole, ainsi que les imperfections d'auditions de l'oreille humaine pour réduire le volume d'information nécessaire.

Dans le domaine de codage de la parole, l'objectif essentiel est de compresser au mieux que possible la parole en réduisant le nombre de bits nécessaires par échantillon de telle sorte que, le signal parole reconstruit au niveau du décodeur, soit aussi audible que l'original. Le but essentiel des experts est d'optimiser les algorithmes de codage afin d'assurer une bonne qualité audio en tenant compte des contraintes de temps d'exécution et de consommation en bande passante et espace mémoire. Pour cela, il faut trouver un compromis entre la qualité et la taille des données.

Pour bien monter ce compromis, on a procédé à l'étude des trois codeurs :

- Le codeur LOG-PCM G.711 (Mu Law) issu des recommandations de type G de l'ITU-T qui code souvent une méthode simple à 64 kbit/s offrant une bonne qualité audio, mais divise la taille des données par deux seulement.
- Le codeur ILBC créé par Global IP Sound, approuvé par la standardisation IETF, basé sur un codage synthèse (LPC), a un débit de 13.33 kbit/s avec une longueur de trame d'encodage de 30 ms et atteint un débit 15.20 kbit/s avec une longueur de trame de 20 ms.
- Le codeur SPEEX né du projet OGG, basé sur un codage synthèse (CELP) qui est un codeur libre et sans brevet, est spécialisé et optimisé pour la voix humaine.

Dans ce chapitre, on commencera par traiter quelques notions, telles que les principales méthodes de codage, définir les attributs des codeurs ainsi que les outils de mesure des performances des codeurs, ensuite on présentera les codeurs cités ci-dessus.

III.2 Généralité sur le codage de la parole

III.2.1 Système de compression élémentaire

Les trois opérations successives nécessaires à la numérisation de la parole sont :

- **L'échantillonnage** : qui consiste à prendre des points du signal analogique au fur et à mesure qu'il se déroule. L'échantillonnage du signal analogique à une certaine précision, cette précision étant caractérisée par le nombre de bits utilisés pour coder l'amplitude de chaque échantillon (voir la figure III-1 [6]). Il est clair que le choix de la fréquence et de nombre de bits utilisés répond à un compromis débit/qualité du signal codé. Plus grande est la qualité souhaitée, plus important est le débit obtenu après échantillonnage.

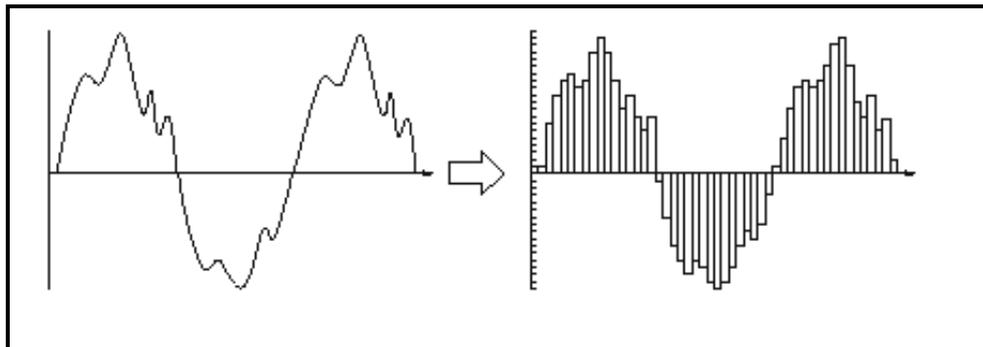


Figure III.1 Processus d'échantillonnage

Le théorème de l'échantillonnage de Nyquist établit qu'un signal analogique peut être reconstruit à partir des échantillons numérisés si la fréquence d'échantillonnage est au moins deux fois la bande passante du signal original. L'oreille humaine est capable de percevoir une gamme de fréquences de 20 Hz à 20 kHz environ. Correspondant à une bande passante de 20 kHz. C'est pourquoi le codage audio dit de haute qualité utilise des fréquences

d'échantillonnages supérieurs à 40 kHz (44,1 kHz pour le codage CD, 48 kHz pour le codage DAT).

- **La quantification** : Il consiste à représenter un échantillon à une valeur numérique au moyen d'une loi de correspondance. Cette phase consiste à déterminer la loi de correspondance de telle sorte que la valeur des signaux est le plus de signification possible, on distingue :

1. La quantification scalaire : Il fait correspondre à chaque paramètre (extrait après analyse) une valeur appartenant à un ensemble fini de nombres.
2. La quantification vectorielle : Elle attribue à un ensemble de paramètres un à vecteur choisi dans un répertoire. La quantification vectorielle est définie comme une méthode de suppression des redondances. Elle utilise quatre propriétés dépendantes des paramètres vectoriels :

- ✓ Dépendance linéaire (corrélation des échantillons)
- ✓ Dépendance non linéaire
- ✓ Forme de la fonction de densité de probabilité
- ✓ Dimension des vecteurs

- **Le codage** : traduit les nombres obtenus après quantification, en séquence de nombres binaires qui seront transmis au décodeur. D'où au final une diminution du débit binaire est une résistance plus forte aux erreurs de transmission. La réalisation d'un système de codage dépend de quatre paramètres :

- Ø Les caractéristiques du signal parole.
- Ø Débit de transmission (débit binaire).
- Ø La qualité de la parole synthétique.
- Ø Le coût du système.

Nous allons présenter les éléments constituant d'un système de compression de parole suivant la figure III-2 :



Figure III.2 Système de compression élémentaire.

- Ø L'élément d'entrée, l'analyseur, analyse le signal de la parole filtré et échantillonné auparavant, la sortie est un vecteur x de paramètres caractéristique de la parole non quantifiée.
- Ø Le codeur quantifie, code le vecteur x en un nouveau vecteur y plus fiable pour la transmission.
- Ø Le canal de transmission transmet le vecteur y des éléments codés.
- Ø Le décodeur à la réception décode le vecteur $Y(n)$ reçu et fournit un ensemble de paramètres dans le vecteur $X(n)$.
- Ø Le synthétiseur utilise ce vecteur de paramètres $X(n)$ pour reconstruire le signal de parole.

III.2.2 Classification des codeurs [6]

L'algorithme de codage le plus simple est celui qui revient seulement à échantillonner un signal analogique et à quantifier les échantillons (c'est-à-dire convertir des valeurs réelles en valeur de précision finie). Ce codage appelé PCM (Pulse Codec Modulation) correspondant à la norme G.711, est utilisé pour coder la voix dans le réseau téléphonique. La bande passante d'une paire torsadée étant d'environ 3.5 kHz, la fréquence d'échantillonnage a donc été fixée à 8 kHz afin de respecter le théorème de Nyquist. La quantification est faite avec une échelle logarithmique sur 8 bits. Le codage PCM est à la base d'une famille de codages différentiels largement utilisés, que nous décrivons dans les titres suivants :

Les codeurs peuvent être classés suivant l'algorithme utilisé, il en résulte trois catégories :

- § Les codeurs différentiels.
- § Les codeurs par synthèse.
- § Les codeurs entropiques.

1. **Codeurs différentiels** : le codage différentiel est basé sur l'observation que les échantillons successifs d'une source audio sont fortement corrélés. Il semble donc judicieux d'encoder non pas les échantillons eux-mêmes, mais la différence entre des échantillons successifs.

Exemple de codeurs différentiel : PDCM (Différential PCM), ADPCM (Adaptive Différential PCM), ADM (Adaptive Différential Modulation).

Principe de codage DPCM : la structure d'un codeur DPCM est donnée à la figure III-3. La variable x représente le signal d'entrée, et r représente la différence entre x et une valeur prédite. Les échantillons successifs étant corrélés, il est possible de prédire l'échantillon $n+1$ à partir des échantillons n , et également $n-1$, $n-2$, etc. On peut donc n'encoder que la différence entre un échantillon X_n et la prédiction de X_n . Dans le cas du codage DPCM, la prédiction de X_n est simplement la valeur de l'échantillon précédant X_{n-1} . Dans un cas plus général, on peut utiliser une prédiction linéaire. On obtient un codage de type ADPCM.

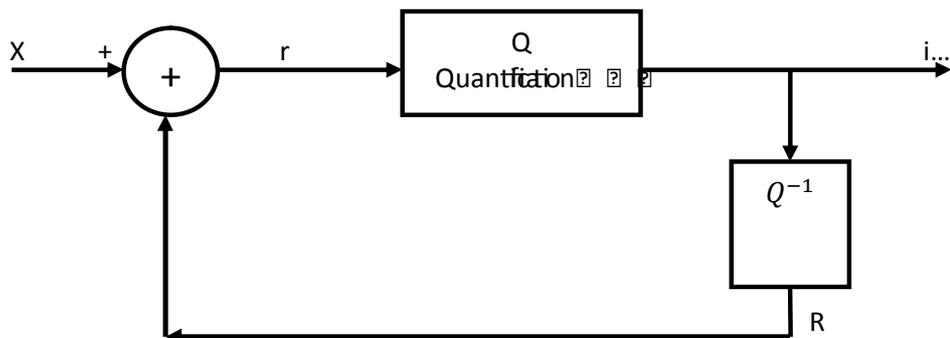


Figure III.3 Principe du DPCM

2. **Codeur par synthèse** : le codage par synthèse est très différent du codage différentiel. L'idée n'est plus de manipuler les échantillons pour éliminer les redondances et corrélations entre échantillons, mais au contraire de considérer des blocs d'échantillons et de construire un modèle qui produit des échantillons statistiquement identiques (ou près) des échantillons

originaux. Ce modèle donc permet de synthétiser des échantillons avec des propriétés statistiques données, d'où le nom de codage par synthèse.

Exemple de codeurs par synthèse : LPC (Linear Predictive Coding), CELP (Code Excited Linear Predictive) et codeur GSM.

