



ÉCOLE DOCTORALE DE MATHÉMATIQUES ET  
INFORMATIQUE

ÉCOLE SUPÉRIEURE POLYTECHNIQUE

Année : 2015

N ° d'ordre :

# THÈSE DE DOCTORAT

Spécialité : Télécommunications

Présentée par :

Samuel OUYA

Titre : **ÉTUDE DE LA CONVERGENCE DES SERVICES DE  
TÉLÉCOMMUNICATIONS ET SES APPLICATIONS AUX  
ORGANISATIONS VIRTUELLES**

Soutenue le 08 janvier 2016 devant le jury composé de :

M. Hamidou DATHE	Professeur, FST-UCAD, Dakar, Sénégal	Président
M. Ousmane THIARE	Professeur, UGB, Saint-Louis, Sénégal	Rapporteur
M. Roger Marcelin FAYE	Professeur, ESP-UCAD, Dakar, Sénégal	Rapporteur
M. Michel BABRI	Maître de Conférences, INPHB, Côte d'Ivoire	Rapporteur
M. Moussa LO	Professeur, UGB, Saint-Louis, Sénégal	Examineur
M. Claude LISHOU	Professeur, ESP-UCAD, Dakar, Sénégal	Directeur de thèse
M. Mary Teuw NIANE	Professeur, UGB, Saint-Louis, Sénégal	Directeur de thèse

## Dédicaces

Je dédie ce travail à tous les mathématiciens et à tous ceux qui œuvrent pour l'émergence de l'Afrique.

## Remerciements

Merci Pr. Mary Teuw NIANE pour les verbes « contrôler, identifier, reconnaître, et converger » que vous m'avez appris à conjuguer depuis mon DEA, en passant par ma thèse de Mathématiques et maintenant cette présente thèse.

J'adresse mes sincères remerciements au Pr. Claude LISHOU pour son soutien tout au long de ces trois années de thèse. Je ne saurais dire combien nos échanges et ses nombreux conseils et idées m'ont été précieux.

Je remercie sincèrement le Pr. Hamidou DATHE pour son soutien et pour avoir accepté de présider mon jury.

Je remercie vivement les professeurs Michel BABRI, Ousmane THIARE et Roger Marcellin FAYE d'avoir accepté d'être les rapporteurs de ma thèse. Leurs remarques et conseils, tous très constructifs, m'ont beaucoup aidé.

Je suis très reconnaissant au Pr. Moussa LO d'avoir accepté de participer à mon jury de thèse et le remercie sincèrement.

Je souhaite également remercier tous mes collègues et étudiants des Laboratoires LIRT et LTI de l'UCAD sans oublier mes collègues de l'Université Virtuelle du Sénégal pour la bonne ambiance de travail.

Enfin, je remercie mes étudiants Marième Dieng FALL et Kalidou SY qui ont pris de leur temps pour m'assister dans la mise en forme de ce présent document .

Bien sûr, je ne peux terminer sans remercier mes proches de tout cœur et notamment mon épouse et mes enfants qui, au cours de ces trois années de thèse, m'ont toujours soutenu et encouragé, comme d'habitude.

# Résumé

L'Afrique a su exploiter les services à valeur ajoutée autour des SMS et USSD pour faciliter le quotidien des utilisateurs de terminaux mobiles en proposant l'accès à des services e-Banking, de paiement, de consultation de cours de marché agricole, etc. à partir des terminaux mobiles. Les messages de ces services sont transportés via le réseau de signalisation sémaphore SS7.

Or, depuis juin 2015, beaucoup de pays Africains sont passés à la Télévision Numérique Terrestre libérant ainsi des fréquences UHF très précieuses et de très bonnes caractéristiques permettant le déploiement de réseau mobile full IP offrant des débits pouvant dépasser 100Mbps.

Ce développement d'Internet mobile à haut débit, d'une part, crée les conditions pour que le protocole de signalisation Voix Sur IP (VoIP) SIP, qui s'est imposé dans le monde Internet, soit massivement utilisé par les terminaux mobiles pour l'accès à des services à valeur ajoutée et des communications multimédia, et d'autre part, favorise le développement de la voix sur LTE (VoLTE) qui s'appuie sur le réseau IMS utilisant le protocole SIP pour offrir des services multimédia de télécommunications (audio, vidéo, messagerie instantanée, etc.).

Par ailleurs, on assiste à l'émergence de la technologie WebRTC permettant à des utilisateurs d'Internet d'utiliser de simples navigateurs pour accéder à des services multimédia à partir de leurs ordinateurs ou de leurs terminaux mobiles tels que smartphones ou tablettes sans avoir besoin d'installer des greffons.

Ces évolutions technologiques vont certainement impacter sur la manière de concevoir des services adaptés aux besoins des populations.

C'est précisément, l'objectif de cette thèse qui se propose de montrer :

Dans sa première partie que :

- (i) L'environnement de développement de services de télécoms en Afrique se doit d'évoluer pour tenir compte des nouvelles opportunités offertes par les réseaux mobiles full IP haut débit ;
- (ii) Des simplifications des architectures des réseaux de télécoms peuvent être opérées pour tenir comptes des spécificités des pays en voie de développement ;
- (iii) Des technologies susceptibles de créer des chaînes de télévisions numériques véritablement interactives existent et sont à développer pour accompagner l'enseignement supérieur et la santé dans les pays aspirant à l'émergence.

Dans sa deuxième partie que :

- (i) Des modèles de collaboration peuvent être mis en œuvre autour de cloud computing pour permettre aux pays Africains d'offrir des formations de qualité à la jeunesse dans le

contexte de la massification des établissements d'enseignement supérieur.

- (ii) Des modèles de services autour de WebRTC, ne nécessitant pas forcément l'accès à Internet, peuvent être mis en œuvre pour favoriser les travaux pratiques dans les domaines de l'enseignement, des langues et Sciences Technologies Ingénierie Mathématiques (STEM) aussi bien dans les universités classiques faisant face à la massification des effectifs d'étudiants que dans les universités virtuelles qui sont en gestation en Afrique.

**Mots-clés :** Services Télécoms, simplification architecture, Télévision interactive, cloud computing, WebRTC, STEM, Travaux pratiques, massification effectifs

# Table des matières

Introduction générale de la thèse	27
<b>I Étude de la convergence des services de télécommunications</b>	<b>30</b>
<b>1 Évolution des réseaux de mobiles</b>	<b>33</b>
1.1 Introduction	33
1.2 Le réseau GSM, le service GPRS et l'évolution EDGE	34
1.2.1 Le réseau GSM	34
1.2.2 Le service GPRS	36
1.2.3 Service SMS classique	38
1.3 Le réseau 3G ou UMTS et les évolutions HSPA	42
1.3.1 Le réseau 3G	42
1.3.2 Les évolutions HSPA de l'UMTS	43
1.4 Le réseau NGN	43
1.4.1 Architecture du réseau NGN	43
1.4.2 Les composants du réseau	44
1.4.3 Le transport de la signalisation	44
1.4.4 Le transport de la voix	45
1.5 Le réseau 4G : LTE (Long Term Evolution)	46
1.5.1 Architecture du réseau LTE	47
1.5.2 Qualité de service (QoS) et bearers	51
1.5.3 Les interfaces	52
1.5.4 Les protocoles radio	53
1.5.5 La fonction CSFB	53
1.6 Le réseau IMS et la voix sur LTE (VoLTE)	55
1.6.1 L'IMS (IP Multimedia Subsystem)	55
1.6.2 VoLTE	59
1.7 Conclusion	60
<b>2 Étude avancée du réseau IMS : Architecture et services</b>	<b>61</b>
2.1 Introduction	61
2.2 Étude du protocole SIP	61
2.2.1 Historique de la normalisation de SIP	61
2.2.2 Avantages du protocole SIP	61

2.2.3	Architecture du protocole SIP . . . . .	62
2.2.4	L'adressage SIP . . . . .	63
2.2.5	Méthodes et codes d'état . . . . .	64
2.3	Présentation de l'IMS . . . . .	66
2.3.1	Historique de la normalisation de l'IMS . . . . .	66
2.3.2	L'architecture IMS . . . . .	67
2.4	Fonctionnement détaillé de l'IMS : Interopérabilité IMS-SIP . . . . .	72
2.4.1	Gestion des profils/identités utilisateurs . . . . .	72
2.4.2	Authentification dans l'IMS . . . . .	74
2.4.3	Enregistrement d'un terminal dans le réseau IMS . . . . .	75
2.4.4	Mise en communication de deux utilisateurs . . . . .	76
2.4.5	Modification des sessions . . . . .	78
2.4.6	Fin d'enregistrement . . . . .	78
2.5	L'architecture de service IMS . . . . .	78
2.5.1	Serveurs d'applications SIP . . . . .	79
2.5.2	IM-SSF : IM-Service Switching Function . . . . .	80
2.5.3	OSA SCS : Open Service Access Service Capability Server . . . . .	81
2.5.4	SCIM : Service Capability Interaction Manager . . . . .	83
2.6	Services standards : Enablers . . . . .	83
2.6.1	Présence . . . . .	84
2.6.2	Messagerie . . . . .	84
2.6.3	Conférence . . . . .	84
2.6.4	Multimedia Telephony (MMTel) . . . . .	84
2.6.5	Rich Communication Suite (RCS) . . . . .	85
2.6.6	Push-To-Talk over Cellular(PoC) . . . . .	85
2.7	Taxation . . . . .	86
2.7.1	Taxation hors-ligne . . . . .	86
2.7.2	Taxation en ligne . . . . .	87
2.8	Mise en place d'une plateforme de service IMS . . . . .	88
2.8.1	Le cœur du réseau : OpenIMSCore . . . . .	88
2.8.2	Plateforme de développement de services pour IMS : Mobicents . . . . .	93
2.8.3	L'intégration de la plate-forme de développement à OpenIMSCore . . . . .	98
2.9	Quelques exemples d'applications compatibles IMS . . . . .	99
2.9.1	Application E-banking . . . . .	99
2.9.2	Application de prise de Rendez-vous . . . . .	103
2.10	Conclusion . . . . .	104
<b>3</b>	<b>Intégration de WebRTC dans IMS</b> . . . . .	<b>105</b>
3.1	Introduction . . . . .	105
3.2	Présentation du WebRTC . . . . .	105
3.2.1	Définition . . . . .	105
3.2.2	Étude de protocoles intervenant dans WebRTC . . . . .	106

3.3	Problématiques du WebRTC et solutions potentielles . . . . .	112
3.3.1	Sécurité des applications . . . . .	112
3.3.2	Transiter par des firewalls ou le NAT . . . . .	112
3.4	Description générale de la norme WebRTC . . . . .	112
3.4.1	PeerConnection . . . . .	113
3.4.2	DataChannels . . . . .	114
3.4.3	MediaStream . . . . .	115
3.4.4	Multiplexage media / data . . . . .	116
3.4.5	Signalisation . . . . .	116
3.4.6	Transport de Medias . . . . .	117
3.4.7	Translation d'adresses . . . . .	117
3.5	Les applications WebRTC . . . . .	117
3.6	Les codecs . . . . .	117
3.6.1	Codecs WebRTC . . . . .	117
3.6.2	Codecs IMS . . . . .	118
3.6.3	Interopérabilité WebRTC-IMS . . . . .	118
3.7	Transcodage . . . . .	119
3.8	Proposition d'une architecture de test . . . . .	120
3.8.1	Mise en place de WRTC-GW (WebRTC Gateway) . . . . .	120
3.8.2	Tests et résultats . . . . .	122
3.9	Conclusion . . . . .	126
<b>4</b>	<b>Proposition d'un modèle technologique de réseaux de télécoms dans un contexte E-learning</b>	<b>127</b>
4.1	Introduction . . . . .	127
4.2	Télévision Numérique Terrestre (TNT) . . . . .	128
4.2.1	Dividende numérique . . . . .	128
4.2.2	Problème de connectivité dans les zones rurales . . . . .	129
4.3	Principe de la technologie TV White SPACE (TVWS) . . . . .	129
4.3.1	Les points forts de TVWS . . . . .	129
4.3.2	Architecture d'un réseau de données avec les TV white space . . . . .	129
4.3.3	Les différents éléments du réseau . . . . .	130
4.3.4	Quelques cas d'utilisation de TVWS en Afrique . . . . .	131
4.4	Proposition d'une architecture de télévision interactive . . . . .	131
4.4.1	Présentation des normes DVB et HBBTV . . . . .	131
4.4.2	Télévision interactive . . . . .	134
4.5	Proposition de solution de SMS sur IP et 4G . . . . .	137
4.5.1	Conception de la solution . . . . .	137
4.5.2	Proposition d'une solution de SMS sur LTE/EPC . . . . .	142
4.5.3	Présentation des tests de validité . . . . .	143
4.6	Proposition d'une solution de réseau 4G optimisé . . . . .	148
4.6.1	Présentation du système de billing . . . . .	148

4.6.2	Présentation du processus de communication en IMS : IFC . . . . .	153
4.6.3	Architecture de déploiement de notre solution . . . . .	154
4.6.4	Conclusion . . . . .	163
4.7	Conclusion . . . . .	164

## **II Applications des services de télécoms aux organisations virtuelles** **167**

<b>1</b>	<b>Organisations virtuelles</b>	<b>170</b>
1.1	Introduction . . . . .	170
1.2	Organisations virtuelles dans le domaine de l'enseignement . . . . .	171
1.2.1	Formations à distance . . . . .	171
1.2.2	Formations Ouvertes et À Distance . . . . .	171
1.2.3	Université virtuelle . . . . .	171
1.2.4	Campus numérique . . . . .	171
1.2.5	Modèles pédagogiques des universités numériques . . . . .	172
1.3	Organisations virtuelles autour des logiciels libres . . . . .	174
1.3.1	Les logiciels libres . . . . .	175
1.3.2	Modèle organisationnel et technologique des logiciels libres . . . . .	180
1.3.3	Les logiciels libres, une solution pour l'enseignement supérieur en Afrique	182
1.3.4	L'expérience de mise en place d'une université numérique à partir de logiciels libres . . . . .	187
1.4	Quelques projets d'interconnexion en Afrique . . . . .	193
1.4.1	AfricaConnect . . . . .	193
1.4.2	Padtice . . . . .	194
1.4.3	SnRER . . . . .	195
1.5	Conclusion . . . . .	196
<b>2</b>	<b>Organisations virtuelles et protocoles</b>	<b>197</b>
2.1	Introduction . . . . .	197
2.2	Étude du protocole XMPP et de PubSub . . . . .	197
2.2.1	Le coeur du protocole XMPP . . . . .	197
2.2.2	Les extensions du protocole XMPP . . . . .	198
2.2.3	L'architecture XMPP . . . . .	198
2.2.4	Quelques exemples de serveurs XMPP open source . . . . .	202
2.2.5	Les bibliothèques clientes de XMPP . . . . .	202
2.2.6	Message Oriented Middleware (MOM)et organisations virtuelles . . . . .	203
2.2.7	PubSub : Principe de fonctionnement et exemples . . . . .	203
2.3	Étude de quelques protocoles VDI : RDP, SPICE . . . . .	206
2.3.1	Introduction à Virtualbox . . . . .	206
2.3.2	Quelques fonctionnalités avancées de Virtualbox . . . . .	206

2.3.3	Introduction au protocole RDP . . . . .	207
2.3.4	Versions et fonctionnalités de RDP . . . . .	207
2.3.5	Implémentations libres du protocole RDP . . . . .	208
2.3.6	Introduction au protocole SPICE . . . . .	209
2.3.7	Architecture de SPICE . . . . .	209
2.3.8	Fonctionnement de SPICE et gestion des périphériques . . . . .	210
2.3.9	Étude comparative de systèmes de virtualisation sous Linux : Xen, Qemu, KVM . . . . .	213
2.4	Conclusion . . . . .	220
<b>3</b>	<b>Proposition d'un modèle d'organisations virtuelles basé sur le Cloud Computing</b>	<b>221</b>
3.1	Introduction . . . . .	221
3.2	Cloud Computing . . . . .	221
3.2.1	Historique et définition . . . . .	221
3.2.2	Les caractéristiques principales du Cloud Computing . . . . .	222
3.2.3	Modèles de délivrance et de déploiement de services . . . . .	223
3.2.4	Les principaux composants du Cloud . . . . .	225
3.2.5	Avantages et inconvénients du Cloud . . . . .	226
3.3	Présentation de l'outil OpenStack . . . . .	228
3.4	Proposition d'un modèle . . . . .	231
3.4.1	Les motivations du modèle proposé . . . . .	231
3.4.2	Proposition du modèle . . . . .	232
3.5	Preuve de la faisabilité et de la pertinence techniques du modèle . . . . .	233
3.5.1	Plateforme de travaux pratiques de l'UCAD . . . . .	233
3.5.2	Plateforme de travaux pratiques de l'UADB . . . . .	236
3.5.3	Pérennité du modèle : Proposition d'un système de facturation . . . . .	237
3.6	Méthode d'authentification décentralisée pour l'accès aux ressources de l'organisation virtuelle . . . . .	239
3.6.1	Conception de la solution . . . . .	239
3.6.2	Utilisation de shibboleth dans la gestion . . . . .	241
3.7	Découverte dynamique des ressources disponibles . . . . .	247
3.7.1	Cas d'utilisation du système de publication dynamique des ressources . . . . .	249
3.7.2	Description de quelques cas d'utilisation de l'application . . . . .	250
3.7.3	Format d'échange de message entre XMPP et l'application . . . . .	251
3.7.4	Réalisation . . . . .	251
3.8	Conclusion . . . . .	253
<b>4</b>	<b>Proposition de plateformes WebRTC pour l'amélioration du système d'enseignement des universités</b>	<b>255</b>
4.1	Introduction . . . . .	255

4.2	Proposition d'une plateforme WebRTC d'appui au système d'enseignement des universités dans un contexte de connexion Internet limitée . . . . .	255
4.2.1	État de l'art sur le WebRTC du point de vue du développement logiciel .	256
4.2.2	Fonctionnalités du système . . . . .	259
4.2.3	Présentation de l'architecture de la solution . . . . .	259
4.2.4	Description de quelques cas d'utilisation de la plateforme . . . . .	261
4.2.5	Résultats . . . . .	263
4.3	Solutions de collaboration en environnement e-learning . . . . .	266
4.3.1	Impact de l'intégration du WebRTC dans les plateformes e-learning des universités . . . . .	266
4.3.2	Proposition d'une plateforme de développement logiciel collaboratif pour les universités numériques . . . . .	272
4.4	Conclusion . . . . .	280
	<b>Conclusion générale de la thèse</b>	<b>282</b>

# Table des figures

I.1.1	Architecture du GSM [1]	34
I.1.2	Architecture GPRS [1]	37
I.1.3	Les entités de base de base du service SMS	39
I.1.4	Service de base SM MO	40
I.1.5	Service de base SM MT	41
I.1.6	Architecture d'un réseau 3G	42
I.1.7	Architecture physique du réseau NGN [1]	44
I.1.8	Le transport de la signalisation [1]	45
I.1.9	Le transport de la voix	46
I.1.10	L'architecture physique du réseau EPS [1]	47
I.1.11	Architecture EPS	49
I.1.12	Bearer et qualité de service	52
I.1.13	Interfaces LTE	53
I.1.14	L'interface SGs : le lien entre les réseaux	54
I.1.15	Positionnement du réseau IMS [2]	56
I.1.16	Le transfert de la signalisation téléphonique [3]	57
I.1.17	Le transfert de la voix [3]	58
I.1.18	L'impact de de la fonction SRVCC sur l'architecture IMS	59
I.1.19	Communication entre deux mobiles 4G	59
I.1.20	Communication 4G-RTCP	60
I.2.1	Architecture de SIP [4]	63
I.2.2	Schéma global de l'architecture en couches de l'IMS [5]	68
I.2.3	Relations entre IMSU, IMPI et IMPU selon le 3GPP R5 [6]	73
I.2.4	Relations entre IMSU, IMPI et IMPU selon le 3GPP R6 [6]	73
I.2.5	Procédure d'enregistrement dans l'IMS [4]	75
I.2.6	Mise en communication de deux utilisateurs [6]	77
I.2.7	Architecture du plan de service IMS	79
I.2.8	Architecture de service CAMEL et interaction avec IMS	81
I.2.9	Architecture OSA [7]	82
I.2.10	Passerelles Parlay/ParlayX [8]	83
I.2.11	Architecture des entités de tarification hors-ligne	87
I.2.12	Architecture des entités de tarification en ligne	87
I.2.13	Architecture de la plateforme implémentée	92
I.2.14	Architecture J2EE	94

I.2.15	Architecture JSLEE . . . . .	95
I.2.16	Intégration Mobicents/JBoss à OpenIMSCore . . . . .	98
I.2.17	Connexion IMS . . . . .	102
I.2.18	Ajout de l'adresse SIP de l'application . . . . .	102
I.2.19	Échanges avec le système . . . . .	102
I.2.20	Utilisateur connecté depuis IMSDroid (installé sur un téléphone Android) . . . . .	103
I.2.21	Client prenant un rendez-vous depuis UCT IMS client . . . . .	104
I.2.22	Client prenant rendez-vous depuis IMSDroid . . . . .	104
I.3.1	Déroulement d'une session WebSocket . . . . .	106
I.3.2	Architecture d'une application WebRTC . . . . .	112
I.3.3	Établissement d'une connexion entre deux clients utilisant WebRTC . . . . .	113
I.3.4	Structure de la pile de protocoles utilisée par WebRTC dans un échange de données . . . . .	114
I.3.5	Structure de la pile de protocoles utilisée par WebRTC dans un échange de médias . . . . .	115
I.3.6	Schéma d'une communication entre clients SIP et clients WebRTC . . . . .	116
I.3.7	Architecture et modèle de référence WebRTC-IMS [9] [10] . . . . .	119
I.3.8	Architecture de test . . . . .	120
I.3.9	Activation du Websocket sous Asterisk . . . . .	121
I.3.10	Activation de la signalisation Websocket sous Asterisk . . . . .	121
I.3.11	Configuration d'un client Sipml5 . . . . .	121
I.3.12	Appel WebRTC - IMS . . . . .	122
I.3.13	Communication entre un utilisateur WebRTC et un utilisateur IMS . . . . .	122
I.3.14	Interface d'Asterisk montrant la mise en communication WebRTC - IMS . . . . .	123
I.3.15	Appel IMS → WebRTC . . . . .	123
I.3.16	La mise en communication réussie d'un utilisateur IMS et d'un utilisateur WebRTC . . . . .	124
I.3.17	Appel WebRTC → Réseau fixe . . . . .	124
I.3.18	Mise en communication réussie entre un utilisateur WebRTC et un utilisateur fixe . . . . .	125
I.3.19	Interface du SBC montrant la mise en communication WebRTC - Fixe . . . . .	125
I.4.1	Dividende numérique . . . . .	128
I.4.2	Répartition des fréquences issues du dividende numérique au Sénégal . . . . .	128
I.4.3	Architecture d'un réseau de données avec TVWS [11] . . . . .	130
I.4.4	Architecture du protocole HBBTV . . . . .	134
I.4.5	Interactivité enseignant et étudiants . . . . .	135
I.4.6	Interactivité enseignant et étudiants développeurs . . . . .	135
I.4.7	Ettus N210 . . . . .	136
I.4.8	Plateforme d'expérimentations DVB de Yann BACHY . . . . .	136
I.4.9	Superposition HbbTV [12] . . . . .	137

I.4.10	Cas du protocole SIMPLE [13]	138
I.4.11	Architecture du SMSC-IP	138
I.4.12	Intégration d'un SMSC	139
I.4.13	Stockage de message vers un destinataire non rattaché au réseau	140
I.4.14	Transmission du message en absence	140
I.4.15	L'architecture du service SMS Group Broadcast	141
I.4.16	Transfert de message d'un utilisateur SIP vers un utilisateur 2G	142
I.4.17	Principe de fonctionnement d'un service à valeur ajoutée	142
I.4.18	SMS over SGs	143
I.4.19	Interopérabilité de notre solution avec le réseau d'un opérateur 4G	143
I.4.20	Interopérabilité de notre solution avec le réseau d'un opérateur VoLTE	143
I.4.21	Connexion de l'utilisateur Samuel	144
I.4.22	Vérification de la liste de présence	144
I.4.23	Envoi d'un SMS à un utilisateur déconnecté	144
I.4.24	Réception d'un message transmis en absence	144
I.4.25	Suppression d'un message transmis de la base de données	144
I.4.26	Envoi d'un SMS de UA vers SIP	145
I.4.27	SVA : Demande de bulletin de notes	146
I.4.28	L'interface d'accueil du client SIP WEB	146
I.4.29	Appel à partir de la plateforme	147
I.4.30	Envoi d'un message depuis la plateforme vers un softphone	147
I.4.31	Envoi et réception depuis un softphone	147
I.4.32	Algorithme du système de Billing	148
I.4.33	Architecture d'un système de facturation moderne	149
I.4.34	Plan de facturation	151
I.4.35	Les composants d'un IFC [14][15]	154
I.4.36	Architecture de notre solution	155
I.4.37	Processus de facturation	155
I.4.38	Création d'un AS	156
I.4.39	Création du Trigger Point	156
I.4.40	Création d'un IFC	157
I.4.41	Création d'un SP	157
I.4.42	Ajout des clients dans le profile billing	158
I.4.43	Modification du IMSU des clients	158
I.4.44	Page d'authentification du Billing	159
I.4.45	Fonctionnalités du système de billing	159
I.4.46	Création de compte pour l'utilisateur Maryteuw sur le billing	160
I.4.47	Création de compte pour l'utilisateur Claude sur le billing	160
I.4.48	Correspondance entre l'IMSU de Maryteuw et son compte de facturation	160
I.4.49	Correspondance entre l'IMSU de Claude et son compte de facturation	161
I.4.50	Liste des correspondances entre comptes IMS et CALLERID billing	161

I.4.51	Définition des tarifs . . . . .	161
I.4.52	Ajout des tarifs dans le plan de numérotation des utilisateurs . . . . .	162
I.4.53	État du crédit des clients . . . . .	162
I.4.54	CDR de l'appel du 790010001 vers le 790010002 . . . . .	163
I.4.55	État du crédit des clients après appel . . . . .	163
II.1.1	Classification des licences . . . . .	180
II.1.2	Capture des fichiers de résolution directe du serveur DNS sous Debian . . . . .	183
II.1.3	Capture des fichiers de résolution directe du serveur DNS sous Windows Server2008 . . . . .	184
II.1.4	Environnement numérique de travail . . . . .	188
II.1.5	Espace personnel d'un utilisateur connecté à Esup Portal . . . . .	189
II.1.6	Interface de Moodle montrant l'ensemble des filières de l'UVS en 2014 . . . . .	189
II.1.7	Interface d'identification de Shibboleth . . . . .	191
II.1.8	Owncloud . . . . .	192
II.1.9	BigBlueButton . . . . .	193
II.1.10	Réseau WACREN . . . . .	194
II.1.11	Cartographie du réseau Fibre optique de l'ADIE . . . . .	196
II.2.1	Architecture Client/serveur XMPP . . . . .	200
II.2.2	Présence . . . . .	200
II.2.3	Message . . . . .	201
II.2.4	Paquet IP . . . . .	201
II.2.5	Réponse du serveur . . . . .	201
II.2.6	Principe de fonctionnement PubSub . . . . .	204
II.2.7	Architecture SPICE . . . . .	210
II.2.8	Composants de SPICE . . . . .	211
II.2.9	La bibliothèque Libvirt . . . . .	219
II.3.1	Les principaux modèles de délivrance de services . . . . .	224
II.3.2	Architecture d'OpenStack . . . . .	229
II.3.3	Architecture du modèle proposé . . . . .	232
II.3.4	Réseau virtuel de l'UCAD . . . . .	233
II.3.5	Les instances des machines virtuelles de l'UCAD . . . . .	234
II.3.6	Interface Web d'accès à la plateforme . . . . .	234
II.3.7	Accès d'un étudiant à la plateforme de TP . . . . .	235
II.3.8	Interface montrant les utilisateurs connectés dans OpenIMScore . . . . .	235
II.3.9	Appel vidéo entre deux utilisateurs . . . . .	236
II.3.10	Réseau virtuel de l'UADB . . . . .	236
II.3.11	Les instances des machines virtuelles de l'UADB . . . . .	237
II.3.12	Interaction entre CloudKitty et OpenStack . . . . .	237
II.3.13	Architecture d'intégration d'une solution de facturation dans une OV basée sur le cloud . . . . .	238

II.3.14	Les services facturés . . . . .	238
II.3.15	Facturation lors du lancement de l'instance . . . . .	239
II.3.16	Principe d'authentification . . . . .	240
II.3.17	The basic shibboleth authentication process . . . . .	241
II.3.18	Architecture d'intégration de Shibboleth dans le cloud . . . . .	242
II.3.19	Processus d'authentification keystone . . . . .	243
II.3.20	Keystone avec HTTPD comme moteur d'authentification . . . . .	243
II.3.21	Hébergement de keystone dans apache . . . . .	244
II.3.22	Shibbolisation dashboard (1) . . . . .	245
II.3.23	Shibbolisation dashboard (2) . . . . .	245
II.3.24	Service de découverte . . . . .	247
II.3.25	Accès à l'infrastructure Cloud . . . . .	247
II.3.26	Architecture du système de publication dynamique des ressources . . . . .	248
II.3.27	Diagramme des cas d'utilisation . . . . .	250
II.3.28	Exemple de profil super admin . . . . .	252
II.3.29	Profil d'un admin local . . . . .	252
II.3.30	Statut des machines virtuelles . . . . .	253
II.3.31	Prise de contrôle d'une machine . . . . .	253
II.4.1	Architecture générale de la technologie WebRTC . . . . .	257
II.4.2	Processus de communication WebRTC . . . . .	258
II.4.3	Identifiant WebRTC d'un utilisateur . . . . .	258
II.4.4	Fonctionnalité du système E-learning . . . . .	259
II.4.5	Architecture du système . . . . .	260
II.4.6	Structure logicielle du serveur de signalisation . . . . .	260
II.4.7	Création d'une nouvelle salle de réunion . . . . .	261
II.4.8	Ajout d'un utilisateur dans une chambre de conférence . . . . .	262
II.4.9	Partage d'écran entre étudiant et enseignant . . . . .	263
II.4.10	Visioconférence entre l'enseignant et les étudiants . . . . .	264
II.4.11	Partage d'écran de l'enseignant . . . . .	264
II.4.12	Microscope optique interfacé à un ordinateur . . . . .	265
II.4.13	Transfert de fichiers . . . . .	265
II.4.14	Architecture Moodle . . . . .	267
II.4.15	Architecture de la solution . . . . .	268
II.4.16	Processus de mapping . . . . .	269
II.4.17	Fonctionnement du plugin . . . . .	269
II.4.18	Classe virtuelle . . . . .	270
II.4.19	Tableau blanc . . . . .	270
II.4.20	Éditeur Mathématiques . . . . .	271
II.4.21	Transfert de fichiers . . . . .	271
II.4.22	Réception de fichiers . . . . .	272
II.4.23	Architecture réseau de l'environnement . . . . .	274

II.4.24	Ajout de la conférence audio/vidéo dans Saros . . . . .	275
II.4.25	Diagramme décrivant l'assignation d'une VM . . . . .	275
II.4.26	Diagramme décrivant le travail collaboratif . . . . .	276
II.4.27	Développement de logiciel collaboratif . . . . .	277
II.4.28	Les membres de session de la TP . . . . .	277
II.4.29	Discussion dans la conférence XMPP . . . . .	278
II.4.30	Session audio/vidéo . . . . .	278
II.4.31	Gestion des utilisateurs . . . . .	279
II.4.32	Assignation de machine virtuelle . . . . .	279
II.4.33	Interface utilisateur . . . . .	280

# Liste des tableaux

I.1.1	Plan de fréquences du réseau GSM . . . . .	36
I.1.2	Caractéristiques des terminaux . . . . .	48
I.3.1	Comparaison entre les trois protocoles de transport SCTP, TCP et UDP . . .	109
I.4.1	Tableau récapitulatif du plan de facturation . . . . .	153
II.1.1	Rôles d'un enseignant dans une formation à distance . . . . .	172
II.1.2	Quelques logiciels libres dans le domaine de l'éducation . . . . .	185
II.2.1	Plugins Openfire . . . . .	202
II.3.1	Choix des universités dans le modèle proposé . . . . .	231
II.4.1	Fichiers Moodle . . . . .	268

# Liste des abréviations

3GPP	Third Génération Partnership Project
AAA	Authentication, Authorization, Accounting
AAL2	ATM Adaptation Layer
ADIE	Agence de l'Informatique de l'Etat
AKA	Authentication and Key Agreement
AMBR	Aggregate Maximum Bit Rate
AMR	Adaptative MultiRate
AN	Access Network
API	Application Program Interfaces
AS	Application Server
ATGW	Access Transfer Gateway
ATM	Asynchronous Transfert Mode
AuC	Authentication Center
AUTN	Authentication Token
BAS	Billing Application Server
BGCF	Breakout Gateway Control Function
BSC	Base Station Controller
BSD	Berkley Software Distribution
BSS	Base Station Subsystem
BSSAP	BSS Application Part
BSSMAP	BSS Management Application Part
BTS	Base Tranceiver Station
CAP	CAMEL Application Part
CDF	Charging Data Function

CDF	Charging Data Function
CDP	Contrat de performance
CDR	Charging Data Record
CK	Chifferring Key
CM	Call Management
CN	Core Network
CNAES	Concertation Nationale Sur l'Avenir de l'Enseignement Supérieur
COPS	Common Open Policy Service
CPCP	Conference Policy Control Protocol
CRBT	Customized Ring Back Tone
CS	Circuit Service
CSA	Common Scrambling Algorithm
CSCF	Call Session Control Function
CSE	CAMEL Service Environlent
CSFB	Circuit Switched FallBack
DHCP	Dynamic Host Control Protocol
DTLS	Datagramme Transport Layer Protocol
DVB	Digital Video Broadcasting
DVB-C	DVB Cable
DVB-S	DVB Satellie
DVB-T	DVB Terrestrial
EDGE	Enhanced Data Rates for GSM Evolution
EIR	Equipment Identity Register
ENO	Espace Numérique Ouvert
ENT	Environnemnt Numérique de Travail
EPS	Universal Integrated Circuit Card
ESA	Eastern and Southern Africa
ETSI	European Telecommunications Standards Institute
eUTRAN	Evolved Universal Terrestrial Radio Access Network
FAD	Formation À Distance

## LISTE DES ABRÉVIATIONS

FDMA	Frequency Division Multiple Access
FOAD	Formation Ouverte et À Distance
FSF	Free Foundation Software
GBC	Group Broadcast Center
GBE	Group Broadcast Entity
GBR	Guaranteed Bit Rate
GGSN	Gateway GPRS Support Node
GMSC	Gateway MSC
GPL	GNU Public Licence
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
GSS	GPRS Sub-System
HBBTV	Hybrig Broadcast Broadband TV
HLR	Home Location Register
HSDPA	High Speed Downlink Packet Access
HSPA	High Speed Packet Access
HSS	Home Subscriber Server
HSUPA	High Speed Uplink Packet Access
HTTP	HyperText Transfer Protocol
HTTPD	Hypertext Transfer Protocol Daemon
I-CSCF	Interrogating CSCF
IaaS	Infrastructure as a Service
ICE	Interactive Connectivity Establishment
ICS	IMS Centralized Services
IDE	Integrated Development Environment
IDP	Identity Provider
IETF	Internet Engineering Task Force
IFC	Initial Filter Criteria
IK	Integrity Key
IMEI	International Mobile Equipment Identity

IMPU	IMS Public User Identity
IMS	IP Multimedia Subsystem
IMSI	International Mobile Subscriber Identity
INAP	Intelligent Network Application Part
IP	Internet Protocol
ISIM	IP multimedia Services Identity Module
ISUP	ISDN User Part
ITU	International Telecommunication Union
IWMSC	Interworking MSC For Short Message Service
J2EE	Java 2 Enterprise Edition
JDK	Java Development Kit
JDT	Java Development Tools
JPA	Java Persistence API
JSF	Java Server Faces
JSLEE	JAIN Service Logic Execution Environment
JVM	Java Virtual Machine
KVM	Kernel-based Virtual Machine
LAI	Location Area Identity
LCD	Least Cost Dialing
LCR	Least Cost Routing
LTE	Long Term Evolution
M3UA	MTP 3 User Adaptation
MAC	Medium Access Control
MAP	Mobile Application Part
ME	Mobile Equipment
MGCF	Media Gateway Control Function
MGCP	Media Gateway Control Protocol
MGW	Media Gateway
MIMO	Multiple-Input Multiple-Output
MM	Mobility Management

## LISTE DES ABRÉVIATIONS

MMTel	Multimedia Telephony
MOM	Message Oriented Middleware
Moodle	Modular Object-Oriented Dynamic Learning Environment
MRF	Media Resource Function
MS	Mobile Station
MSC	Mobile service Switching Center
MSISDN	Mobile Station ISDN Number
MSRN	Mobile Station Roaming Number
MTI	Mandatory To Implement
NAS	Non Access Stratum
NASS	Network Attachment SubSystem
NAT	Network Address Translation
NGN	Next Generation Network
NIST	National Institute of Standards and Technology
NREN	National Research and Education Network
NSS	Network Switching Subsystem
OCS	Online Charging System
OSA	Open Service Access
OSA SCS	Open Service Access Service Capability Server
OV	Organisation Virtuelle
OV	Organisation Virtuelle
P-CSCF	Proxy CSCF
PA	Presence Agent
PaaS	Platform as a Service
PADTICE	Projet d'Appui au Développement des Technologies de l'Information et de la Communication
PCEF	Policy and Charging Enforcement Function
PCRF	Policy and Charging Rules Function
PCU	Packet Control Unit
PDCP	Packet Data Convergence Protocol

PDN	Packet Data Network
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PLMN	Public Land Mobile Network
PoC	Push-To-Talk over Cellular
PS	Packet Service
PSK	Phase Shift Keying
PSTN	Public Switched Telephone Network
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase-Shift Keying
RACS	Ressource Admission Control SubSystem
RCP	Rich Client Platform
RCS	Rich Communication Suite
RDP	Remote Desktop Protocol
RES	RESponse
RFC	Request For Comments
RLC	Radio Link Control
RNC	Radio Network Controller
RRC	Radio Resource Control
RREN	regional Research and Education Network
RTC	Réseau Téléphonique Commuté
RTCP	Real Time transport Control Protocol
RTP	Real-time Transport Protocol
S-CSCF	Serving CSCF
SaaS	Software as a Service
SAML	Security Assertion Markup Language
SBC	Session Border Controller
SCC AS	Service Centralization and Continuity Application Server
SCF	Services Capability Function

## LISTE DES ABRÉVIATIONS

SCIM	Service Capability Interaction Management
SCS	Service Capability Server
SCTP	Stream Control Transmission Protocol
SDH	Synchronous Digital Hierarchy
SDP	Session Description Protocol
SDR	Software Defined Radio
SER	SIP Express Router
SGBDR	Système de Gestion de Bases de Données Relationnelles
SGSN	Service GPRS Support Node
SGW	Serving Gateway
SIGESR	Système d'Information et de Gestion de l'Enseignement Supérieur et de la Recherche
SIGTRAN	Signalling Transport over IP
SIM	Subscriber Identity Module
SIP	Session Initiation Protocol
SM-MO	Short Message Mobile Originated
SM-MT	Short Message Mobile Terminated
SME	Short Message Entity
SMPP	Short Message Peer to Peer
SMS	Short Message Service
SMSC	SMS Center
SMTP	Simple Mail Transfert Protocol
SPI	Software, Platform, Infrastructure
SPICE	Simple Protocol for Independent Computing Environments
SPT	Service Point Trigger
SRTCP	Secure RTCP
SRTP	Secure RP
SRVCC	Single Radio Voice Call Continuity
SS7	Signaling System #7
SSF	Service Switching Function

STEM	Science, Technology, Engineering, and Mathematics
STUN	Simple Traversal of UDP through NATs
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
TIC	Technologies de l'Information et de la Communication
TICE	Technologies de l'Information et de la Communication pour l'Enseignement
TLS	Transport Layer Security
TMSI	Temporary Mobile Subscriber Identity
TNT	Télévision Numérique Terrestre
TRAU	Transcoder and Rate Adaptation Unit
TSC	Tandem Switching Center
TURN	Traversal Using Relays around NAT
TVWS	TV White Spaces
UAC	User Agent Client
UADB	Université Assane DIOP de Bambey
UAS	User Agent Server
UCAD	Université Cheikh Anta DIOP de Dakar
UDP	User Datagram protocol
UE	User Equipment
UHF	Ultra High Frequency
UICC	Universal Integrated Circuit Card
UMTS	Universal Mobile Telecommunications System
UNESCO	United Nations Educational, Scientific and Cultural Organization
URI	Universal Ressource Identifier
USIM	Universal Subscriber Identity Module
USSD	Unstructured Supplementary Service Data
UTRAN	Universal Terrestrial Radio Access Network
UVS	Université Virtuelle du Sénégal
VDI	Virtual Desktop Infrastructure

## LISTE DES ABRÉVIATIONS

VLR	Visitor Location Register
VMSC	Visited MSC
VNC	Virtual Network Computing
VoIP	Voice over Internet Protocol
VoLTE	Voice over LTE
WACREN	West Africa and Center Research and Education Network
WCDMA	Wideband Code Division Multiple Access
WebRTC	Web Real-Time Communication
WG MMUSIC	Work Group Multiparty Multimedia Session Control
WiFi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WSML	Web services Markup Language
XDM	XML Document Management
XDMS	XCAP Document Management server
XEP	XMPP Extension Protocol
XMPP	Extensible Messaging and Presence Protocol
XRES	eXcepted RESponse
XSF	XMPP Standard Foundation

# Introduction générale de la thèse

L'accès des populations au haut débit peut être vu aujourd'hui comme l'un des facteurs de déclenchement de l'émergence d'un pays.