Le codage LPC : Le codage LPC (Linear Predictive Coding) consiste à synthétiser des échantillons à partir d'un modèle de système de production vocale et d'une excitation. Pour la voix humaine, le système de production vocale est l'ensemble des poumons-cordes vocales-trachée-gorge-bouche-lèvres. En pratique, on modélise ce système par un ensemble de cylindres de diamètres différents, 10 dans le cas de [LPC-10], excités par un signal qui soit une sinusoïde. Voisées ou non voisée. Un signal voisé correspond à une lettre comme « a » ou « u ». Un signal non voisé correspond à une lettre comme « r » ou « s ».

Le processus de synthèse à deux phases qu'ils sont : l'identification de la fonction d'excitation et l'identification des diamètres, des cylindres (ou de façons plus générales l'identification de la fonction de transfert du modèle vocal). On obtient donc le schéma de la figure III-4.

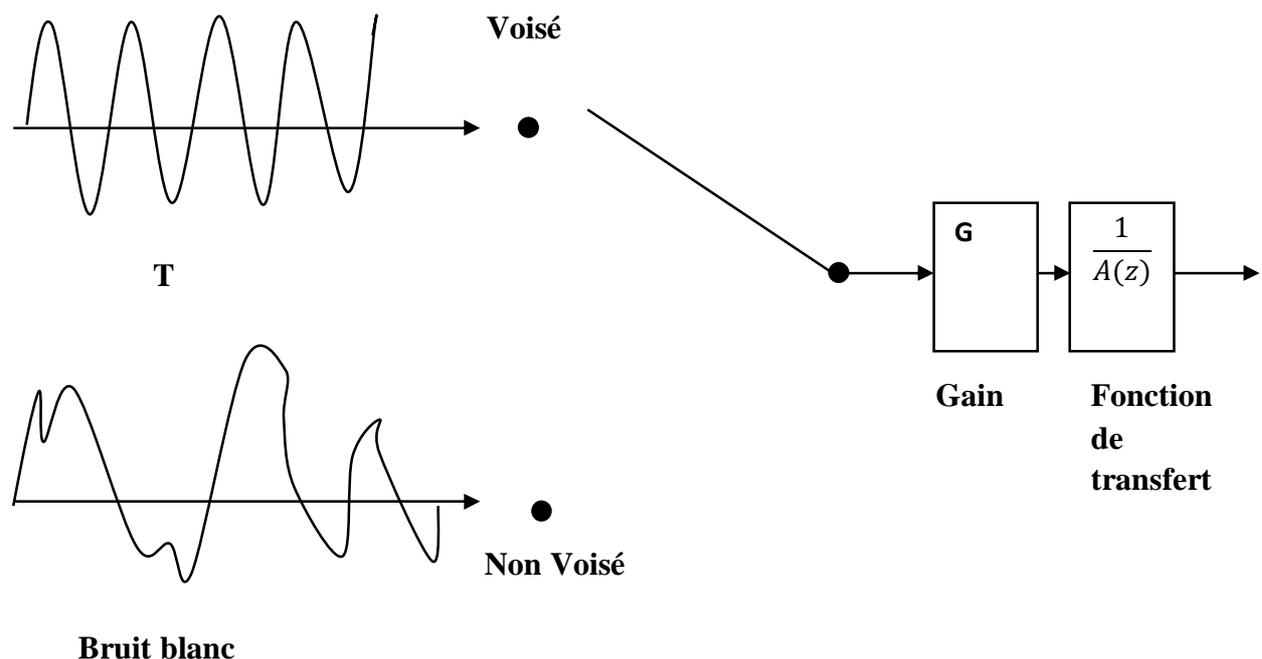


Figure III.4 Principe de LPC

Une fois la décision-signal voisé ou non voisé-prise, il reste à déterminer les coefficients de réflexion du signal. On utilise pour cela l'algorithme de Durbin qui est appliqué au signal filtré afin d'en obtenir une prédiction linéaire qui minimise l'erreur quadratique moyenne définie par [9] :

$$E(A) = \frac{1}{N} \sum e^2(n) \quad \text{Ou} \quad e(n) = x(n) - \sum a_i x(n-i) \quad \text{pour } n=0 \dots N$$

Les a_i obtenus sont les coefficients de la fonction de transfert (aussi appelés coefficients de réflexion). Le gain est obtenu par un simple calcul d'énergie. A ce moment, l'identification est complètement terminée.

Le codage CELP : Le CELP (Code Excited Linear Predictive) correspondant au standard fed_std 1016 est une extension du codage LPC. Il comporte toujours deux phases correspondant aux fonctions d'excitation et de transfert est identique à celle faite avec LPC. Par contre, la fonction de transfert n'est pas seulement un bruit blanc ou une sinusoïde, mais une combinaison linéaire de fonction stochastique (c'est-à-dire de bruit) et périodique. L'identification de ces fonctions est très coûteuse en temps CPU (et d'ailleurs, les codeurs CELP sont généralement implémentés avec l'aide de cartes spécifiques de traitement de signal), mais la qualité obtenue est bien meilleure que celle de LPC.

3. **Codeurs Entropie :** Le codage entropique exploite les propriétés statistiques des coefficients quantifiés pour diminuer le débit de transmission en utilisant des mots courts pour représenter les évènements les plus probables et des mots plus longs pour les occurrences rares.

III.2.3 Evaluation des codeurs

On désire concevoir une application logicielle audio sur un LAN. Il est donc important de comparer et d'évaluer les codeurs décrits dans la section précédente pour une telle application. Trois paramètres semblent importants pour évaluer un codeur, à savoir son débit, le coût CPU nécessaire au codage/décodage et la quantité de l'audio obtenue après codage et décodage.

- **Débit des codeurs** : On définit le débit binaire d'une représentation digitale en bits par échantillon, ou bien bits par seconde notée, exemple : 8 bits par échantillons $f_e = 8$ kHz, alors le débit = 64 kbps. Le tableau ci-dessous résume le débit de différents codeurs [6] :

Codeur	Débit (kbps)
PCM	64
ADPCM3	24
ADPCM2	16
GSM	13,3
LPC	5,6

Tableau III.1 Débit des codeurs

- **Coût CPU des codeurs** : le coût CPU des différents codeurs a été évalué en faisant le codage et décodage d'un même fichier audio contenant des échantillons à 8 kHz d'un dialogue en langue française. Pour chaque codeur le fichier est codé/décodé 100.000 fois en mesurant le temps du début et fin du processus pour ensuite calculer un temps moyen de codage/décodage il faut noter ici que dans le calcul de ce temps moyen, on n'exclut pas le temps où notre processus n'a pas le CPU (entrelacement de processus propre aux systèmes d'exploitation à temps partagé).

Le tableau III-2 montre le coût de codage/décodage en ms d'une trame de 40 ms (soit 320 octets) pour les plateformes SunSparc Ultra à 143 MHz utilisant Solaris 2.5 et pc HP à 33 MHz (Intel) utilisant FreeBSD 2.1.0.

Codeur	SUltra	PC
PCM	0,004/0,003	0,024/0,023
ADPCM2	0,088/0,079	0,983/0,971
ADPCM3	0,094/0,083	1,095/1,022
LPC	1,053/0,824	15,274/13,622
GSM	9,438/2,151	44,414/13,709

Tableau III.2 Coût CPU des codeurs (ms)

Les résultats obtenus en matière de temps de codage et décodage sont très différents selon qu'il s'agit d'un codage de type différentiel ou d'un codage de type synthèse. En effet, les codages différentiels ne nécessitent que quelques opérations algébriques, ainsi que des opérations de permutation de bits et de déplacement dans des tableaux.

Les codages par synthèse au contraire n'effectuent pas moins de 3 calculs d'auto corrélations et 2 calculs de l'algorithme de Dublin, ceci pour chaque fenêtre de 20 ms. Mais vu la puissance des machines actuelles, cela est peu pénalisant pour la VoIP.

III.3 Qualité des codeurs

La qualité subjective d'un codeur, c'est-à-dire la qualité auditive perçue par un utilisateur, est très difficile à évaluer et il n'existe pas de mesure objective fiable de la qualité subjective. On utilise donc une mesure grossière, mais pratique, qui est le rapport signal/bruit SNR (Signal to Noise Ratio).

Le SNR exprimé en décibels (dB) est défini ci-dessous, où X_i est le $i^{\text{ème}}$ échantillon du signal d'entrée, \bar{X}_i est le $i^{\text{ème}}$ échantillon du signal reconstruit [8].

$$SNR = 10 \log_{10} \left(\frac{\frac{1}{N} \sum_{i=0}^{N-1} X_i^2}{\frac{1}{N} \sum_{i=0}^{N-1} (X_i - \bar{X}_i)^2} \right)$$

Dans le cas d'un signal obtenu par un micro-ordinateur et un échantillonnage, la différence entre un échantillon X_i et la valeur du signal de parole au temps i génère un bruit appelé bruit de quantification. On définit σ_x^2 et $\sigma_{B_q}^2$ comme variance du signal et la variance du bruit de quantification. Cette dernière valeur est évaluée entre 30 et 35 dB [8] :

$$\text{Bruit de quantification} = 10 \log_{10} \left(\frac{\sigma_x^2}{\sigma_{B_q}^2} \right) \cong 30 \text{ à } 35 \text{ dB}$$

La valeur, du rapport signal sur bruit, où le signal est maintenant le signal analogique non échantillonné, est donc maintenant [8] :

$$SNR \text{ avec bruit de quantification} = 10 \log_{10} \left(\frac{\sigma_x^2}{\sigma_x^2 + \sigma_{B_q}^2} \right)$$

III.4 Présentation du codeur G.711 [6]

Basée sur une approche de quantification non PCM, G.711 est une norme de compression audio de l'ITU-T simple, point de vue temps calculé et implémentation, basée sur la loi μ (utilisée aux USA, Japon et notamment la conception qu'on fera dans le prochain chapitre) et la loi A (utilisée en Europe). Son principe repose donc sur une grille de quantification non linéaire, permettant de diminuer le rapport signal/bruit de l'erreur de quantification pour les sons de faibles amplitudes.