Or depuis quelques années, les technologies et services de télécommunications ne cessent d'évoluer vers tout IP.

Mieux encore, le dividende Numérique issu du passage à la télévision numérique offre la possibilité d'avoir des réseaux de télécommunications offrant des débits de plus en plus élevés et à faible latence. Ce qui ouvre la voie au développement des services temps réel interactifs qui peuvent être utilisés dans les secteurs tels que l'enseignement et la santé.

Dans beaucoup de pays Africains, le coût d'accès aux réseaux de télécommunications reste élevé et des opérateurs de télécommunications le justifient en partie par le coût important de déploiement d'un réseau et d'acquisition de services Télécoms.

Or l'Afrique est l'un des continents qui a su mettre en valeur des services de Télécommunications adaptés à ses besoins à travers ce qu'on appelle les services à valeur ajoutée.

Dans le contexte Africain où dans beaucoup de pays, on est convaincu que l'émergence passe par une bonne formation des jeunes, quels seraient les éléments sur lesquels agir pour résoudre le problème d'accès à moindre coût aux services réseaux des télécommunications ?

Quelles orientations prendre pour continuer à fournir des services innovants à valeur ajoutée pour l'Afrique ?

Comment combiner le haut débit et la télévision numérique pour rendre la distance de plus en plus négligeable pour des secteurs tels que l'enseignement supérieur et la santé ?

Comment exploiter les nouveaux concepts autour des réseaux Télécoms pour contribuer à l'amélioration des services rendus aux étudiants dans les domaines des STEM et des langues ?

Telles sont les principales questions abordées dans cette thèse qui est composée de deux parties :

La première traite de l'étude de la convergence des réseaux de télécommunications dont les contributions sont essentiellement la proposition :

- d'un système de centre SMS sur IP intégrant des services à valeur ajoutée interopérable avec les systèmes des opérateurs classiques des télécommunications
- d'un centre SMS pour la 4<sup>ème</sup> génération de réseau des Télécommunications
- d'une architecture simplifiée d'un réseau d'opérateur des Télécommunications dans un contexte de services à l'enseignement supérieur et la santé
- d'un modèle de chaîne de télévision Numérique à forte interactivité

La deuxième partie traite des applications de la convergence des réseaux des télécommunications

aux organisations virtuelles autour de l'enseignement supérieur dont les contributions sont essentiellement la proposition :

- d'une plateforme des Travaux Pratiques dans le domaine des STEM basée sur le cloud computing
- d'une méthode décentralisée d'accès à des ressources pédagogiques dans le cloud computing
- d'une méthode de découverte dynamique des ressources pédagogiques dans le cloud computing
- des solutions d'amélioration de l'enseignement basées sur la technologie WebRTC

## Première partie

# Étude de la convergence des services de télécommunications

# Introduction générale

Ces dernières années, on assiste à la convergence de l'informatique, des télécommunications et de l'audiovisuel. En effet, l'usage de réseaux IP se généralise dans les trois domaines précités à savoir l'informatique, les télécommunications et l'audiovisuel. Des services se développent dans le monde IP et grâce à Internet mobile, les utilisateurs des terminaux mobiles bénéficient presque de tous les services qui n'étaient accessibles que via des PC. Dans beaucoup de pays, on passe de la télévision analogique à la télévision numérique libérant ainsi un ensemble de fréquences appelé dividende numérique ayant des propriétés intéressantes en terme de transmission des données .

D'autre part, on assiste à la baisse des prix des supports fibre optique permettant de bâtir un réseau à grande couverture et à bande passante presque illimitée.

Ainsi, il est possible pour une grande entreprise répartie sur plusieurs villes dans un même pays ou des pays limitrophes de communiquer dans un réseau IP propre avec la possibilité de s'interconnecter à Internet.

Dans presque tous les pays du monde et même ceux en voie de développement, des conditions favorables à la mise en place des réseaux haut débit sont réunies grâce à la baisse des coûts de déploiement de la fibre optique et la possibilité de l'utilisation du dividende numérique.

Or dans ces pays en voie de développement, plus particulièrement ceux d'Afrique, les moyens financiers sont limités et les développeurs de logiciels ont eu des inspirations qui leur ont permis de créer des services à valeur ajoutée adaptés, utilisant comme support le réseau de signalisation sémaphore SS7 des télécoms, pour régler des problèmes des utilisateurs de terminaux mobiles.

Nous remarquons que le développement du haut débit mobile ouvre la voie à la signalisation SIP de voix sur IP si bien qu'on se demande s'il n'est pas judicieux de se focaliser sur ce protocole pour acheminer les SMS et fournir des services à valeur ajoutée. Quand on examine la communication sur IP, on se rend compte que le souci des concepteurs des communications sur IP est toujours de simplifier l'usage des services aux utilisateurs.

En téléphonie sur IP, on est passé de terminaux lourds appelés des softphones à l'intégration de flash dans les navigateurs pour en faire des téléphones IP.

Mais l'usage de flash n'est pas si simple pour un utilisateur lambda ; on voudrait que le navigateur soit un softphone sans l'usage d'un quelconque plugin. Cela a poussé Google à concevoir le WebRTC et à le rendre open source.

Désormais des navigateurs peuvent être utilisés pour accéder à des services multimédia de la même manière que les utilisateurs du réseau IMS qui utilisent le protocole SIP et dont la partie accès peut être les réseaux classiques connus tels que le Wifi, le réseau 2G, les réseaux

3G ou le réseau classique fixe de téléphonie RTC.

Il se pose donc la question de la pertinence de l'intégration des réseaux IMS et WebRTC. Le dividende numérique favorisant forcément l'émergence du haut débit IP et les facilités qu'offrent le WebRTC ne pourraient-ils pas être utilisés pour créer des chaînes de télévision interactives au service de l'éducation et la santé dans les pays en voie de développement ?

Il s'agit dans cette première partie de thèse d'étudier minutieusement dans le chapitre 1 , l'évolution des réseaux mobiles pour valider certaines de nos hypothèses.

Dans le chapitre 2 , il est question de cerner le réseau IMS ainsi que les normes de ses services car ce réseau semble important dans la convergence des réseaux des télécoms d'aujourd'hui.

Le chapitre 3 examine la question de l'interopérabilité IMS - WebRTC pour en tirer des services innovants pour les utilisateurs des terminaux mobiles, des ordinateurs, dans le but de favoriser le développement de l'économie numérique qui peut être un des leviers pour l'émergence d'un pays.

Enfin, le chapitre 4 tire les retombées des trois premiers chapitres pour proposer dans un premier temps des solutions optimales de réseaux mobiles en termes de SMS et de services à valeur ajoutée et dans un deuxième temps des solutions de télévisions interactives qui peuvent impulser les secteurs de l'économie numérique, de l'éducation et de la santé.

# Chapitre 1

## Évolution des réseaux de mobiles

### 1.1 Introduction

Un réseau est un groupe d'entités en communication. Le réseau de télécommunications est né de la volonté d'assurer la communication à distance, par la transmission électrique de la voix. Le réseau informatique ou réseau de données est un ensemble d'équipements (nœuds et terminaux) reliés entre eux pour échanger des informations. Développés séparément à leurs débuts respectivement par le monde des télécommunications (commutation de circuits) et de l'informatique (commutation de paquets), ils ont convergé pour donner un réseau unique appelé simplement le réseau de communications.

L'évolution des technologies et le développement de la téléphonie mobile ont conduit vers des besoins de types nouveaux et croissants en termes de services et d'applications. Cela a conduit à des évolutions (ou générations) de la communication.

Les services fournis par les réseaux de télécommunications sont mis en œuvre selon deux modes :

- Le mode circuit CS (Circuit Service) : Ce mode est caractérisé par l'attribution d'une ressource dédiée à un flux.
- Le mode paquet PS (Paquet Service) : Ce mode, qui permet uniquement la transmission de données, est caractérisé par l'attribution d'une ressource partagée par plusieurs flux.

On peut noter d'emblée que l'architecture des réseaux de mobiles fait apparaître deux sous-systèmes :

- Le réseau d'accès AN (Access Network) qui se charge d'attribuer la ressource radioélectrique aux mobiles et qui est fortement impacté par les évolutions successives.
- Le cœur du réseau CN (Core Network) connecte les réseaux d'accès et les réseaux tiers :
  - les réseaux fixes téléphoniques PSTN (Public Switched Telephone Network) ;
  - les réseaux fixes de données PDN (Packet Data Network) ;
  - les réseaux de mobiles PLMN (Public Land Mobile Network) ;

Le présent chapitre présente les évolutions de l'architecture des réseaux de mobiles et décrit les différentes technologies déployées.

On y présente notamment les technologies :

- 2G : le réseau GSM, le service GPRS et l'évolution EDGE
- 3G : le réseau UMTS et ses évolutions HSPA
- 4G : le réseau EPS
- IMS et VoLTE

## 1.2 Le réseau GSM, le service GPRS et l'évolution EDGE

### 1.2.1 Le réseau GSM

#### 1.2.1.1 Architecture GSM

la figure I.1.1 présente l'architecture générale d'un réseau GSM.

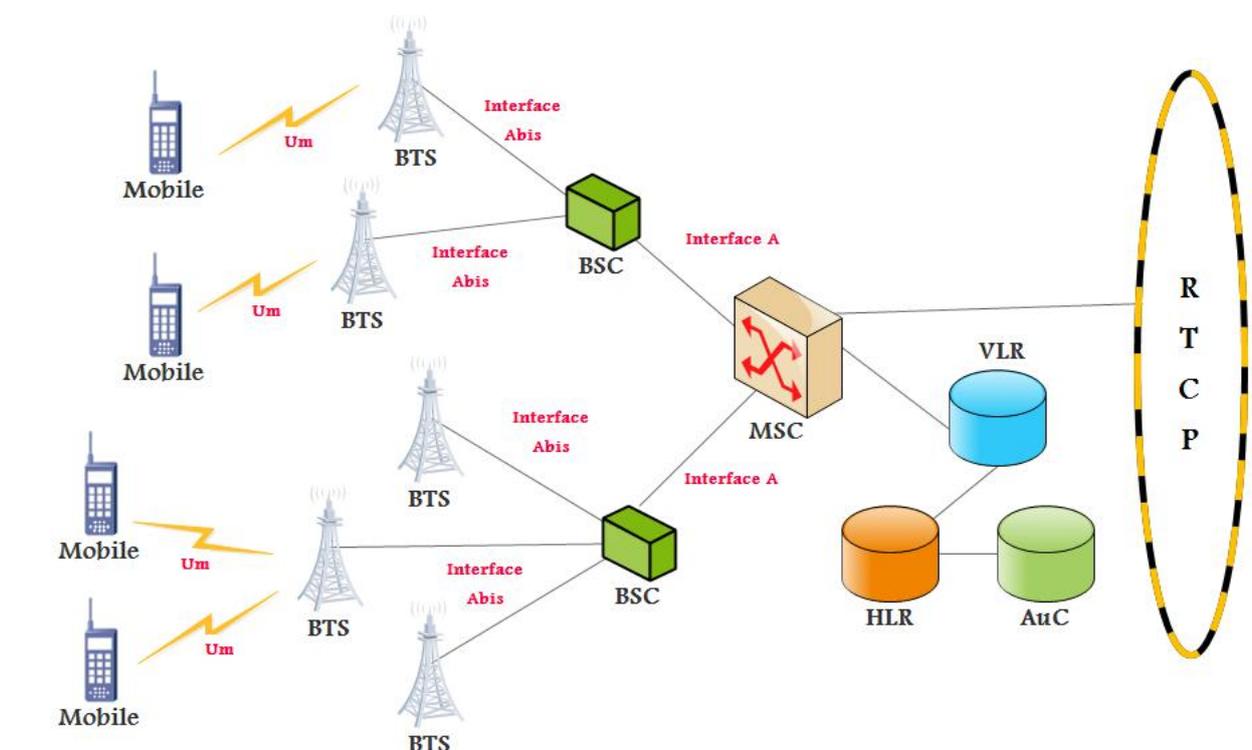


FIGURE I.1.1 – Architecture du GSM [1]

#### 1.2.1.2 Les composants

Le réseau de mobiles GSM est composé de deux sous-systèmes [1].

- Le sous-système radio BSS (Base Station Subsystem) : Il assure la transmission radioélectrique avec le mobile, gère les ressources radio et permet la mobilité du mobile. Il est constitué essentiellement des stations radioélectriques BTS, de contrôleurs de stations radioélectriques BSC et d'équipements de transcodage TRAU
- Le sous-système NSS : Il permet le traitement de l'appel pour l'établissement de la communication ainsi que la gestion de l'itinérance et de la mobilité. Le sous-système NSS est constitué des commutateurs téléphoniques MSC , GMSC (Gateway MSC) et TSC

(Tandem Switching Center) et des bases de données HLR (Home Location Register), AuC (Authentication Center) et EIR (Equipment Identity Register).

### **Le mobile**

Le mobile MS (Mobile Station) permet à l'utilisateur d'établir des communications téléphoniques ou d'envoyer et de recevoir des SMS. Il comprend un terminal et une carte SIM (Subscriber Identity Module). Chaque terminal est identifié de manière unique par un IMEI (International Mobile Equipment Identity) attribué par le constructeur.

La carte SIM identifie l'utilisateur, permet l'établissement de la communication et contient le numéro IMSI (International Mobile Subscriber Identity) identifiant l'abonné à l'intérieur du réseau GSM.

### **Le sous-système radio**

La BTS est un ensemble d'émetteurs-récepteurs ayant en charge la transmission radioélectrique avec les mobiles. Elle permet l'accès à la couche physique par :

- accès multiple à une fréquence en mode FDMA (Frequency Division Multiple Access)
- accès multiple à un intervalle de temps en mode TDMA (Time Division Multiple Access)

Le nombre de porteuses et le rayon de couverture d'une BTS dépendent du type de zone (urbaine ou rurale).

La BSC effectue le contrôle des BTS et utilise les mesures radioélectriques effectuées par les BTS pour contrôler les puissances émises par le mobile.

Elle gère aussi la mobilité à travers le handover.

Le TRAU est un équipement qui réalise le transcodage de la parole. Il permet la conversion du format G.711 à 64kbits dans le format utilisé dans le sous réseau radio.

### **Le sous-système réseau**

Le MSC est un commutateur temporel de circuits à 64 kbits/s. Il gère les messages de la signalisation échangés entre le mobile et le NSS. Il gère aussi le transfert des messages courts (SMS) et exécute le handover.

Le GMSC est un MSC qui effectue l'interface avec les réseaux téléphoniques fixes ou mobiles de autres opérateurs.

Le TSC est un commutateur temporel de circuits à 64 kbits/s permettant le transit entre deux commutateurs MSC.

Le VLR est une base de données qui mémorise les données de l'utilisateur présent dans la zone géographique couverte par un ou plusieurs MSC. Les données stockées par le VLR proviennent en partie du HLR et complétées par l'identité provisoire de l'utilisateur TMSI , par MSRN (Mobile Station Roaming Number) attribué à l'utilisateur dans le NSS et par des informations sur la localisation du mobile LAI.

Le HLR est une base de données qui gère les caractéristiques de chaque abonné :

- l'identité IMSI de l'abonné utilisé par le réseau
- le numéro d'annuaire de l'abonné MSISDN
- le profil de l'abonnement comme les services supplémentaires ou l'autorisation d'appel international

L'AuC est une base de données stockant une clé secrète utilisée pour l'authentification et le chiffrement de données d'un utilisateur.

L'EIR est une base de données contenant les identités des terminaux IMEI. Ces informations peuvent servir entre autres à retrouver les téléphones volés.

Le réseau de signalisation SS7 est un réseau de données transportant les messages échangés entre les équipements du sous-système réseau et concerne le protocole ISUP (ISDN User Part) pour le traitement d'appel, le protocole MAP (Mobile Application Part) pour la gestion de l'itinérance et de la mobilité et le protocole INAP (Intelligent Network Application Part) pour la gestion de l'interrogation d'un point de contrôle de service.

### 1.2.1.3 Le plan de fréquences

L'espacement entre les canaux est égal à 200 kHz, autorisant un nombre de canaux de 124 pour GSM 900, 374 pour DCS 1800.

TABLEAU I.1.1 – Plan de fréquences du réseau GSM

Système	Bande de fréquence	
	MS ⇒ BTS	BTS ⇒ MS
GSM 900 standard	890 - 915 MHz	935 - 960 MHz
DCS 1800	1710 - 1785 MHz	1805 - 1880 MHz

### 1.2.2 Le service GPRS

Le service que nous présentons ci-après permet l'ajout au GSM du sous-système GSS permettant l'accès de certains terminaux mobiles à Internet ou à des services de type USSD.

### 1.2.2.1 Architecture

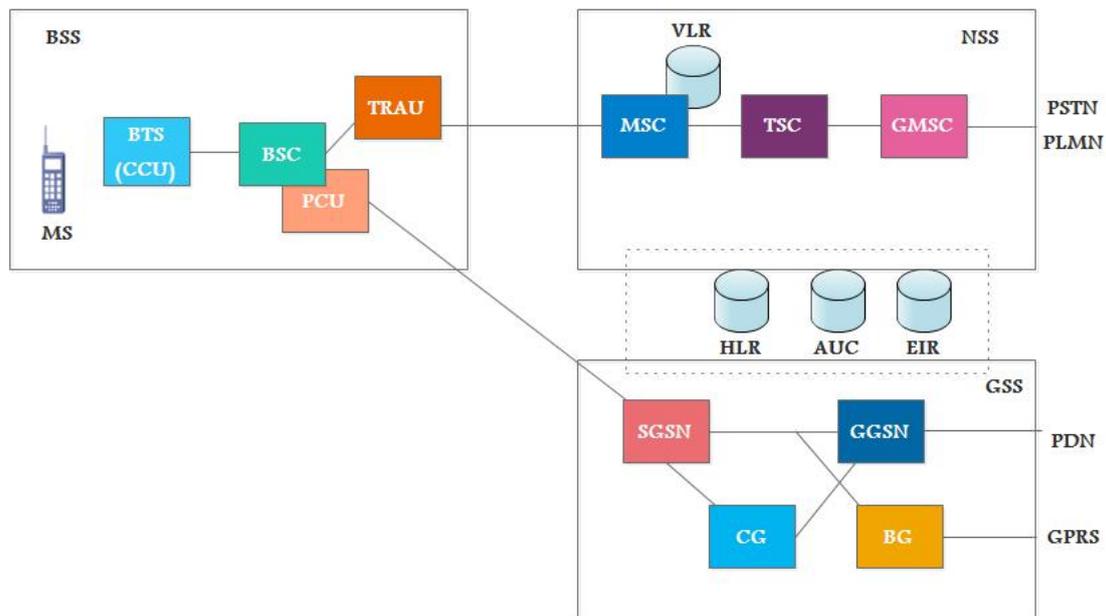


FIGURE I.1.2 – Architecture GPRS [1]

### 1.2.2.2 Les composants

L'introduction du GPRS [16] ne représente pas une mise à niveau importante sur l'infrastructure du réseau GSM existant. L'impact concerne essentiellement l'addition de nouvelles entités dans le réseau.

- le SGSN (Service GPRS Support Node) et le GGSN (Gateway GPRS Support Node) qui constituent le réseau GSS (GPRS Sub-System) et qui assurent le routage des paquets
- la fonction PCU (Packet Control Unit) introduite dans le BSS qui assure l'attribution de la ressource radioélectrique au mobile et l'interface avec le SGSN.

Le SGSN assure les fonctions de routage des paquets entre le BSS et le GGSN et de gestion des utilisateurs (gestion de l'itinérance et de la mobilité). Le GGSN assure le routage des paquets vers les réseaux de données externes (par exemple le réseau internet).

La gestion de la localisation est assurée par le HLR (Home Location Register) comme pour un réseau GSM en mode circuit. Il contient les informations de correspondance entre l'identifiant IMSI (International Mobile Subscriber Identity) et le contexte PDP (Packet Data Protocol) de l'utilisateur. Le contexte PDP contient les différents attributs de l'utilisateur (par exemple son adresse IP).

L'interconnexion entre le BSS et le SGSN est fournie par un réseau de données de type Frame Relay, offrant un débit maximal de 2Mbit/s. L'interconnexion entre les SGSN d'une part et le GGSN d'autre part est fournie par un réseau de données de type IP.

Le sous-système GSS est connecté aux différentes entités déployées pour le réseau GSM : BSS, MSC, HLR et aux réseaux extérieurs à travers les interfaces suivantes :

- l’interface Gb, entre le BSS (fonction PCU) et le SGSN, utilisant le protocole Frame Relay ;
- l’interface Gr, entre le SGSN et le MSC, utilisant une extension du protocole MAP (Mobile Application Part) ;
- l’interface Gs, entre le SGSN et le MSC, utilisant une extension du protocole BSSMAP, pour des opérations communes au GSM et au GPRS ;
- l’interface Gd, entre le SGSN et le SMSC, utilisant une extension du protocole MAP, pour l’envoi des SMS (Short Message Service) via le réseau GPRS ;
- l’interface Gc entre le GGSN et le HLR, utilisant une extension du protocole MAP ;
- l’interface Gi, entre le GGSN et les réseaux de données, utilisant le protocole IP.

### 1.2.2.3 L’évolution EDGE

EDGE [17] a été introduit dans les réseaux GSM et GPRS pour offrir :

- soit un service en mode circuit ECSD (Enhanced Circuit-Switched Data)
- soit un service en mode paquet EGPRS (Enhanced GPRS)

L’évolution de la fonction EDGE est l’augmentation du débit sur l’interface radioélectrique en conservant la même largeur de canal radioélectrique (200KHz). Le débit maximal offert est égal à 384kbit/s (théoriquement 473,6kbit/s) lorsque huit intervalles de temps sont concaténés. Cette augmentation de débit est rendue possible grâce à un nouveau type de modulation 8-PSK ( Phase Shift Keying à 8 états de phase) et à un nouveau mécanisme d’adaptation de la liaison vis-à-vis des perturbations.

## 1.2.3 Service SMS classique

### 1.2.3.1 Architecture SMS point à point

L’architecture de ce service [18] est constituée des entités suivantes :

- **Gateway MSC For Short Message Service (SMS-GMSC)** : Il s’agit d’une fonction capable d’une part de recevoir un message court d’une entité SMSC et d’interroger le HLR afin de déterminer la localisation de la station mobile destinataire et d’autre part de délivrer le message court au MSC auquel est rattaché cette station mobile destinataire. Ce MSC de rattachement est aussi appelé VMSC (Visited MSC).
- **Interworking MSC For Short Message Service (SMS-IWMSC)** : Il s’agit d’une fonction capable de recevoir un message court d’un MSC et de le soumettre à un SMSC.
- **Short Message Service Centre (SMSC)** : Cette fonction est responsable du stockage / relaying d’un message court.
- **Short Message Entity (SME)** est une entité extérieure au réseau GSM pouvant émettre / recevoir des messages courts. Il s’agit d’un serveur spécialisé ou d’un micro-ordinateur.

Généralement, les produits SMSCs vendus mettent en œuvre les fonctions SMS-GMSC, SMS-IWMSC et SMSC. Un produit SMSC comporte une interface normalisée côté réseau GSM (SMS-GMSC ou SMS-IWMSC) reposant sur le protocole de signalisation MAP (Mobile Appli-

cation Part) et une interface non-normalisée côté SME, SMPP (Short Message Peer to Peer). Les messages MAP sont transportés par le réseau SS7 (Signaling System n° 7).

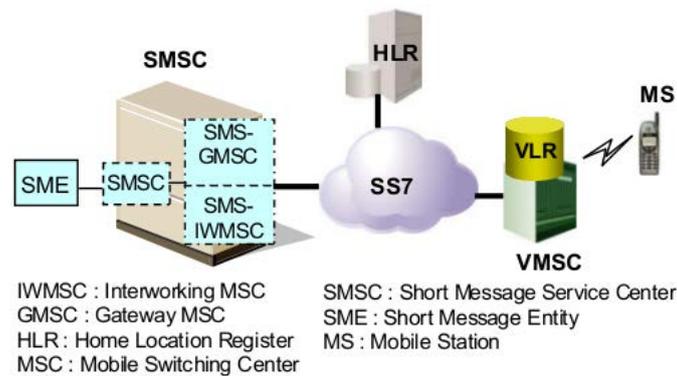


FIGURE I.1.3 – Les entités de base de base du service SMS

### 1.2.3.2 Procédure de transfert SMS point à point

Les procédures de transfert de messages courts sont similaires à celles relatives à l'établissement d'appels téléphoniques, à ceci près qu'aucun circuit de parole n'est réservé.

La transmission du message court est prise en charge par le réseau SS7.

Le service message court point-à-point consiste en deux services de base :

- SM MT (Short Message Mobile Terminated Point-to-Point) : Il dénote la capacité du réseau GSM à transférer un message court soumis par le SMSC à une station mobile et celle de fournir un rapport de livraison indiquant la bonne livraison ou toute erreur ayant pu survenir. Dans ce dernier cas, un mécanisme pour la livraison ultérieure du message court est prévu.
- SM MO (Short Message Mobile Originated Point-to-Point) : Il dénote la capacité du réseau GSM à transférer un message court soumis par la station mobile (MS, Mobile station) à une autre station mobile ou à un SME via un SMSC, et celle de fournir un rapport de livraison indiquant la bonne livraison ou toute erreur ayant pu survenir.

### Service SM-MO (Short Message Mobile Originated)

Avec le service SM-MO, la station mobile envoie un message court au SMSC. dans ce cas, le cheminement logique des messages courts est le suivant : MS → MSC → IWMSC → SMSC (Figure I.1.4).

Lorsque l'utilisateur mobile souhaite envoyer un message court, il doit indiquer l'adresse du destinataire et l'adresse du SMSC.

1. L'émetteur remet le message court à son MSC/VLR de rattachement (VMSC/VLR, Visited MSC/VLR) à travers la demande SMS-SUBMIT.
2. Le MSC émet un message MAP-SEND-INFO-FOR-MO-SMS à son VLR pour lui demander le numéro de téléphone (MSISDN, Mobile Station ISDN Number) de l'émetteur et pour vérifier qu'aucune restriction n'est imposée à cet émetteur.

3. Le VLR retourne alors une réponse MAP-SEND-INFO-FOR-SMS-ack.
4. Si la réponse est positive, le MSC émet le message MAP-MO-FORWARD-SHORT-MESSAGE à la fonction SMS-IWMSC à travers le réseau SS7. Ce message contient l'adresse du SMSC, les numéros MSISDN de l'émetteur et du destinataire, et le message court. Le message court est donc véhiculé dans une transaction MAP.
5. La fonction SMS-IWMSC le retransmet à son tour au SMSC. Le SMSC stocke le message et les adresses dans sa mémoire.
6. Le SMSC retourne une réponse (rapport de livraison) au SMS-IWMSC.
7. Ce rapport est inclus dans le message MAP-MO-FORWARD-SHORT-MESSAGE-ack retourné par le SMS-IWMSC au MSC.
8. Le MSC retourne à l'émetteur un message SMS-STATUS-REPORT.

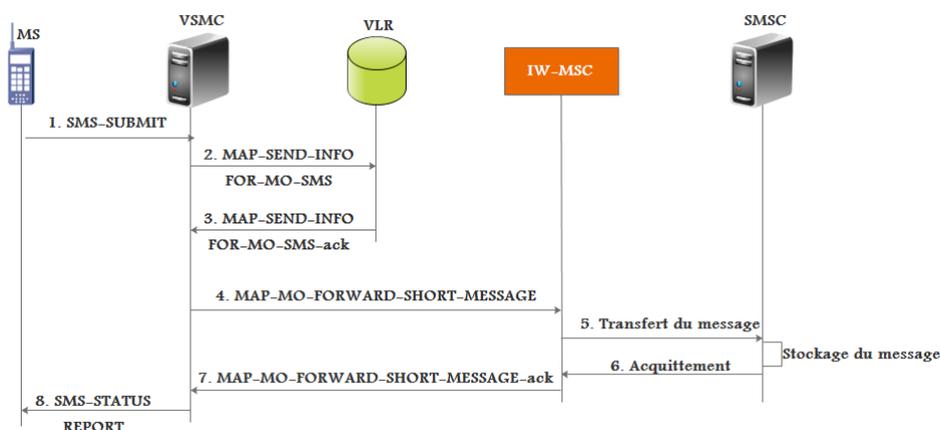


FIGURE I.1.4 – Service de base SM MO

### Service SM-MT (Short Message Mobile Terminated)

Avec le service SM MT, une station mobile reçoit un message court du SMSC. L'émetteur du message peut être une autre station mobile ou toute entité externe au réseau (un PC).

Le cheminement logique des messages courts dans le cas du service de base SM MT est le suivant : SMSC → SMS-GMSC → MSC → MS (Figure I.1.5).

Décrivons-le plus précisément :

1. La fonction SMS-GMSC reçoit un message court du SMSC.
2. Cette fonction demande des informations de routage du message au HLR à travers la requête MAP-SEND-ROUTING-INFO-FOR-SM, informations qui lui permettent de relayer le message au MSC approprié (MSC auquel est rattachée la station mobile destinataire). Cette requête contient notamment le numéro MSISDN du destinataire.
3. Le HLR utilise ce numéro pour rechercher les informations de routage qu'il retourne au SMS-GMSC à travers la réponse MAP-SEND-ROUTING-INFO-FOR-SM-ack. Cette réponse contient l'IMSI du destinataire et l'adresse du MSC de rattachement.
4. Le SMS-GMSC délivre le message court au MSC à travers une requête MAP-MT-FORWARD-SHORT-MESSAGE.

5. Le MSC émet la requête MAP-SEND-INFO-FOR-MT-SMS à son VLR en vue d'obtenir des informations relatives au destinataire. Le paramètre passé dans cette requête est l'IMSI du destinataire.
6. A partir de l'IMSI fourni par le MSC, le VLR identifie la zone de localisation (LA, Location Area) du mobile destinataire. Le VLR lance alors une procédure de paging (MAP\_\_PAGE), technique consistant à effectuer une recherche sur l'ensemble de la zone où est susceptible de se trouver le mobile demandé. Si le VLR ne connaît pas l'identité du destinataire, un message MAP-SEARCH-FOR-SUBSCRIBER est alors émis afin de lancer la procédure de paging sur toutes les LAs dépendant du MSC. Dans l'exemple montré à la figure I.1.5, l'identification du mobile destinataire est supposée connue. La procédure de paging est initiée par le VLR mais effectuée par le MSC.
7. Le MSC effectue la procédure de paging sur la zone de localisation du destinataire.
8. La station mobile destinataire répond positivement.
9. Le VLR retourne une réponse MAP-SEND-INFO-FOR-MT-SMS-ack au MSC, autorisant ce dernier à relayer le message court à la station mobile destinataire.
10. Le MSC achemine le message court au destinataire via le message SMS-DELIVER et reçoit un acquittement SMS-STATUS-REPORT.
11. Le MSC inclut ce rapport dans la réponse MAP-MT-FORWARD-SHORT-MESSAGE-ack retourné au SMS-GMSC.
12. Le SMS-GMSC passe le rapport au SMSC.

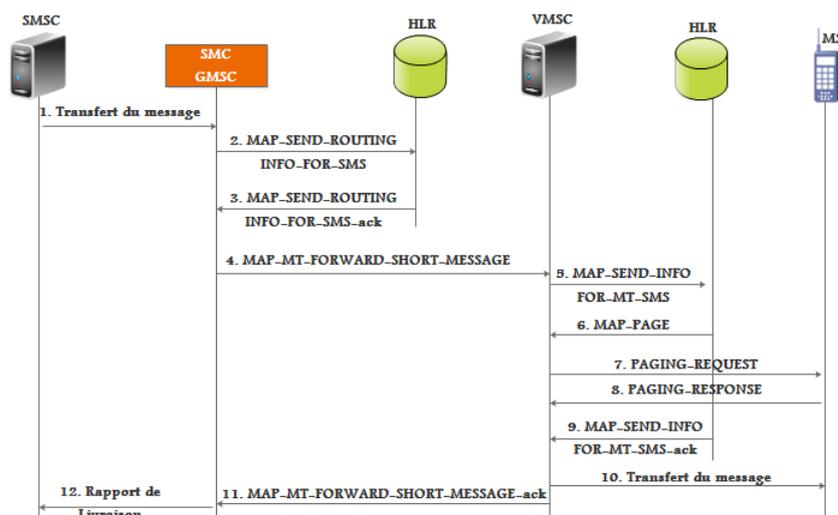


FIGURE I.1.5 – Service de base SM MT

## 1.3 Le réseau 3G ou UMTS et les évolutions HSPA

### 1.3.1 Le réseau 3G

Le réseau de mobile UMTS [19] fournit principalement les services téléphoniques en mode circuit (CS) et les services de transmission de données en mode paquet (PS).

Le réseau UMTS est composé de deux sous-systèmes :

- le sous-système NSS qui est commun aux réseaux GSM et à l'UMTS pour les services orientés circuit.
- Le réseau d'accès UTRAN a une topologie identique à celle du BSS des réseaux GSM

Le codec à débit variable AMR (Adaptative MultiRate) est utilisé.

#### 1.3.1.1 Architecture du réseau 3G

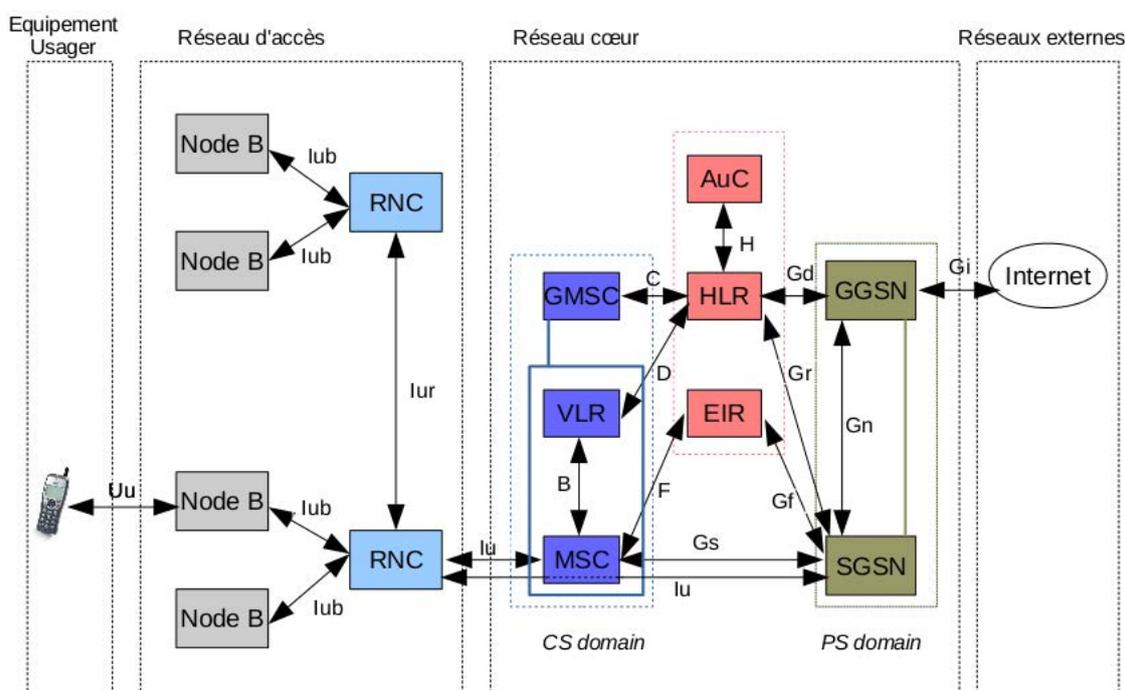


FIGURE I.1.6 – Architecture d'un réseau 3G

#### 1.3.1.2 Caractéristiques du réseau 3G

L'Universal Mobile Telecommunications System (UMTS) est un système cellulaire numérique de communication avec les mobiles ou entre mobiles, destiné à offrir une large gamme de services de voix, de données et d'images ainsi que l'accès à l'Internet.

Il présente ces caractéristiques :

- Il est basé sur la technologie WCDMA
- Il est standardisé par le 3GPP en Europe
- Il permet un débit théorique maximal de 1.920 Mbps avec une mobilité faible et proche de l'antenne et un débit de 384 Kbps en mouvement
- Il s'appuie sur de nouvelles antennes **Noeud B (Node B)** qui est l'équivalent de la BTS et un contrôleur de noeud nommé **RNC (Radio Network Controller)**

- Le Node B et le RNC forment la partie radio UMTS nommée **UTRAN (UMTS Terrestrial Radio Access Network)**
- Le terminal mobile **UE (User Equipment)** est constitué du **ME (Mobile Equipment)**, smartphone ou tablette, et de l'**USIM** qui représente l'abonnement souscrit.

### 1.3.1.3 Les classes de services

Dans l'UMTS, on définit les classes de services suivantes :

La classe de service conversationnelle réservée à la téléphonie ou la visioconférence. Dans cette classe, les trois paramètres de QoS que sont la perte, le retard et la gigue sont gérés.

La classe de service Streaming, est utilisée pour les signaux unidirectionnels en temps réel. Le retard n'est pas très important pour des applications utilisant cette classe de service mais par contre cette classe est très sensible aux erreurs de transmission contrairement à la classe de service conversationnelle.

La classe de service interactive est réservée aux applications pour lesquelles un utilisateur entretient un dialogue avec un serveur.

La classe de service arrière-plan est utilisée pour les applications de l'Internet (messagerie électronique, transfert de fichiers, ...)

## 1.3.2 Les évolutions HSPA de l'UMTS

L'évolution HSDPA (High Speed Downlink Packet Acces) permet d'augmenter le débit dans le sens descendant en agrégeant les canaux et en augmentant le nombre de bits par symbole pour la modulation. La ressource du sens descendant est partagée entre les différents mobiles.

L'évolution HSUPA(High Speed Uplink Packet Access) permet d'augmenter le débit dans le sens montant en agrégeant les canaux.

L'évolution HSPA+ conserve le mode de fonctionnement HSDPA et HSUPA mais utilise plutôt un système MIMO dans le sens descendant.

## 1.4 Le réseau NGN

A un moment donné de l'évolution des réseaux de télécoms, le MSC classique utilise le mode circuit pour fournir des services. L'interconnexion entre les entités du sous-système NSS est réalisé par un réseau de transmission SDH (Synchronous Digital Hierarchy).

Le réseau NGN contrairement au coeur du réseau NSS qui utilise un réseau de transmission SDH utilise un réseau IP pour l'interconnexion des entités dans le coeur du réseau.

### 1.4.1 Architecture du réseau NGN

Dans le réseau NGN [20], les noeuds MSC et GMSC(Gateway MSC) sont décomposés en deux entités, d'une part le MSC Server ou le GMSC Server et d'autre part les passerelles MGW(Media Gateway) et le SGW(Signalling Gateway)I.1.7.

### 1.4.2 Les composants du réseau

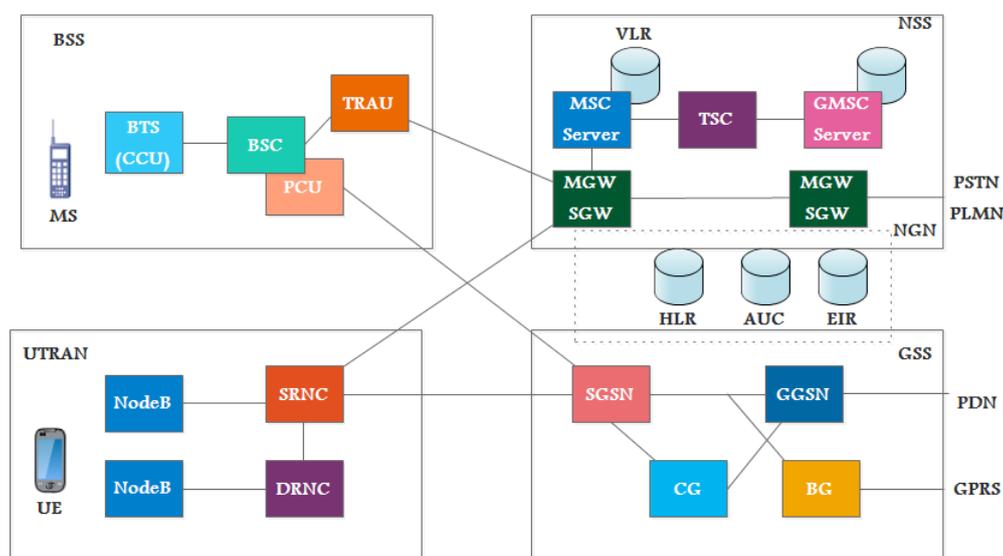


FIGURE I.1.7 – Architecture physique du réseau NGN [1]

Le MSC Server assure le traitement des fonctions de gestion de la communication et de la mobilité. Il termine la signalisation échangée entre les entités suivantes :

- Le mobile pour la signalisation CM (Call Management) ou MM (Mobility Management)
- les réseaux d'accès BSS pour la signalisation BSSAP (BSS Application Part) ou UTRAN pour la signalisation RANAP (Radio Access Network Application Part)
- Les réseaux PSTN ou PLMN pour la signalisation ISUP (ISDN User Part)

La signalisation BSSAP transporte d'une part la signalisation CM et MM échangée entre le MSC Server et le mobile, et d'autre part la signalisation BSSMAP (BSS Management Application Part) échangée entre le BSC et le MSC Server.

### 1.4.3 Le transport de la signalisation

Le modèle SS7 est appliqué avec le réseau d'accès BSS, ainsi qu'aux interfaces avec les réseaux tiers. Le modèle SIGTRAN (Signalling Transport over IP) est utilisé sur les interfaces internes du réseau NGN, entre les SGW et les MSC Server. La conversion entre le modèle SS7 et le modèle SIGTRAN est effectuée par le SGW.

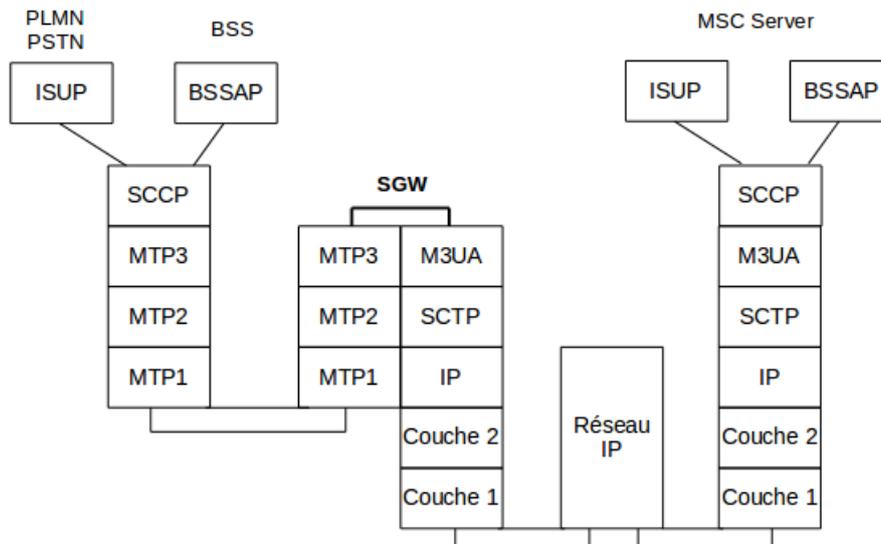


FIGURE I.1.8 – Le transport de la signalisation [1]

Le modèle SIGTRAN définit deux couches protocolaires de transport de la signalisation SS7 (Signalling System) sur IP :

- le protocole de transport SCTP (Stream Control Transmission Protocol)
- le protocole d'adaptation M3UA (MTP 3 User Adaptation) aux protocoles de signalisation échangés avec les réseaux d'accès ou les réseaux tiers. Le protocole M3UA peut être également utilisé pour le transport de la signalisation MAP (Mobile Application Part) échangée avec les bases de données (HLR, AuC, EIR) ou la signalisation CAP (CAMEL Application Part) utilisée par les entités du réseau intelligent.

#### 1.4.4 Le transport de la voix

Le MGW effectue une conversion de la structure des données multimédia (voix, vidéo, données) entre d'une part le réseau de données IP et d'autre part les réseaux d'accès tiers.

A l'interface avec le réseau d'accès BSS ou avec les réseaux tiers, la structure TDM (Time Division Multiplexing) multiplexe directement les données issues du codec G.711 dans une trame G.704.

A l'interface avec le réseau d'accès UTRAN, les données issues du codec AMR (Adaptative MultiRate) sont encapsulées par les protocoles AAL2 (ATM Adaptation Layer), ATM et SDH.

A l'interface des réseaux tiers, le MGW convertit également le format du codec AMR utilisé dans les réseaux mobiles UMTS en un codec G.711, normalisé pour l'interconnexion avec les réseaux PLMN et PSTN.

A l'interface du réseau de données IP, la voix est encapsulée par les protocoles RTP (Real-time Transport Protocol), UDP (User Datagram protocol) et IP (Internet Protocol).

L'utilisation du protocole UDP ne permet pas de détecter et corriger les pertes de paquets dans le réseau de données IP. Ce choix est cependant dicté par l'impossibilité d'adopter le protocole TCP qui régule le flux de la source, ce qui induit une réduction automatique du débit en cas de congestion.

Le protocole est RTP utilisé par les applications temps réel (voix, vidéo). Il est complémentaire du protocole UDP auquel il est associé. Il fournit les fonctions suivantes :

- L'identification du type de codec ;

- La numérotation des segments RTP ;
- L’horodatage des segments RTP ;

La contribution du protocole RTP à l’amélioration de la qualité vocale est double :

- il permet grâce à la numérotation des segments RTP, la détection de segments perdus. Un mécanisme propre à chaque codec permet de masquer cette perte à l’utilisateur ;
- il permet grâce à l’horodatage des segments RTP, la correction de la gigue produite par le réseau des données IP.

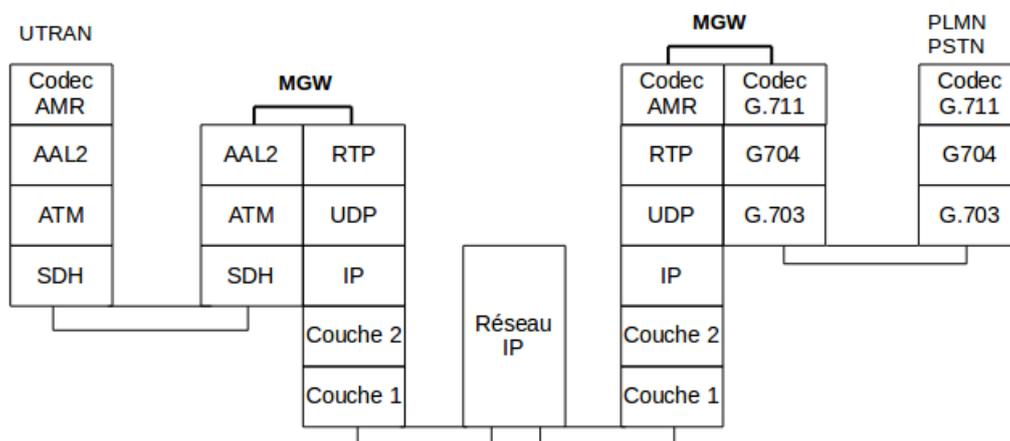


FIGURE I.1.9 – Le transport de la voix

[1]

## 1.5 Le réseau 4G : LTE (Long Term Evolution)

### Introduction

Le LTE (Long Term Evolution)[21] est un projet mené par l’organisme de standardisation 3GPP visant à rédiger les normes techniques de la quatrième génération en téléphonie mobile. Elle permet le transfert de données à très haut débit, avec une portée plus importante, un nombre d’appels par cellule supérieur et une latence plus faible.

Le LTE utilise des bandes de fréquences hertziennes d’une largeur pouvant varier de 1,4 MHz à 20 MHz, permettant ainsi d’obtenir (pour une bande 20 MHz) un débit binaire théorique pouvant atteindre 300 Mbit/s en liaison descendante à partager entre les utilisateurs mobiles d’une même cellule. La « vraie 4G », appelée LTE Advanced offrira un débit descendant pouvant atteindre 1 Gbit/s, ce débit nécessitera l’utilisation de bandes de fréquences de 2x100 MHz de largeur qui sont définies dans les versions 10 et 11 (3GPP release 10 et 11) de la norme LTE Advanced. La plage de fréquences supportées par le LTE et le LTE-Advanced est large, elle comprend celles historiquement attribuées au GSM et à l’UMTS (800 MHz à 2.6 GHz) et de nouvelles bandes, le tout s’étalant de 600 MHz à 3,8 GHz.

Pour les opérateurs, le LTE implique de modifier le cœur du réseau et les émetteurs radio. Il faut également développer des terminaux mobiles adaptés. En termes de vocabulaire, le futur réseau s’appelle EPS (Evolved Packet System). Il est constitué d’un nouveau réseau d’accès appelé LTE (Long Term Evolution) et d’un nouveau réseau cœur appelé SAE (System Architecture Evolution).

### 1.5.1 Architecture du réseau LTE

La figure I.1.10 ci-dessous permet d'avoir une bonne idée de l'architecture réseau LTE/EPC.

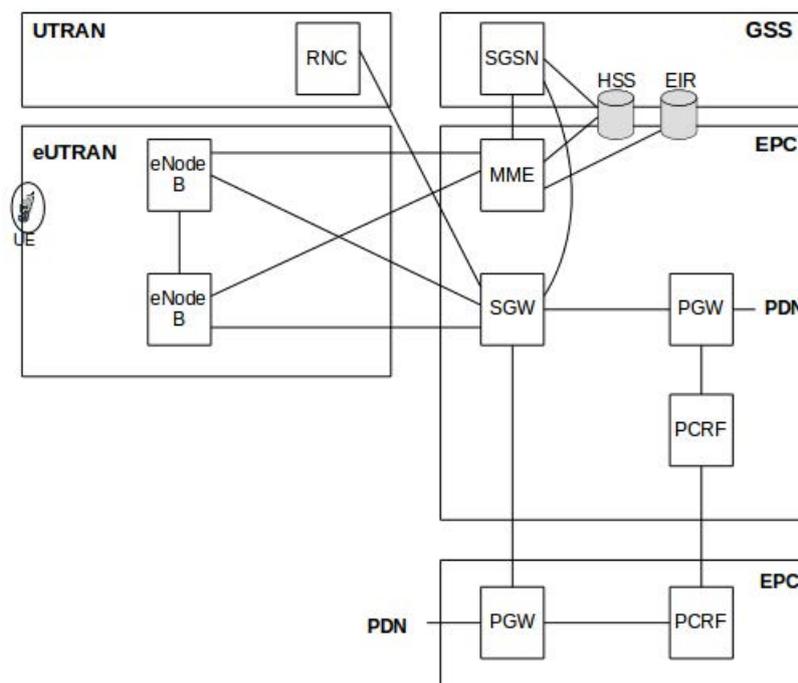


FIGURE I.1.10 – L'architecture physique du réseau EPS [1]

Les réseaux LTE sont des réseaux cellulaires constitués de milliers de cellules radio qui utilisent les mêmes fréquences hertziennes, y compris dans les cellules radio mitoyennes.

Ceci permet d'affecter à chaque cellule une largeur spectrale plus importante qu'en 3G, variant de 1.4 MHz à 20 MHz et donc d'avoir une bande passante plus importante et plus de débit dans chaque cellule.

Le réseau est constitué de deux parties : une partie radio eUTRAN (Evolved Universal Terrestrial Radio Access Network) ou tout simplement LTE et un cœur de réseau « EPC » (Evolved Packet Core). En plus des deux parties on peut ajouter le terminal utilisateur appelé ici User Equipment (UE).

#### 1.5.1.1 Le terminal ou User Equipment (UE)

L'architecture interne de l'UE pour le LTE est identique à celle utilisée par UMTS et le GSM qui est en fait un équipement mobile, ça peut être un téléphone (Smartphone), une tablette, une clé-modem USB ou tout autre type d'équipement fixe ou mobile (GPS, ordinateur, écran vidéo...). L'UE comprend les modules importants suivants :

- **Mobile Termination (MT)** :gère les fonctions de communication
- **Terminal Equipment (TE)** :gère les flux de données
- **Universal Integrated Circuit Card (UICC)** :aussi appelé carte SIM LTE (ou SIM 4G), il exécute une entité appelée Universal Subscriber identity Module (USIM). Un USIM stocke les données spécifiques à l'utilisateur très semblables à la carte SIM 3G. Il conserve des informations sur le numéro de téléphone de l'utilisateur, l'identité du réseau domestique, les clés de sécurité,...

Le 3GPP et l'ETSI dans les normes release 8 (version 8), ont défini cinq classes de terminaux LTE correspondant aux débits maximaux (montant et descendant) que doit supporter l'équipement, et au type d'antenne qu'il intègre. Tout terminal, quelle que soit sa catégorie, doit être capable de s'adapter aux six largeurs spectrales allant de 1,4 à 20 MHz, définies par le 3GPP. Les débits de données listés dans le tableau I.1.2 supposent une largeur de bande de 20 MHz.

TABLEAU I.1.2 – Caractéristiques des terminaux

<b>Catégories de terminaux LTE (3GPP release 8)</b>						
Catégorie		1	2	3	4	5
Débits crête (Mbit/s)	Descendant	10	50	100	150	300
	Montant	5	25	50	50	75
<b>Caractéristiques fonctionnelles</b>						
Bande passante radio		1,4 à 20MHz				
Modulations	Descendante	QPSK, 16QAM, 64QAM				
	Montante	QPSK, 16QAM		QPSK, 16QAM, 64QAM		
<b>Antennes</b>						
MIMO 2×2		Non	Oui			
MIMO 4×4		Non		Oui		

Le LTE consiste donc à envoyer et recevoir les données avec plusieurs antennes simultanément. La technologie MIMO (Multiple-Input Multiple-Output) est donc utilisée. Aujourd'hui, la technologie MIMO est présente dans certaines connexions WiFi, ainsi que pour certaines antennes 3G Dual-Carrier (jusqu'à 84 Mbps par cellule dans ce cas précis). L'acheminement des données dépendra de la (diversité spatiale ou multiplexage spatiale), mais en doublant le nombre d'antennes output (au niveau du relais) on double l'efficacité spectrale permettant ainsi de supporter plus d'utilisateurs. Du côté du terminal, si l'on double le nombre d'antennes en réception (input), on ira jusqu'à doubler les performances et le débit en downlink.

### 1.5.1.2 La partie radio eUTRAN

La partie radio du réseau, appelée « eUTRAN » ou LTE intègre le RNC (Radio Network Controller) du 2G (GERAN) et 3G (UMTS) à la station de base eNode B. Elle est donc composée des eNode B, d'antennes locales ou distantes, de liaisons en fibres optiques vers les antennes distantes et des liaisons IP reliant les eNode B entre eux (liens X2) et avec le cœur de réseau (liens S1) via un réseau de backhaul.

#### L'eNode B

L'eNodeB est capable de gérer plusieurs cellules logiques. Il gère entièrement le lien radio en effectuant les tâches suivantes :

- Modulation/Démodulation : il utilise, comme HSDPA (High Speed Downlink Packet Access) et WiMAX (Worldwide Interoperability for Microwave Access), les modulations QPSK (Quadrature Phase-Shift Keying), 16QAM (Quadrature Amplitude Modulation) et 64QAM ;
- Codage/décodage ;
- Contrôle des ressources : création, modification ou destruction de canaux ou bearers (dirigé par le MME), allocation de nouveaux canaux,...
- Gestion de la mobilité radio : décision de lancer une procédure de handover et choix de son type en utilisant l'interface X2, ou S1 ;
- Compression et chiffrement des en-têtes IP sur l'interface radio ;

- Gestion de la détection et correction d’erreur de transfert au niveau de la couche 2 du modèle OSI (Open Systems Interconnection) ;
- Sélection du MME de l’utilisateur au moment de sa connexion.

### 1.5.1.3 Le cœur du réseau EPC

Le cœur de réseau appelé « EPC » (Evolved Packet Core) utilise des technologies « full IP », c’est-à-dire basées sur les protocoles Internet pour la signalisation, pour le transport de la voix et des données. Ce cœur de réseau permet l’interconnexion via des routeurs avec les autres eNodeB distants, les réseaux des autres opérateurs mobiles, les réseaux de téléphonie fixe et le réseau Internet. L’utilisation du protocole IP de bout-en bout dans le cœur de réseau permet des temps de latence réduits pour l’accès internet et les appels vocaux LTE.

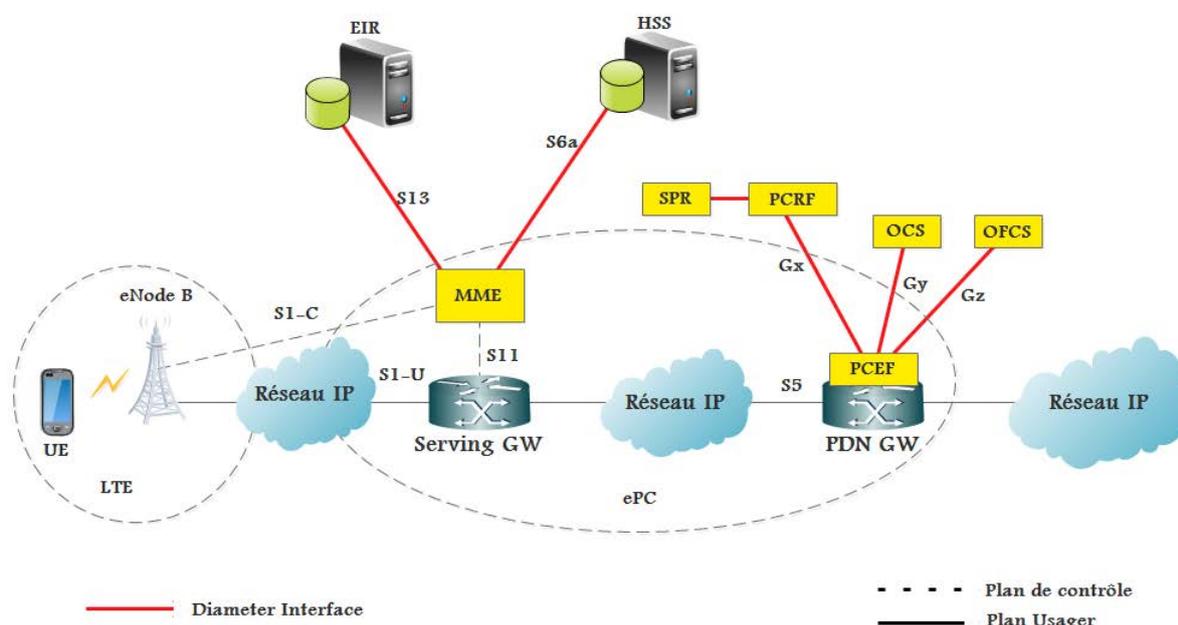


FIGURE I.1.11 – Architecture EPS

### La passerelle PDN (Packet Data Network)

La passerelle PDN (ou PGW, PDN Gateway) est le point d’ancrage IP qui fournit les points d’entrée et de sortie du trafic de données entre le réseau de l’opérateur mobile et l’équipement de l’utilisateur, autrement dit son appareil mobile. Elle contient les paramètres requis pour les canaux de transmission (bearers), assure la connexion de l’UE à l’Internet, au cœur IMS (IP Multimedia Subsystem) et aux autres réseaux de données. Elle incarne le volet services du réseau de l’opérateur mobile, où est réalisée une grande partie du traitement des paquets. Ses fonctions incluent, pour chaque UE et chaque service :

- la création et la suppression de bearers ;
- l’inspection des paquets et le filtrage (déterminer quel type de bearer doit être utilisé) ;
- la mise en œuvre des politiques de qualité de service et d’aide à la facturation associés à chaque bearer, en fonction des services achetés par l’utilisateur ;

- la comptabilisation du trafic de l'UE en vue de la facturation ;
- le reporting.

### La passerelle SGW (Serving Gateway)

La passerelle SGW assure les fonctions suivantes :

- Gestion de la mobilité de l'utilisateur (veiller à la continuité de la transmission des paquets même si l'utilisateur change d'eNODE) ;
- Routage et le transfert des paquets de données downlink vers l'eNodeB actuel de l'UE ;
- Participe configuration des canaux bearers et attribue les priorités ;
- Gère et stocke les informations de bearers ;
- Déclenche le paging (par exemple, notification et sortie de veille des ressources radio) lorsque des données en liaison descendante arrivent sur l'UE.

Il peut être lié à plusieurs PGW pour permettre aux utilisateurs d'être connectés simultanément à différents PDN. Dans le cas où l'UE est en mode **idle** (veille) le SGW peut être amené à mettre en cache les paquets downlink lui étant destinés, le temps pour le MME de « réveiller » l'UE et de construire les bearers radio.

### Le MME (Mobility Management Entity)

L'entité de gestion de mobilité ou MME fonctionne entièrement dans le plan de contrôle où elle émet et reçoit des instructions relatives au réseau, aux services et aux applications. Elle remplit plusieurs rôles clés tels que :

- Affecter chaque UE à la passerelle SGW qui convient ;
- Orchestrer l'établissement des bearers ;
- Gérer la négociation de la qualité de service (QoS) ;
- Suivre la trace de l'UE et affecter/optimiser les ressources réseau quand ce dernier se déplace (gère la mobilité) ;
- Gestion du roaming en mettant à jour la localisation de l'UE avec le HSS.
- Authentifier les UEs en communiquant avec le serveur d'abonnés résidentiels HSS (Home Subscriber Server) ;
- Gérer la sécurité et le cryptage des communications entre l'appareil de l'utilisateur et le réseau.
- Sélection du SGSN lors du handover avec les réseaux d'accès 2G et 3G.

### Le HSS (Home Subscriber Server)

Il permet de stocker des informations d'abonnement pouvant servir au contrôle des appels (qualité de service) et à la gestion de session des utilisateurs réalisé par le MME. Il entrepose, pour l'identification des utilisateurs, la numérotation et le profil des services auxquels ils sont abonnés. Il contient aussi les informations de souscription pour les autres réseaux, comme le GSM, le GPRS, l'UMTS et l'IMS. Il gère la mobilité du UE dans le sens où il enregistre leur localisation pour le MME ou d'autres nœuds de technologie différente comme le SGSN.

### EIR (Equipment Identity Register)

C'est la base de données comportant les informations de sécurité relatives à un mobile. C'est à partir de cet équipement qu'un opérateur peut bloquer un mobile volé.

## PCRF (Policy and Charging Rules Function)

Il gère la QoS et fournit les politiques de contrôle et de la facturation des flux IP au PGW afin que ce dernier puisse réaliser le blocage/l'autorisation/la dégradation des flux de service IP et la taxation de ces flux. Il se trouve sur le plan de contrôle et ne fait donc que notifier les autres nœuds des décisions qu'il prend.

## PCEF (Policy and Charging Enforcement Function)

L'entité PCEF dispose d'une interface de taxation avec l'OCS (Online Charging System) pour la taxation online des flux de services IP consommés par l'utilisateur et une interface avec l'OFCS (Offline Charging System) pour la taxation offline des flux de services IP de l'utilisateur. Le PCEF obtient des crédits de l'OCS et soumet des tickets de taxation à l'OFCS.

## 1.5.2 Qualité de service (QoS) et bearers

### 1.5.2.1 Qualité de service

La qualité de service (QoS) ou Quality of service (QoS) est la capacité de transmission dans de bonnes conditions d'un certain nombre de paquets dans une connexion entre un émetteur et un récepteur, et cela peut être présenté sous plusieurs termes tels que la disponibilité, le débit, le délai de transmission, la gigue, le taux de perte de paquets,...

Plusieurs applications peuvent être en cours d'exécution dans un UE à tout moment, chacune ayant des exigences de qualité de service (VoIP, navigation Web ou le téléchargement d'un fichier au même moment). La VoIP a des exigences plus strictes en matière de qualité de service en termes de retard et gigue que la navigation sur le Web et FTP, tandis que le second nécessite un taux de perte de paquets beaucoup plus faible. Afin de supporter les exigences de qualité de service, différents bearers sont mis en place au sein de l'EPS, chacun étant associé à une qualité de service. La QoS est fournie à l'utilisateur à travers les bearers par l'intermédiaire du PGW et du MME.

### 1.5.2.2 Les bearers

Les bearers sont des canaux logiques établis entre l'UE et la PGW et qui agrègent plusieurs flux data transportés entre les deux entités. Comme dans les réseaux UMTS, chaque service peut posséder un bearer, cependant plusieurs services ou connexions peuvent emprunter le même bearer qui lui, peut ne pas être alloué à un et un seul service. Dans le LTE il existe deux types de bearers :

- **Bearers par Défaut (Default Bearers)** : Chaque UE, à sa connexion crée son unique bearer par défaut, celui-ci est toujours présent, même quand l'UE est en mode idle. Tous les trafics provenant ou allant vers l'UE et qui n'appartiennent à aucun autre bearer, passent par lui.
- **Bearers Dédiés (Dedicated Bearers)** : Ils sont créés, modifiés ou détruits à la demande de l'UE ou du réseau. Ils sont généralement réservés à un service en particulier, mais rien n'empêche un opérateur de permettre que plusieurs connexions, plusieurs types de trafic ou plusieurs services passent par un même bearer. Il existe deux sous-types de bearer dédié :
  1. **Non-GBR (Guaranteed Bit Rate)** : Leur débit est limité par les Aggregate Maximum Bit Rate (AMBR).
  2. **GBR** : Ils possèdent des paramètres de QoS supplémentaires et leurs bandes passantes ne sont pas limitées par les AMBR.

La figure I.1.12 montre l'allocation de paramètres de QoS pour un service spécifique (voix sur IP, vidéo, ...).



FIGURE I.1.12 – Bearers et qualité de service

### 1.5.3 Les interfaces

La fourniture d'un réseau auto-optimisant est l'un des objectifs clés du réseau LTE. Il permet d'obtenir de meilleures performances et une meilleure rentabilité, en particulier dans des environnements à propagation radio changeante. C'est la pierre angulaire du système LTE dès ses débuts et est le concept autour duquel toutes les procédures ou interfaces S1 et X2 ont été conçus.

L'interface X2 connecte un eNodeB à un autre eNodeB et l'interface S1 relie un eNodeB à l'Evolved Packet Core (EPC). Les deux interfaces sont basées sur du IP et incluent des piles de protocole de plan usager et de plan de contrôle séparés.

Le plan de contrôle de X2 (X2-CP) permet la signalisation entre eNodeB tandis que le plan usager de X2 (X2-UP) permet le transfert de données d'application entre eNodeB.

Le plan de contrôle de S1 (S1-MME) permet la signalisation avec le MME alors que le plan usager de S1 (S1-UP) permet le transfert de données de l'application à travers la passerelle de service SGW. Les protocoles d'application ont été spécifiés pour définir les procédures de signalisation et les messages qui peuvent être envoyés à travers les interfaces X2 et S1 (c'est-à-dire X2-AP et S1-AP).

La figure I.1.13 ci-après présente les interfaces entre les certaines entités du réseau LTE/EPC.

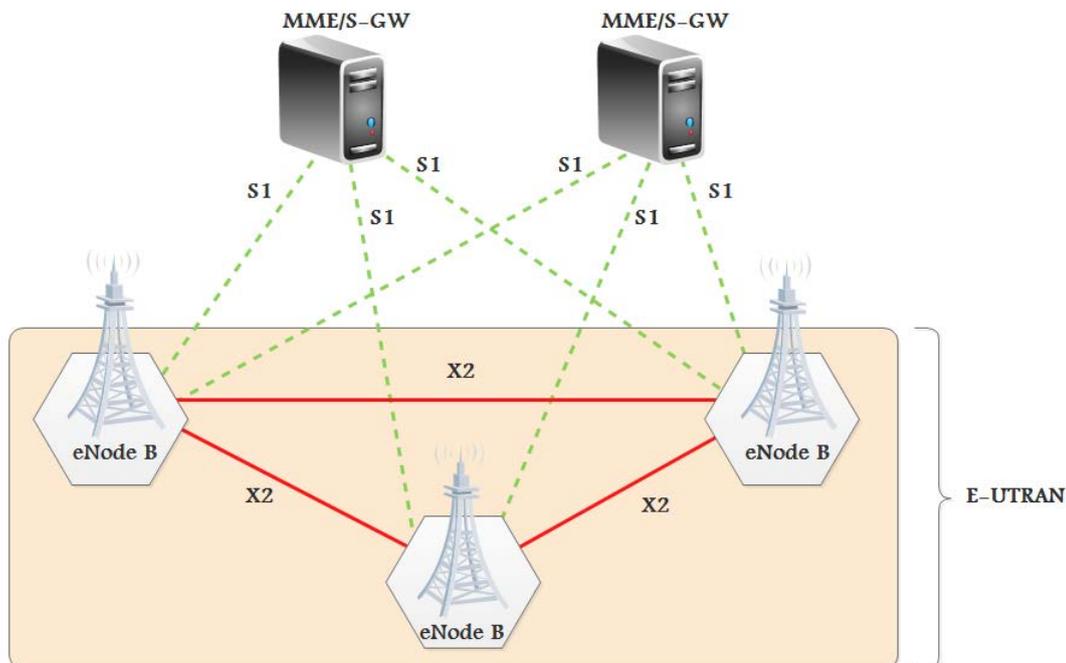


FIGURE I.1.13 – Interfaces LTE

### 1.5.4 Les protocoles radio

Le rôle des protocoles d'interface radio LTE est de mettre en place, reconfigurer et libérer les bearers radio qui fournissent les moyens de transfert aux bearers EPS. Les couches de protocole d'interface radio LTE au-dessus de la couche physique comprennent les protocoles de couche 2 : le contrôle d'accès au support ou Medium Access Control (MAC), le contrôle de liaison radio ou Radio Link Control (RLC) et le protocole de convergence de paquets de données ou Packet Data Convergence Protocol (PDCP). La couche 3 est constituée du protocole de commande de ressource radio ou Radio Resource Control (RRC) qui fait partie du plan de contrôle. La couche de protocole du dessus (pour le plan de contrôle) est le protocole NAS (Non Access Stratum) qui se termine dans la partie de réseau central.

### 1.5.5 La fonction CSFB

Le CSFB (Circuit Switched FallBack)[22] permet à un mobile attaché au réseau 4G de recevoir une alerte (paging) transmise par le réseau 2G/3G faisant suite à un appel entrant permettant ainsi d'établir la communication en mode circuit. De même, le mobile attaché au réseau 4G, et désirant établir un appel téléphonique sortant, doit, au préalable, être transféré sur le réseau 2G/3G en mode CS. A la fin de la communication, l'UE revient en mode paquet sur le 4G.

Pour que le Cœur de réseau 4G (EPC : Evolved Packet Core) soit compatible avec la technologie CSFB, il est nécessaire que ce dernier puisse communiquer avec le cœur de réseau en commutation de circuit CS-Core du réseau 2G/3G. En effet, le MME (mobility Management Entity) doit pouvoir contacter le MSC (Mobile Switch Center) et la VLR afin de donner procuration au réseau 2G/3G de la gestion de la mobilité. L'interface utilisée se nomme SGs, et fait référence, en reprenant son rôle, à l'interface Gs existante entre le SGSN et le MSC dans le réseau 3G.

Lorsque l'appel est accepté, la technologie CSFB utilise à nouveau l'interface SGs pour informer le

réseau LTE de l'acceptation de l'appel. L'acquittement est donc transmis par le réseau en Commutation de Circuit (CS) vers le réseau LTE en empruntant l'interface SGs.

La structure protocolaire des données échangées sur l'interface SGs est décrite à la figure I.1.14.

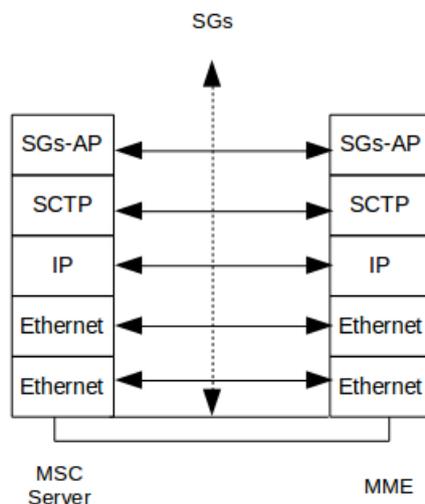


FIGURE I.1.14 – L'interface SGs : le lien entre les réseaux

Les messages SGs-AP permettent de remplir les fonctions suivantes :

- l'attachement du mobile auprès de l'entité MSC Server ;
- l'émission d'un *paging* par l'entité MSC Server pour avertir le mobile d'un appel téléphonique entrant.

## Conclusion

La technologie LTE (4G) qui est une évolution de la 3G apporte des améliorations significatives dans les télécommunications car il est maintenant possible avec la 4G d'obtenir du haut débit sur des terminaux mobiles. Elle offre une architecture simplifiée et une nouvelle technologie d'accès aux données ainsi qu'une infrastructure simple et facile à entretenir pour les opérateurs, en même temps elle a permis de proposer des nouvelles offres jusque-là impossible de mettre en œuvre. Pour les clients, la 4G a permis d'obtenir des débits élevés et de bénéficier de nouveaux services de qualité. Cependant le déploiement de la 4G reste inégal à travers le monde et cela entraîne un problème de comptabilité entre les pays. Aussi, les utilisateurs doivent s'équiper des terminaux capables de supporter la 4G, ce qui constitue une charge supplémentaire pour les clients. Une autre difficulté est que la 4G n'a été conçue que pour acheminer des données. La mise en œuvre de la voix sur LTE constitue donc un défi majeur à relever pour les opérateurs. Avec l'explosion de l'utilisation de nouveaux types de terminaux (Smartphones, tablettes...), donc de la consommation de données, le risque de congestion des réseaux 4G pourrait rapidement poser problème. Il est donc impératif d'augmenter les capacités actuelles de la LTE et de l'améliorer, c'est-à-dire d'augmenter le débit par cellule pour augmenter le débit moyen par utilisateur et permettre un grand nombre d'utilisateurs par cellule. Le LTE-Advanced constitue un espoir et un domaine de recherche dans ce sens et vers lequel les chercheurs doivent se lancer afin d'améliorer l'actuel LTE.