La norme de compression G.711 est aujourd'hui moins utilisée qu'à ses débuts à cause de sa gourmandise en termes de bande passante sur le réseau, à la faveur d'autres codecs de compression audio. Elle reste tout de même très utilisée notamment chez le grand public.

Caractéristiques

- Échantillonnage : 8000 Hz pour une bande passante du téléphone entre 300 et 3400 Hz.
- Bande passante sur le réseau : 64 ou 56 Kbps
- Type de codage : MIC-Modulation par Impulsion et Codage-

III.5 Présentation de codeur ILBC [6]

ILBC (Internet Low Bitrate Codec) est conçu pour les bandes passantes étroites et a deux débits binaires. Un débit binaire de 13,33 kbps a une longueur de trame d'encodage de 30 ms. ILBC utilise un algorithme d'encodage de type LPC (Linéaire Predictive Coding). Lorsque le codec fonctionne sur des blocs d'encodage de longueur de 20 ms. Il produit 304

bits par bloc. De même pour des blocs de longueur de 30 ms il produit 400 bits par bloc. Les deux modes pour les différentes longueurs de trame opèrent d'une manière similaire.

Le codec ILBC est parfaitement adapté aux communications vocales sur IP. Le codec gère sans problème le cas de perte de trames qui réduisent la qualité de la communication. Ce standard de codec à faible débit binaire, exploite la dépendance entre les trames des paroles. Cette technologie unique offre à l'ILBC une vraie robustesse contre la perte de paquets et les délais.

Certaines des applications pour lesquelles ce codeur est convenable sont : la communication en temps réel (VoIP), la vidéoconférence et streaming audio.

Caractéristiques

- Echantillonnage : 8000 Hz pour une bande passante du téléphone entre 300 et 3400 Hz.
- Bande passante sur le réseau : 13,33 ou 15,20 kbps
- Type de codage : LPC-Linear Predictive coding-

III.6 Présentation de codeur SPEEX [6]

SPEEX est un codec libre de compression audio. SPEEX fait partie de projet GNU et est disponible sous la licence BSD. Le projet SPEEX vise à réduire la barrière d'entrée pour les applications voix, en offrant une alternative gratuite aux coûteux codecs audio des propriétaires. En outre, SPEEX est bien adapté aux applications internet et fournit des fonctionnalités qui ne sont pas présentes dans la plupart des autres codes. Actuellement : Linphone, Astrisk sont quelques-uns des projets utilisant SPEEX.

Caractéristique

- Echantillonnage : 8000 Hz pour une bande passante du téléphone entre 300 et 3400 Hz.
- Bande passante sur le réseau : une bande passante variable de 2 kbps à 44Kbps.
- Type de codage : CELP- Code Excited Linear Predictive-

III.7 Discussion

Objectif de codage est la réduction du volume de l'information pour chaque trame d'entrée au processus de codage tout en gardant une qualité qui satisfait un besoin spécifique. Dans ce chapitre, nous avons introduit le codage de la parole en définissant le domaine et les types des codeurs utilisés dans la conception de l'application VoIP.

IV.1 Préambule

Dans les chapitres précédents, on a décrit les différents protocoles relatifs à la signalisation des appels et au transport de données, ainsi que les méthodes de compression de la voix utilisées pour la réalisation d'une application VoIP.

Dans ce chapitre, on commencera par le choix de la méthode de conception UML (unified modeling language), ensuite, on décrira l'architecture générale de l'application et les différents composants qui la constituent et leur fonctionnement.

IV.2 Choix de la méthode de conception [7] [10]

Parmi plusieurs méthodes de conception qui existent, notre choix s'est reposé sur la méthode UML (unified modeling language) pour les raisons suivantes :

- ✓ UML est un support de communication performant, qui facilite la représentation et la compréhension de solution objet. Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation de solutions.
- ✓ UML comble une lacune importante des technologies objet. Il permet d'exprimer et d'élaborer des modèles objets, indépendamment de tout langage de programmation.
- ✓ UML permet non seulement de représenter et de manipuler les concepts objet, il sous-entend une démarche d'analyse qui permet de concevoir une solution objet de manière itérative, grâce aux diagrammes, qui supportent l'abstraction.
- ✓ Peut-être intégré à n'importe quel processus de développement logiciel de manière transparente.

IV.3 Architecture générale de l'application

La figure IV.1 illustre l'architecture proposée et les interactions entre les différents composants

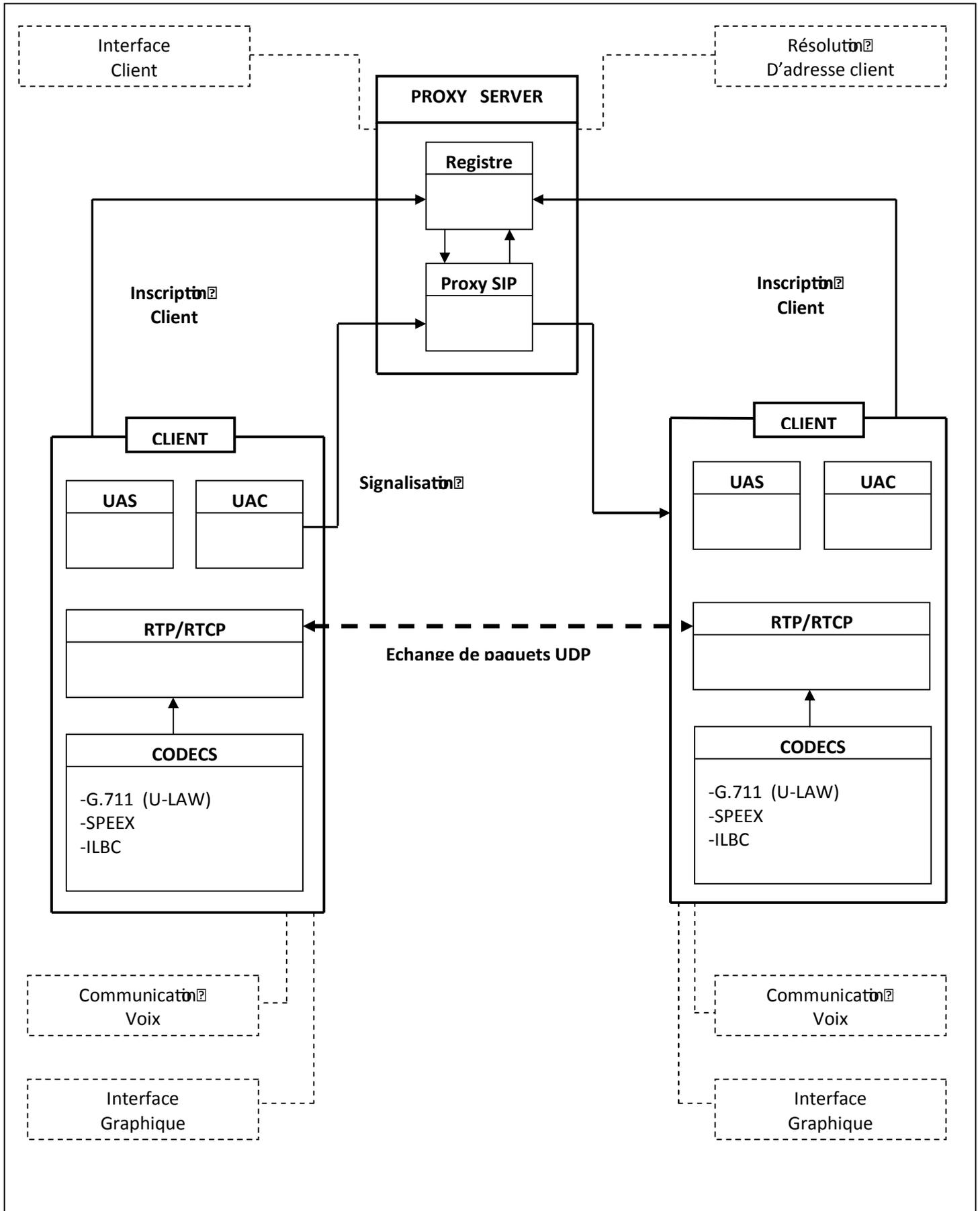


Figure IV.1 Architecture générale de l'application

- ✓ L'entité « **CLIENT** » est conçue pour exécuter une session de conversation sur le protocole internet IP. Elle se charge de la création et de la terminaison d'une conférence, les composants UAC et UAS se chargent de la signalisation SIP entre clients, de plus les composants RTP/RTCP et CODEC se chargent de l'échange audio.
- ✓ L'entité « Proxy Server » quand à elle s'occupe de la gestion des enregistrements des « CLIENT » et permet la résolution des adresses lors de la transmission des requêtes de signalisation. Cependant, cette partie ne fait pas partie du projet.

La figure IV.2 illustre les détails d'un modèle de dialogue entre deux entités « CLIENT » ainsi que l'enregistrement auprès du « Proxy Server ».

- ✓ La connexion UDP1 : cette connexion relie chaque client avec le Proxy Server afin qu'un client connecté puisse établir la communication avec lui.
- ✓ La connexion UDP2 : cette connexion permet de réaliser la signalisation entre deux clients dans le but d'établir et de gérer le dialogue vocal sur un canal UDP.
- ✓ La connexion UDP : permet l'échange de paquets RTP entre les deux clients participants à la connexion UDP2.