## 1.6 Le réseau IMS et la voix sur LTE (VoLTE)

La voix téléphonique constitue la plus grande partie sinon l'essentiel du chiffre d'affaire des opérateurs téléphoniques. Or il se trouve que le réseau EPS bien que fournissant un réseau haut débit à commutation de paquets n'a pas prévu la prise en compte native de la voix téléphonique mise à part la voix sur IP classique. A ce jour les appels téléphoniques passés sous couverture 4G exploitent la technologie 3G ou 2G. Il faut dire que le LTE repose entièrement sur un réseau IP, incompatible avec le fonctionnement des communications dites de circuit tel qu'elles existent depuis l'avènement du standard GSM. Il devient impératif pour les opérateurs de corriger ce problème en développant des méthodes pour pouvoir assurer la transmission des appels téléphoniques traditionnels en 4G.

### 1.6.1 L'IMS (IP Multimédia Subsystem)

Le sous-système multimédia IP ou IMS est un réseau 3GPP (Release 5) à part entière qui communique avec le cœur réseau EPC et avec les domaines paquet de l'UMTS et GSM (il fait abstraction du réseau d'accès) afin de contrôler les services temps réel IP multimédia tel que la voix sur IP [6]. L'IMS est un système complexe qui nécessite des investissements importants de la part de l'opérateur. Cependant il est très approprié pour l'acheminement de la voix et SMS sur LTE et peut être considéré comme une version sophistiquée du VoIP de 3GPP.

Ses composants les plus importants sont les fonctions de commande de session d'appel ou Call State Control Function (CSCF) dont il existe trois types :

- **S-CSCF** (Server Call State Control Function) : c'est le point d'accès dans le réseau IMS d'un UE. Il est chargé d'initier, de contrôler et de maintenir la session des terminaux.
- **P-CSCF** (Proxy Call State Control Function) : C'est le point d'entrée et de contact à IMS pour les UEs. Il transmet des messages de signalisation SIP au S-CSCF résidentiel de l'utilisateur, compresse les messages de signalisation que l'UE échange avec l'IMS de manière à réduire leur charge sur le réseau de transport de LTE et sécurise les messages en les cryptant et en protégeant leur intégrité. Il communique également avec le PCRF afin de garantir la qualité de service.
- **I-CSCF** (Interrogating Call State Control Function) : C'est le premier point de contact pour les messages de signalisation qui arrivent d'un autre IMS.

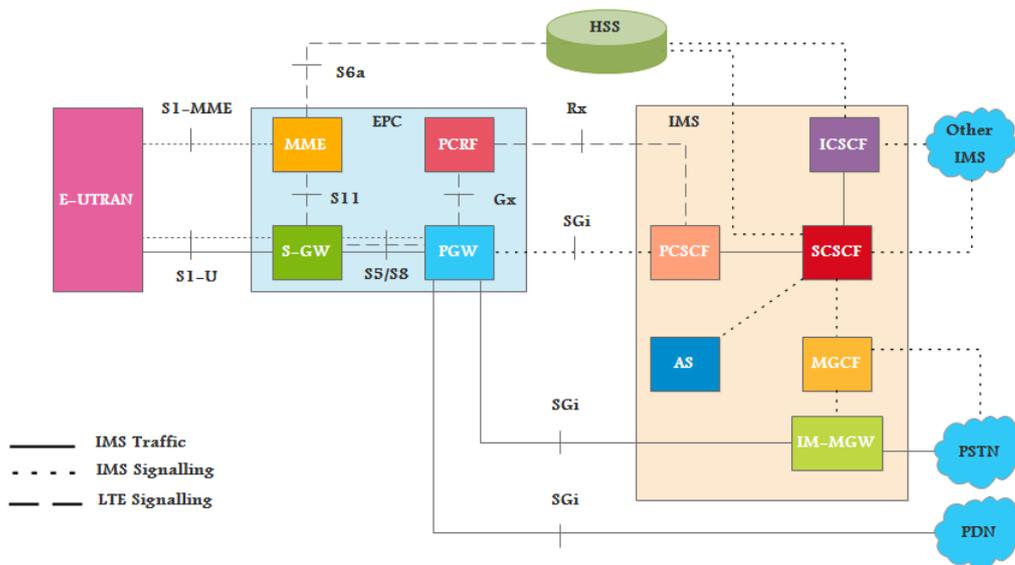


FIGURE I.1.15 – Positionnement du réseau IMS [2]

La passerelle IMS Multimedia Gateway (IMS-MGW) et la passerelle Media Gateway Control Function (MGCF) permettent à l'IMS de communiquer avec les réseaux à commutation de circuits traditionnels tels que le Réseau Téléphonique Commuté (RTC). La passerelle IMS-MGW convertit les flux VoIP en flux mode circuit circuits et vice-versa, tandis que le MGCF effectue la même tâche pour les messages de signalisation, contrôle la passerelle IMS-MGW et est elle-même contrôlée par le S-CSCF. Les serveurs d'application (AS) fournissent à l'utilisateur des services complémentaires tels que la messagerie vocale, et le serveur d'abonnés ou Home Subscriber Server (HSS) est une base de données centrale qui stocke les identités et les services auxquels les abonnés ont souscrits et est partagée avec le cœur réseau EPC. La figure I.1.15 ci-dessus montre l'interaction entre le réseau IMS et les autres types de réseaux des opérateurs. Le protocole de signalisation principal de l'IMS est le protocole d'initiation de session ou Session Initiation Protocol (SIP), un protocole de l'IETF qui est largement utilisé par d'autres systèmes VoIP. Les éléments du réseau IMS utilisent ce protocole pour communiquer entre eux et avec le l'UE pour réaliser des tâches telles que l'enregistrement de l'UE et la mise en place d'un appel. Les messages SIP peuvent contenir des informations intégrés écrites à l'aide du protocole de description de session (SDP), qui définit les propriétés des médias tels que le débit et les codecs pris en charge. L'UE s'identifie auprès de l'IMS en utilisant une identité privée. Il s'identifie au monde extérieur à l'aide d'un ou de plusieurs identités publiques, qui agissent comme des numéros de téléphone ou adresses e-mail. Ces identités sont stockées dans un module appelé ISIM (IP multimedia Services Identity Module), qui est une application de l'UICC (Universal Integrated Circuit Card) similaire à la carte USIM.

### 1.6.1.1 La fonction SRVCC

Le SRVCC [23][24] est une fonction particulière du réseau IMS qui assure le maintien de la communication en cas de perte de signal 4G par le terminal en transférant la communication en mode circuit sur du 2G ou 3G en cas de handover inter-système PS-CS (Packet Service-Circuit Service). C'est en fait un sous-ensemble de la fonction ICS (IMS Centralized Services) qui définit un contrôle unique de la signalisation téléphonique basé uniquement sur les mécanismes de l'IMS et est utilisé pour mettre

en place le support de la voix . Le SRVCC s'effectue en deux étapes :

- Le transfert de technologie d'accès radio en utilisant les protocoles définis
- Le transfert de la session nécessaire pour déplacer le contrôle d'accès et les médias de l'EPC vers le réseau circuit concerné.

### 1.6.1.2 L'impact du SRVCC sur les réseaux de mobiles

L'entité MME (Mobility Management Entity) du réseau EPS est impactée par la fonction SRVCC. Elle effectue les fonctions suivantes :

- elle sépare le support (bearer) de la voix des autres supports ne transportant pas la voix.
- elle initialise, via l'interface Sv, la procédure SRVCC pour le handover de la voix vers la cellule cible du réseau GSM ou UMTS ;
- elle coordonne le handover du mode PS vers le mode CS pour la voix et éventuellement le handover du mode PS vers le mode PS pour les autres flux

L'entité MSC (Mobile Switching Center) Server du réseau GSM ou UMTS est également impactée par la fonction SRVCC. Elle réalise les fonctions suivantes :

- elle s'assure de la disponibilité des ressources dans le réseau GSM ou UMTS avant d'exécuter le handover ;
- elle coordonne l'exécution du *handover* et le transfert de la communication téléphonique ;
- elle démarre la procédure de transfert de la communication téléphonique.

Le transfert de la communication téléphonique concerne le transfert de la signalisation téléphonique (figure I.1.16) et de la voix (figure I.1.17).

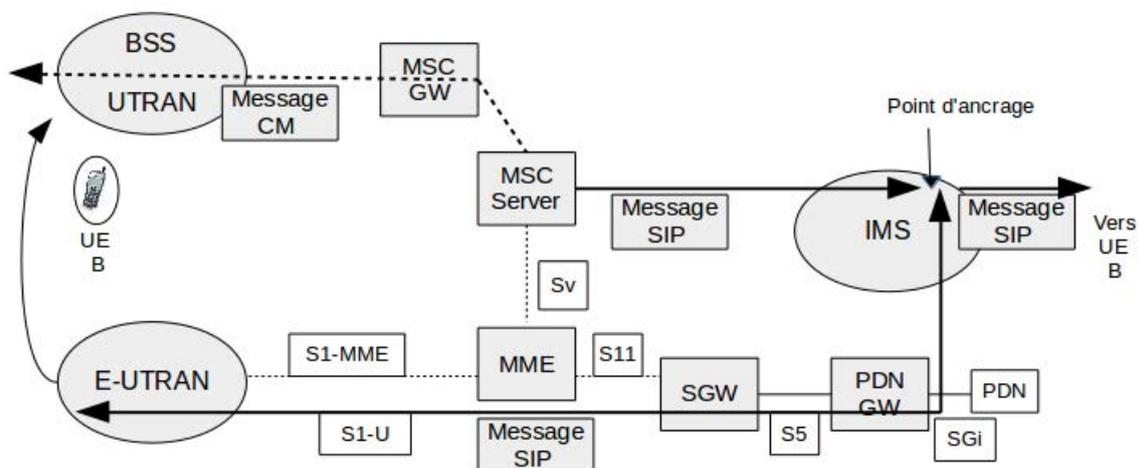


FIGURE I.1.16 – Le transfert de la signalisation téléphonique [3]

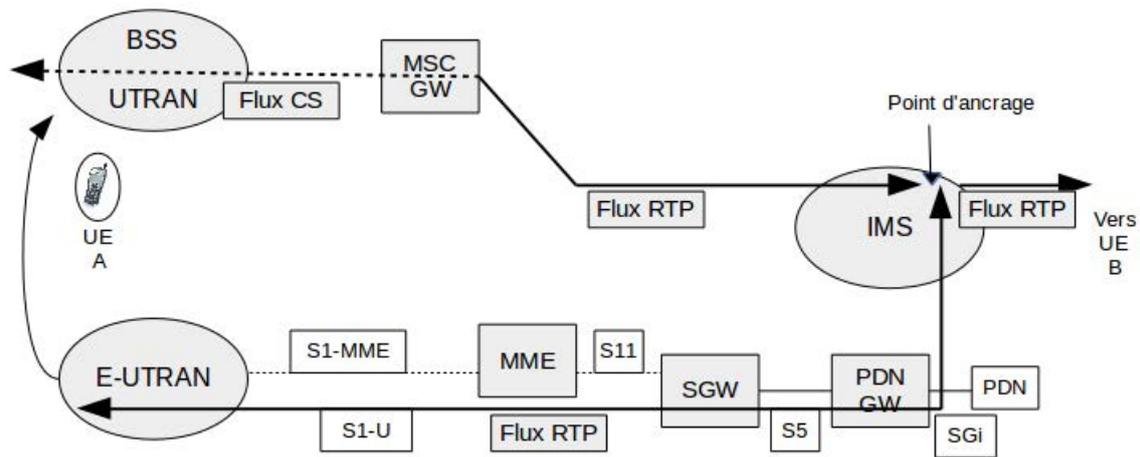


FIGURE I.1.17 – Le transfert de la voix [3]

Le transfert de la signalisation téléphonique consiste à transférer la signalisation SIP (Session Initiation Protocol) échangée entre le mobile et un point d'ancrage dans le réseau IMS vers la signalisation composée du protocole CM (Call Management) échangée entre le mobile et l'entité MSC Server, et du protocole SIP échangé entre l'entité MSC Server et le point d'ancrage dans le réseau IMS.

Le transfert de la voix consiste à transférer le flux RTP (Real-Time Protocol) établi initialement entre le mobile et le point d'ancrage dans le réseau IMS. Ce transport est effectué vers un aboutement du support CS et du flux RTP :

- le flux CS est établi entre le mobile et l'entité MSC GW (MSC Gateway)
- le flux RTP est établi entre l'entité MSC GW et le point d'ancrage.

### 1.6.1.3 L'impact de la fonction SRVCC sur le réseau IMS

Le mécanisme SRVCC se traduit au niveau du réseau IMS par l'introduction de trois entités (figure I.1.18) :

- le serveur d'application SCC AS (Service Centralization and Continuity Application Server) qui assure le contrôle du mécanisme SRVCC ;
- l'entité ATCF (Access Transfer Control Function) qui assure le point d'ancrage pour la signalisation SIP. L'entité ATCF est insérée dans le chemin de la signalisation entre les entités de contrôle de sessions CSCF (Call Session Control Function), d'une part l'entité P-CSCF (Proxy-CSCF) et, d'autre part, l'entité I-CSCF (Interrogating-CSCF) ou l'entité S-CSCF (Serving-CSCF) ;
- l'entité ATGW (Access Transfer Gateway) qui assure le point d'ancrage pour le flux RTP. L'entité ATGW est insérée dans le chemin de la voix.

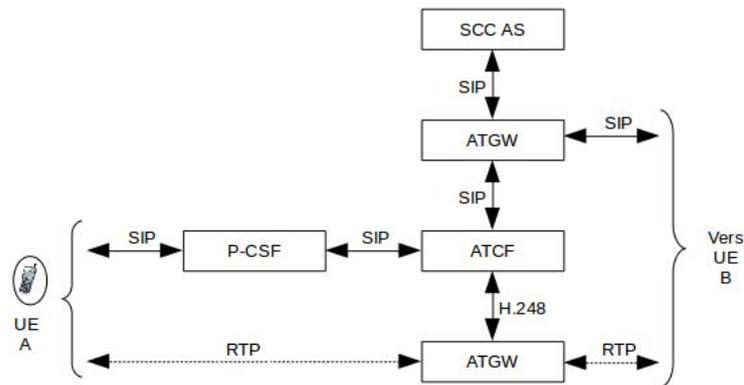


FIGURE I.1.18 – L'impact de de la fonction SRVCC sur l'architecture IMS

## 1.6.2 VoLTE

La VoLTE [25] consiste en une mise en œuvre par l'association du réseau EPS pour le transport des flux (voix et signalisation téléphonique) et du réseau IMS [IP (Internet Protocol) Multimedia Sub-system] pour le traitement de la signalisation téléphonique grâce à la fonction SRVCC décrite dans le paragraphe ci-dessus.

Le réseau EPS ne fournit donc pas le service téléphonique car il ne traite pas la signalisation. Il fonctionne en mode PS (Packet Service) et il effectue uniquement le transport de paquets IP. De ce fait, il transfère seulement les paquets IP contenant de la voix ou de la signalisation téléphonique.

Donc le réseau EPS se charge de la localisation des UE, d'établir les sessions et la création des bearers conformes aux QoS liés au contrat de l'abonné avec l'opérateur. Le réseau IMS de l'opérateur nominal de chaque mobile traite la signalisation téléphonique SIP, effectue le routage de la signalisation téléphonique et fournit les compléments de service téléphonique (comme le transfert d'appel). La voix est ainsi directement transférée entre les réseaux 4G.

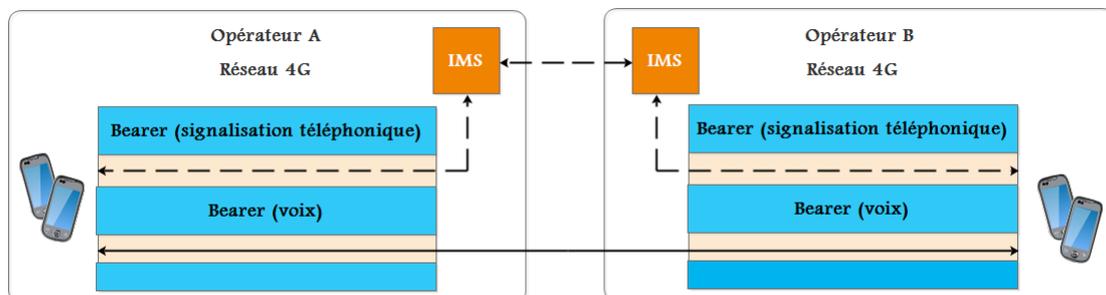


FIGURE I.1.19 – Communication entre deux mobiles 4G

Il est aussi possible d'établir une communication téléphonique avec les utilisateurs des réseaux PSTN/PLMN car l'IMS fournit l'entité MGCF (Media Gateway Control Function) qui permet le transcoding de la signalisation SIP en SIGTRAN et SS7 utilisés dans les réseaux fixe et le IMS-GW qui fait de même pour les flux VoIP.

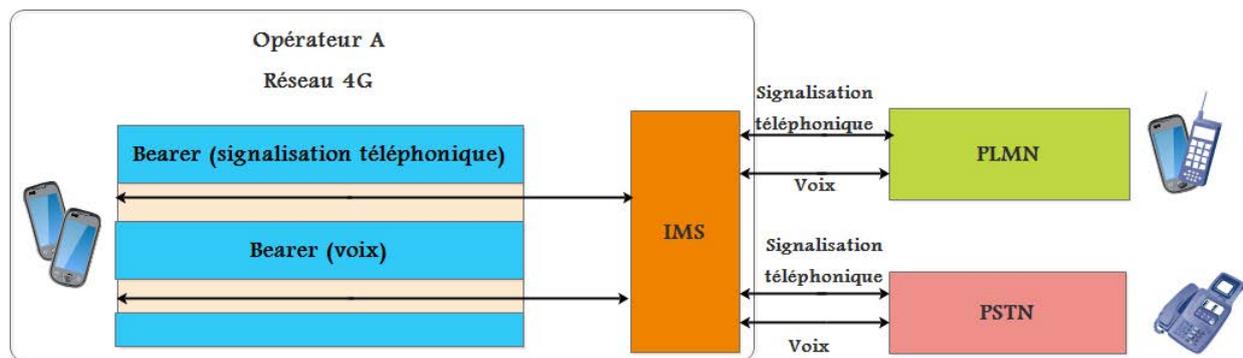


FIGURE I.1.20 – Communication 4G-RTCP

## 1.7 Conclusion

Ce chapitre a présenté les différentes évolutions des réseaux de mobiles. Nous pouvons noter que les évolutions consistent soit à modifier la partie accès ou la partie cœur.

Le NGN a consisté à modifier le mode de transport du cœur du réseau en passant du SDH en IP.

Le réseau UMTS a permis d'atteindre des débits de l'ordre de 2 Mbps rendant possible l'utilisation de la visiophonie entre autres.

Avec les besoins de plus en plus croissants des utilisateurs en termes de débits, le réseau LTE/EPC a été conçu en tout IP.

Ce réseau initialement n'avait pas pour vocation de gérer la signalisation téléphonique. Mais grâce au réseau IMS et l'introduction de la fonction SRVCC, la voix sur LTE est devenue une réalité.

La voix sur LTE impacte sur l'architecture du réseau IMS par l'introduction de trois entités qui sont SCC AS, ATCF et ATGW.

Il est à noter que c'est le serveur d'application SCC AS qui gère la procédure de SRVCC rendant possible la voix sur LTE. D'où l'importance d'avoir une maîtrise de la conception des serveurs AS à utiliser dans le plan service de l'IMS.

# Chapitre 2

## Étude avancée du réseau IMS : Architecture et services

### 2.1 Introduction

Le chapitre précédent nous a permis de nous rendre compte de l'importance de l'IMS dans le devenir des réseaux mobiles.

Ce réseau IMS utilise principalement le protocole SIP pour la signalisation.

Sa couche application permet l'accès à des services riches de communication et les services à valeur ajoutée.

Le présent chapitre présente le protocole SIP, l'architecture détaillée de l'IMS, l'architecture de services IMS, les modèles de taxation IMS et quelques résultats de développement de services IMS.

### 2.2 Étude du protocole SIP

Ce chapitre propose une première approche du protocole SIP (Session Initiation Protocol). Il présente une sélection simplifiée mais néanmoins complète des connaissances essentielles sur ce protocole et les applications qui en résultent.

#### 2.2.1 Historique de la normalisation de SIP

Le protocole SIP [5] a été normalisé pour la première fois en 1997 par le groupe de travail WG MMUSIC (Work Group Multiparty Multimedia Session Control) de l'IETF. En 1999, le même groupe proposa la deuxième version du protocole (RFC 2543). Celle-ci constitue la version majeure. Elle est a été largement revue, complétée et corrigée jusqu'en juin 2002 (RFC 3261). Des compléments au protocole ont été définis dans les RFC 3262 à 3265.

#### 2.2.2 Avantages du protocole SIP

##### 2.2.2.1 Compatibilité

L'un des grands atouts de SIP est sa capacité à s'intégrer à d'autres protocoles standards du monde IP. En tant que standard ouvert, il offre un service modulaire, prévu pour fonctionner avec différentes

applications, telles que la téléphonie, la messagerie instantanée, la vidéoconférence, la réalité virtuelle ou même le jeu vidéo.

SIP peut notamment se déployer ou s'intégrer aux protocoles suivants : RTP (Real-time Transport Protocol), qui se charge du transport des flux temps réel, RTCP (Real-time Transport Control Protocol), fournit des informations dynamiques sur l'état du réseau, RTSP (Real-Time Streaming Protocol), RFC 2326, pour contrôler la diffusion de flux multimédias en temps réel, SDP (Session Description Protocol), RFC 2327, qui fournit la description d'une session, c'est-à-dire les paramètres utilisés dans une communication SIP.

HTTP (HyperText Transfer Protocol), RFC 2616, pour le traitement des pages Web sur Internet (on peut inclure des adresses SIP directement dans des pages Web). MGCP (Media Gateway Control Protocol), pour le contrôle des passerelles assurant la connectivité entre un réseau IP et un réseau téléphonique.

### 2.2.2.2 Modularité

Le protocole SIP est indépendant de la couche transport. De ce fait, Les protocoles TCP et UDP sont en mesure de supporter l'envoi de messages SIP

### 2.2.2.3 Simplicité

SIP utilise un langage textuel très proche des protocoles HTTP et SMTP, ce qui facilite son intégration à Internet. Par comparaison, le protocole H.323 utilise ASN.1, qui est un langage compilé.

Les avantages d'un langage textuel sont les suivants :

- Les traitements de commandes sont rapidement interprétés car ils ne nécessitent pas une compilation préalable.
- L'implémentation de nouveaux services ne nécessite pas de compilateur pour interpréter les commandes.

## 2.2.3 Architecture du protocole SIP

L'architecture du protocole SIP s'articule principalement autour de cinq entités : terminal utilisateur, serveur d'enregistrement, serveur de localisation, serveur de redirection, serveur proxy. Cette architecture est entièrement logicielle.

### 2.2.3.1 Terminal

Le terminal est l'élément dont dispose l'utilisateur pour appeler et être appelé. Il permet de composer des numéros de téléphone. Il peut se présenter sous la forme d'un composant matériel (un téléphone) ou d'un composant logiciel (un programme lancé à partir d'un ordinateur).

Le terminal est appelé UA (User Agent). Il est constitué de deux sous-entités :

- Une partie cliente, appelée UAC (User Agent Client), chargée d'émettre les requêtes. C'est l'UAC qui initie un appel.
- Une partie serveur, appelée UAS (User Agent Server), qui est en écoute, reçoit et traite les requêtes. C'est l'UAS qui répond à un appel.

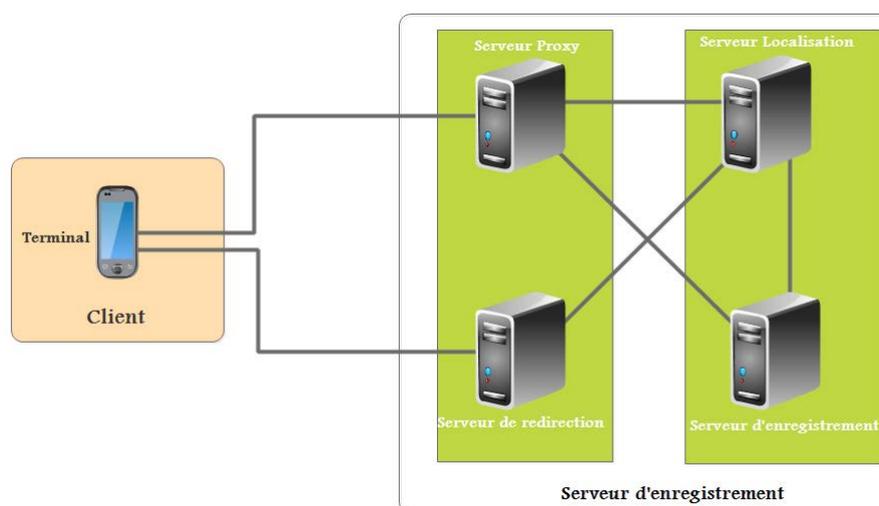


FIGURE I.2.1 – Architecture de SIP [4]

### 2.2.3.2 Serveur d'enregistrement

Deux terminaux peuvent communiquer entre eux sans passer par un serveur d'enregistrement, à la condition que l'appelant connaisse l'adresse IP de l'appelé. Cette contrainte est fastidieuse, car un utilisateur peut être mobile et donc ne pas avoir d'adresse IP fixe.

En outre, l'adresse IP peut être fournie de manière dynamique par un serveur DHCP.

Le serveur d'enregistrement (Registrar Server) offre un moyen de localiser un correspondant avec souplesse, tout en gérant la mobilité de l'utilisateur.

### 2.2.3.3 Serveur de localisation

Le serveur de localisation (Location Server) joue un rôle complémentaire par rapport au serveur d'enregistrement en permettant la localisation de l'abonné.

Ce serveur contient la base de données de l'ensemble des abonnés qu'il gère. Cette base est mise à jour par le serveur d'enregistrement.

### 2.2.3.4 Serveur de redirection

Le serveur de redirection (Redirect Server) agit comme un intermédiaire entre le terminal client et le serveur de localisation. Il est sollicité par le terminal client pour contacter le serveur délocalisation afin de déterminer la position courante d'un utilisateur.

### 2.2.3.5 Serveur Proxy

Le serveur proxy s'occupe de la localisation d'un correspondant, initie, maintient et termine une session vers un correspondant. Il peut également effectuer des traitements spécifiques sur des requêtes.

## 2.2.4 L'adressage SIP

L'adressage consiste à localiser les utilisateurs dans un réseau. C'est une des étapes indispensables pour permettre à un utilisateur d'en joindre un autre.

Pour localiser les utilisateurs, il faut pouvoir les identifier de manière unique. SIP propose des moyens très performants pour nommer les utilisateurs, grâce au concept d'URI, classique sur Internet. Un URI (Universal Resource Identifier) définit une syntaxe permettant de désigner de manière unique, formelle et normalisée une ressource, qu'il s'agisse d'un document textuel, audio, vidéo ou plus généralement d'une entité logique ou physique.

Un URI est formé d'une chaîne de caractères. Sa syntaxe a été normalisée par l'IETF en août 1998 dans la RFC 2396 puis révisée de nombreuses fois, notamment dans la RFC 2396bis, et reprise en janvier 2005 dans la RFC 3986.

Le format d'une adresse SIP se présente comme suit :

```
sip :identifiant[ :mot_de_passe]@serveur[ ?paramètres]
```

- Le mot-clé sip spécifie le protocole à utiliser pour la communication.
- La partie identifiant définit le nom ou le numéro de l'utilisateur.
- La partie mot\_de\_passe est facultative. Il est obligatoire lorsqu'on veut s'authentifier auprès d'un serveur.
- La partie serveur spécifie le serveur chargé du compte SIP dont l'identifiant précède l'arobase. Le serveur est indiqué par son adresse IP ou par un nom qui sera résolu par DNS.
- La partie paramètres est facultative. Les paramètres permettent soit de modifier le comportement par défaut (par exemple, en modifiant les protocoles de transport ou les ports, ou encore le TTL par défaut)

## 2.2.5 Méthodes et codes d'état

Une réponse à une requête (méthode) est caractérisée, par un code et un motif, appelés **code d'état** et **reason phrase** respectivement.

### 2.2.5.1 Codes d'état

Un code d'état est un entier codé sur 3 chiffres indiquant un résultat à l'issue de la réception d'une requête. Ce résultat est précisé par une phrase, text-based (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. On distingue actuellement six classes de codes d'état :

- Codes 1xx : réponses informatives
- Codes 2xx : réponses réussies
- Codes 3xx : réponses de redirection
- Codes 4xx : erreurs du client
- Codes 5xx : erreurs du serveur
- Codes 6xx : erreurs générales

### 2.2.5.2 Principales méthodes

#### INVITE (Invitation à une session)

La méthode INVITE indique que l'application ou utilisateur est invité à participer à une session. Le **Corps du message** décrit cette session (média supportés par l'appelant entre autres). En cas de réponse favorable à l'invitation, l'invité doit spécifier également les médias qu'il supporte dans son

Corps du message. Un serveur est tenu de répondre à une telle invitation, identifiée par son CALL-ID, par une **réponse OK (code 200)**.

### **ACK (Acquittement de messages)**

La méthode ACK confirme que le client a reçu une réponse définitive à une requête INVITE. Les réponses de code 2xx sont acquittées par les UAC et les autres types de réponses définitives par les premiers PS ou UAC les ayant reçues. La requête ACK possède, dans son Corps de message, une description définitive de la session que doit utiliser l'appelé.

### **OPTIONS (Indication de paramètres)**

La méthode OPTIONS est utilisée afin d'interroger les capacités et l'état d'un User agent ou d'un serveur. La réponse contient ses capacités (type de média étant supporté, méthodes supportées, langue supportée) ou le fait que l'UA soit indisponible. Si l'UAS ne supporte pas cette méthode, il renvoie une **réponse Busy (code 600)** au PS.

### **BYE (Mettre fin à une session)**

La méthode BYE est utilisée par l'UAS de l'appelé pour signaler au PS local qu'il ne souhaite plus participer à la session. Si la requête INVITE contient une URL CONTACT, l'appelé envoie la requête BYE directement à cette adresse plutôt qu'à l'URL FROM. Cette méthode doit être supportée par les PS et devrait être supportée par les RS et UAS.

### **REGISTER (Enregistrement d'une entité auprès du serveur Registrar)**

C'est la méthode utilisée par le client pour enregistrer l'adresse listée dans l'URL TO par le serveur auquel il est relié. Les requêtes sont traitées par le client dans l'ordre où elles arrivent et celui-ci devra éviter d'envoyer une nouvelle requête REGISTER tant qu'il n'aura pas traité la précédente. Le client doit définir une adresse d'enregistrement du type Utilisateur@Domaine . Cette méthode assure un service de localisation.

### **CANCEL (Mettre fin à une session pendante)**

La requête CANCEL est envoyée par un UAC ou un PS pour annuler une requête non validée par une réponse finale d'état. Par exemple, un PS peut envoyer une requête CANCEL aux destinataires n'ayant pas retourné une réponse finale après avoir reçu une réponse 2xx ou 6xx du PS.

#### **2.2.5.3 Extensions du protocole SIP**

### **SUBSCRIBE**

Une entité SIP peut souscrire à un événement afin d'être notifiée de son occurrence. La requête SUBSCRIBE permet la souscription alors que la requête NOTIFY est utilisée afin de notifier (RFC3265). La méthode PUBLISH permet quant à elle de publier son état.

## REFER

La méthode REFER (RFC3515) renvoie le récepteur vers une ressource identifiée dans la méthode. REFER permet d'émuler différents services ou applications dont le transfert d'appel.

## MESSAGE

La méthode MESSAGE (RFC3428) a été proposée comme extension au protocole SIP afin de permettre le transfert de messages instantanés. La messagerie instantanée (IM, **Instant Messaging**) consiste en l'échange de messages entre usagers en pseudo temps réel. Cette nouvelle méthode hérite de toutes les fonctions offertes par le protocole SIP telles que le routage et la sécurité. La requête MESSAGE peut transporter plusieurs types de contenus en s'appuyant sur le codage MIME.

## INFO

La méthode INFO (RFC2976) permet de transférer des informations de signalisation durant l'appel. Parmi les exemples d'information figurent les digits DTMF, les informations relatives à la taxation d'un appel, des images, etc.

## PRACK

Les réponses finales 2XX, 3XX, 4XX, 5XX et 6XX à une requête INVITE sont acquittées par la requête ACK alors que les réponses provisoires de type 1XX ne sont pas acquittées. Or, certaines réponses provisoires telles que 180 Ringing sont critiques et leur réception est essentielle pour la détermination de l'état de l'appel, notamment lors de l'interconnexion avec le RTCP. La méthode PRACK (RFC3262) a donc été définie afin d'acquitter la réception de réponses provisoires, de type 1XX.

## UPDATE

La méthode UPDATE (RFC3311) permet à un terminal SIP de mettre à jour les paramètres d'une session multimédia. La méthode UPDATE peut être envoyée avant que la session soit établie. UPDATE est donc particulièrement utile lorsqu'il s'agit de mettre à jour des paramètres de session avant son établissement.

## 2.3 Présentation de l'IMS

Le but de ce paragraphe est de présenter les instances de normalisation de l'IMS, les architectures de réseau et de service IMS avec les concepts sous-jacents, les entités impliquées et leurs fonctionnalités.

### 2.3.1 Historique de la normalisation de l'IMS

Le développement de l'IMS s'est appuyé sur les différents réseaux existants. Sa conception a tenu compte des différentes technologies. Il s'est imprégné des tendances en cours, des besoins et des recommandations.

Au départ, la question était de savoir comment faire évoluer les réseaux cellulaires, c'est à-dire, déterminer les protocoles et architectures des futurs réseaux de télécommunications.

Le problème est que, suivant l'endroit où l'on se trouve, les standards utilisés dans les réseaux cellulaires sont différents.

A cet effet, l'organisme de normalisation international IMT-2000 de l'UIT a constitué en 1998 deux groupes de travaux : le 3GPP, en charge de réfléchir aux évolutions de la norme GSM, et le 3GPP2 (Third Génération Partnership Project 2), en charge de réfléchir aux évolutions de la norme CDMA 2000.

Le 3GPP a la particularité de référencer ses documents de travail sous le nom de spécifications techniques, ou TS (Technical Specifications), et de rapports techniques (ou TR (Technical Reports)). Lorsqu'un ensemble de TS et de TR est complet et cohérent pour former un tout décrivant exhaustivement les fonctionnalités attendues, l'ensemble est classé sous le nom de Release. L'IMS a été défini dans la Release numéro 5 du 3GPP.

En 2004, quatre groupes de travaux de l'ETSI (SPAN, Signaling System Ndeg 7, R1 et Typhon) ont fusionné pour donner naissance au groupe de travail TISPAN (Telecommunications and Internet converged Services and Protocols for Advanced Networking), qui œuvre dans le cadre du 3GPP pour l'IMS.

En 2005, TISPAN a proposé une extension importante de l'IMS à destination des réseaux filaires, lesquels, avant cela, n'étaient pas concernés par l'IMS. La première version de l'IMS adoptée par l'ETSI a été proposée cette même année. Ces travaux ont été repris et incorporés dans l'IMS dès 2006, avec la publication de la Release 6 de l'IMS. La deuxième version a été proposée en 2007 (ajout de la VoD, d'IPTV et de la QoS). La troisième (ajout de la mobilité généralisée et de la convergence) vit le jour en 2009.

Parallèlement au développement de l'architecture IMS, un autre groupe, créé en 2002 et nommé OMA (Open Mobile Alliance), s'est impliqué dans le développement de l'IMS pour la définition de services plus évolués que ceux prévus par le 3GPP et le 3GPP2.

### 2.3.2 L'architecture IMS

IMS repose sur une architecture modulaire qui permet de distinguer des niveaux de traitements différents. Quatre couches peuvent être identifiées (accès, transport, contrôle, application), chacune d'elles étant liée à un domaine spécifique comme le montre la figure I.2.2 ci-dessous.