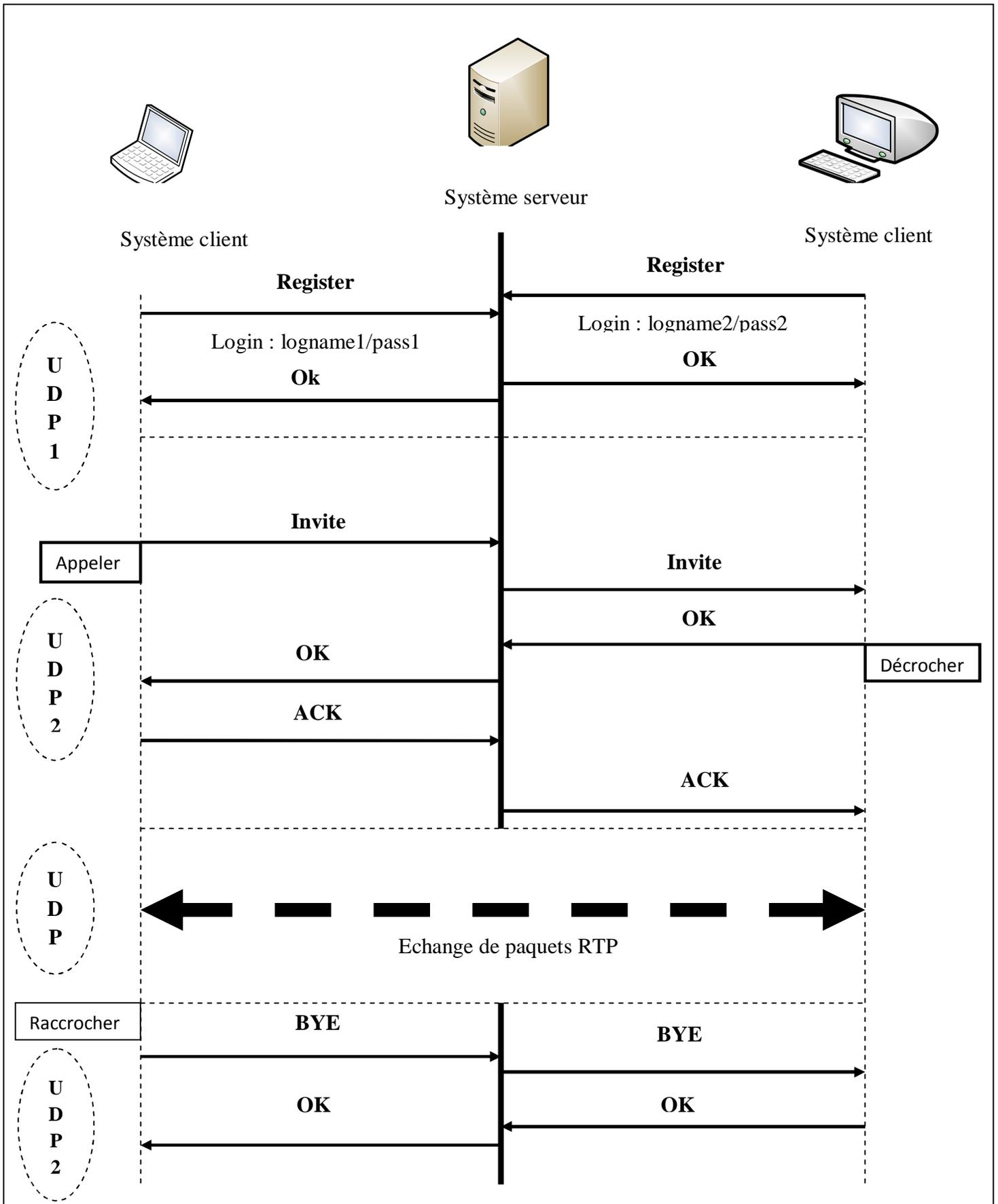


Figure IV.2 modèle de dialogue

IV.4 Etude conceptuelle

On a utilisé le formalisme UML afin de donner une vue générale claire et simple de cette application indépendamment de tout langage de programmation. Pour cela, on va présenter les besoins des utilisateurs et du système sous forme diagrammes de cas d'utilisation, diagrammes d'états ; pour aboutir à un diagramme de classes.

IV.4.1 Spécification des besoins par des cas d'utilisations

Le cas d'utilisation (use case) est la solution pour représenter le modèle conceptuel, correspond à un objectif de système, motivé par un besoin d'un ou plusieurs acteurs. L'ensemble des cas d'utilisation décrit les objectifs du système.

La spécification des besoins définit le contour du système à modéliser et identifie les fonctionnalités principales de système.

Les éléments de base des cas d'utilisation :

- ✓ Acteur : entité externe qui agit sur le système (opérateur, autre système).
 - § L'acteur peut consulter ou modifier l'état du système.
 - § En réponse à l'action d'un acteur, le système fournit en retour un service correspondant à l'action de l'acteur.
- ✓ Un cas d'utilisation : action réalisée par le système fournie en réponse à une action d'un acteur.
 - § Les cas d'utilisation peuvent être structurés et organisés en paquetages. Dans notre cas, nous allons définir cas d'utilisations utilisateur et système.

Vue utilisateur :

Suivant le formalisme UML, la figure IV.3 représente l'ensemble des cas d'utilisation de l'utilisateur client et notre architecture.

Utilisateur Client

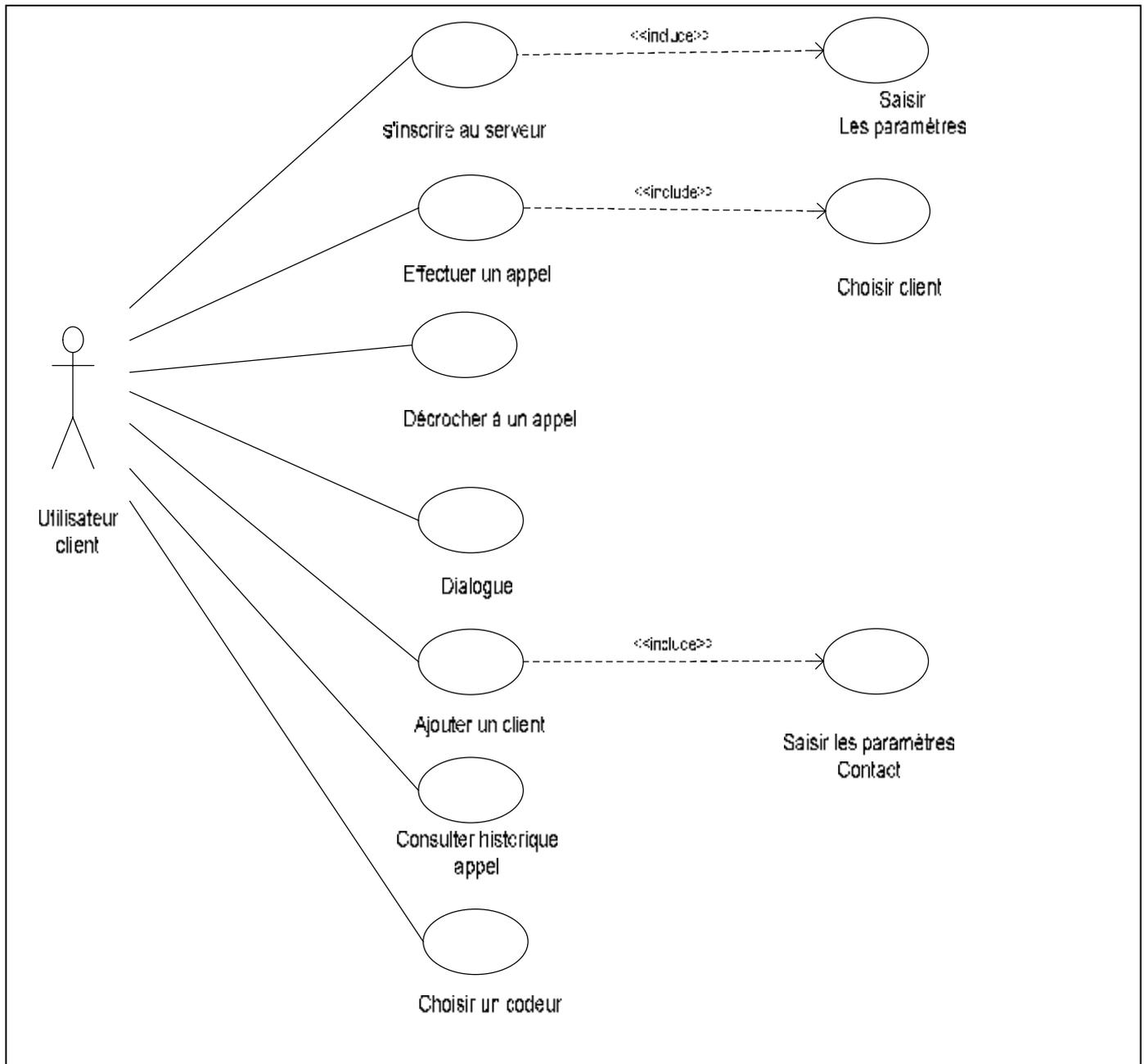


Figure IV.3 Diagramme de cas d'utilisation d'un client

Cas d'utilisation : S'inscrire au serveur

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur demande l'inscription.

Scénario de succès : l'utilisateur s'inscrit au serveur.

Ce cas d'utilisation inclut le cas d'utilisateur « saisir les paramètres ».

Cas d'utilisation : Saisir les paramètres

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur saisit les paramètres.

Scénario de succès : l'utilisateur saisit les paramètres de connexion au serveur SIP.

Cas d'utilisation : Effectuer un appel

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur effectue un appel vers un autre client.

Scénario de succès :

- l'utilisateur effectue un appel.
- L'utilisateur reçoit la réponse du client auquel il a demandé la connexion.

Ce cas d'utilisation inclut le cas d'utilisateur « choisir le client ».

Cas d'utilisation : Choisir le client

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur choisit un client dans la liste client.

Scénario de succès : l'utilisateur choisit le client.

Cas d'utilisation : décrocher à un appel

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur accepte la connexion entrante.

Scénario de succès :

- Le système notifie l'utilisateur d'une demande de connexion.

- L'utilisateur répond à la demande de connexion.

Cas d'utilisation : Dialoguer

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur commence le dialogue.

Scénario de succès : l'utilisateur dialogue avec le client qui accepte sa demande de connexion.

Cas d'utilisation : Ajouter un client

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur demande d'ajouter un client.

Scénario de succès : l'utilisateur ajoute un client à sa liste de contacts.

Ce cas d'utilisation inclut le cas d'utilisation « Saisir les paramètres de contact ».

Cas d'utilisation : saisir les paramètres contact

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur saisit les paramètres d'un contact.

Scénario de succès : l'utilisateur saisit : Nom-Prénom-Adresse SIP (URL).

Cas d'utilisation : Consulter historique appel

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur consulte l'historique des appels.

Scénario de succès : Affichage de l'historique des appels.

Cas d'utilisation : Choisir Codec

Description :

Acteur : Utilisateur Client.

Événement déclencheur : l'utilisateur choisit un type de codage parmi ceux existants.

Scénario de succès : un codec est choisi pour l'encodage du message.

Vue système**Paquetages des cas d'utilisation système**

Le paquetage est un mécanisme général de regroupement d'éléments en UML, qui est principalement utilisé en analyse et conception objet pour regrouper des classes et des associations. Les package sont des espaces de noms : deux élément ne peuvent pas porter le même nom au sein de même package. Par contre, deux éléments appartenant à des packages différents peuvent porter le même nom. La structuration d'un modèle en packages est une activité délicate. Elle doit s'appuyer sur deux principes fondamentaux : cohérence et indépendance.

Les cas d'utilisation système sans regrouper dans deux packages pour donner une vue globale de l'application, ensuite on détaillera ces cas pour chaque paquetage.

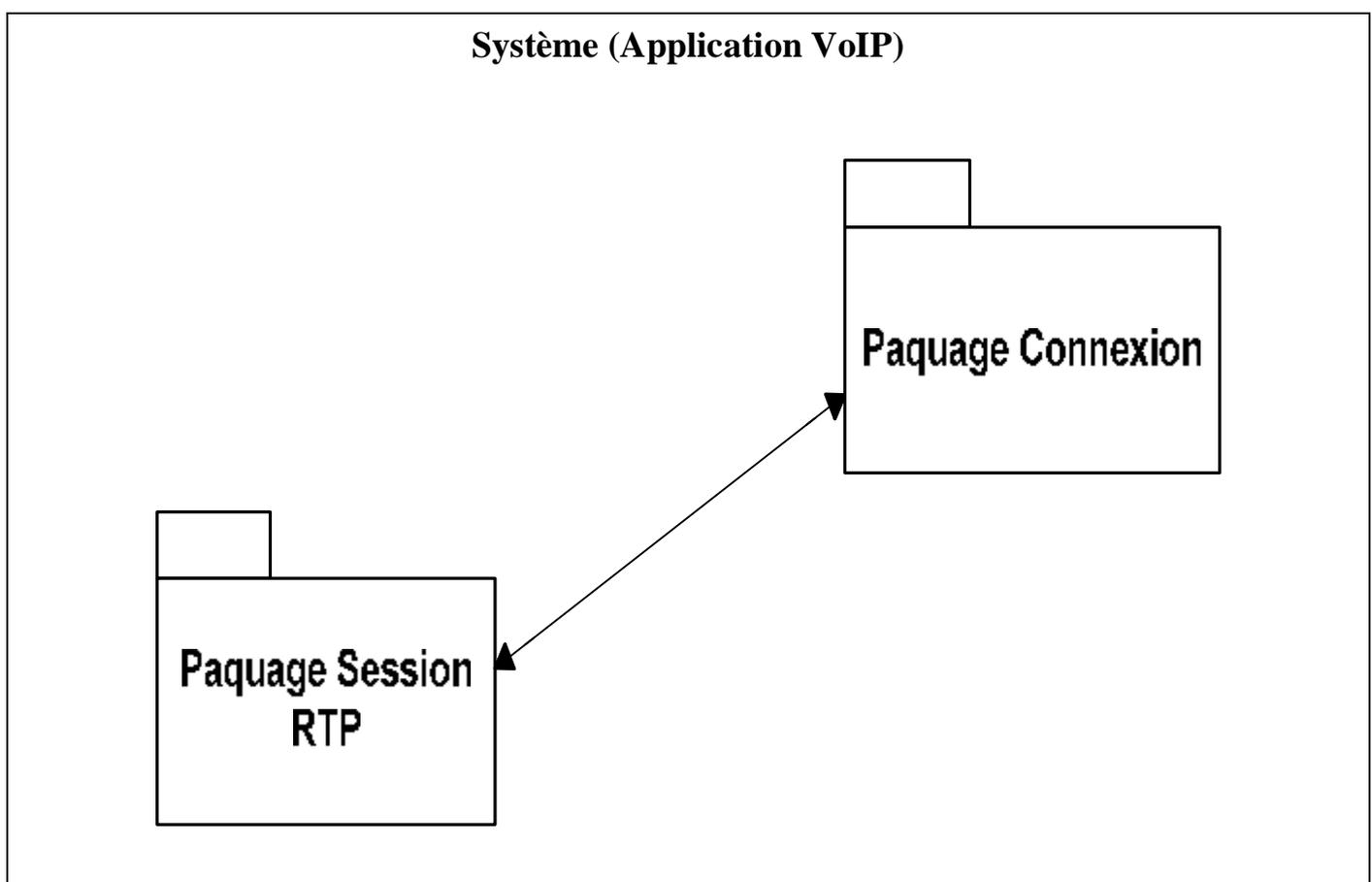


Figure IV.4 Paquetage de haut niveau des cas d'utilisation vue système

Paquetage Session RTP

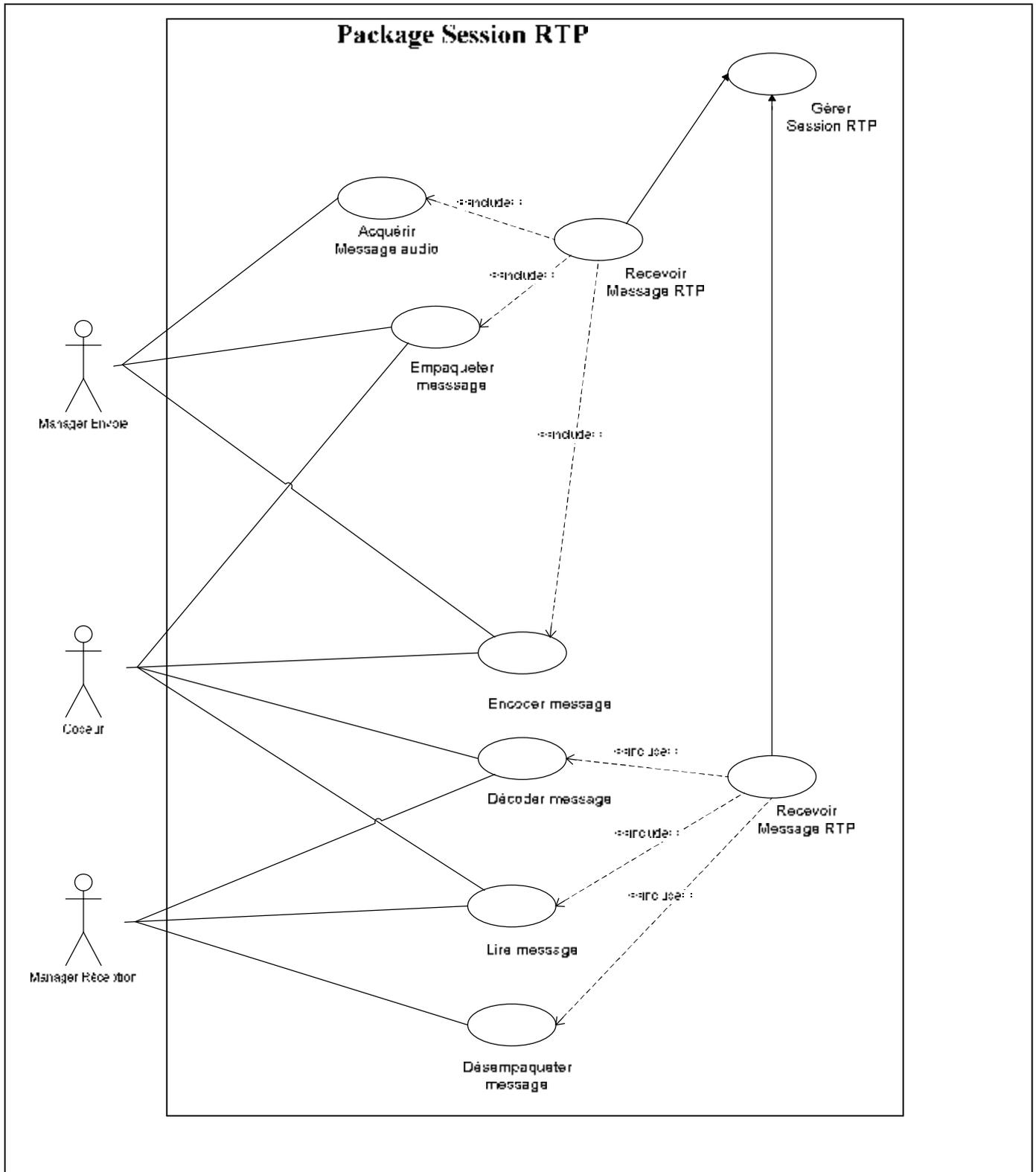


Figure IV.5 Diagramme de cas d'utilisation du paquetage Session RTP

Cas d'utilisation : Envoie message RTP

Description :

Acteur : système.