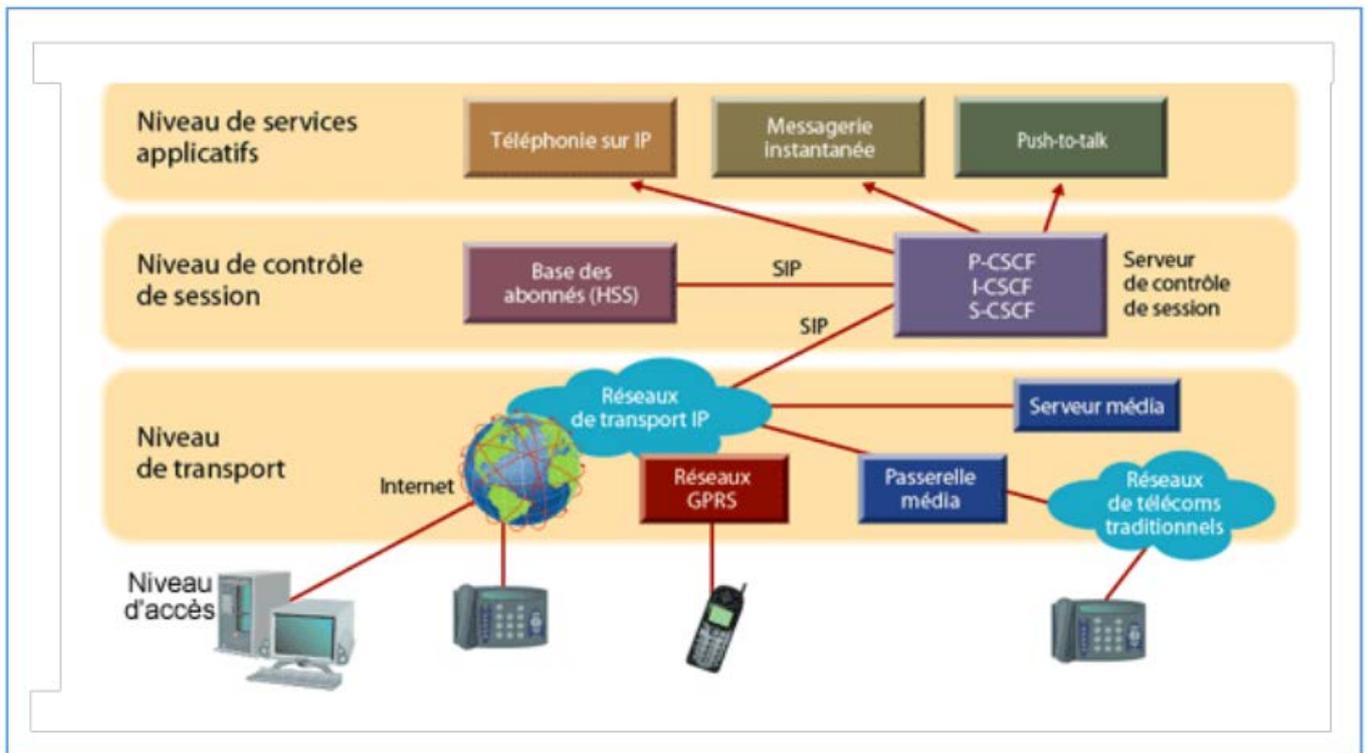


FIGURE I.2.2 – Schéma global de l'architecture en couches de l'IMS [5]

Chaque couche est indépendante. De ce fait, il est possible, par exemple, d'ajouter librement de nouveaux services dans la couche applicative, sans tenir compte du réseau d'accès que les utilisateurs ont employé, ni du terminal qu'ils ont utilisé.

### 2.3.2.1 La couche physique

La couche physique est une couche d'accès et l'architecture IMS permet un large choix de terminaux aux utilisateurs. Tous les systèmes comme des ordinateurs, des téléphones mobiles, des PDAs, des téléphones fixes numériques, sont capables de se connecter à l'architecture IMS via le réseau. Même les téléphones traditionnels analogiques sont capables de se connecter à l'architecture IMS par le biais de passerelles, pour ainsi obtenir une adresse IP, condition sine qua non à une connexion au réseau cœur IMS. Ce procédé intervient dans les couches supérieures.

### 2.3.2.2 La couche transport

C'est le réseau IP qui est utilisé dans cette couche. Elle est formée d'un maillage de commutateurs et de routeurs qui assurent, dans le réseau IP, le routage des données multimédias, à l'exclusion des informations de signalisation, qui sont à la charge de la couche de contrôle, décrite plus loin.

C'est ici que les terminaux se connectent au réseau IP, indépendamment du média de transmission. Sont alors traitées les technologies d'accès Wi-fi, DSL, câble, GPRS, UMTS, etc. Un des principaux enjeux au niveau de cette couche est de réaliser la convergence entre réseau à routage par paquet et réseaux à routage par circuit, ou autrement dit entre réseau téléphonique commuté et réseau utilisant TCP/IP.

Elle dispose en outre de deux sous-systèmes particuliers :

- **NASS** (Network Attachment SubSystem) pour la configuration du réseau. Il permet de disposer dans le réseau de l'équivalent d'un serveur DHCP, afin d'attribuer une adresse IP à l'utilisateur, et d'un client, pour les authentications de l'utilisateur, réalisées sur son profil.
- **RACS** (Ressource Admission Control SubSystem) pour le contrôle d'admission. Il permet d'allouer les ressources sollicitées par les applications en effectuant, à la demande, des réservations.

### 2.3.2.3 La couche contrôle

La couche contrôle comprend les éléments (entités) clés correspondant au cœur du réseau IMS. Toutes ces entités constituent le socle de l'architecture IMS et, à ce titre, sont indispensables pour le fonctionnement d'un réseau IMS. Elles sont des entités logiques, ce qui signifie que, malgré leur distinction fonctionnelle, rien n'empêche de les implémenter (toutes ou certaines) au sein d'un même équipement.

#### Le Home Subscriber Server (HSS)

Le HSS est la base de données des abonnés de l'IMS et des services associés (à l'instar du HLR pour les réseaux mobiles). Les comptes utilisateurs, leurs profils et droits d'accès, le nom du S-CSCF associé sont stockés dans cette base. Il communique avec le CSCF pour fournir, temporairement, une copie du profil utilisateur.

#### Le Call Session Control Function

Il existe trois types de CSCF : Proxy CSCF, Interrogating CSCF et Serving CSCF.

#### Le Proxy CSCF (P-CSCF)

Un Proxy-CSCF (P-CSCF) est un proxy SIP qui est le point de contact entre un terminal et le réseau IMS. Ses missions consistent notamment à contrôler l'accès et à établir une connexion sécurisée avec le terminal. Son adresse est découverte par l'utilisateur lors d'une phase de « CSCF discovery ». Il agit comme intermédiaire entre l'abonné et le I-CSCF.

Certains réseaux peuvent utiliser un Session Border Controller (SBC) pour cette fonction. Le P-CSCF possède à sa base un SBC spécialisé pour l'interface utilisateur-réseau qui non seulement protège le réseau, mais aussi le terminal IMS. L'utilisation de SBC supplémentaires entre le terminal IMS et le P-CSCF est donc inutile et n'est pas réalisable du fait que la signalisation est cryptée à partir de cette liaison. Le terminal peut découvrir son P-CSCF à l'aide d'un serveur DHCP.

#### L'Interrogating CSCF (I-CSCF)

L'interrogating CSCF a comme principales fonctions de déterminer le S-CSCF auquel l'abonné peut se connecter et transmettre les messages entre le P-CSCF et le S-CSCF, un peu comme une passerelle.

Les fonctions réalisées par cette entité sont :

- la localisation du S-CSCF concerné par la session par consultation de la base HSS (load balancing de S-CSCF),
- la garantie de sécurité entre le Visited network et le Home network.

### **Le Serving CSCF (S-CSCF)**

Le Serving CSCF est l'équipement qui a pour rôle de finaliser l'authentification de l'utilisateur et lui fournir les services opérationnels. Il fournit des informations de routage, de facturation, maintient l'état de la session en contrôlant un timer, interroge le HSS pour vérifier les droits utilisateurs vis-à-vis d'un service, etc.

### **Policy Decision Function (PDF)**

Dans l'objectif d'assurer la mise en œuvre de politiques de provisionning, de routage ou de QoS, la release 5 des spécifications 3GPP a prévu l'utilisation d'une plate forme de distribution de politiques conforme COPS. Dans ce contexte, la fonction PDF est une entité fonctionnelle dont le rôle est d'assurer la distribution des politiques de services locale (*Service Based Local Policy* : SBLP). A cet effet, la release 5 a intégré cette fonction dans le P-CSCF, tandis que pour la release 6 l'entité PDF est prise en charge par un bloc fonctionnel indépendant du P-CSCF.

### **Passerelles et Contrôle de passerelles**

Les Gateways (passerelles) ont un rôle essentiel : elles assurent non seulement l'acheminement du trafic, mais aussi l'inter fonctionnement avec les réseaux externes et avec les divers réseaux d'accès.

### **La Média Gateway (MG)**

La Media Gateway est située au niveau du transport des flux média entre le réseau RTC et les réseaux en mode paquet, ou entre le cœur de réseau NGN et les réseaux d'accès. Elle a pour rôle le codage et la mise en paquets du flux média reçu du RTC et vice versa (conversion du trafic TDM IP) et aussi la transmission, suivant les instructions du Media Gateway Controller, des flux média reçus de part et d'autre.

### **Signalling Gateway (SG)**

La fonction Signalling Gateway est de convertir la signalisation échangée entre le réseau NGN et le réseau externe interconnecté selon un format compréhensible par les équipements chargés de la traiter, mais sans l'interpréter (ce rôle étant dévolu au Media Gateway Controller). Notamment, elle assure l'adaptation de la signalisation par rapport au protocole de transport utilisé. Cette fonction est souvent implémentée physiquement dans le même équipement que la Media Gateway, d'où le fait que ce dernier terme est parfois employé abusivement pour recouvrir les deux fonctions MG + SG.x

### **Breakout Gateway Control Function (BGCF)**

La fonction de contrôle de passerelles de dérivation (BGCF) détermine le réseau de destination pour lequel le réseau PSTN doit se connecter ; elle choisit détermine le MGCF local ou la BGCF homologue et fournit la sécurité par l'autorisation des réseaux homologues.

### **Media Gateway Control Function (MGCF)**

Le contrôle de ces passerelles est assuré par la MGCF qui assure les fonctions suivantes : Conversion du protocole ISUP provenant du RTCP en protocole SIP, Contrôle les parties de l'appel qui maintient le contrôle de connexion pour les canaux media, Communique et dialogue avec la CSCF, Sélectionne le CSCF en fonction du routing number pour les appels entrants.

## La couche application

Tout en haut de l'architecture IMS se trouve la couche applicative ou la couche de service. Les trois couches précédemment vues permettent de définir une plate-forme unifiée et standardisée, qui va ainsi permettre aux fournisseurs de service d'offrir un large panel de services multimédia. Tout service est exécuté par un serveur applicatif, en liaison avec les équipements de la couche de session par l'intermédiaire des protocoles SIP et Diameter, assurant ainsi la sécurité des utilisations.

Il y a trois plates-formes de services standardisées : (1) *SIP application server* (SIP-AS), (2) *Open Service Access* (OSA), *Service Capability Server* (SCS), (3) et **IP Multimedia Service Switching Function** (IM-SSF).

Les services offerts par ces plateformes sont des services à valeurs ajoutées (**value-added services** VAS) ou des services spécifiques à l'opérateur. L'OSA SCS et L'IM-SSF ne sont pas des serveurs d'applications proprement dit. Ce sont plus des passerelles d'accès aux autres environnements de services. L'OSA SCS et l'IM-SSF interfacent respectivement avec l'OSA application server et le **CAMEL Service Environment** (CSE). Du point de vue du S-CSCF, cependant, ils présentent tous le même comportement de l'interface ISC.

## Protocoles utilisés dans l'IMS

### SIP

SIP est le protocole fédérateur de l'architecture IMS. Il est en quelque sorte la glue qui permet aux différents composants de communiquer entre eux de manière homogène.

Son choix, par rapport à tout autre protocole de signalisation, n'est pas anodin puisqu'il pérennise SIP au détriment de H.323, jugé trop lourd et trop coûteux. Il marque ainsi la confiance des industriels et du monde de la recherche dans le protocole SIP. Bien sûr, son utilisation implique celle des protocoles associés, en particulier SDP, pour la description des sessions, et RTP/RTCP, pour le transport en temps réel des flux de données multimédias.

### Diameter

Diameter, décrit dans la RFC 3588 de l'IETF, est un protocole de type AAA (Authentication, Authorization, Accounting). Issu du protocole RADIUS (Remote Authentication Dial-In User Service), Diameter constitue une évolution de la technologie AAA. Il est employé pour la sécurisation des communications, notamment lors de l'enregistrement des utilisateurs et lorsque les profils utilisateur sont transférés d'une base de données vers les serveurs de traitement. Il fonctionne en mode client/serveur, donc sous la forme de requêtes et de réponses.

### COPS

COPS (Common Open Policy Service) est un protocole flexible permettant la mise en place de politiques par une entité centrale appelée PDP (Policy Decision Point), qui sont appliquées sous formes de règles dans des entités appelées PEP (Policy Enforcement Point). COPS sert notamment à permettre aux opérateurs de garantir une qualité de service dans une architecture IMS.

## 2.4 Fonctionnement détaillé de l'IMS : Interopérabilité IMS-SIP

### 2.4.1 Gestion des profils/identités utilisateurs

#### 2.4.1.1 Les types d'identités dans l'IMS

Comme les réseaux télécoms de base, le réseau IMS utilise aussi la notion d'identités publique et privée [4].

L'identité publique d'un utilisateur désignée ici sous le vocable IMPU (Public User Identity) désigne un URI (SIP ou de type téléphonique) affecté à l'utilisateur et utilisé par les correspondants de ce dernier pour le désigner, et pour le joindre.

Selon le type d'URI, le format de l'IMPU peut être :

- **URI SIP**

[sip :\[utilisateur\]@\[domaine\].\[com\]](#)

Voici un exemple : [sip :ndiaye@esp.sn](#)

- **TEL URI**

Il s'agit d'un numéro de téléphone conventionnel.

Voici un exemple : **+221338692400**

Dans la terminologie télécoms, ce concept est l'équivalent du MSISDN (Mobile Subscriber ISDN Number).

L'IMPI (Private User Identity), l'identité privée de l'utilisateur, est utilisée pour identifier et authentifier un abonné et ne joue aucun rôle dans le routage des messages SIP.

Cette identité privée est fournie par l'opérateur du réseau IMS et n'est en générale pas connue de l'utilisateur. L'IMPI est généralement stockée dans une ISIM (IMS Subscriber Identity Module). Un équivalent dans les réseaux mobiles est la carte SIM.

L'adresse d'une IMPI doit respecter le format suivant : [utilisateur]@[domaine]

L'IMS utilise aussi une autre identité : l'IMSU (IMS Subscription). Il s'agit d'un identifiant utilisé pour la facturation. C'est, en général, le nom complet de l'utilisateur. Par exemple, on peut attribuer à IMSU la valeur Claude Lishou ou un numéro par exemple 790010011.

#### 2.4.1.2 Relations entre les différents types d'entités

Dans le terminal de l'utilisateur est stocké une Public et une Private User Identity. Le HSS, qui stocke l'ensemble des informations sur l'utilisateur, dispose d'une table contenant les Public et Private User Identities de l'utilisateur. Le HSS et le S-CSCF lient aussi adresse publique et privée.

Le lien entre adresses privées et publiques diffèrent entre les releases 5 et 6 du standard IMS du 3GPP. En effet dans la release 5, un utilisateur possède une unique adresse privée qui peut être liée à plusieurs adresses publiques comme la montre la figure I.2.3 alors que dans la release 6, un utilisateur peut avoir plusieurs adresses privées et deux adresses privées peuvent partager une même adresse publique comme le montre la figure I.2.4. Le fait d'avoir plusieurs adresses privées permet par exemple d'avoir des cartes SIMs différentes pour différents terminaux.

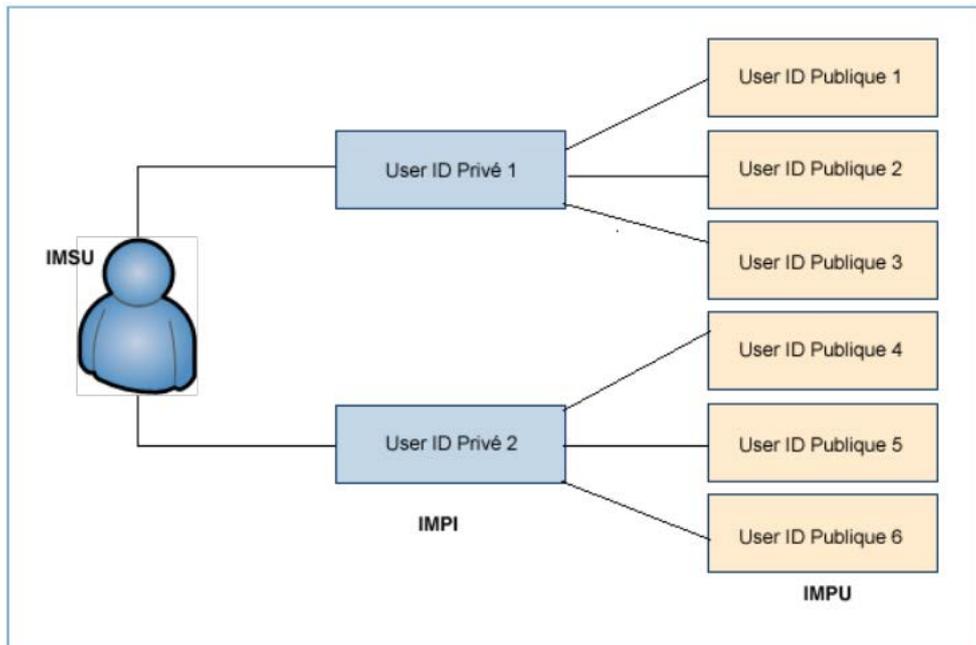


FIGURE I.2.3 – Relations entre IMSU, IMPI et IMPU selon le 3GPP R5 [6]

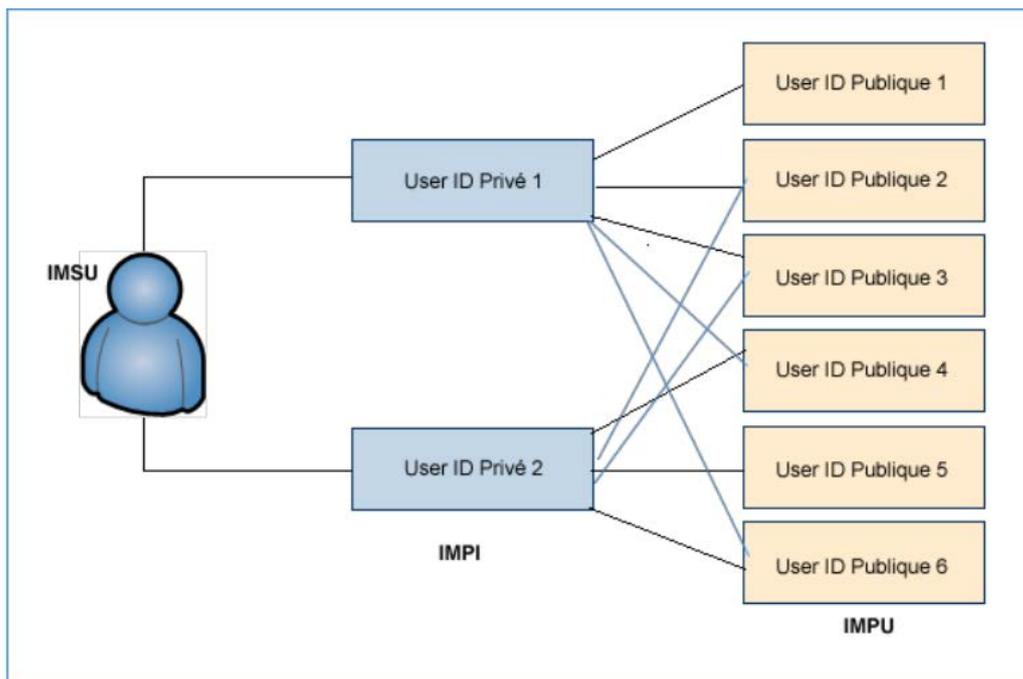


FIGURE I.2.4 – Relations entre IMSU, IMPI et IMPU selon le 3GPP R6 [6]

### 2.4.1.3 Gestion des identités

Les terminaux IMS sont identifiés par la présence d'une carte à puce appelée UICC (Universal Integrated Circuit Card), qui comporte un ensemble d'informations d'identification et de personnalisation. L'interface de communication entre la carte UICC et le terminal est standardisée, mais la carte UICC est générique et peut comporter toutes sortes d'applications, notamment les applications classiques suivantes : SIM (Subscriber Identity Module), USIM (Universal Subscriber Identity Module), ISIM (IP multimedia Services Identity Module).

Ces trois applications ne sont évidemment pas les seules. Elles peuvent être utilisées en même temps par le terminal pour fournir plusieurs de ces données enregistrées, selon les connexions (SIM pour les réseaux GSM, USIM pour les réseaux UMTS, etc.).

Par contre, il est indispensable de disposer au moins de l'application ISIM ou USIM pour se connecter dans un réseau du 3GPP. En ce qui concerne la version IMS du 3GPP2, la notion d'UICC est absente, mais ces mêmes applications USIM et ISIM peuvent être fournies comme une configuration du terminal de l'utilisateur.

#### 2.4.1.4 Notions de réseau visité et de roaming

Avec l'architecture IMS, il est possible d'utiliser de multiples technologies d'accès. Par ailleurs, les zones d'accès géographiques peuvent être aussi larges que possible. Autrement dit, le terminal IMS de l'utilisateur doit pouvoir se connecter de partout.

Puisqu'un opérateur possède une infrastructure physique forcément limitée, il convient de prendre en compte la possibilité d'exploiter des réseaux tiers, c'est-à-dire de permettre à un abonné d'un opérateur d'utiliser des réseaux d'accès physiques qui appartiennent à d'autres opérateurs que le sien. Pour cela, on introduit le concept de réseau visité : lorsqu'un utilisateur utilise un réseau qui n'est pas celui de l'opérateur auquel il a souscrit un contrat de services. La connexion de l'utilisateur à un réseau visité est soumise aux accords qui lient son opérateur à l'opérateur qui exploite le réseau visité.

On parle de **roaming** pour désigner la **possibilité** qu'a un utilisateur de se connecter à un réseau qui n'est pas celui de son opérateur. De même, les accords de roaming décrivent les conditions contractuelles qui lient un opérateur, exploitant une base de données de clients, à un opérateur, exploitant un réseau d'accès avec une couverture géographique.

### 2.4.2 Authentification dans l'IMS

Le mécanisme utilisé pour l'authentification dans l'IMS s'appelle IMS-AKA (Authentication and Key Agreement). C'est un mécanisme d'authentification qui repose sur l'utilisation d'un secret partagé entre l'utilisateur (sur l'ISIM ou à défaut l'USIM) et le réseau IMS (au niveau du HSS). Le HSS, toujours localisé dans le réseau « home » de l'utilisateur, contient un quintuple d'authentification : un challenge aléatoire, RAND, un jeton d'authentification, AUTN (contenant un numéro de séquence et un identifiant MAC), une réponse attendue XRES, une clé de cryptage, CK, et une clé d'intégrité, IK, qui assurent l'intégrité et la confidentialité des messages SIP échangés entre le terminal et le P-CSCF.

Au cours de l'enregistrement, le S-CSCF transmet les paramètres RAND et AUTN au terminal qui s'en sert pour calculer une réponse attendue XMAC. L'authentification est mutuelle entre l'utilisateur et le réseau (S-CSCF) :

- a) le terminal compare son XMAC avec le MAC reçu du réseau : le réseau est authentifié s'ils sont identiques ;
- b) le réseau compare son XRES avec le RES calculé par le terminal : l'utilisateur est authentifié s'ils sont identiques.

Lorsque seule l'USIM est utilisée, il faut dériver les paramètres utiles à l'IMS (qui seraient normalement sur l'ISIM) depuis l'IMSI (identifiant de l'utilisateur pour l'UMTS) stocké sur l'USIM :

- a) l'identité privée de l'utilisateur qui est une identité publique temporaire qui ne sert que lors de l'enregistrement, les identités publiques associées peuvent en revanche être utilisées ;

- b) une identité publique de l'utilisateur ;
- c) le nom de domaine.

Par ailleurs, un mécanisme a été introduit pour assurer l'authentification des terminaux qui accèdent à l'IMS sans USIM ni ISIM, en l'occurrence, les terminaux 2G qui n'implémentent pas de fonction IPSec. Ce mécanisme associe l'adresse IP du terminal et les identités de l'utilisateur (identité privée, identités publiques) au niveau du S-CSCF, ce qui permet de vérifier que les messages SIP reçus au S-CSCF ne proviennent pas d'un autre utilisateur.

### 2.4.3 Enregistrement d'un terminal dans le réseau IMS

L'enregistrement d'un utilisateur dans le réseau est la première action réalisée par un terminal, dès sa mise en route. Elle est indispensable puisqu'elle permet à la fois d'appeler et d'être joignable par ses correspondants. La méthode associée à cette fonctionnalité est REGISTRER, du protocole de signalisation SIP. La figure I.2.5 illustre les étapes temporelles liées au scénario d'enregistrement.

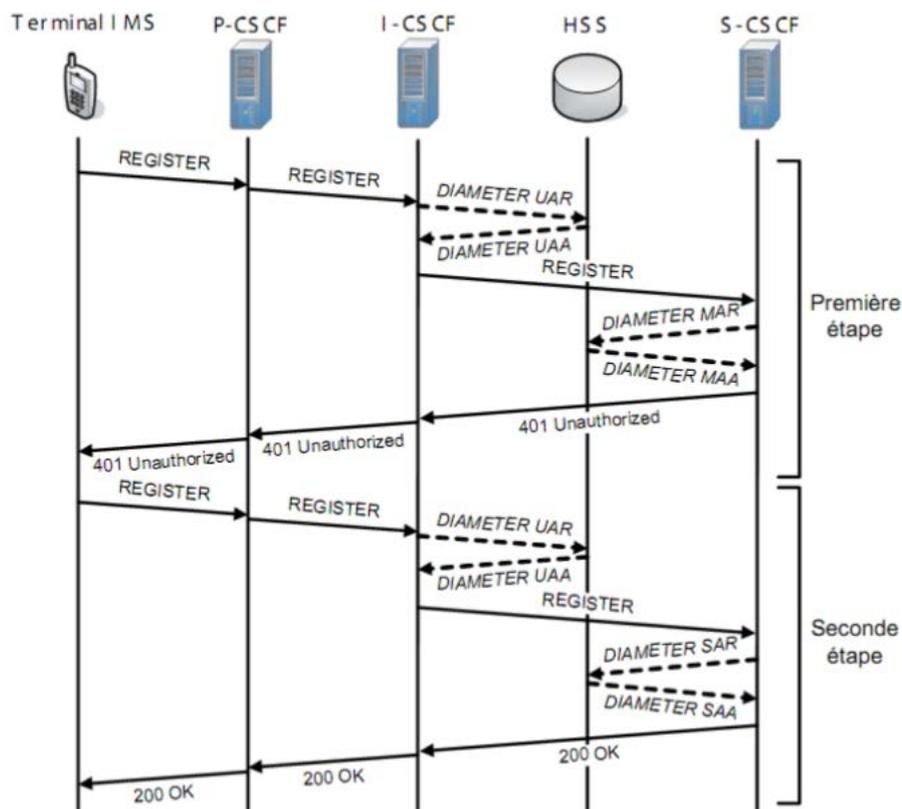


FIGURE I.2.5 – Procédure d'enregistrement dans l'IMS [4]

Dans un premier temps, le terminal IMS envoie sa requête d'enregistrement au serveur P-CSCF. Celui-ci ne connaît pas nécessairement le serveur I-CSCF. Pour localiser ce dernier, le serveur P-CSCF procède à une requête DNS à partir du nom de domaine fourni par l'utilisateur.

Une fois localisé, le P-CSCF joint l'I-CSCF en lui fournissant la requête de l'utilisateur, dans laquelle il a ajouté un champ d'entête P - VISITED-NETWORK-ID. Ce champ contient un identifiant du réseau dans lequel le P-CSCF se trouve. Il permettra au S-CSCF de vérifier que le réseau visité bénéficie d'un accord de roaming avec l'opérateur de l'utilisateur.

Un autre champ ajouté par le P-CSCF est le champ d'en-tête PATH qui spécifie l'adresse SIP du P-CSCF. Cette information permettra de retourner la réponse à cette requête via ce même serveur P-CSCF.

Lorsque le I-CSCF reçoit la requête, il ignore si elle concerne un utilisateur qui est déjà enregistré ou s'il s'agit d'un nouvel enregistrement. Le I-CSCF utilise le protocole Diameter pour contacter la base de données HSS et lui demander, à partir des identités publiques et privées contenues dans le message de requête REGISTER, d'authentifier l'utilisateur (requête UAR).

Ensuite, lorsque le client reçoit le message de réponse 401, il prépare automatiquement une réponse adaptée. Cette réponse est générée dans une nouvelle requête d'enregistrement REGISTER. Elle est envoyée en suivant le même processus d'acheminement que le premier message de requête : le terminal client ne connaît que le serveur P-CSCF, qui, lui-même, ne peut s'adresser qu'au I-CSCF, qui, à son tour, sous-traite la demande auprès du serveur S-CSCF.

Si les paramètres d'authentification sont valides, le serveur S-CSCF en informe le HSS (par un message SAR).

Le HSS répond ensuite au S-CSCF en lui envoyant le profil complet de l'utilisateur, qui est stocké temporairement et servira à paramétrer et personnaliser les services de ce dernier.

Pour terminer, le serveur S-CSCF envoie un message de réponse 200 OK au terminal pour lui notifier le succès de l'opération d'enregistrement.

#### **2.4.4 Mise en communication de deux utilisateurs**

Une communication implique une première étape de recherche des abonnés, de vérification des autorisations d'accès, puis de mise en relation entre les correspondants. La méthode associée à cette dernière fonctionnalité est INVITE avec le protocole de signalisation SIP.

La figure I.2.6 ci-dessous illustre la mise en communication de deux utilisateurs.

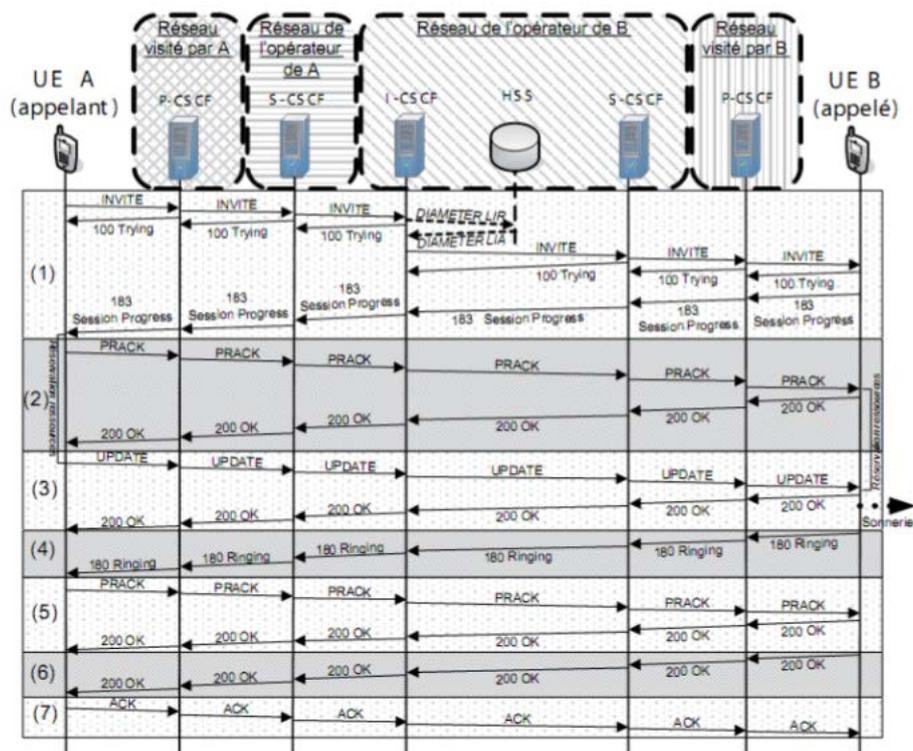


FIGURE I.2.6 – Mise en communication de deux utilisateurs [6]

On considère que l'appelant et l'appelé ont des opérateurs différents et se trouvent dans des réseaux visités.

Ce scénario de mise en relation de deux terminaux peut être découpé en sept grandes étapes :

1. L'appelant (A) envoie un message d'invitation à l'appelé (B), pour proposer de négocier les paramètres de la communication. A doit solliciter le terminal de B et, pour cela, il s'adresse au serveur I-CSCF de B, qui le localise après une requête Diameter. Le terminal de B lui envoie deux réponses temporaires : une réponse 100 pour indiquer la tentative et une réponse 183.
2. Pour rassurer son correspondant qu'il a bien reçu la réponse 183, A doit impérativement envoyer à B un acquittement temporaire.
3. Le terminal A négocie ensuite les paramètres de qualité de service pour garantir sa communication dans le réseau. Cette étape permet aux utilisateurs d'établir une communication avec une bande passante garantie, et donc un service de qualité. Le terminal de B vérifie ensuite que lui aussi a réservé les ressources nécessaires à la communication dans le réseau et valide la requête par sa réponse.
4. Dès cet instant, le terminal de B commence à sonner. Cette étape complète les réponses temporaires à la requête d'invitation par une réponse 180, elle aussi temporaire.
5. Pour s'assurer que cette réponse est bien reçue du terminal A, ce dernier doit confirmer la réception par une requête d'acquittement, qui attend elle-même une réponse.
6. Dès que l'utilisateur du terminal B a répondu, la réponse définitive 200 est envoyée à la requête initiale d'invitation.
7. La requête d'acquittement finale valide l'initialisation de la communication, qui peut dès lors débiter pour permettre aux terminaux de s'échanger des flux de données multimédias.

### 2.4.5 Modification des sessions

Parmi les services offerts par l'IMS, la session multimédia apparaît comme le service de base qui permet de mettre en communication deux terminaux pour échanger un ou plusieurs flux de média. Des serveurs d'application peuvent être contactés lors de l'établissement de la session. Il est également possible d'établir une session multimédia entre un terminal et un serveur d'application via le réseau IMS.

L'appelant ou l'appelé peut demander une modification de la session en cours à tout moment. La demande de modification de session est exprimée par le terminal via le message « SIP UPDATE » ou un message de type « re-INVITE » qui contient une nouvelle description pour la session en cours.

Une modification de session peut consister à ajouter ou supprimer un ou plusieurs médias, à changer de codec sur un ou plusieurs médias, à changer de numéro de port pour un ou plusieurs flux de médias, à changer la bande passante pour un ou plusieurs médias.

### 2.4.6 Fin d'enregistrement

Le terminal demande notamment la fin d'enregistrement quand l'utilisateur ne souhaite plus accéder à ses services. Cette procédure démarre avec l'envoi d'un message SIP REGISTER depuis le terminal. La fin d'enregistrement peut également survenir à l'initiative du réseau. Dans ce cas, on utilise le message « SIP NOTIFY » en lieu et place de « SIP REGISTER ».

Lors d'une fin d'enregistrement, le terminal doit mettre fin à ses sessions en cours, le HSS supprime l'adresse du S-CSCF stockée pour l'utilisateur, les CSCF et le terminal suppriment les données d'enregistrement correspondant à l'adresse publique et l'adresse IP pour lesquelles la fin d'enregistrement est demandée. Enfin, le terminal et le P-CSCF suppriment l'association de sécurité établie entre eux pendant l'enregistrement, s'il ne reste plus d'enregistrement actif pour l'utilisateur.

## 2.5 L'architecture de service IMS

Dans cette section, nous examinerons en détails la couche service de IMS. Le plan de service de l'IMS est conçu pour fournir la prochaine génération d'applications avec SIP et fonctionner avec les plates-formes de services existantes. Les éléments du plan de service appelés serveurs d'applications (SA) ont la capacité de prendre en charge la logique applicative d'un service. Ils peuvent fournir les services multimédia à valeur ajoutée tels que la présence, le service push-to-talk, etc. Les fonctions principales des SA sont :

- la possibilité de traiter et modifier une session SIP entrante
- la capacité d'initier des requêtes SIP
- la capacité d'envoyer des informations de comptabilisation aux organes de taxation.

Ils peuvent en outre fonctionner comme une passerelle ou fournir une fonction d'interopérabilité avec un serveur natif -du réseau intelligent- ou un serveur non-SIP. Ils peuvent aussi jouer un rôle de coordination de la logique de service entre plusieurs serveurs. Quel que soit leur rôle, les SA communiquent avec le S-CSCF par le protocole SIP au travers de l'interface IMS Service Control (ISC). Ils ont également accès aux informations d'abonné stockées dans le serveur HSS. Les SA se retrouvent aussi bien dans le réseau nominal de l'abonné que dans celui d'un fournisseur tiers de services.

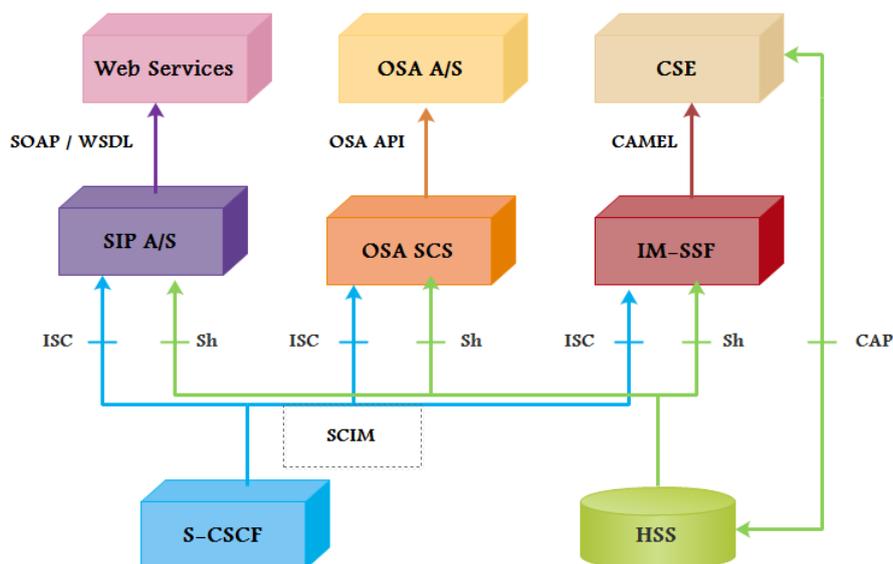


FIGURE I.2.7 – Architecture du plan de service IMS [26]

### 2.5.1 Serveurs d'applications SIP

Les applications IMS sont généralement mis en œuvre sur des serveurs d'application SIP. Cependant certains vendeurs de serveurs d'applications peuvent ajouter une pile SIP aux plateformes existantes afin de minimiser les coûts de reconstruction. Par ailleurs certains peuvent choisir de déployer leurs applications sur une plateforme convergente agissant comme un serveur d'application qui supporte plusieurs technologies (SIP, Web Services).

Un serveur d'application peut utiliser quatre modes de fonctionnement pour le traitement des requêtes SIP. Des services peuvent être conçus en utilisant une combinaison de ces modes entre le serveur d'application et l'entité S-CSCF. Dans le mode de fonctionnement "Terminating UA ou Redirect Server", la requête SIP reçue par le S-CSCF est routée au serveur d'application qui joue le rôle de "Terminating UA" ou "Redirect Server". Si le mode est "Terminating UA", le serveur d'application termine la requête SIP et retourne une réponse à l'émetteur de la requête SIP. Si le mode est "Redirect Server", alors le serveur d'application retourne une réponse de redirection à l'émetteur. Cette réponse contient l'adresse où la destination peut être contactée. Dans le mode de fonctionnement "Originating UA", le serveur d'application joue le rôle de "Originating UA" et génère une requête SIP qu'il émet au S-CSCF ; ce dernier l'envoie vers la destination.