Événement déclencheur : message prêt à transmettre.

Scénario de succès :

- Le système récupère le message à transmettre.
- Le système transmet le message.

Ce cas d'utilisation inclus le cas d'utilisation « Acquérir message audio », « Encoder message », « Empaqueter message ».

Cas d'utilisation : Acquérir message audio

Description :

Acteur : ManagerEnvoie.

Événement déclencheur : l'enregistreur du ManagerEnvoie lance l'acquisition de message.

Scénario de succès :

- L'enregistreur demande les ressources système nécessaires à l'acquisition.
- Le système alloue à l'enregistreur ces ressources.
- ManagerEnvoie lance l'acquisition.

Cas d'utilisation : EncoderMessage

Description :

Acteur : Encodeur, ManagerEnvoie.

Événement déclencheur : Le ManagerEnvoie appelle l'encodeur choisit, à la fin de l'acquisition du message.

Scénario de succès :

- Passage du message à coder à l'Encodeur.
- L'Encodeur encode le message.

Cas d'utilisation : empaquetage message

Description :

Acteur : ManagerEnvoie, Codeur.

Événement déclencheur : ManagerEnvoie appelle l'empaqueteur à la fin du codage du message.

Scénario de succès :

- L'encodeur passe le message codé au ManagerEnvoie.
- Le ManagerEnvoie empaquette le message encodé.

Cas d'utilisation : recevoir message RTP

Description :

Acteur : système.

Événement déclencheur : le système est notifié d'une réception.

Scénario de succès : le système récupère le message reçu.

Ce cas d'utilisation inclut le cas d'utilisation « Déempaqueter message », « Décoder message » et « lire message ».

Cas d'utilisation : Déempaqueter message

Description :

Acteur : ManagerRéception.

Événement déclencheur : le système appelle ManagerRéception pour la dépaquetisation après réception du message RTP.

Scénario de succès : ManagerRéception dépaquetise le paquet reçu.

Cas d'utilisation : Décoder message

Description :

Acteur : Décodeur, ManagerRéception.

Événement déclencheur : le ManagerRéception appelle le décodeur à la fin de l'acquisition de message.

Scénario de succès :

Package Connexion

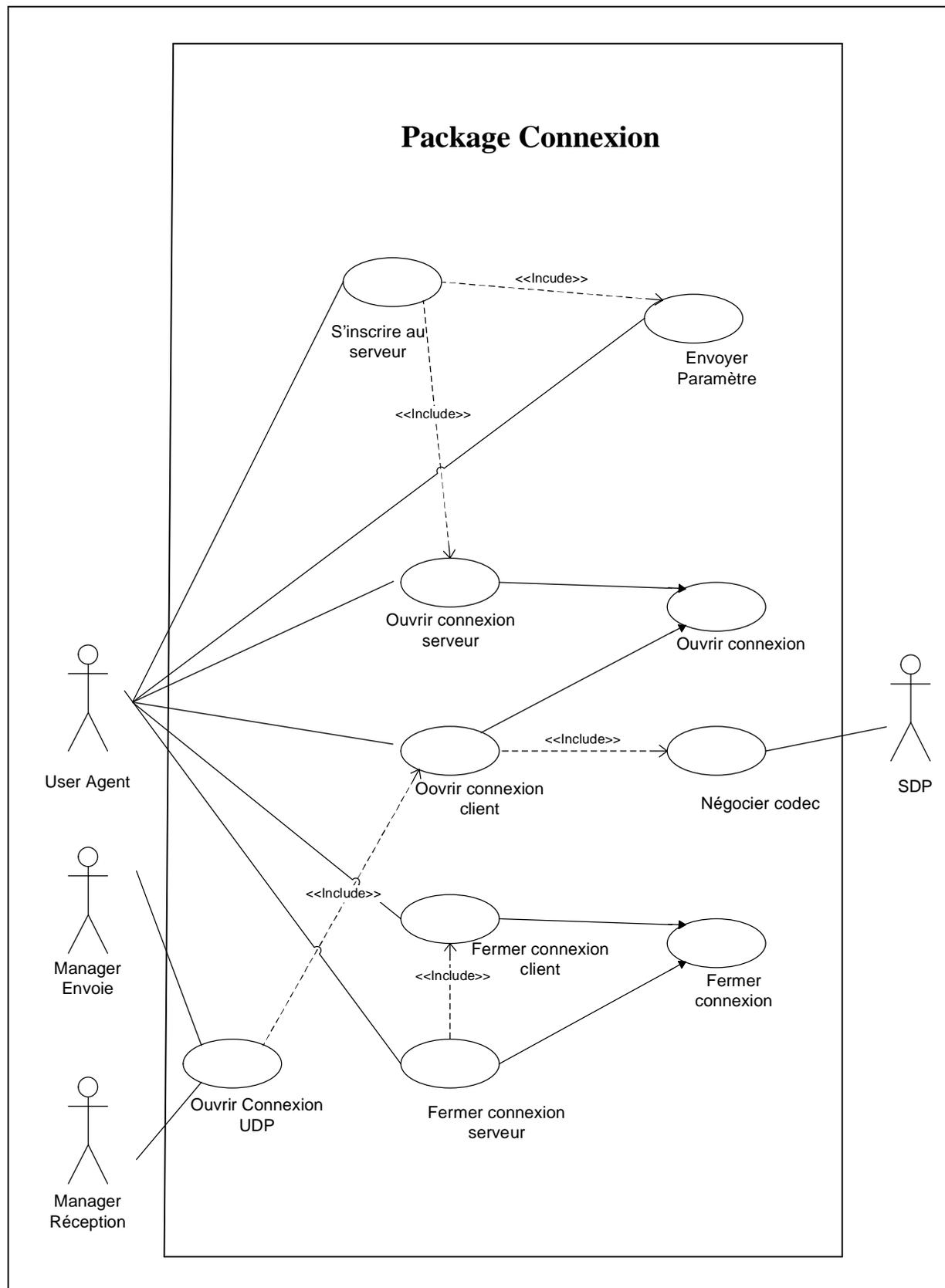


Figure IV.6 Diagramme de cas d'utilisation du paquetage connexion

Cas d'utilisation : inscrire au serveur

Description :

Acteur : UserAgent.

Événement déclencheur : UserAgent inscrit le système auprès du serveur.

Scénario de succès :

- UserAgent envoie les paramètres au serveur
- Connexion établie au serveur

Cas d'utilisation inclut le cas d'utilisation « Envoyer paramètres » et « Ouvrir connexion serveur ».

Cas d'utilisation : Envoyer paramètres

Description :

Acteur : UserAgent.

Événement déclencheur : UserAgent envoie les paramètres au serveur.

Scénario de succès : UserAgent envoie les paramètres requis par le serveur.

Cas d'utilisation : Ouvrir connexion serveur

Description :

Acteur : UserAgent.

Événement déclencheur : UserAgent ouvre une connexion avec le serveur.

Scénario de succès :

- UserAgent demande la connexion au serveur.
- Le serveur enregistre la connexion.

Cas d'utilisation : Ouvrir connexion Client

Description :

Acteur : UserAgent.

Événement déclencheur : UserAgent ouvre la connexion avec un client.

Scénario de succès :

- UserAgent ouvre une connexion.
- Lancement du dialogue après notification de l'acceptation d'UserAgent.

Ce cas d'utilisation inclut le cas d'utilisation « Négocier codec ».

Les cas « Ouvrir connexion serveur » et « Ouvrir connexion Client » se généralisent au cas « Ouvrir connexion ».

Cas d'utilisation : Fermer connexion serveur

Description :

Acteur : UserAgent.

Événement déclencheur : UserAgent ferme la connexion avec le serveur.

Scénario de succès :

- UserAgent ferme la connexion avec le serveur.
- Le serveur enregistre la déconnexion.

Ce cas d'utilisation inclut le cas d'utilisation « Fermer connexion client ».

Cas d'utilisation : Fermer connexion client

Description :

Acteur : UserAgent.

Événement déclencheur : UserAgent ferme la connexion avec le client.

Scénario de succès : UserAgent ferme la connexion avec le client.

Les cas « Fermer connexion serveur » et « Fermer connexion client » se généralisent au cas « Fermer connexion ».

Cas d'utilisation : Ouvrir connexion UDP

Description :

Acteur : ManagerEnvoie, ManagerRéception.

Événement déclencheur : ouverture d'une connexion UDP avec un client.

Scénario de succès : Début d'échange RTP entre client.

Ce cas d'utilisation inclut le cas d'utilisation « Ouvrir connexion client ».

Cas d'utilisation : Négocier codec

Description :

Acteur : SDP.

Événement déclencheur : SDP négocier le codec en commun à utiliser.

Scénario de succès : paramétrage des clients sur un type de codec.

IV.4 .2 Spécification des besoins par des états transitions

Ils ont pour rôle de représenter les traitements (opérations) qui vont gérer le domaine étudié. Ils définissent l'enchaînement des états de classe et font donc apparaître l'ordonnement des travaux.

Le diagramme d'états transition est associé à une classe pour laquelle on gère différents états : il permet de représenter tous les états possibles ainsi que les événements qui provoquent les changements d'états.

Les éléments de base des états transition

- ✓ Etat : Un état correspond à une situation durable dans laquelle se trouvent les objets d'une classe. On lui associe les règles de gestion et les activités particulières.
- ✓ Evénements et transitions : Un objet passe d'un état à un autre suite à des événements. Une transition est que l'état 2 est possible si certains événements sont vérifiés.