Ce mode de fonctionnement correspond à l'initiation de la session par le serveur d'application (click-to-dial). Dans le mode de fonctionnement "Proxy", la requête SIP reçue par le S-CSCF est relayée au serveur d'application jouant le rôle de "Proxy". Le serveur d'application peut alors modifier le contenu de la méthode SIP reçue et la retourner au S-CSCF qui la relaye à la destination. Ce mode s'applique par exemple aux services de traduction de numéro ou de renvoi d'appel sophistiqué en fonction de l'heure du jour, du jour de la semaine, du numéro de l'appelant, etc. Dans le mode de fonctionnement "B2BUA", la requête SIP reçue par le S-CSCF est relayée au serveur d'application qui génère une nouvelle session SIP qu'il envoie au S-CSCF qui la relaye à la destination. Le serveur d'application joue le rôle de "Back-To-Back User Agent" (B2BUA) et gère ainsi plusieurs legs dans la

session indépendamment les uns des autres. Ce mode est adapté pour un service comme le Customized Ring Back Tone (CRBT). Le serveur d'application correspondant gère trois sessions (trois legs) : la première avec l'appelant, une seconde avec l'appelé et une troisième avec le MRF. Lorsque l'appelé est alerté, il joint les legs relatifs à l'appelant et au MRF ; ce dernier joue un message personnalisé à l'appelant. Dès que l'appelé décroche, le serveur d'application libère le leg relatif au MRF et joint les legs appelant et appelé.

### 2.5.2 IM-SSF : IM-Service Switching Function

La fonction de commutation de service IP Multimedia (IM-SSF) permet aux opérateurs mobiles d'utiliser leur infrastructure de services existante – celle des réseaux 2G par exemple – avec leur réseau IMS afin d'éviter un sur-coût de redéveloppement et de redéploiement des applications et services. L'IM-SSF est spécialement conçu pour permettre aux réseaux IMS de réutiliser la suite d'applications CAMEL2 déjà déployées. Au nombre de ces applications, figurent les plateformes de facturation des abonnés prépayés. Ces implémentations très coûteuses, supportent une part importante des opérations génératrices de revenus pour les opérateurs. Il serait prohibitif de devoir reproduire et de maintenir un nouveau réseau prépayé.

Techniquement, l'IM-SSF traduit les messages SIP provenant du S-CSCF en leurs équivalents CAMEL. Ainsi, l'IM-SSF apparaît comme un SA SIP au S-CSCF et un gsmSSF (GSM Service Switching Function) au gsmSCF (GSM Service Control Service), qui est un SA de type CAMEL. En plus de la traduction des messages entre l'IMS et un environnement CAMEL, l'IM-SSF possède également des fonctionnalités d'un réseau CAMEL pour faciliter l'interopérabilité. Ces caractéristiques incluent la gestion des appels sur le modèle du GSM et le maintien des points de détection de déclenchement, des clés de service, et l'adressage des SA (gsmSCF). Tout comme les SA SIP, l'IM-SSF peut interagir avec le serveur HSS par le biais de l'interface Si pour extraire des données de profil utilisateur et de service.

Il en résulte que l'IM-SSF assure une interopérabilité entre les réseaux intelligents existants et IMS pour la gestion des services évitant ainsi aux opérateurs des coûts supplémentaires liés au redéploiement des services.

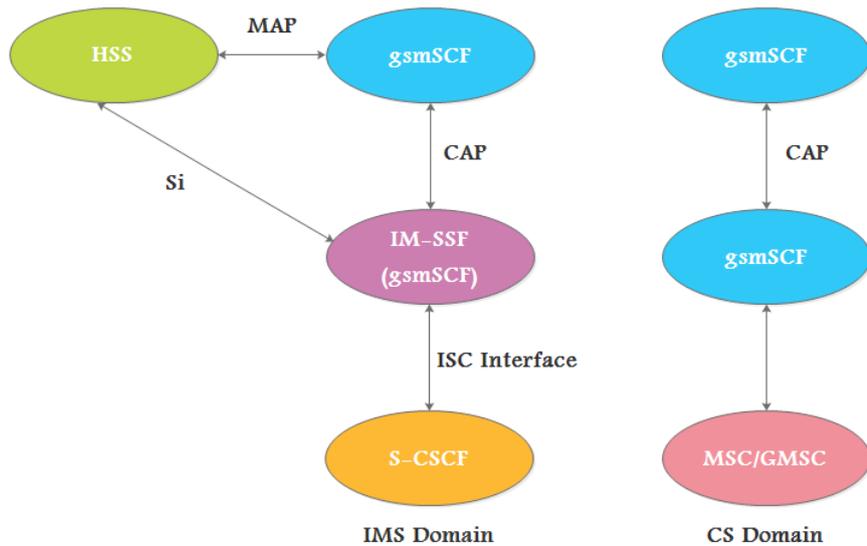


FIGURE I.2.8 – Architecture de service CAMEL et interaction avec IMS

### 2.5.3 OSA SCS : Open Service Access Service Capability Server

Les groupes de travail Open Service Access (OSA) et Parlay ont contribué aux spécifications du 3GPP portant sur l'exposition des services IMS. Au nombre des contributions de ces groupes figurent les spécifications sur les relations entre opérateurs et tiers fournisseurs de services ainsi que les développeurs. Les termes OSA et Parlay sont utilisés (dans les cercles de télécommunications) pour désigner une architecture qui permet aux développeurs d'accéder et de faire usage des fonctionnalités d'un réseau d'opérateur de façon abstraite par le biais des API ouvertes et standardisées. OSA définit un ensemble de serveurs de capacité de service (SCS). Chacun d'eux est une fonction qui donne une vue abstraite d'un élément de réseau et fournit une interface OSA (Parlay) pour les applications.

Ces SCS sont généralement déployés au sein d'une passerelle qui fournit des fonctionnalités supplémentaires telles que la sécurité, les accords de niveau de service (SLA), la taxation et la gestion des comptes. Ces fonctionnalités additionnelles constituent dans la terminologie OSA le framework ; il tourne sur un SCS dédié. Ce framework spécifie aussi les SCSs disponibles et assure l'authentification entre une application OSA et les SCSs. Au sein des SCSs figurent des Services Capability Function (SCF). Une SCF constitue l'abstraction d'un service particulier offert par le réseau et accessible au travers des APIs Parlay. Les SCFs sont spécifiées via des interfaces avec leurs méthodes propres. Un seul et même SCS peut être vu par les applications comme une ou plusieurs SCFs. Les SCSs implémentent les spécifications Parlay sur l'interface OSA de la passerelle, et se chargent d'effectuer la traduction de ces APIs vers les protocoles de télécommunications spécifiques présentes au plus bas niveau, cachant de cette manière la complexité du réseau aux applications. La figure I.2.9 résume l'architecture d'une passerelle OSA.

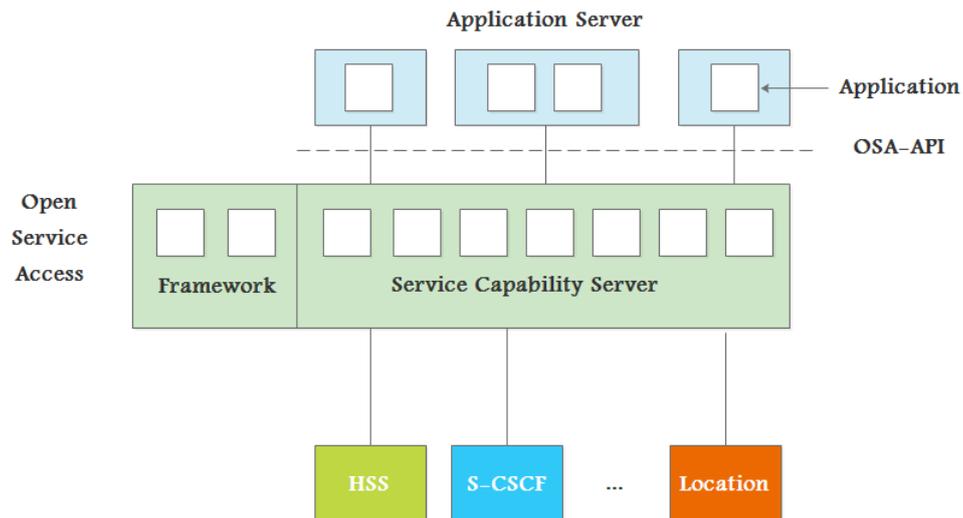


FIGURE I.2.9 – Architecture OSA [7]

Il y a un dernier point à ajouter au sujet des passerelles OSA. Dans l'écosystème Internet, les développeurs utilisent beaucoup la norme Web services Markup Language (WSML) désigné plus simplement sous le vocable Services Web pour le développement d'applications. Parlay est conçu pour des applications temps réel comme la téléphonie alors WSML est utilisé pour les services de type données. Bien que l'API OSA facilite le développement d'application télécoms, il n'en demeure pas moins que le développeur doit posséder une certaine connaissance du monde des télécommunications. Pour allier ces deux mondes, une version Services Web de Parlay, appelée Parlay X, a été défini pour faciliter l'accès aux ressources des opérateurs aux développeurs. Cela a conduit au développement des passerelles Parlay X. Il existe deux façons de déployer ce type de passerelle. L'une des méthodes consiste à disposer d'une passerelle Parlay X autonome embarquant un framework. Cette configuration est appropriée aux opérateurs voulant utiliser uniquement des Services Web.

L'autre approche consiste à mettre la passerelle Parlay X au dessus d'une passerelle OSA comme le montre la figure I.2.10. Dans ce cas le framework n'est plus embarqué dans la passerelle Parlay X, il se retrouve plutôt dans la passerelle OSA. La passerelle Parlay X interagit avec la passerelle OSA par le biais de l'API OSA. D'une certaine façon, elle se comporte comme un serveur d'application OSA. Les applications peuvent exploiter et invoquer les Services Web Parlay X au travers de protocole d'échange de messages basés sur XML tels que SOAP (Simple Object Access Protocol).

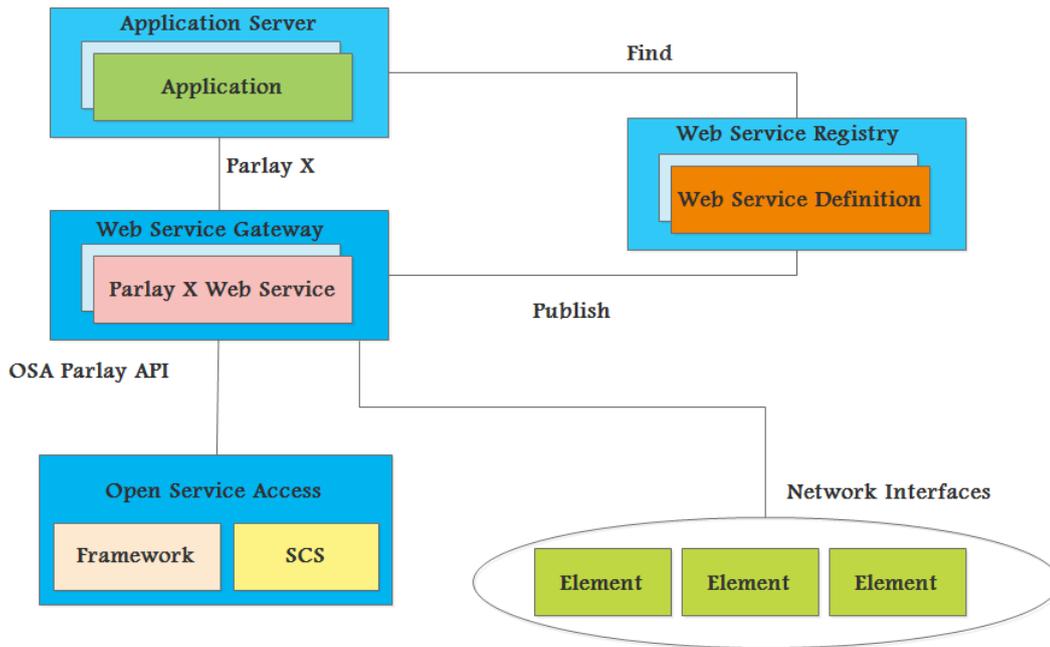


FIGURE I.2.10 – Passerelles Parlay/ParlayX [8]

L'interconnexion entre le S-CSCF et les SCSs de la passerelle OSA se fait par l'interface ISC. Ainsi, la passerelle OSA a l'apparence d'un SIP AS pour le S-CSCF (semblable au comportement de IM-SSF).

## 2.5.4 SCIM : Service Capability Interaction Manager

Le Service Capability Interaction Manager (SCIM) est une entité de la couche application introduite dans la release 5 des spécifications 3GPP sur l'IMS. Le SCIM se charge de coordonner l'exécution des services potentiellement conflictuels. Son but initial est la gestion de l'interaction entre les serveurs d'applications pour un service donné. Il agit ainsi comme un routeur d'applications et fournit une flexibilité accrue pour le chaînage de services grâce à sa capacité de composition. Les fonctionnalités du SCIM ne sont pas clairement exposées dans la spécification afférente. Il peut donc être implémenté suivant plusieurs conceptions. Le SCIM peut être déclenché plusieurs fois au cours d'une session, avant ou après un AS, dans le but d'exécuter des services supplémentaires, d'effectuer quelques manipulations, consulter des bases de données externes, ou d'effectuer toute tâche faisant partie du processus de composition. La composition constitue le mécanisme liant plusieurs AS pour fournir un service d'une valeur ajoutée plus importante.

## 2.6 Services standards : Enablers

Les services comme la téléphonie et la messagerie sont des services classiques. Ils méritent bien l'appellation de services dans le sens où ils sont visibles et directement utilisables par les abonnés contrairement aux fonctionnalités cachées telles que le routage et la transcodage. Cependant quand un service devient omniprésent, son utilité va au au-delà du rôle de service et peut faciliter le développement et le déploiement de nouveaux services plus évolués. Dans cette section il s'agit de présenter ces *facilitateurs* nommés *enablers*.

### 2.6.1 Présence

Le service de présence, d'utilité courante, entre souvent dans le cadre de la messagerie instantanée. Le terme présence fait référence à trois aspects : la présence en elle-même, la disponibilité et l'humeur. Ainsi un usager peut partager ces informations en l'occurrence son statut à ses contacts afin qu'ils communiquent avec lui à un moment approprié. Le standard SIMPLE (Sip for Instant Messaging and Presence Leverage Extensions) régit le fonctionnement de ce service. L'architecture de présence est bâtie autour de 3 fonctions principales : l'entité de présence nommé *presentity*, le serveur de présence et les observateurs (*watcher*). L'entité de présence est celui qui soumet son statut au serveur de présence. Le serveur de présence est le dépôt central qui stocke les informations de présence des abonnés et de leur contacts. Il fournit également ces informations aux entités observatrices. Le *watcher* est l'entité qui souscrit aux statuts de présence des abonnés qui l'intéresse. Il reçoit les notifications de changements d'état de A noter que ce service peut être combiné avec d'autres services tels que la voix. En effet, il ne sert qu'à diffuser les informations de présence d'un abonné.

### 2.6.2 Messagerie

La messagerie instantanée constitue la pierre angulaire du service de messagerie. Autour d'elle, se greffent la présence et un serveur XDM (XML Document Management) lui donnant ainsi une valeur ajoutée. Le serveur XDM est utilisé pour conserver les groupes de contacts et de discussion d'un abonné. La capacité de service Messaging fonctionne selon deux principaux modes :

- Pager-mode messaging : Des messages SIP contenant les données à échanger sont routés de manière asynchrone entre l'émetteur et le récepteur avec la méthode SIP MESSAGE.
- Session-mode messaging : Une session IMS est établie pour la session de donnée (de type Tchat). Le protocole MSRP (Message Session Relay Protocol) est alors utilisé pour transporter les données entre usagers et non pas le protocole SIP. Les données peuvent être du texte ou des documents multimédia.

A noter, l'existence d'un troisième mode appelé *larger mode* qui combine les avantages des deux modes précédents. Pour ce mode, il n'y a pas de session et le message est transporté par le protocole MSRP. Il est unidirectionnel et adapté pour l'envoi des données dépassant 1300 octets.

### 2.6.3 Conférence

Une conférence est une conversation entre plusieurs participants. Ce service concerne à la fois la voix, la vidéo et le texte. Le service de conférence fonctionne selon deux modes :

- Le mode Ad-hoc qui permet de créer des conférences à la demande. Il s'agit de conférences non planifiées et de courte durée.
- Le mode pre-arranged (planifié) qui permet de créer des conférences à l'avance en utilisant le protocole Conference Policy Control Protocol (CPCP). Il spécifie un schéma XML qui énumère les éléments d'information de politique de conférence permettant à l'utilisateur de définir sa politique de conférence.

### 2.6.4 Multimedia Telephony (MMTel)

Le cœur de réseau IMS ne fournit pas de base un service de téléphonie, il est essentiellement un système qui permet de retrouver et de connecter des machines (clients et serveurs SIP). MMTel est

l'équivalent du service de téléphonie comme nous le connaissons dans les réseaux GSM et RTC, par contre il est mis en œuvre dans un contexte de transport IP. MMTel est un serveur d'application SIP. La capacité de service Multimedia Telephony permet des communications conversationnelles entre deux ou plusieurs participants. Cette capacité inclut les services supplémentaires comparables à ceux fournis par le domaine circuit tels que le renvoi d'appel, le signal d'appel, la mise en garde, le rappel automatique sur occupation, les listes noire et blanche etc. La fonctionnalité MMTel permet à un utilisateur d'effectuer des appels téléphoniques multimédia en utilisant le service de téléphonie standard IMS. MMTel prend en charge la voix, la vidéo, le texte et supporte le protocole MSRP pour la messagerie instantanée et l'échange de fichiers. Les médias peuvent être ajoutées, modifiées ou supprimées pendant un appel. Il fournit également des fonctionnalités telles que la conférence téléphonique, la présentation d'identité (de l'appelant).

### 2.6.5 Rich Communication Suite (RCS)

RCS signifie Rich Communications Suite et est un standard défini par la GSM Association (GSMA) qui est un consortium mondial rassemblant de nombreux acteurs des télécommunications, allant des opérateurs téléphoniques, aux constructeurs en passant par des développeurs. Au départ, Rich Communication Suite a été imaginé à l'instar des réseaux sociaux pour exploiter la communauté que représente le carnet d'adresses du téléphone, mais cette idée a évolué pour maintenant être une norme internationale. Aujourd'hui, cette technologie est à version 5.0 et voici ce qu'elle propose :

- Un carnet d'adresses enrichi qui va permettre aux utilisateurs de partager leur présence (Disponible, Occupé), un message d'humeur (statut), une photo ou encore leur lien web favori et qui sera automatiquement synchronisé sur les téléphones des personnes avec qui il a partagé son « profil », sa fiche contact.
- Le Enriched Call qui va permettre de partager pendant l'appel, une vidéo ou une photo prise en direct ou déjà enregistrée dans le téléphone.
- Le Enhanced Messaging qui consiste à retrouver une interface évoluée dans l'affichage des messages : sous forme de chat, sans différence entre un SMS et un MMS avec des fonctionnalités d'envoi de fichier. Le Rich Messaging est basé sur l'enabler OMA pour la messagerie instantanée OMA SIMPLE IM.
- Ce service est disponible avec le même identifiant (numéro de téléphone) depuis le terminal mobile ainsi que depuis des logiciels RCS sur ordinateur. La fonctionnalité de présence est enrichie avec un accès aux plateformes de réseaux sociaux.
- Et surtout : la possibilité d'ajouter n'importe quelle autre fonctionnalité comme la géolocalisation ou encore le guidage GPS etc. . .

En résumé, l'apport global de RCS est dans sa capacité à fournir un accès à de multiples fonctionnalités depuis une seule interface telle que le carnet d'adresse. RCS n'est pas une nouvelle technologie, il fait une agrégation des enablers définis par l'OMA.

### 2.6.6 Push-To-Talk over Cellular(PoC)

La capacité de service Push-to-talk over Cellular (PoC, Push To Talk over Cellular) consiste à utiliser son téléphone comme un talkie-walkie, simplement en poussant un bouton pour dialoguer les uns avec les autres. La technologie se veut l'équivalent voix du SMS. Le service PoC permet la transmission de messages vocaux entre utilisateurs mobiles sur les réseaux de données. L'utilisateur

sélectionne un ou plusieurs correspondants dans son carnet d'adresses, puis presse un bouton sur son terminal pour enregistrer son message vocal. Le message est ensuite encodé puis transmis par paquet RTP/UDP/IP via le réseau d'accès large bande mobile. La transmission de messages par ce biais introduit un délai de latence qui n'autorise pas, en théorie, des échanges vocaux en "quasi-temps réel", mais qui, en pratique, pourrait voir le service PoC utilisé plutôt pour des services de type "messagerie instantanée vocale". Le fonctionnement de ce service requiert un ensemble d'éléments à savoir : un serveur PoC, un client PoC, un serveur de présence et un serveur XDM.

## 2.7 Taxation

La taxation est probablement la fonction la plus importante pour un fournisseur de service car il soutient toute la structure économique de ce dernier. L'IMS offre de nouveaux modèles de tarification [27] ouvrant de nouvelles perspectives de modèles d'affaire aussi bien pour les opérateurs que les développeurs. La capacité de fournir une taxation sur la base de session, ou d'événements, ou d'un service est l'un des principaux avantages de IMS. Deux modes distincts de taxation seront présentés : la taxation online et offline. Ces deux modes sont plus connus sous les appellations d'offres prepaid et postpaid. Voici les définitions des tarifications online et offline tirés de la spécification TS32.240 :

**Tarification Online** C'est un mécanisme de tarification pour lequel les informations collectées sur l'utilisation des ressources impactent en temps réel le service rendu. Par conséquent une interaction directe est requise entre le système de tarification et les couches média, session et contrôle de service. On entend par interaction directe, par exemple, que le système de tarification puisse constater un défaut de crédit dans le compte d'un utilisateur, et donc un message est généré pour notifier à l'abonné qu'il doit recharger son compte. Cela doit se faire dans un délai raisonnable au cours de l'établissement d'un appel.

**Tarification Offline** C'est un mécanisme de tarification pour lequel les informations collectées sur l'usage des ressources n'affectent pas en temps réel le service rendu. Cela signifie que les Charging Data Record (CDR) sont collectés et traités ultérieurement.

### 2.7.1 Taxation hors-ligne

Comme on peut le voir sur la figure I.2.11, bon nombre de nœuds de l'architecture IMS, autant au cœur qu'à la couche application, sont en mesure de produire des informations de tarification. Ces dernières sont délivrées par l'interface Rf utilisant le protocole Diameter, spécifié dans la RFC 3588 de l'IETF. Les entités et interfaces se présentent comme suit :

- Charging Data Function (CDF), cette fonction recueille les informations de taxation provenant de divers nœuds. L'adresse du CDF utilisé est obtenue depuis la configuration (statique) du nœud ou à partir de l'entête P-Charging-Vector. Le P-Charging-Vector est une nouvelle entête introduite par le 3gpp pour la signalisation SIP. Pour les abonnés itinérants, le CDF du réseau visité est utilisé.
- Charging Gateway Function, cette fonction fait l'agrégation des fichiers de tarification, les formate et les envoie au système de facturation.
- Rf, cette interface transporte les enregistrements de taxation (CDR)
- Ga, elle transfère les CDR au CGF
- Bi, cette interface transporte les fichiers de tarification au système de facturation.

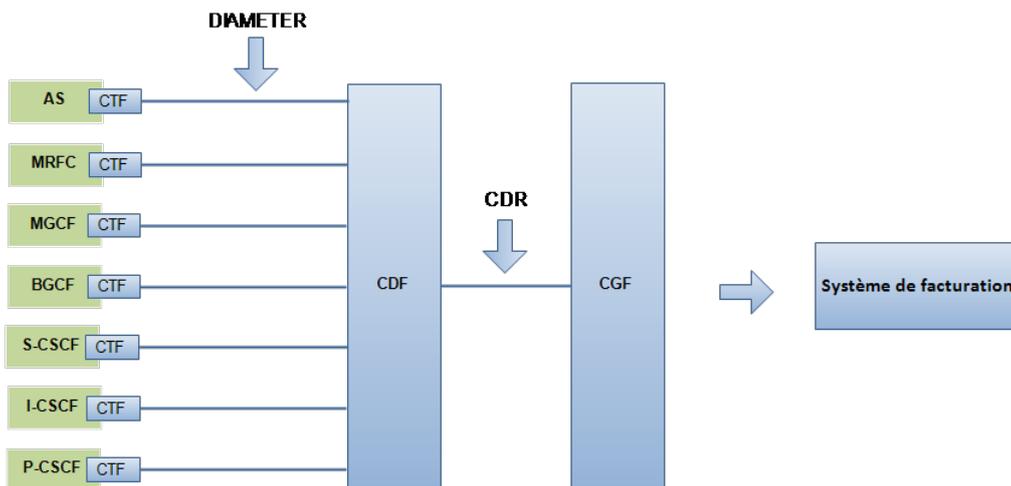


FIGURE I.2.11 – Architecture des entités de tarification hors-ligne

Supposons qu'un abonné A émet un appel vocal vers un autre B. Le serveur gérant la communication enverra un requête ACR (Accounting Request) à l'établissement de la session, périodiquement au cours de la session et à la fin de la session. Le CDF répondra avec des messages ACA (Accounting Answer).

## 2.7.2 Taxation en ligne

La plupart des mécanismes précédemment présentés s'appliquent aussi pour la taxation en ligne. Le CGF est l'intermédiaire entre le système de taxation en ligne – Online Charging System (OCS) – et le système de facturation. Diameter est le protocole exploité par l'interface Ro. La seule différence avec la taxation hors-ligne, est que les nœuds (serveur d'application, etc) fournissant les services doivent attendre les réponses à leur requêtes de taxation pour traiter la signalisation ou l'invocation des services. Les réponses sous forme de Credit Control Answer (CCA) comportent des messages OK ou non, mais aussi les informations relatives à l'abonné ayant initié la session. Ces informations concernent par exemple, le nombre de minutes restantes pour un appel en fonction du solde de l'abonné. Avant de consommer une ressource, le serveur d'application prenant en charge l'abonné initie une requête (Credit Control Request) à l'endroit de l'OCS pour servir le dit abonné. Lorsque le crédit demandé auprès de l'OCS arrive à terme, il fait une nouvelle demande. En cas de non-consommation complète d'un crédit alloué, le reste est retourné à l'OCS. Aussi s'il n'y a plus de crédit dans le compte de l'abonné, la communication est coupée.

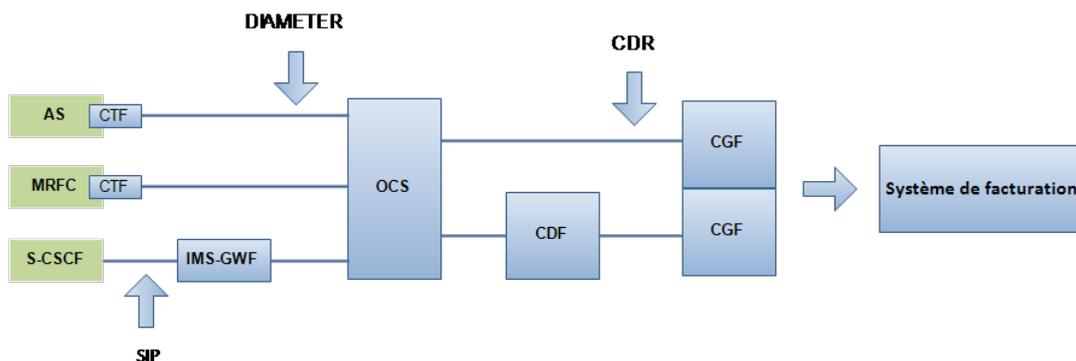


FIGURE I.2.12 – Architecture des entités de tarification en ligne

## 2.8 Mise en place d'une plateforme de service IMS

Pour pouvoir développer des services IMS et les tester, nous allons utiliser la plateforme open source OpenIMSCore pour le déploiement d'un cœur de réseau IMS et la plateforme open source VoIP Mobicents certifiée JSLEE.

### 2.8.1 Le cœur du réseau : OpenIMSCore

Open IMS est un cœur de réseau IMS basé sur la solution open source SIP Express Router (SER). Il a été développé par l'institut Fraunhofer FOKUS en Allemagne et les premières versions sont apparues à partir de 2006 et sont destinées à des plateformes du monde Linux. OpenIMSCore est une implémentation Open Source de la norme 3GPP IMS.

Ce projet a été lancé pour promouvoir l'adoption de la technologie IMS dans les réseaux de télécommunications de prochaine génération, et amener le développement de nouveaux services basés sur IMS. OpenIMSCore implémente les différentes fonctions qui forment ensemble les éléments de base d'une architecture IMS/NGN. Il s'agit des fonctions de contrôle de session d'un cœur de réseau IMS (serveurs CSCF - P-CSCF, I-CSCF et S-CSCF) fonction HSS permettant donc de provisionner un certain nombre d'utilisateurs et de leur associer un profil de service permettant la mise en œuvre de l'invocation de services si chers à l'architecture IMS. Tous les composants sont basés sur des logiciels open source, tels que SER (SIP Express Router) ou MySQL.

Le projet OpenIMSCore est avant tout conçu pour la recherche et le développement (fournisseurs de matériel de télécommunications, opérateurs de réseaux, projets de recherche universitaire).

#### 2.8.1.1 Configuration

OpenIMSCore est basée sur une architecture composée de quatre processus séparés. Le premier processus correspond à l'interface homme machine s'exécutant sur l'ordinateur de l'utilisateur ; le deuxième processus correspond à la couche session qui assure le contrôle de l'appel et qui gère l'établissement et la fin d'une session entre différents participants ; le troisième processus stocke les données dans le serveur de base de données et le dernier processus regroupe les serveurs d'application. Il est recommandé que les différents processus soient situés sur des serveurs séparés afin de mieux percevoir les interactions. Cependant, il est possible d'installer les éléments de contrôle, la base de données et les serveurs d'applications sur un même serveur. La décision finale devra être prise en tenant compte de la situation particulière d'un opérateur, comme, par exemple, le nombre d'utilisateurs, la charge de travail des machines, l'utilisation projetée du système, etc.

Le choix d'un serveur unique est un choix de simplicité et d'économie matérielle. Chacune des fonctions doit donc être connue du DNS, lui-même configuré sur le serveur commun. Le choix d'avoir un cœur de réseau gérant un domaine particulier implémenté sur une seule machine peut permettre de dupliquer aisément cette configuration afin d'étendre l'infrastructure à d'autres domaines IMS.

Pour la mise en place du cœur du réseau, la distribution de linux retenue est Debian (dans sa version Squeeze 6.0.3). Il s'agit d'une distribution avancée de Linux et très stable. C'est également une distribution adaptée à la recherche.

### 2.8.1.2 Étapes nécessaires à la mise en place d'OpenIMSCore

La mise en place de l'application OpenIMSCore passe par plusieurs étapes.

#### – Pré-requis (Outils et librairies)

Des outils et librairies sont nécessaires pour la mise en place d'OpenIMSCore. Le tout premier est celui permettant de récupérer la dernière version de l'application. Il s'agit de subversion (svn). Une partie de l'application est écrite en c, cela nécessite des outils de compilation et d'installation tels que GCC3/4, make, ant. L'autre partie est écrite en JAVA et requiert JDK (Java Development Kit) (version supérieure ou égale à 1.5). Il est également utile d'installer un système de gestion de bases de données relationnelles (SGBDR), tel que MySQL, pour la base de données HSS ; par ailleurs, un serveur DNS est requis pour la configuration du domaine.

#### – Création des dossiers d'installation

Le code source d'OpenIMSCore est pré-configuré pour fonctionner selon un chemin de fichier standard à savoir /opt/OpenIMSCore/. Toute l'installation sera effectuée dans ce dossier. Il faut le créer manuellement. Il est également utile de créer manuellement deux autres dossiers dans ce dossier principal : ser\_ims pour y placer les serveurs CSCF et FHoSS pour y placer le serveur FHoSS.

#### – Récupération des codes sources

Pour bénéficier des dernières mises à jour du code de l'application, on utilise l'outil svn pour récupérer les codes sources. Ces codes doivent être placés dans les dossiers correspondants. Par exemple, pour récupérer les serveurs CSCF, on lance la commande

```
svn checkout http://svn.berlios.de/svnroot/repos/openimscore/ser\_ims/trunk opt/OpenIMSCore/ser_ims/
```

#### – Compilation d'openIMSCore

Pour compiler les serveurs CSCF, on se place dans le dossier opt/OpenIMSCore/ser\_ims et on lance la commande *make install-libs all*.

En ce qui concerne le serveur FHoSS, on se place d'abord dans le dossier correspondant. On renseigne ensuite la variable d'environnement JAVA\_HOME. Enfin, on lance successivement les commandes *ant compile* pour la compilation et **ant deploy** pour le déploiement.

#### – Mise en place des bases de données

En dehors de la base de données HSS, OpenIMSCore utilise aussi une base nommée icscf. La mise en place de la première nécessite l'exécution de deux scripts : hss\_db.sql et userdata.sql. Pour la seconde, on exécute le script icscf.sql.

#### – Configuration de l'environnement DNS

La configuration initiale d'openIMSCore permet uniquement un fonctionnement en localhost. Le domaine par défaut est « open-ims.test ».

Un exemple de fichier de zone DNS est présent dans le répertoire d'installation d'openIMSCore.

On copie ce fichier et on le place dans le répertoire de configuration du DNS (/etc/bind/). Il faut s'assurer que le named.conf fasse appel au named.conf.local. Il faut ensuite configurer le client DNS

textitnomenclatureDSNDomain Name Service (/etc/resolv.conf).

– **Fichiers de configuration et de lancement des serveurs d'openIMSCore**

A ce niveau, on copie les fichiers de configuration .cfg et .xml, ainsi que les scripts de lancement .sh des serveurs x-cscf dans le répertoire d'openIMSCore (/opt/OpenIMSCore/).

– **Lancement d'openIMSCore**

On se place dans le dossier /opt/OpenIMSCore/ pour lancer en parallèle les 3 fichiers pcsf.sh, icscf.sh et scscf.sh correspondant aux trois serveurs du CSCF.

Quant au serveur FHoSS, il faut s'assurer que la variable d'environnement JAVA\_HOME est bien renseignée avant de lancer le serveur grâce au script FHoSS/deploy/startup.sh.

On peut vérifier que le FHoSS est correctement lancé en accédant à l'interface.

<http://localhost:8080/>.

– **Création d'un script de démarrage automatique**

Il peut être utile de créer un script qui lance de manière automatique les différents serveurs et qui s'occupe de faire les configurations nécessaires avant le lancement des serveurs (par exemple mettre à jour/exporter la variable JAVA\_HOME, etc.).

## Remarques

L'installation du cœur de l'IMS, telle que décrite ci-dessus est souvent un peu complexe à réaliser. Cela est dû à deux grandes raisons :

- a) il faut pouvoir adapter les bibliothèques à la distribution de Linux sur laquelle l'installation doit avoir lieu ;
- b) la configuration du domaine et des adresses des serveurs suivant un contexte donné n'est pas toujours aisée ;

### 2.8.1.3 Les terminaux

#### Uctimsclient

Le client IMS Uctimsclient a été conçu pour être utilisé en conjonction avec Open IMS. C'est donc une solution intéressante pour compléter notre réseau cœur car elle a été de ce fait testée et validée avec ce dernier. Le client Uct a été développé par un groupe de recherche de l'Université du Cap en Afrique du Sud. La dernière version présente encore quelques dysfonctionnements, notamment dans la gestion des informations de présence, mais permet une émulation très correcte des caractéristiques principales d'un terminal IMS.

## Boghe

Boghe est un client IMS open source similaire au client propriétaire Mercurio qui est une des toutes premières solutions commerciales d'un client IMS.

Ses caractéristiques sont plutôt riches incluant notamment une gestion de serveurs XDMS (XCAP Document Management server) dès sa version gratuite d'évaluation (version Bronze). Je regrette cependant que cette même version ne supporte les mécanismes de pré-condition (dit QoS) et que par conséquent son interopérabilité avec le client Uct nécessite de désactiver cette fonction. Son usage est toutefois globalement agréable et son support de Windows et Windows mobile permet de mixer les environnements (Windows + Linux).

## Monster

Monster n'est à proprement parler pas un simple client mais plutôt un réel framework permettant de développer des applications clientes destinées au monde mobile (IMS en autre). Il a également été développé par l'institut Fraunhofer FOKUS.

## Imsdroid

IMSDroid est la première source de toutes les fonctionnalités ouverts SIP / IMS client pour Android. C'est aussi le seul client SIP gratuit Android, qui supporte les appels vidéo directs.

IMSDroid est la première application open source totalement 3GPP IMS client pour les appareils Android (1.5 et versions ultérieures). Le but principal du projet est d'exposer les caractéristiques de *doubango* et d'offrir un client IMS à la communauté open source. Doubango est une expérimentation, open source, 3GPP IMS / LTE qui prend en compte les systèmes embarqués (Android, Windows Mobile, Symbian, iPhone, etc.) et les systèmes de bureau (Windows XP/Vista/7, Mac OS X, Linux, etc.) et est écrit en ANSI-C pour faciliter la portabilité. Le cadre a été soigneusement conçu pour travailler efficacement sur les systèmes embarqués avec une mémoire limitée et la puissance de calcul faible.

### 2.8.1.4 Les services

#### Le serveur applicatif OpenSIPS

OpenSIPS est une implémentation Open Source d'un serveur SIP. Il s'agit en fait d'une instantiation de la solution OpenSER rebaptisé pour l'occasion.

OpenSIPS est plus qu'un simple proxy SIP car il inclut des fonctionnalités de niveau applicatif et c'est justement dans ce cadre que qu'il est utilisé ici. Grâce à son moteur de routage très souple et à une conception très modulaire (mais de configuration un peu obscure ... on ne peut pas tout avoir), il peut être utilisé comme un vrai serveur de présence. Il remplit donc ce rôle dans l'architecture actuelle et sera perçu par le cœur de réseau IMS comme un serveur applicatif dont les caractéristiques (d'invocation et droits d'accès) seront configurées dans le HSS.