Afin de mieux comprendre le fonctionnement SIP de l'application pour l'établissement de connexion avec le client et l'établissement de connexion avec proxy SIP on a réalisé deux diagrammes d'états pour l>UserAgentClient (UAC) et l>UserAgentServer (UAS).

REGISTER : l'UAC transmet cette requête permettant à l'application de s'enregistrer auprès d'un serveur SIP. Peut se faire avec ou sans authentification selon le Registrar du Serveur SIP.

- Un serveur SIP peut exiger une authentification de type WWW (Word Wide Web), le message SIP sera envoyé avec un nouveau champ dans lequel aura été générée un message haché grâce à MD5 (Message Digest 5).

P.A.R (Proxy_Authentication_Required): réponse envoyée par le registrar indiquant qu'il exige une authentification.

OK : cette réponse est reçue après un enregistrement avec succès auprès du serveur SIP.

Connexion de l'application au client SIP (établissement de l'appel)

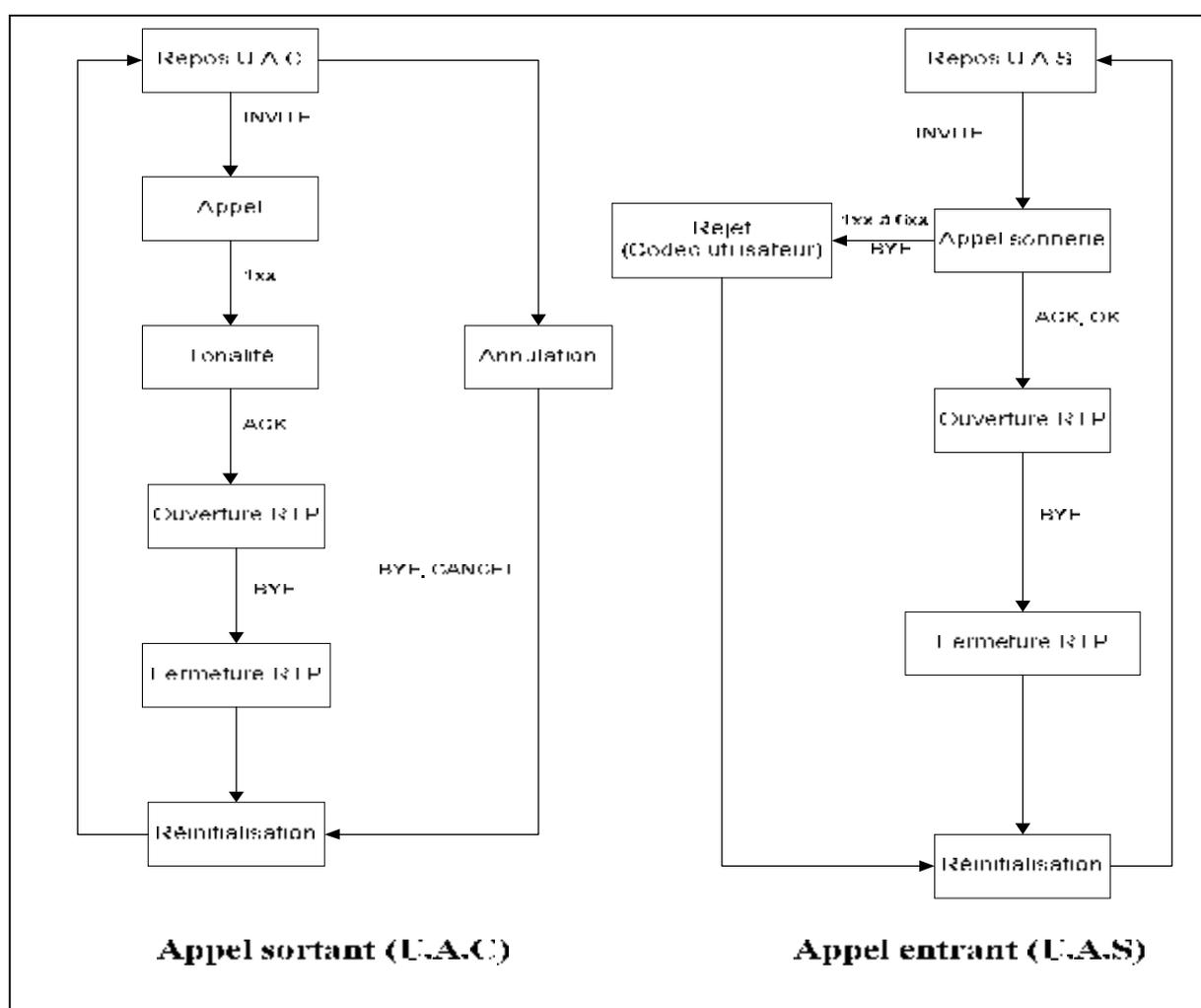


Figure IV.8 Diagramme d'états de l'U.A.C et de l'U.A.S lors de l'établissement d'un appel

Les requêtes et réponses SIP déclenchant les événements qui provoquent un changement d'états de l'UAC et l'UAS sont :

INVITE : envoyé par l'UAC de l'appelant, permettant d'appeler un utilisateur grâce à son URL SIP, pour être reçu par l'UAS de l'appelé, indiquant qu'un utilisateur essaye de le joindre. La requête INVITE contient un message SDP (Session Description Protocol) qui est chargé de décrire l'IP, le port utilisé pour la réception RTP ainsi qu'une liste de codecs (audio) présente dans l'application.

1xx : message notifiant que la requête INVITE a été reçu et continue d'être traitée.

OK : envoyée en réponse pour INVITE, CANCEL ou REGISTER.

ACK : cette requête est envoyée par l'UAC après réception de réponse OK à une requête INVITE. ACK est reçu par l'UAS. Et marque le début de la réception RTP.

BYE : cette requête marque la fin d'un appel (donc la réinitialisation de l'UAC et l'UAS ainsi que la fermeture des flux RTP).

CANCEL : cette requête permet d'annuler une requête INVITE précédemment envoyée. (Exemple : lorsque l'utilisateur se rend compte qu'il appelle le mauvais utilisateur).

IV.4.3 Identification de classes

D'après la spécification des besoins précédente schématisée par des diagrammes, nous avons un certain nombre d'acteurs principaux et secondaires, les différents cas de figures nous ont permis d'aboutir par la suite à identifier les classes candidates suivantes :

GUIUtilisateur : c'est la classe qui représente la fenêtre principale de l'interface graphique. Elle permet notamment d'instancier la classe UserAgent.

GUICompte : cette classe représente une nouvelle fenêtre permettant de paramétrer le compte d'utilisateur de l'application.

GUINContact : cette classe représente une nouvelle fenêtre permettant d'ajouter un nouveau contact à sa liste de contacts (Utilisateurs appelés).

GUIContacts : cette classe représente une nouvelle fenêtre permettant d'afficher une liste de contacts cliquables (à appeler).

GUIHistorique : cette classe permet d'afficher une fenêtre contenant l'historique des appels effectués et reçus par l'utilisateur.

InterfaceHistorique : cette classe permet de sauvegarder et charger l'historique des appels dans un fichier.

interfaceCompte : cette classe permet de sauvegarder et charger les paramètres du compte d'un utilisateur dans un fichier.

UserAgent : cette classe représente l'UserAgent de l'application (UA). Elle est le point de réception et de départ des messages SIP et est chargée de déclencher les méthodes adéquates selon les requêtes ou réponses reçues.

UserAgentClient : c'est la classe qui implémente l'UAC de l'application. Permet de produire et envoyer les requêtes SIP telles que les messages : INVITE, REGISTER, BYE, CANCEL. Elle instancie aussi les objets permettant l'émission et la réception des flux RTP.

UserAgentServer : c'est la classe qui implémente l'UAS de l'application. Cette class permet de traiter et envoyer les réponses aux requêtes SIP reçues. Elle permet aussi l'analyse le message SDP (chois des codecs) et instancie les objets permettant l'émission et la réception des flux RTP.

SDP (Session Description Protocol) : cette classe permet de traiter les messages SDP : extraire des paramètres, codecs.

MD5 (Message Digest 5) : implémente l'authentification en utilisant l'algorithme de hachage MD5.

ManagerEnvoie : implémente l'émission d'un flux RTP à travers une capture audio.

ManagerReception : implémente l'émission d'un RTP, et la recréation donc du flux de sortie.

Codeur : cette classe permet de charger les encodeurs et décodeurs (ILBC, SPEEX et G.711).

Encodeur : permet d'effectuer l'encodage audio.

Décodeur : permet d'effectuer le décodage audio.

IV.4.4 Identification et documentation des relations entre les classes

Les relations modélisées sont de deux types, on trouve des relations binaires (simples ou réflexives) entre deux classes (acteurs du système), mais aussi des relations n-aires (généralement ternaires), mettant en jeu plus de deux classes (acteurs), des relations de ce genre sont appelées classes d'association dans une modélisation UML.

Ces relations expriment aussi bien l'aspect connexion (communication) entre les objets du domaine, que les relations de transfert et traitement des messages audio entre ces mêmes objets.

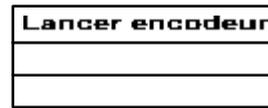
1. Initier connexion (relation binaire) : exprime l'ouverture d'une connexion entre deux objets à travers une requête.
2. Annuler connexion (relation binaire) : exprime la fermeture d'une connexion entre deux objets à travers une requête.
3. Accepter connexion (relation binaire) : exprime l'acceptation d'établissement d'une connexion entre deux objets suite à une requête.
4. Refuser connexion (relation binaire) : exprime le refus d'établissement d'une connexion entre deux objets.
5. Demander connexion (relation binaire) : exprime la demande d'établissement d'une connexion entre deux objets à travers une requête.
6. Lancer acquisition et empaquetage (relation binaire) : représente le lancement de l'acquisition de la voix, et sa transformation en paquet RTP après encodage.
7. Lancer déempaquetage et lecture (relation binaire) représentent le début de la transformation du paquet reçu en message codé, et sa lecture après décodage.
8. Lancer encodage (classe d'association) : représente le début du codage du message vocal acquis précédemment.
9. Lancer décodage (classe d'association) : représente le début de la transformation du message issu du déempaquetage du format codé vers le format original.

Cet ensemble de relations (associations) entre acteurs du domaine conduit à un diagramme de classes représentant les classes de système étudié, et qui donne une vue sur l'application.

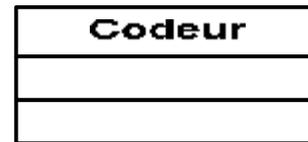
IV.4.5 Diagramme de classes

Légende :

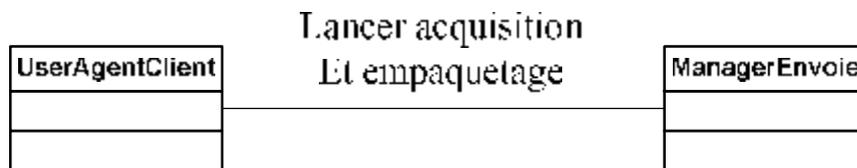
- Classe d'association (lancer encodage)



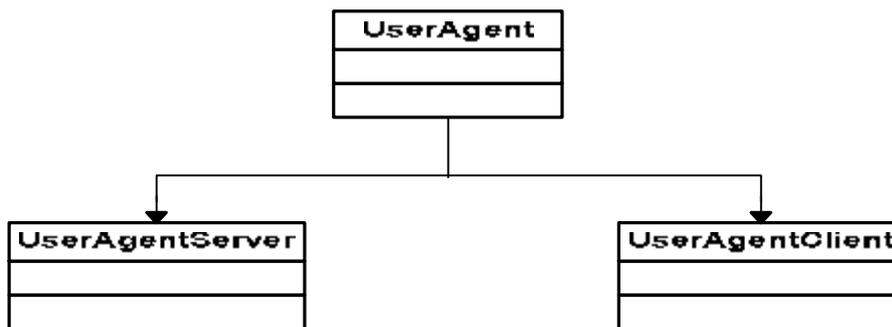
- Classe codeur



- Association (lancer acquisition et empaquetage)



- Association de spécialisation multiple (spécialisation de l'UA)



- Association d'utilisation (appelle)



IV.4.6 Déroulement de l'application

Le déroulement va mettre en évidence le séquençement de cette application qui va suivre les étapes suivantes dans son exécution. Soit le scénario suivant qui met en relation deux systèmes clients (appelant et appelé) :

1. Les deux clients saisissent les paramètres de connexion (GUICompte, InterfaceCompte : Login, mot de passe, domaine ou adresse IP du serveur SIP).
2. Les deux clients lancent une connexion vers le serveur SIP (UserAgent : envoie d'une requête REGISTER).
3. Le système serveur SIP accepte les connexions entrantes, les clients sont donc enregistrés auprès du serveur SIP.
4. Chaque client saisit les paramètres de l'autre client dans sa liste de contact (GUINContact : saisir le nom et URL SIP du contact).
5. L'appelant choisit l'appelé dans la liste des contacts et lance une demande de connexion. (GUIContacts : listes des contacts, UserAgent : envoie d'une requête INVITE).
6. L'appelé quand à lui reçoit la demande de connexion faite par l'appelant (UserAgent : reçoit la requête INVITE) dans le cas d'une acceptation de processus suivant se déclenche :
 - a) décrocher à l'appel entrant (UserAgent : envoie une réponse OK à l'appelant).
 - b) Ouverture d'une Session RTP au niveau de l'appelé (ManagerEnvoie et ManagerRéception).
7. L'appelant reçoit la confirmation d'acceptation de connexion, le processus suivant se déclenche :
 - a) Ouverture d'une session RTP au niveau de l'appelant.
 - b) Notification du début de la connexion RTP (UserAgent : envoie d'une requête ACK à l'appelé).
8. Durant le dialogue, le processus suivant se déroule :
 - a) L'enregistreur lance l'acquisition du message vocal et le transmet au codeur.
 - b) Le Codeur encode le message et le transmet à l'empaqueteur.
 - c) L'empaqueteur met en paquets RTP le message codé, le paquet est alors prêt à l'envoi.
 - d) ManagerEnvoie envoie à l'appelé le paquet RTP via le canal UDP.

- e) A la réception, le paquet reçu est désempaqueté, il en résulte un message transmis au codeur.
 - f) Le Codeur décode le message et l'envoie à la lecture.
 - g) Le lecteur lance la lecture du message décodé.
9. Lorsque l'un des deux systèmes clients désire mettre fin au dialogue, il lance alors la fermeture de connexion client (UserAgent : envoi d'une requete BYE).

IV.5 Discussion

Dans ce chapitre, on a défini l'architecture générale de l'application, puis l'identification des différents diagrammes et classes avec le formalisme UML, afin de donner une vue simple et générale de l'application.

Actuellement, il est évident que la voix sur IP va continuer de se développer dans les prochaines années. Le marché de la voix sur IP est très jeune, mais se développe à une vitesse fulgurante. C'est aujourd'hui que les entreprises doivent investir dans la voix sur IP si elles veulent y jouer un rôle majeur. Le fait que IP est maintenant un protocole très répandu, et a fait ses preuves et que beaucoup d'entreprises disposent des avantages de la téléphonie IP, car elle demande un investissement relativement faible pour son déploiement.

A travers ce travail, nous avons fait une conception d'une application voix sur IP, en faisant le transfert de la voix via le protocole RTP, adoptant la technique de signalisation SIP.

Notre travail s'est fait en deux étapes successives et complémentaires : état de l'art et conception.

La première s'occupait d'éclaircir et de définir les principes généraux de la voix sur IP, les protocoles de transport, les standards de signalisation ainsi que les algorithmes de codage de la parole adéquats pour mener à bien la conception aussi complexe que le transfert de la voix en temps réel.

La seconde étape avait pour but de donner un modèle UML décrivant entièrement l'architecture de l'application suivant les besoins spécifiques aux utilisateurs et au système.

On peut ainsi vraisemblablement penser que le protocole IP deviendra un jour un standard unique permettant l'interopérabilité des réseaux mondialisés. C'est pourquoi l'intégration de la voix sur IP n'est qu'une étape vers EoIP : Everything over IP.

On propose en perspective pour notre travail, la réalisation de cette application avec un langage de programmation orienté objet tel que le Java, d'ajouter un module de conférence vidéo et d'intégrer la messagerie instantanée dans le SIP afin de concevoir un système de communication complet.

ARP	Address Resolution Protocol
ADM	Adaptive Delta Modulation
ADPCM	Adaptive Differential Pulse Code Modulation
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CELP	Code Exited Linear Predective
CoS	Class of Service
DNS	Domain Name System
DTMF	Dual Tone Multi Frequency
FTP	File Transfert Protocol
GSM	Globel System for Mobile
HTTP	Hyper Text Transfert Protocol
ICMP	Internet Control Message Protocol
ITU-T	International Telecommunication Union-Telecommunication sector
IEEE	Institute of Electrical and Electronics Engineers
ILBC	Internet Low Britate Codec
IETF	Internet Engineering Task Force
LAN	Local Area Network
Log-PCM	Logarithmic PCM
LPC	Linear Prediction Coding
MPLS	Multi Protocol Label Switching
MPEG	Moving Pictures Expert Group
MCU	Multipoint Control Unit
NT	New Technology
OSI	Open systems Interconnection
PCM	Pulse Code Modulation
PABX	Private Automatic Branch eXchange
PDCM	Differential PCM
POP	Post Office Protocol
QoS	Qualité of Services
RTP	Reel Time Protocol
RTCP	Reel Time Control Protocol
RNIS	Réseau Numérique à Intégration de Services (ISDN en anglais)
RAS	Registration Admission Status
RSVP	Resource reSerVation Protocol

RTC	Réseau téléphonique Commuté
SIP	Session Internet Protocol
SDH	Synchronous Digital Hierarchy
SAP	Session Advertisement Protocol
SMTP	Simple Mail Transfert Protocol
SNR	Signal to Noise Ration
SDP	Session Description Protocol
TCP/IP	Transmission Control protocol/ Internet Protocol
ToIP	Telephone Over Internet Protocol
UDP	User Datagram Protocol
UML	Unified Modeling Language
URL	Uniform Ressource Locator
UTF	UCS Transformation Format
UAS	User Agent Service
UAC	User Agent Client
VoIP	Voce Over Internet Protocol

[1] **Ali ELGHATTAS**, « *Architecture et fonction d'un réseau informatique* », 2008.

Lien : www.tripalais.on.ma

[2] **GUY Pujolle** « *les réseaux* ».Éditions Eyrolles, 5^e édition, 2006.

[3] Site internet : www.framip.com

[4] Site internet : www.lyceefaurcade.fr.fm

[5] **GUY Pujolle and Laurent Ouakil**, « *Téléphonie sur IP* ». 2^e Édition Eyrolles, 2008.

[6] Site internet : www.sop.inria.fr/rodeo/avega/phd/phd-html/node8.html. INRIA.

[7] **CHRISTIAN Soutou** « *UML2 pour les bases de données* ».Éditions Eyrolles, 2007.

[8] **N. S. Jayant**, « *P. Noll Digital Coding of Waweforms* ». Edition Prentice Hall, 1989.

[9] **L. R. Rabiner, R. W. Schafer** « *Digital Processing of Speech Signal* ». Edition Prentice-Hall, Englewood Cliffs, N.J., 1978.

[10] **OULAGHA Amine**, « *l'implémentation d'une plateforme pour une application Voice over IP* ».UMMTO, département d'informatique, Promotion 2008.