#### Le serveur de document XML OpenXCAP

OpenXCAP est une implémentation open source d'un serveur XCAP. Grâce au protocole XCAP un client IMS peut lire, écrire et modifier des données de configuration de service stockées au format XML sur un serveur (de type XDMS). Un serveur XCAP peut donc être utilisé par des clients IMS

pour gérer des listes de contacts et la politique d'autorisation d'accès à des informations de type présence. Le serveur OpenXCAP a été développé pour fonctionner de pair avec le serveur OpenSIPS utilisé en tant que serveur de présence (Presence Agent PA).

### Le serveur d'application de télévision sur IP UCT Advanced IPTv

L'application UCT Advanced IPTv [28] est une mise en œuvre conforme aux normes d'un service IMS basée IPTV. Il a été créé un groupe de recherche de l'Université de Cape Town.

Le projet est à ses débuts et en phase d'expérimentation, mais les résultats déjà sont très encourageants.

#### 2.8.1.5 Vue globale de la plateforme IMS et des services

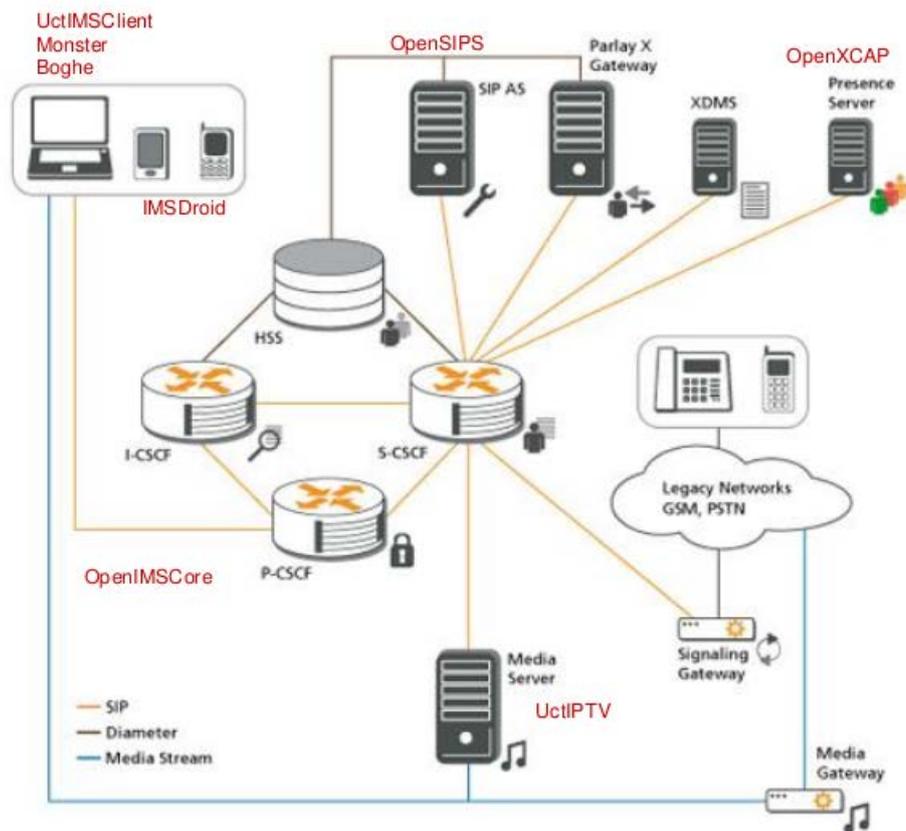


FIGURE I.2.13 – Architecture de la plateforme implémentée

L'architecture IMS déployée est une vue simplifiée d'une architecture. Entièrement basée sur des composants open source, elle adresse principalement les mécanismes de routage de cœur de réseau mais permet cependant d'y intégrer des services.

## 2.8.2 Plateforme de développement de services pour IMS : Mobicents

### 2.8.2.1 Mobicents

Mobicents [29] est une plateforme ouverte de VoIP de source entièrement certifiée pour JSLEE. Mobicents met en application également certains des dispositifs proposés de JAIN SLEE 1.1. Mobicents adapte les applications de télécoms à un environnement composant robuste de modèle et d'exécution. Il complète J2EE pour permettre la convergence de la voix, de la vidéo et des données dans des applications intelligentes de prochaine génération.

Mobicents est la source ouverte la plus populaire SIP Serveur d'application pour la plateforme de Java. Il facilite l'exécution de nouveaux services dans une manière simple et rapide ; il permet le développement d'une plateforme orientée vers le marché et rentable. En même temps, Mobicents favorise un environnement standard pour le marché libre naissant de créateurs d'application et tire profit (et visibilité d'élasticité) des programmeurs mondiaux doués. L'installation est facile et bien-expliquée et le projet inclut un éventail d'exemples intéressant.

Mobicents est distribué dans le cadre d'une politique duelle flexible de permis. Le permis de distribution par défaut est GNU.

Mobicents est la réunion de quatre composants ou outils à savoir : J2EE, JSLEE, SIP servlets Mobicents et Media Server.

## Environnement J2EE

Java 2 Enterprise Edition (J2EE) est une spécification pour la technique Java de Sun plus particulièrement destinée aux applications d'entreprise. Ces applications sont considérées dans une approche multi-niveaux. Dans ce but, toute implémentation de cette spécification contient un ensemble d'extensions au framework Java standard (JSE, Java Standard Edition) afin de faciliter la création d'applications réparties.

Pour ce faire, Java EE définit les éléments suivants :

- a) Une plate-forme (Java EE Platform), pour héberger et exécuter les applications.
- b) Une suite de tests (Java EE Compatibility Test Suite) pour vérifier la compatibilité.
- c) Une réalisation de référence (Java EE Reference Implementation), qui est GlassFish.
- d) Un catalogue de bonnes pratiques (Java EE BluePrints).

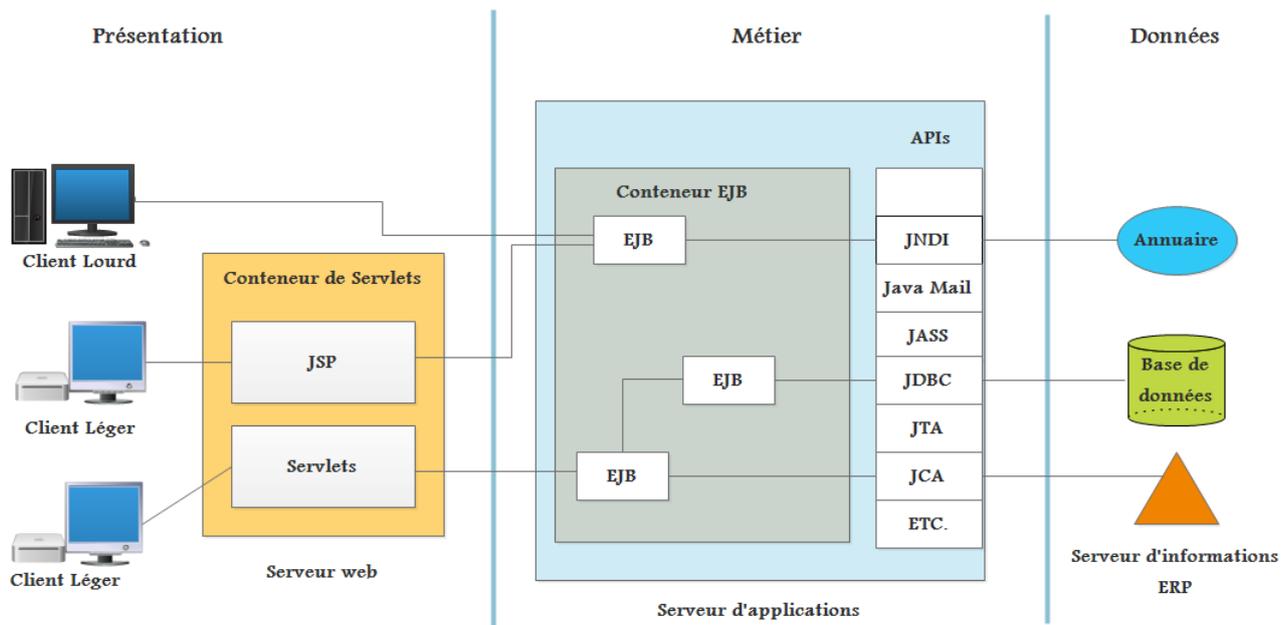


FIGURE I.2.14 – Architecture J2EE

## Environnement JSLEE

JAIN est un ensemble d'API (Application Program Interfaces) JAVA qui permet de développer rapidement de nouveaux services pour des réseaux de télécommunication voix ou données, indépendamment des serveurs utilisés (matériel). De plus, JAIN étant basé sur la plateforme JAVA, il introduit la portabilité des services entre systèmes et permet des accès sécurisés aux ressources des différents réseaux.

La technologie JAIN change radicalement le marché des télécommunications en permettant le passage de systèmes fermés et propriétaires à des systèmes ouverts offrant une interconnexion totale des différents réseaux existant (PSTN, IP, ATM, GSM, WLAN). JAIN se compose de plusieurs groupes agissant chacun à un niveau différent et offrant chacun leurs avantages :

- a) JAIN Protocol API
- b) JAIN Call Control API (JCC)
- c) JAIN Service Logic Execution Environment (JSLEE)
- d) JAIN Service Creation Environment (JSCE)

JSLEE est un environnement d'exécution qui offre une abstraction par rapport à ce qui l'entoure.

- a) Il permet au développeur de se concentrer sur le service qu'il désire créer sans se préoccuper de l'infrastructure et des protocoles de signalisations utilisés.
- b) Les services sont totalement indépendants des changements du réseau et des protocoles utilisés.
- c) Tout service développé pour cet environnement fonctionnera dans un autre environnement JSLEE.
- d) Il est possible de créer rapidement des nouveaux services en combinant des services déjà existant et déjà créer pour fonctionner sur JSLEE.

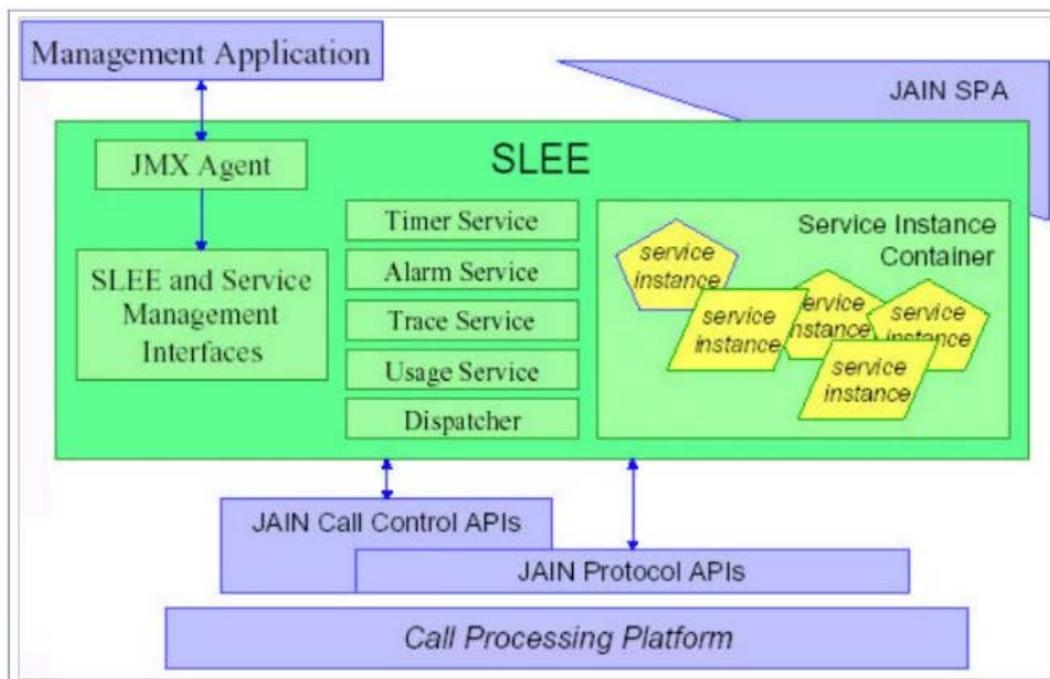


FIGURE I.2.15 – Architecture JSLEE

Il existe deux possibilités pour créer des services pour JAIN SLEE. La première consiste à utiliser un éditeur conventionnel ou un outil de développement pour JAVA, puis de compiler le service et de le déployer dans l'environnement SLEE. La seconde solution consiste à utiliser l'outil JSCE proposé par JAIN

### Comparaison entre J2EE et JSLEE

- a) Les implémentations J2EE sont utilisées pour fournir de hautes performances dans des **réseaux d'entreprise**, mais elles ne sont pas capables de fournir les performances requises pour un **réseau de communication** ;
- b) Les implémentations JSLEE sont utilisées pour fournir de hautes performances dans les **réseaux de communication**, mais elles ne sont pas capables de fournir les performances requises pour un **réseau d'entreprises** ;
- c) JSLEE est comparable à la plateforme J2EE mais ne fait pas partie de ce modèle de composants.
- d) JSLEE est conçu pour les applications orientées événement qui requièrent de faibles temps de réponses et un haut débit.

### Exemples d'applications

- a) Services réseaux
  - Services prépayés
  - VPN (Réseaux Privés Virtuels)
  - Messagerie instantanée (antispam pour les SMS)
- b) Contrôles réseaux
  - IP Multimedia Subsystem (IMS)
  - Contrôle d'appel

- Réseaux de nouvelle génération (Voix sur IP, 3G, etc.)

## Les SIP servlets Mobicents

La solution SIP servlets de Mobicents est une plate-forme ouverte permettant de développer et déployer des applications SIP portables et des services Web Java EE convergents. Il s'agit de la première implémentation open source certifiée conforme de la Servlet SIP 1.1 (JSR 289 Spec).

La spécification SIP Servlets offre la possibilité de mélanger les servlets SIP avec HTTP Servlets API ou J2EE pour mettre en œuvre des interactions de composants multimédia avec des applications d'entreprise nouvelles ou existantes.

Il tire partie du modèle populaire, Servlets, qui est familière aux développeurs servlets HTTP. Il utilise le même modèle de développement et définit SIP des objets SIP de haut niveau, similaires à ceux HTTP Servlets, et des outils d'aide (tels que Proxy et Retour à User Agent Retour) pour faciliter le développement d'applications SIP.

En outre, elle collabore avec les servlets HTTP d'une manière à leur permettre d'être en mesure de lancer des appels ou des parts de données. Il est cependant différent de servlets HTTP parce que les servlets SIP sont asynchrones, n'ont pas racine du contexte et des multiples servlets SIP peuvent recevoir le même message SIP.

## L'interface Diameter Mobicents

Mobicents propose une stack Diameter permettant de compléter un environnement de développement afin de permettre au serveur applicatif de pouvoir s'interfacer avec le HSS (Interface Sh). Celle-ci est fournie pré-intégrée avec l'environnement JBoss et SIP Servlets. Cette interface permettra de développer des services nécessitant un accès à la base de données HSS.

### 2.8.2.2 Media Server

Mobicents Media Server est un logiciel open source, la mise en œuvre du logiciel basée sur l'élément de réseau VoIP responsables de la manipulation médiatique. Il supporte les interfaces IP et TDM, et peut agir à la fois comme un serveur média et en tant que Media Gateway. Il fournit un support pour les services à la fois distribués et centralisés, y compris circuit de commutation voix / vidéo, des annonces et des tonalités.

Dans le monde moderne VoIP, les fournisseurs offrent des services hautement personnalisés qui allient des supports tels que audio, vidéo et messagerie instantanée. Ces services nécessitent des supports dédiés et personnalisés des capacités de traitement. Pour y parvenir, le réseau VoIP sépare les fonctions de traitement des médias dans un nœud dédié qui est responsable de traitement des médias seulement, alors que toute l'intelligence est exécutée par le contrôleur d'appel séparée. Le nœud de traitement du flux média est appelé un serveur de médias. Le nœud de manipulation des médias doit également être en mesure de combler le fossé entre les anciens systèmes traditionnels et modernes des réseaux VoIP. Ce nœud est généralement appelé un Media Gateway.

Mobicents Media Server peut fonctionner avec le serveur d'application JAIN SLEE Mobicents. Cependant, le Media Server est livré avec une interface standard de Telco MGCP, qui permet d'utiliser un autre contrôleur d'appel, si désiré.

Le Media Server est implémenté en utilisant le noyau JBoss Microcontainer, ce qui permet d'archiver un maximum de flexibilité. Il donne la possibilité d'adopter le Media Server pour des actions tâche

spécifique, et / ou étendre les fonctions du Media Server en installant des supports supplémentaires de traitement des composants.

Voici quelques spécifications techniques de Media Server :

- a **Codeurs** : G711 (a-Law, u-Law), GSM, SPEEX, G729, DTMF(RFC 2833, INBAND)
- b **Fichiers multimédia** : WAV, SPX, GSM, MP3, ISO
- c **Signalisation et contrôle** : MGCP, Java Media Control API(JSR-309)
- d **Traitement audio** : transcodage audio, atténuateur et conférence ; DTMF, Text-To- Speech, Capacité à générer le silence ou le bruit

### 2.8.2.3 Eclipse

**Eclipse IDE** (Integrated Development Environment) est un IDE dédié au développement de logiciels basés sur **Java** (bien que d'autres langages soient supportés également (**C/C++**, **python**, **php**, **ruby**, etc.)). Il s'agit d'un projet de la **Fondation Eclipse** visant à développer tout un environnement de développement libre, extensible, universel et polyvalent. Son objectif est de produire et fournir divers outils gravitant autour de la réalisation de logiciel, englobant les activités de codage logiciel proprement dites (avec notamment un environnement de développement intégré) mais aussi de modélisation, de conception, de test, de reporting, etc. Son environnement de développement notamment vise à la généralité pour lui permettre de supporter n'importe quel langage de programmation.

La caractéristique essentielle d'Eclipse est l'extensibilité de l'environnement. Plus que de se focaliser sur un environnement de développement Java, les concepteurs d'Eclipse se sont efforcés avant tout de créer un socle applicatif sur lequel viennent se greffer des modules et/ou plugins.

Le projet Eclipse est pour cela organisé en un ensemble cohérent de projets logiciels distincts, sa spécificité tenant à son architecture totalement développée autour de la notion de plugin (en conformité avec la norme OSGi) : toutes les fonctionnalités de l'atelier logiciel doivent être développées en tant que plug-in bâti autour de l'**IDE Eclipse Platform**.

La base de l'environnement de développement intégré que constitue l'EDI Eclipse Platform est composée de :

- a) **Platform Runtime** démarrnant la plateforme et gérant les plugins
- b) **SWT**, la bibliothèque graphique de base de l'EDI
- c) **JFace**, une bibliothèque graphique de plus haut niveau basée sur SWT
- d) **Eclipse Workbench**, la dernière couche graphique permettant de manipuler des composants, tels que des vues, des éditeurs et des perspectives.

Ces composants de base peuvent être réutilisés pour développer des clients lourds indépendants d'Eclipse grâce au projet Eclipse RCP (Rich Client Platform). L'ensemble des outils de développement Java sont ensuite ajoutés en tant que plugins, regroupés dans le projet **Java Development Tools** (JDT). Ces plugins sont architecturés selon les recommandations de OSGi.

### 2.8.2.4 Serveur JBOSS

La spécification EJB 3 dispose d'atouts indéniables pour redorer le blason des EJB. D'une part elle simplifie le processus de développement en allégeant le code, d'autre part les lacunes des EJB Entity sont largement comblées par une nouvelle spécification qui traite spécialement du problème de la persistance : JPA (Java Persistence API).

JBoss Application Server est un serveur d'applications J2EE Libre entièrement écrit en Java, publié sous licence GNU LGPL. Parce que le logiciel est écrit en Java, JBoss Application Server peut être utilisé sur tout système d'exploitation fournissant une machine virtuelle Java (JVM).

JBoss est un serveur d'application J2EE développé à partir de 1999 par un français Marc FLEURY. Ce serveur est écrit en Java et distribué sous licence LGPL.

### 2.8.3 L'intégration de la plate-forme de développement à OpenIMScore

Comme le montre la figure I.2.16 ci-dessous, l'environnement d'exécution est principalement basé sur JavaEE serveur JBoss et Mobicents SIP Servlet. À côté, une partie supplémentaire API Java (également fourni par Mobicents) est intégrée pour l'interfaçage avec le HSS, XDMS et le serveur multimédia.

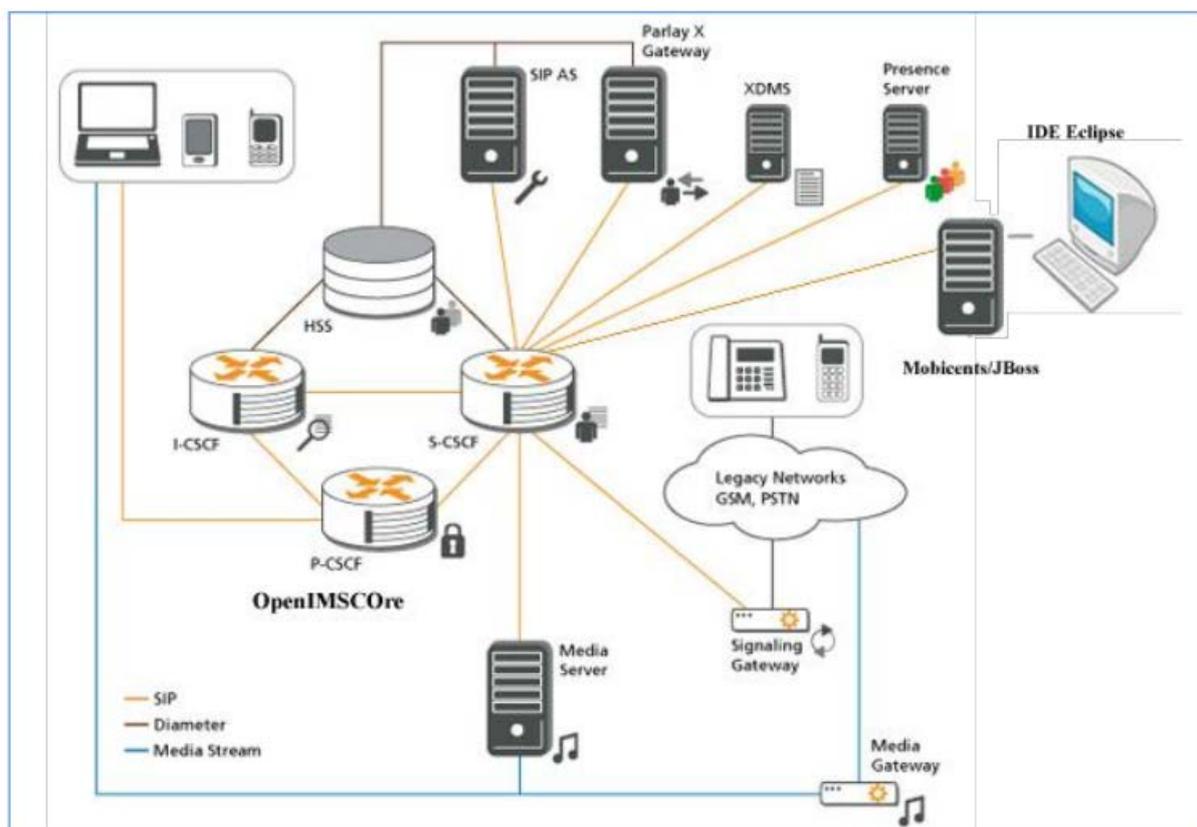


FIGURE I.2.16 – Intégration Mobicents/JBoss à OpenIMScore

L'application de base permet de construire convergé Web / Servlet SIP en utilisant la mise en œuvre de servlet SIP qui donnent toute la flexibilité appropriée requise pour l'interfaçage avec l'IMS de base grâce à l'interface ISC. Une partie importante des codes peuvent également être basés sur des pages JSP pour l'interaction avec les clients HTTP.

La plateforme open source Mobicents SIP servlet + Eclipse + JBoss réunit en son sein les technologies les plus avancées de développement et de déploiement d'application SIP.

Par ailleurs, son intégration aisée à la plate-forme IMS constitue un véritable atout en ce que cela permettra aux chercheurs de faire des simulations concrètes et aux opérateurs de mettre en œuvre de nouvelles approches et concepts ainsi que de nouveaux services et protocoles sans le moindre coût.

Pour confirmer cette possibilité d'intégration des services dans l'IMS grâce à Mobicents, nous avons conçu deux services en environnement Mobicents. Le premier permet de faire des transactions bancaires à partir d'un client IMS et le second, de prendre rendez-vous à partir d'un client IMS.

## 2.9 Quelques exemples d'applications compatibles IMS

### 2.9.1 Application E-banking

Dans le but d'étayer notre étude des divers APIs et environnements de développement de services IMS, nous avons décidé de nous intéresser à une application du domaine E-banking. D'abord, E-Banking est l'abréviation de Electronic Banking, on comprend qu'il s'agit des opérations bancaires en ligne menées par des moyens électroniques. Nous entendons par moyens électroniques, les confirmations de transaction par e-mail ou sms, consultation de solde depuis un téléphone ou sur le site web de votre banque.

L'E-Banking regroupe un ensemble d'éléments tels que le Mobile Money ou Mobile Payment, web banking, etc. Avec le Mobile Banking, on peut réaliser un grand nombre de transactions depuis le téléphone mobile. Les services vont du transfert d'argent au paiement de facture en passant par le virement de salaire. Il existe une catégorisation des transactions mobiles en rapport avec l'E-Banking. On distingue trois branches principales : les services mobiles, le commerce par mobile et le paiement par mobile.

Le Mobile Banking fait partie des services mobiles. C'est l'utilisation du téléphone portable pour fournir des services financiers tels que les échanges d'informations financières et les transactions financières .

Nous sommes intéressés particulièrement par le Mobile Banking en raison des nombreuses facilités qu'il procure dans un environnement bancaire comme celui des pays africains.

En effet, dans ces pays – particulièrement les pays sub-sahariens – le taux de bancarisation est faible. Aussi, une répartition géographique inadéquate des institutions financières n'est pas en faveur des populations éloignées des centres urbains. Il faut également noter que le taux de pénétration des téléphones mobiles est important (près de 69% pour 2015) dans ces pays. Il apparaît que l'on peut venir à bout de ce problème en facilitant l'accès aux services bancaires et financiers aux populations non bancarisées depuis leurs téléphones mobiles. C'est dans cette optique que nous comptons développer une application de transfert d'argent depuis un appareil mobile IMS.

#### 2.9.1.1 Services envisagés

Cette application qui permettra de faire des transferts de personne à personne, de faire un retrait d'argent ou de payer une facture. Pour pouvoir utiliser ce service, un abonné doit d'abord souscrire au service. Ainsi, on pourra lui créer un compte l'identifiant et fournissant des informations sur son avoir, son nom et son adresse SIP. L'adresse SIP est utilisée pour joindre l'utilisateur. Un utilisateur ayant souscrit au service peut faire des transferts d'argent, retirer du liquide soit à la banque ou auprès d'un agent agréé. Dans notre contexte, il y a plusieurs types d'utilisateurs :

- le simple utilisateur qui souscrit au service,
- un agent qui peut encaisser de l'argent en espèce pour créditer le compte d'un utilisateur, ou auprès de qui l'on peut faire des retraits,
- une caissière de banque qui a tous les rôles d'un agent.

Lorsqu'un transfert est effectué, si le récipiendaire n'est pas abonné au service, nous générons un code valide 24 heures avec lequel il peut retirer le montant du transfert auprès d'un agent ou d'un guichet de banque. Par contre, si le récipiendaire est abonné au système, il voit son solde augmenté du montant de la transaction. Un agent est aussi un abonné du service, toutes les transactions qu'il effectue impactent directement son compte personnel. Cependant, la caissière de banque ne dispose que d'un compte virtuel pour s'authentifier auprès du système. Les transactions concernant les paiements de facture sont défalqués du compte de l'utilisateur.

Pour simplifier le développement de notre service, la facturation n'a pas été prise en compte.

Pour ce type de service, nous avons à disposition deux principaux canaux : la messagerie et la voix avec un serveur vocal interactif. Nous envisageons exploiter ces deux médias pour le service. Ainsi, un utilisateur peut depuis n'importe quel terminal SIP faire des transactions et entrer en contact avec sa banque par la messagerie instantanée par exemple, ou un appel vocal.

Mieux, on pourrait interfacer ce système avec les réseaux sociaux. Ces derniers pour la plupart ont des systèmes de messagerie fonctionnant avec le protocole XMPP (eXtensible Messaging and Presence Protocol). Ceci nous conduit à un nouveau concept, celui de la banque sociale et du commercial social. Ainsi les usagers du système qui sont aussi des utilisateurs de réseaux sociaux peuvent accéder au service depuis l'interface du réseau auquel ils sont abonnés. Ils pourront également acheter des biens et effectuer leur paiement en ligne. C'est un transfert de personne à personne comme nous l'avons souligné plus haut. Pour valider les codes émis par le système, la caissière de banque doit disposer d'une interface pour faire les vérifications et servir les clients. Il apparaît clairement que ce service est une application convergente. La convergence est l'une des grandes promesses de IMS.

### 2.9.1.2 Contraintes d'intégrité

Un service de cette nature doit garantir un minimum de sécurité pour ses utilisateurs. Il y va de la qualité du service offert et de la sécurité des avoirs des utilisateurs du système.

Si les messages échangés ne sont pas chiffrés, des individus mal intentionnés pourraient à partir d'outils simples récupérer les informations des comptes de nos utilisateurs. Ces individus pourraient donc se faire passer pour les utilisateurs du système, car possédant les mots de passe, pour faire des transactions frauduleuses.

### 2.9.1.3 Choix des outils

#### Vue architecturale

Le choix des outils s'inspire beaucoup de l'étude effectuée à la section 1.6. Ainsi nous utiliserons Mobicents JAIN SLEE comme serveur d'application. Ces nombreux avantages font de lui un choix pertinent en raison de sa capacité à prendre en charge un grand nombre de protocoles tels que SIP principalement, MGCP (Media Gateway Control Protocol), Diameter, etc. MGCP est le protocole utilisé par le MFRC pour contrôler le serveur de média. Dans cette configuration, l'application jouera le rôle de Media Gateway Controller et Mobicents Media Server sera le Media Gateway. Ce dernier sera utilisé pour jouer les annonces vocales et guider l'utilisateur à travers un menu vocal interactif. SIP étant le protocole sous-jacent de IMS, il est utilisé dans cette application pour supporter la fonctionnalité de messagerie instantanée. Pour ces deux média, il faut installer le MGCP RA et le SIP RA au niveau du serveur Mobicents JSLEE.

En autres outils, nous avons exploité JPA, Hibernate, Maven, la pile protocolaire JAIN-SIP, le framework JSF et une base de données MySQL. Java Persistence API (JPA) est une API qui permet d'annoter les classes java pour faire la correspondance entre les objets JAVA et les tables de la base de données. C'est son implémentation nommée Hibernate que nous avons choisi. Maven est un outil qui facilite la compilation et le déploiement d'application. MySQL est un système de gestion de base de données relationnelles. Elle va servir à sauvegarder les informations des utilisateurs et les transactions effectuées. Le framework JSF(Java Server Faces) est un ensemble d'outils pour développer des applications web. Il nous a été utile pour l'interface d'administration du système et l'interface du caissier.

## Vue fonctionnelle

La vue fonctionnelle correspond à la logique métier de l'application. Selon le médium d'accès, l'utilisateur sera connecté au système soit par la messagerie instantanée ou par un serveur vocal interactif pour effectuer ses transactions.

Dans le cas de la messagerie instantanée, après une connexion réussie au réseau IMS, l'abonné ajoute l'adresse SIP « *mbank@open-ims.test* » dans la liste de ces contacts pour échanger avec l'application. Ce dernier agira en robot et répondra automatiquement aux requêtes. Le format des requêtes en règle générale que nous avons choisi est celui-ci : *numeroCompte motDepasse operation infosSupplementaires*. Ce format est détaillé ci-après selon les opérations à réaliser. Les mots clés identifiant les opérations sont : *solde* pour une demande de solde, *sde* pour payer une facture SDE, *sonatel* pour une facture sonatel, *trans* pour faire un transfert d'argent, *code* pour avoir des informations sur un code généré, *vcode* pour valider un code de transfert afin de remettre de l'espèce au bénéficiaire du transfert et *load* pour créditer un compte. Certaines opérations nécessitent un niveau d'accréditation donné. Par exemple pour effectuer une demande de solde, l'utilisateur saisira ceci : **1111 passer solde** où 1111 est son numéro de compte, passer son mot de passe et solde représente l'opération. Pour effectuer un transfert, l'utilisateur saisira : **1111 passer trans leibniz@open-ims.test** où leibniz@open-ims.test constitue l'adresse SIP du récipiendaire du transfert. Voici une liste des différentes actions possibles :

- *numeroCompte motDePasse solde*
- *numeroCompte motDePasse trans montant @SIPRecepiendaire*
- *numeroCompte motDePasse trans montant @SIPRecepiendaire [nomRecepiendaire prenomRecepiendaire]*
- *numeroCompte motDePasse load montant @SIPRecepiendaire*
- *numeroCompte motDePasse code trxCode*
- *numeroCompte motDePasse vcode trxCode*
- *numeroCompte motDePasse sde montant numeroFacture*
- *numeroCompte motDePasse sonatel montant numeroFacture*
- *numeroCompte motDePasse excaf montant numeroFacture*

Dans cette application, nous avons fait le choix d'une double authentification. Ainsi pour avoir accès au service, le client doit être obligatoirement connecté avec son adresse SIP enregistrée dans le système. Nous vérifions donc la conformité de l'adresse SIP, du mot de passe et du numéro de compte.

Avec l'application web, l'administrateur crée les utilisateurs en renseignant leur données de connexion et le solde initial pour les profils « Agent » et « Abonné ». Le caissier procède à la validation d'un code en le saisissant sur son interface de base. Le système recherche les données associées au code. Il peut ensuite valider la transaction si le code est correct ; au cas contraire il lui est impossible

d'opérer cette validation. Dans tous les cas, le système répondra automatiquement avec le message approprié.

### 2.9.1.4 Présentation des résultats

A travers cette sous-section, nous allons montrer les résultats du développement effectué au fil de quelques captures d'écran illustrant l'utilisation de l'application.

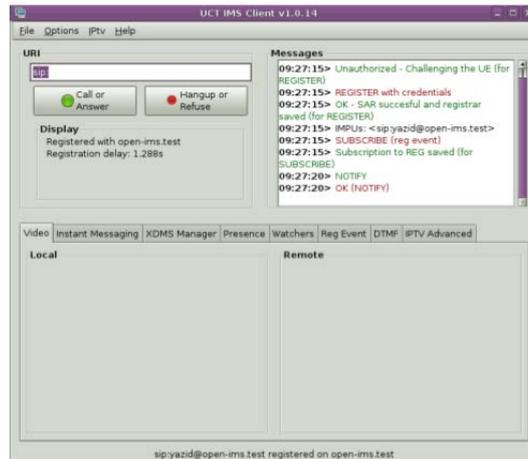


FIGURE I.2.17 – Connexion IMS

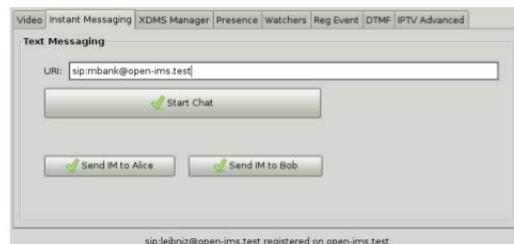


FIGURE I.2.18 – Ajout de l'adresse SIP de l'application

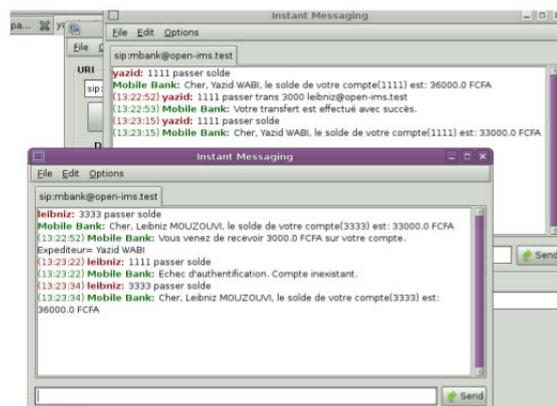


FIGURE I.2.19 – Échanges avec le système

## 2.9.2 Application de prise de Rendez-vous

Le service de gestion de RDV est une application permettant aux utilisateurs de prendre un rendez-vous à travers la messagerie instantanée. Dans des pays comme le nôtre où il est souvent utile de se déplacer jusqu'à l'endroit voulu pour pouvoir prendre un rendez-vous, cette application vient améliorer le quotidien des utilisateurs. Elle peut être utilisée par tous les types d'entreprises.

### 2.9.2.1 Scénario d'utilisation

Le client, après s'être connecté depuis son client SIP (UCT IMS Client, IMSDroid, ...) au réseau IMS (ici OpenIMScore), envoie : « rdv <prénom nom téléphone> » à l'application (rdv@imscore.sn). Celui-ci lui retourne le prochain créneau disponible. Si cela arrange le client, il accepte en répondant par un oui et le serveur lui envoie la confirmation du rdv composé de la date et de l'heure.

Sinon il répond par un non et la dialogue se termine. Ainsi il n'y aura pas de réservation de rendez-vous.

Au niveau de l'administrateur, il y a une interface web pour administrer l'application (gérer la disponibilité, paramétrer les heures de début et de fin ainsi que les créneaux de prise de rendez-vous, consulter les clients et les réservations, ajouter, modifier ou supprimer un autre admin...).

### 2.9.2.2 Présentation des résultats

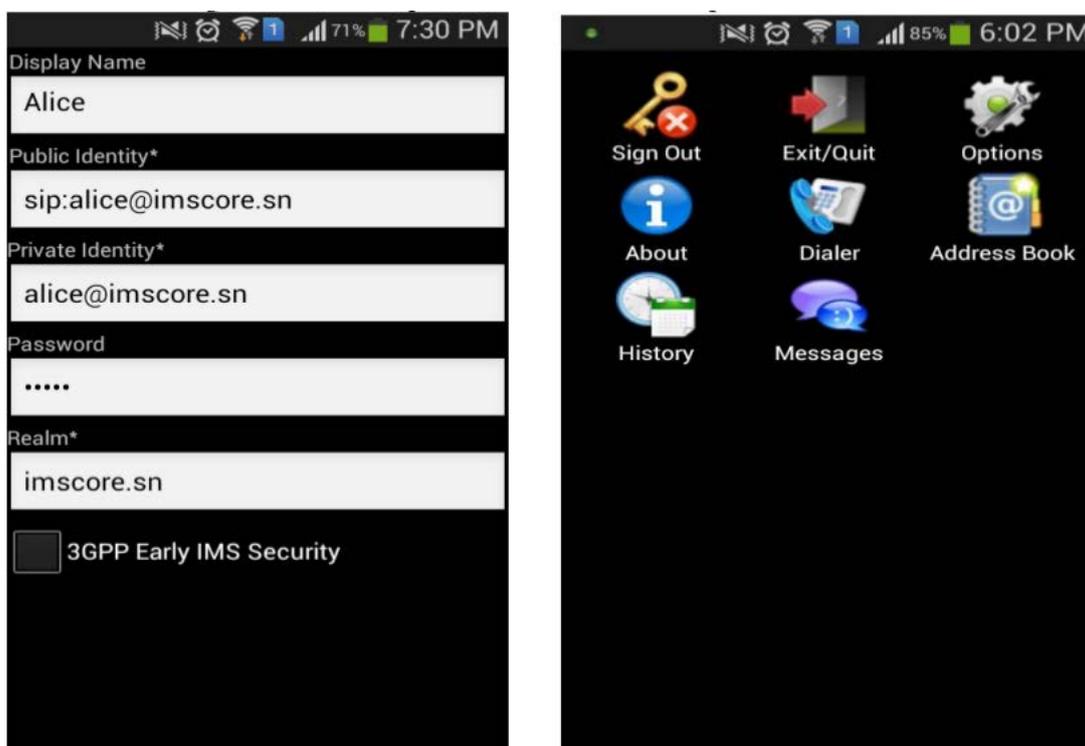


FIGURE I.2.20 – Utilisateur connecté depuis IMSDroid (installé sur un téléphone Android)

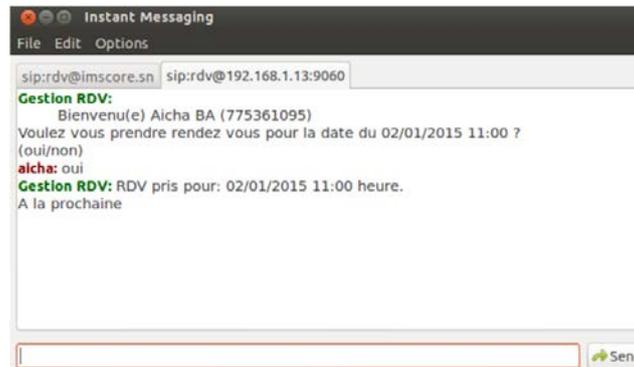


FIGURE I.2.21 – Client prenant un rendez-vous depuis UCT IMS client



FIGURE I.2.22 – Client prenant rendez-vous depuis IMSDroid

## 2.10 Conclusion

La facturation des services dans les réseaux de télécoms est une fonctionnalité importante. Dans le cas de IMS, c'est l'identité IMSU d'un utilisateur qui est utilisée pour la facturation des services utilisés. Ce constat a son importance car il permet d'intégrer un système de facturation externe des services de l'IMS comme le montre notre proposition dans le paragraphe 4.5 de la partie 1 de cette thèse.

L'étude détaillée que nous avons menée dans ce chapitre a permis, entre autres, de bien cerner le rôle de chaque composant du cœur de réseau IMS, les normes de développement et de déploiement de services IMS.

L'expérience que nous avons menée dans ce chapitre concernant le déploiement d'OpenIMSCore comme cœur de réseau IMS et Mobicents comme plateforme de développement et serveur d'application pour IMS ouvre un vaste champ aux développeurs de services à valeur ajoutée dans les réseaux convergents de télécoms.

Les services E-banking, de prise de rendez-vous médicaux, développés dans ce chapitre prouvent qu'on peut implémenter des nouvelles fonctions telles que le SRVCC comme serveur d'application IMS ou encore le développement d'un serveur d'application IP-SM-GW jouant le rôle d'un centre SMS pour les réseaux convergents de télécoms.

# Chapitre 3

## Intégration de WebRTC dans IMS

### 3.1 Introduction

Depuis Juin 2011, l'API WebRTC a été proposé par Google et des développeurs s'y sont intéressés permettant ainsi aux utilisateurs de navigateurs de bénéficier de services riches de communication (textes, audio, video etc.) sans avoir à installer des plugins supplémentaires.

Le réseau IMS, quant à lui, permet aux opérateurs de Télécommunications de fournir presque les mêmes types de services riches de communication que WebRTC. Cependant c'est le réseau IMS qui a été choisi par 3GPP pour comme socle permettant à la 4G LTE/EPC de fournir de la VoLTE.

Donc il est important d'étudier les deux technologies pour voir la possibilité de les intégrer en vue de fournir de services riches de communication aux utilisateurs de terminaux mobiles.

Cette intégration passe par l'analyse fine des composants des deux technologies et certainement aussi par la compatibilité des codecs audio et video utilisés par chacune de ces technologies.

### 3.2 Présentation du WebRTC

#### 3.2.1 Définition

WebRTC (Web Real-Time Communication, littéralement communication web en temps réel) est une technologie permettant les communications directes et en temps réel entre plusieurs navigateurs sans la nécessité d'installer des plugins ou des applications tiers [30]. WebRTC est une interface de programmation (API) JavaScript actuellement au stade de brouillon (Draft) développée au sein du W3C et de l'IETF. Son but est de lier des applications de voix sur IP, le partage de fichiers en pair à pair en s'affranchissant des plugins propriétaires jusqu'alors nécessaires.

Les échanges directs entre navigateurs ne sont pas une nouveauté introduite par le W3C et l'IETF mais les recherches et implémentations précédentes n'étaient pas standards (à cause de l'utilisation de plugins propriétaires et souvent mal documentées tels qu'Adobe Flash ou Microsoft ActiveX). WebRTC est soutenu par Google, Mozilla et Opéra au sein des standards du World Wide Web Consortium (W3C), dont les premières ébauches sont apparues en mai 2011.

## 3.2.2 Étude de protocoles intervenant dans WebRTC

### 3.2.2.1 Fonctionnement de WebSocket

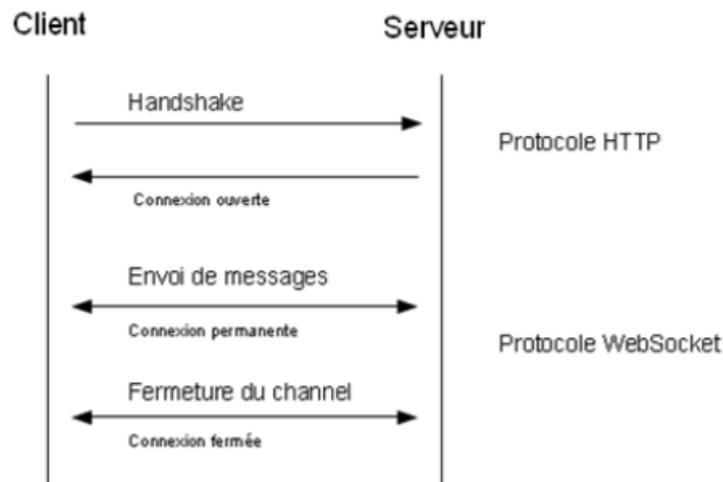


FIGURE I.3.1 – Déroulement d’une session WebSocket

Le protocole WebSocket utilise par défaut les mêmes ports que ceux utilisés par HTTP et HTTPS (80 et 443) et le même schéma pour ce qui est de l’URL : `ws://esp.sn` ou `wss://esp.sn` (avec la surcouverte TLS).

Pour établir une connexion WebSocket, une négociation HTTP spécifique est réalisée entre le client et le serveur. WebSocket utilise HTTP pour la phase d’initialisation de la communication. Ceci dans l’objectif d’assurer une compatibilité avec les infrastructures existantes notamment les proxys et les pare-feu.

Cette phase d’initialisation se nomme le handshake et comprend une requête coté client et une réponse coté serveur : Voir annexes

### 3.2.2.2 Les protocoles de transport

#### RTP/RTCP

Pour transporter la voix ou la vidéo sur IP, le protocole IP (Internet Protocol) au niveau 3 et le protocole UDP (User Datagram Protocol) au niveau 4 sont utilisés.

Mais ces deux protocoles UDP et IP ne suffisent pas à assurer le transport de la voix. En effet, UDP est un protocole sans correction d’erreur, et à aucun moment l’arrivée des paquets dans leur ordre d’émission n’est assurée. Pour le transport de données temps réel telles que la voix ou la vidéo, il est nécessaire d’utiliser deux protocoles supplémentaires : RTP (Real-time Transport Protocol) et RTCP (Real-time Transport Control Protocol)

- **RTP**(Real-Time Transport Protocol) est un protocole de communication informatique permettant le transport de données soumises à des contraintes de temps réel, tels que des flux média audio ou vidéo.
- **RTCP** (Real-time Transport Control Protocol) est un protocole couplé au RTP. C’est un protocole de contrôle des flux RTP, permettant de véhiculer de manière périodique des informations

sur les participants d'une session, les statistiques de bandes passantes et sur la qualité de service. RTP ne transporte aucune information relative au média lui-même.

### Fonctionnement de RTP/RTCP

RTP et RTCP sont des protocoles qui se situent au niveau de l'application et s'appuient sur le protocole de transport UDP. RTP et RTCP peuvent utiliser aussi bien le mode Unicast (point à point) que le mode Multicast (multipoint). RTP et RTCP utilisent des ports différents. RTP utilise un numéro de port pair, et RTCP le numéro de port impair qui suit directement.

Lorsqu'une session RTP est ouverte, alors une session RTCP est aussi ouverte de manière implicite. Les ports RTP et RTCP par défaut sont respectivement 5004 et 5005.

### Les protocoles SRTP/SRTCP

Nous venons de voir appris que RTP/RTCP sont des protocoles de transport de données temps réels ; tandis que RTP gère le transport des données proprement dit (audio vidéo) RTCP gère quant à lui la qualité de service.

Et bien SRTP/SRTCP (Secure RTP/ Secure RTCP) définissent le profil de RTP/RTCP qui a pour but d'apporter le chiffrement, l'authentification et l'intégrité des messages et des données RTP. L'implémentation de ces deux protocoles a pour but de répondre aux contraintes de sécurité posées par les technologies utilisant RTP/RTCP comme la VOIP, WEBRTC...etc.

Pour ce faire les paquets RTP/RTCP sont encapsulés dans le protocole DTLS (Datagramme Transport Layer Protocol).

### Les protocoles TLS (Transport Layer Protocol) / SSL (Secure Socket layer) / DTLS

TLS, tout comme le protocole sur lequel il est basé SSL, sont des protocoles de sécurisation des échanges sur Internet. Le protocole SSL était développé à l'origine par Netscape. L'IETF, en a poursuivi le développement en le rebaptisant Transport Layer Security (TLS). On parle parfois de SSL/TLS pour désigner indifféremment SSL ou TLS.

TLS (ou SSL) fonctionne sur TCP et remplit les fonctions de sécurité suivants : l'authentification du serveur, la confidentialité des données échangées (ou session chiffrée), l'intégrité des données échangées, de manière optionnelle, l'authentification du client (mais dans la réalité celle-ci est souvent assurée par le serveur). Le protocole est très largement utilisé, sa mise en œuvre est facilitée du fait que les protocoles de la couche application, comme HTTP, n'ont pas à être profondément modifiés pour utiliser une connexion sécurisée, mais seulement implémentés au-dessus de SSL/TLS, ce qui pour HTTP a donné le protocole HTTPS.

#### – DTLS (Datagram Transport Layer Protocol)

DTLS est basé sur TLS et est une modification de ce dernier pour fournir les mêmes fonctions de sécurité précédemment citées mais en mode datagramme UDP.

### Le protocole SCTP (Stream Control Transport Protocol)

#### Définition

Les protocoles usuels de transport de l'information dans les réseaux IP sont TCP (*Transmission Control Protocol*) et UDP (*User Datagram Protocol*). Pour répondre aux besoins du transport des

protocoles de signalisation téléphonique sur IP notamment dans le contexte NGN (**N**ext **G**eneration **N**etwork), l'IETF a élaboré un protocole spécifique très fiable, SCTP qui est présent au même niveau que TCP et UDP.

### Fonctionnement de SCTP

SCTP fournit un transport fiable, détecte le rejet, la duplication de données ainsi que les données erronées et retransmet les données corrompues. A ce propos, SCTP gère des temporisateurs plus courts que ceux de TCP car il s'agit de transporter des données de signalisation qui ont des contraintes de temps de livraison plus strictes que celles liées aux données classiques. Alors que dans TCP un flux fait référence à une séquence d'octets, un flux SCTP fait référence à une séquence de messages. SCTP est donc plus simple à interpréter à la réception.

Une des fonctionnalités principales du protocole SCTP est le multi-homing, c'est à dire la capacité pour un endpoint SCTP de supporter plusieurs adresses IP. Ceci est un avantage comparé à TCP. Une connexion TCP est définie par une paire d'adresses de transport (Adresse IP + numéro de port TCP). Chaque endpoint d'une association SCTP fournit à l'autre extrémité une liste d'adresses IP avec un unique numéro de port SCTP.

### Comparaison entre SCTP, TCP et UDP

Les principaux avantages de SCTP par rapport à TCP sont :

- Le support du multihoming qui est directement supporté par SCTP alors qu'avec TCP l'application devra s'appuyer sur plusieurs connexions TCP et gérer le basculement si une connexion TCP est perdue.
- Le support du multi-streaming qui est proposé par SCTP permettant d'émuler plusieurs flots parallèles alors que TCP ne gère qu'un seul flot. Il est possible d'émuler cette fonction au niveau applicatif si TCP est le transport. L'application devra établir plusieurs connexions TCP et répartir la charge sur l'ensemble des connexions.
- La sécurité intrinsèque à SCTP alors que TCP ne fournit aucune sécurité. Avec SCTP il y a authentification au moment de l'établissement de l'association SCTP et par ailleurs chaque paquet transmis par toute entité SCTP émettrice doit contenir une signature qui doit être validée par l'entité SCTP réceptrice.

TABLEAU I.3.1 – Comparaison entre les trois protocoles de transport SCTP, TCP et UDP

Caractéristique	SCTP	TCP	UDP
Transfert de données fiable	Oui	Oui	non
Contrôle de congestion	Oui	Oui	non
Découverte MTU	Oui	Oui	non
Multiplexage de message	Oui	Oui	non
Support du Multi-homing	Oui	Non	non
Support du Multi-streaming	Oui	Non	non
Livraison de données désordonnées	Oui	Non	non
Sécurité	Oui	Non	non
Heartbeat intégré	Oui	Non	non

### 3.2.2.3 Les protocoles STUN, TURN et ICE

#### Le protocole STUN

STUN (« Simple Traversal of UDP through NATs » ou « traversée simple de UDP à travers les NAT ») est un protocole client-serveur permettant à un client UDP situé derrière un routeur NAT (ou de multiples NAT) de découvrir son adresse IP publique ainsi que le type de routeur NAT derrière lequel il est.

En effet, lors de l'établissement d'une session SIP, les clients s'échangent des messages précisant l'adresse IP et le port à utiliser pour transmettre les données. Cependant, un client derrière un routeur NAT transmettra son adresse IP privée. Il ne pourra donc jamais recevoir les données.

#### Fonctionnement de STUN

Le principe du protocole STUN est le suivant :

- Le client envoie une requête au serveur STUN. Dans les en-têtes de cette requête, le routeur a donc substitué une adresse IP publique du réseau local à l'adresse locale de ce client, et généralement a changé aussi le port.
- Dans le corps de sa réponse, le serveur STUN recopie ces informations pour qu'elles parviennent intactes au client : en recevant cette réponse, le routeur n'en modifie à nouveau que les en-têtes, en y remettant l'adresse locale du client et son port utilisé.
- le client initiateur de cette requête reçoit donc en réponse un message dont le contenu lui décrit

la NAT mise en œuvre par son routeur. Ce poste est alors en mesure de faire connaître à quelles adresses IP et sur quels ports de communication on peut lui envoyer des flux RTP et RTCP.

### Le protocole TURN

Ce protocole permet à un serveur STUN de jouer un rôle de proxy. En effet, lorsqu'une NAT dynamique impose des restrictions sur l'adresse et éventuellement le port de destination, il ne sert à rien qu'un serveur STUN informe le poste local du détail de cette NAT mise en œuvre par son routeur.

Car cette NAT n'acceptera comme destination (et donc, en fonctionnant "à l'envers", comme source) que ce serveur STUN, et pas un autre poste devant participer à la communication concernée. La solution est donc que la communication passe par ce serveur.

Pour ce faire, ce serveur doit supporter le protocole TURN, qui est considéré comme une extension du protocole STUN.

### Fonctionnement de TURN

La procédure est la suivante :

- un poste A derrière un routeur mettant en œuvre une telle NAT, et devant communiquer avec un poste B, se connecte donc à un serveur TURN ;
- cette connexion TURN crée une NAT dynamique dans le routeur du réseau du poste A, que le poste A aura pour charge de laisser active, en entretenant cette connexion ;
- ce serveur TURN pourra alors utiliser "à l'envers" cette NAT dynamique, en servant ainsi de proxy pour les flux de communication qui doivent parvenir au poste A ;
- le poste A devra informer le poste B de l'usage de ce proxy : le serveur TURN communique donc au poste A son adresse IP et son port de relais.
- Ces dernières informations sont généralement appelée "adresse relayée". Et bien sûr, il ne s'agit plus de l'adresse IP et du port traduits à la sortie du routeur du poste A, mais de l'adresse IP et du port publics de ce serveur TURN. Dans la pratique, un serveur TURN met aussi en œuvre le protocole STUN (l'inverse n'est pas vrai).

#### 3.2.2.4 Le protocole ICE

Le protocole ICE (Interactive Connectivity Establishment) s'inscrit dans le cadre des protocoles précédemment cités STUN et TURN c'est-à-dire qu'il a été mis en œuvre dans le but de faciliter la communication entre deux clients situés derrière des routeurs NAT ou pare-feu. Il s'agit ici de décrire comment chacun des postes communiquant doit découvrir ses adresses IP et les ports utilisables et en informer l'autre poste, et comment s'effectuent les choix finaux, en cas de possibilités multiples.

### Fonctionnement de ICE

Le principe est le suivant :

**Première étape** : le rassemblement des adresses candidates.

C'est le poste à l'initiative de l'invitation à communiquer qui procède le premier à ce rassemblement :

- ce poste recense ses adresses IP locales, pour chacune il choisit 2 ports (un pour le protocole RTP et un autre pour le protocole RTCP)
- en utilisant les services d'un serveur STUN et éventuellement aussi TURN, le poste A associe à chacune de ces adresses candidates une adresse réflexive, une adresse relayée et des priorités puis l'ensemble est alors encodé selon le protocole SDP, et transmis au poste B,

A la réception de cette invitation, le poste B fait la même chose que dans l'autre sens.

**Deuxième étape** : les tests de connectivité et le choix des connexions.

Chaque poste est alors en possession de ces deux listes d'adresses candidates.

- chaque poste forme une liste de “paires candidates”, en combinant chaque adresse candidate de sa liste avec chaque adresse candidate de l'autre liste.
- chaque poste attribue ensuite à chacune de ces paires une priorité, selon lequel il trie cette liste de paires.

A ce stade, les deux postes sont en possession de la même liste triée de paires candidates.

Puis un des deux postes, qui a le rôle de contrôleur (Ice-Controlling), teste chacune de ces paires par l'envoi d'une requête STUN à l'autre poste. En cas de succès, l'autre poste teste aussi cette connexion. A l'issue de ces tests, les paires nécessaires à la communication envisagée sont choisies par le poste Ice-Controlling.

Le plus souvent, celui des postes qui joue ce rôle d'Ice-Controlling est le poste appelant, qui aura envoyé l'invitation à communiquer.

### 3.2.2.5 Le protocole DIAMETER

Le protocole DIAMETER a été conçu comme une version améliorée du protocole RADIUS. D'ailleurs ces deux protocoles sont des protocoles AAA (Authentication, Authorization, Accounting) et sont utilisés par des procédures d'authentification, d'autorisation et de taxation dans le monde IP.

Plus de 80 applications DIAMETER ont été définies à ce jour pour les télécommunications, notamment par 3GPP pour les architectures EPS (Evolved Packet System) qui représente le réseau 4G, IMS (IP Multimedia Subsystem) qui représente une architecture de service sur IP notamment pour la ToIP, PCC (Policy and Charging Control) permettant le contrôle et la taxation des flux IP du client sur le réseau data mobile, etc...

#### Avantages de Diameter/RADIUS

RADIUS fonctionne sur UDP ce qui ne constitue pas un transport fiable. DIAMETER fonctionne sur TCP ou SCTP permettant de fiabiliser le transport des requêtes/réponses DIAMETER, notamment via des mécanismes de contrôle de flux et de retransmission.

RADIUS est un protocole client-serveur ne permettant pas au serveur d'initier une requête; seul le client peut initier une requête et le serveur retourne une réponse une fois la requête traitée. DIAMETER qui est plutôt un protocole peer to peer permet au client ou au serveur d'initier une requête. Chaque nœud DIAMETER a un rôle dual client et serveur.

**Sécurité au niveau du protocole RADIUS** : Avec RADIUS, le client et le serveur doivent être préconfigurés avec un secret partagé même si la sécurité au niveau IP a été activée. DIAMETER au contraire peut sécuriser la communication entre paires avec des mécanismes standards de sécurité IP tels que IPSec, TLS (Transport Layer Security), ou DTLS (Datagram TLS).

### 3.3 Problématiques du WebRTC et solutions potentielles

#### 3.3.1 Sécurité des applications

Plusieurs problèmes de sécurité se posent lors de l'utilisation de WebRTC :

- JavaScript peut être téléchargé depuis n'importe quel site sans consentement de l'utilisateur.
- Les utilisateurs doivent pouvoir être capables de donner leur agrément à l'utilisation de ressources médias locales comme les caméras et microphones.
- La confidentialité et l'authentification doivent être garanties dans tout échange pour éviter les attaques telles que l'attaque de l'homme du milieu. Les informations privées de l'utilisateur ne doivent pas être dévoilées à des tiers sans le consentement de celui-ci.

Si certains de ces problèmes sont inhérents à toute communication sur l'Internet, d'autres problèmes ont été résolus par l'implémentation de WebRTC. Ainsi les échanges de médias sont sécurisés par le protocole SRTP.

#### 3.3.2 Transiter par des firewalls ou le NAT

WebRTC peut être difficile à utiliser en entreprises dans la mesure où celles-ci ont souvent des politiques de sécurité en informatique incompatibles avec les besoins de l'API.

Les entreprises utilisent de plus en plus des SBC, des pare-feu de niveau application, utilisant un contrôle des flux de signaux et médias. Ces SBC posent des difficultés pour WebRTC dans la mesure où le flux de signaux n'est pas standardisé par l'API et laissé libre de choix à l'implémentation. De plus, les SBC se placent comme intermédiaires dans la transmission du client vers le serveur, mais le protocole DTLS utilisé par WebRTC entraîne le chiffrement des flux de contrôle et de média et ne permet pas leur observation par un tiers.

Si l'un des pairs se situe derrière un NAT, la technique principale utilisée pour gérer ce genre de difficultés est le protocole ICE.

### 3.4 Description générale de la norme WebRTC

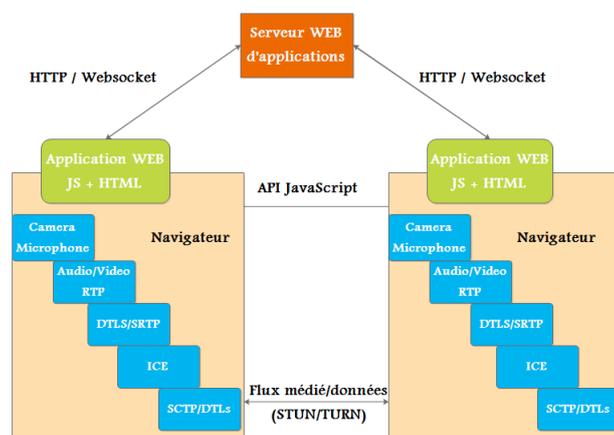


FIGURE I.3.2 – Architecture d'une application WebRTC

L'architecture de l'API WebRTC repose sur une construction triangulaire impliquant un serveur et deux pairs. Les deux navigateurs téléchargent depuis un serveur une application JavaScript vers leur contexte local. Le serveur est utilisé comme point de rendez-vous jusqu'à ce que la connexion directe entre navigateurs soit établie. Le but est d'avoir une application cliente en JavaScript et HTML5 interagissant avec le navigateur au travers de l'API WebRTC.

Les flux échangés peuvent rencontrer divers serveurs qui se chargeront de modifier, traduire ou gérer le signal au besoin, permettant par exemple la traversée de Firewalls, proxys ou NAT.

### 3.4.1 PeerConnection

Afin d'établir une connexion utilisant le standard WebRTC, les navigateurs A et B doivent être connectés simultanément à la page du service et télécharger la page HTML ainsi que le code JavaScript permettant de maintenir la connexion ouverte par HTTPS ou socket. Lorsque le navigateur A souhaite établir la connexion avec B, l'API instancie un objet PeerConnection qui, une fois créé, permet d'établir des flux de médias ou de données. La figure I.3.3 illustre l'établissement d'une connexion entre deux terminaux WebRTC. Il est aussi nécessaire, pour une vidéoconférence par exemple, que les utilisateurs A et B acceptent le partage de leur webcam et/ou de leur microphone.

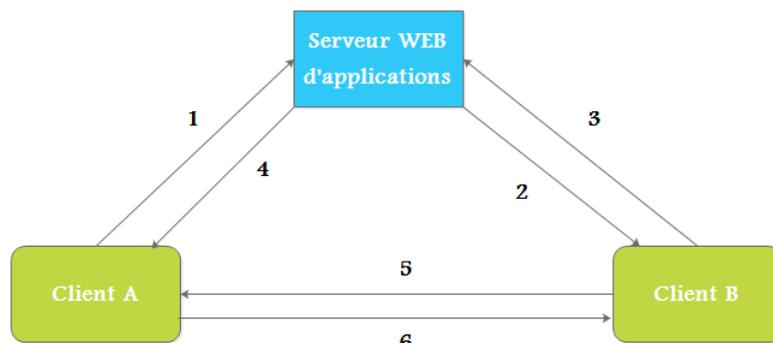


FIGURE I.3.3 – Établissement d'une connexion entre deux clients utilisant WebRTC

1. A demande au serveur une connexion avec B.
2. Le serveur relaie la demande de A.
3. Si B accepte, il envoie une demande de connexion à A.
4. Le serveur relaie la demande à A.
5. Les PeerConnection bidirectionnelles sont établies.

Une fois cet objet PeerConnection créé par A, le navigateur envoie au serveur un paquet contenant les informations sur les médias partagés. Le serveur va décoder ce paquet et envoyer une notification à B. Si la requête est acceptée, le même processus a lieu entre B et A cette fois afin d'établir la connexion bidirectionnelle. Une fois celle-ci établie, les flux de médias ou de données peuvent être ajoutés à la connexion librement.

Cette étape est permise par l'API via Session Description Protocol. Néanmoins, l'utilisation de SDP est en cours de remplacement au sein de la norme WebRTC par le protocole JSEP. Ce processus

utilisant SDP permet la négociation à la fois pour RTP (transport de médias) et pour SCTP (transport de données).

Afin d'assurer la continuité de la connexion en cas de translation d'adresse par NAT et éviter le blocage par les pare-feu (notamment en entreprise), l'objet `PeerConnection` utilise les protocoles STUN et ICE.

### 3.4.2 DataChannels

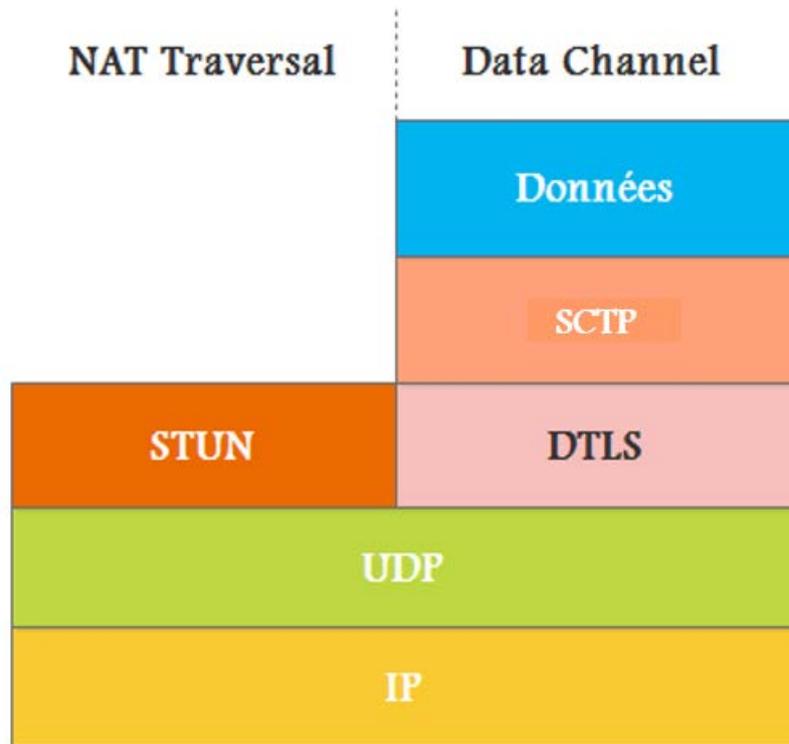


FIGURE I.3.4 – Structure de la pile de protocoles utilisée par WebRTC dans un échange de données

L'API *DataChannels* offre un moyen d'échange bidirectionnel et pair à pair de données telles que des images ou du texte.

Ces canaux de données sont créés entre pairs en utilisant l'objet *PeerConnection*. Ces données autres que les flux médias sont échangées via le protocole SCTP, lui-même encapsulé dans **DTLS**. Cette solution permet au flux de données d'être intégré dans le même paquet que les flux de médias et donc de partager le même numéro de port pour les échanges.

SCTP supporte nativement plusieurs flux de données de façon bidirectionnel au sein d'une association SCTP et gère les priorités. De cette façon, il est possible de favoriser les messages de haute priorité face aux gros objets à la priorité basse. Chaque flux représente une connexion logique unidirectionnelle.

Afin d'assurer la confidentialité et l'authenticité des paquets SCTP échangés, chaque flux repose sur le protocole DTLS.

Au sein d'un canal de données, les applications peuvent transmettre des messages de façon ordonnée ou désordonnée. L'ordre de remise est préservé uniquement dans le cas d'une transmission de paquets ordonnés envoyés sur le même lien de données.

Un flux de données est créé lorsque l'un des pairs appelle une méthode *CreateDataChannel()* pour la première fois après avoir créé un objet *PeerConnection*.

Chaque appel suivant à *CreateDataChannel()* créera un nouveau flux de données au sein de la connexion SCTP existante.

Le protocole DTLS n'a pas pour seul rôle d'encapsuler les paquets SCTP. Dans le cadre d'un multiplexage avec des flux médias, le protocole DTLS encapsule la gestion des clés et la négociation des paramètres pour le protocole SRTP utilisé pour la gestion des flux médias. Il y a donc dépendance du flux média vis-à-vis du flux de données.

### 3.4.3 MediaStream

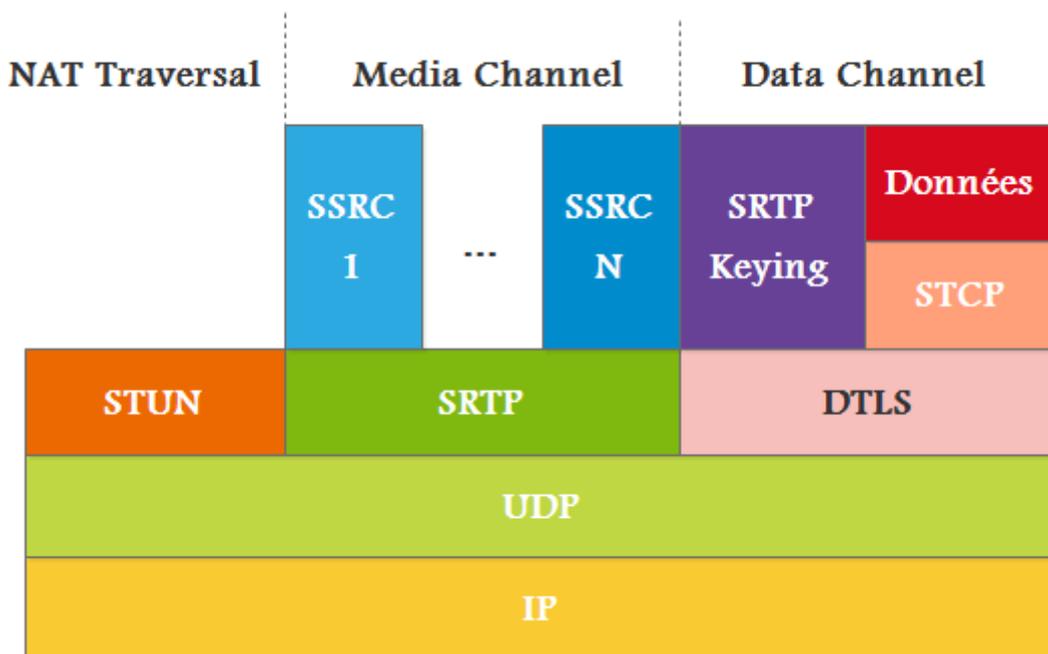


FIGURE I.3.5 – Structure de la pile de protocoles utilisée par WebRTC dans un échange de médias

Un *MediaStream* est une représentation d'un flux de données audio ou vidéo. Il permet la prise en charge des actions sur le flux média telles que l'affichage, l'enregistrement et l'envoi à un pair distant. Un *MediaStream* peut être local ou distant.

Un *MediaStream* local doit demander l'accès aux ressources multimédia de l'utilisateur via la fonction *getUserMedia()*. L'application spécifie le type de média (audio ou vidéo) auquel elle souhaite accéder et le navigateur autorise ou refuse l'accès à la ressource demandée. Une fois que le média n'est plus utilisé, l'application peut révoquer son propre accès avec la méthode *stop()*.

Les flux médias sont transportés par le biais du protocole RTP. La confidentialité, authentification des messages et la protection contre les répétitions est apportée par l'utilisation sécurisée de RTP, SRTP. La gestion des clés pour SRTP est assurée par DTLS et donc le flux de données. Il est donc impossible d'avoir un flux média indépendant d'un flux de données là où l'inverse est envisageable.

Il est possible d'associer plusieurs flux médias sur une même connexion SRTP qui utiliseront des ressources médias différentes ou non. Dans ce cas, les sources de chaque flux sont clairement identifiées

comme des SSRC.

### 3.4.4 Multiplexage media / data

L'API WebRTC prévoit le multiplexage de flux données ou média reposant sur une seule connexion de niveau transport. Ce multiplexage fait que les trois protocoles STUN, SRTP et DTLS coexistent au même niveau du modèle et qu'il est nécessaire de démultiplexer les paquets arrivant. Pour cela, le premier octet indiquant la nature du contenu UDP sert à déterminer de quel protocole il s'agit. Une valeur de 0 ou 1 indique un paquet STUN, une valeur entre 20 et 63 indique un paquet DTLS une valeur de 128 à 191 indique un paquet SRTP.

L'intérêt principal de ce multiplexage est qu'en partageant un même paquet de niveau transport, les flux de médias et de données passent plus facilement les NAT ou pare-feu.

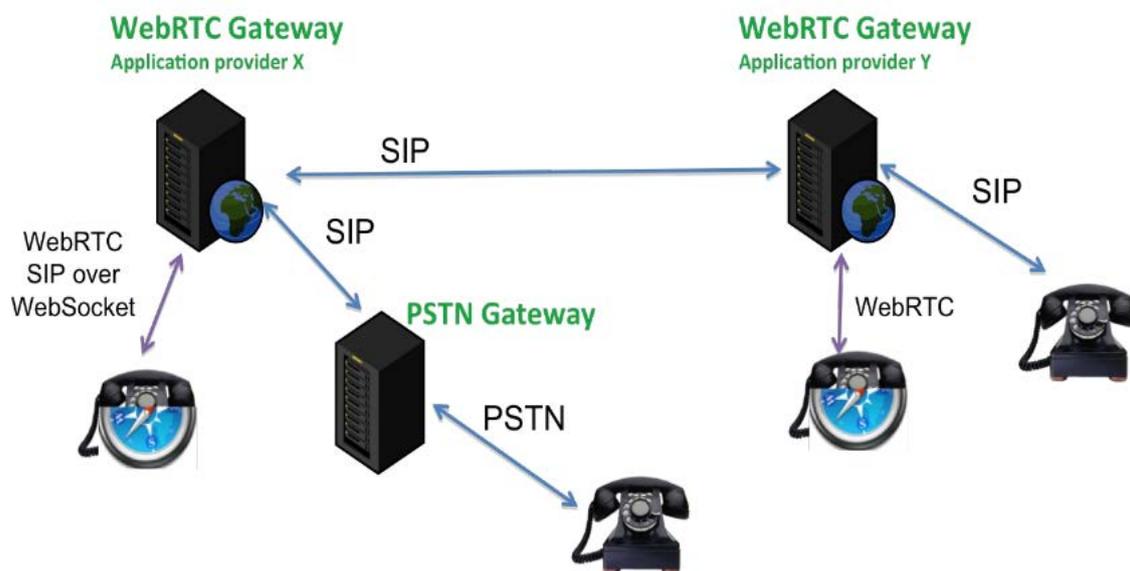


FIGURE I.3.6 – Schéma d'une communication entre clients SIP et clients WebRTC

La passerelle WebRTC permet de faire le lien entre la technologie WebRTC et les technologies de VoIP tel que SIP. En plus d'interconnecter différentes applications (clientes) WebRTC, la passerelle WebRTC permet aussi la communication entre des clients WebRTC (navigateurs) et des clients SIP (téléphone IP ou softphone).

Lorsqu'un client émet un appel, la passerelle WebRTC doit décider si l'appelé est joignable à travers le WebRTC. Selon la réponse la passerelle WebRTC devra ou non traduire l'appel en session SIP, traduction qui se fait en plusieurs étapes.

### 3.4.5 Signalisation

La signalisation pour les applications WebRTC n'a pas encore été normalisée. Toutefois, SIP over WebSocket (RFC 7118) est souvent utilisé à cause de l'adaptabilité de SIP dans la plupart des scénarios de communication mais aussi la disponibilité de logiciels open source comme JsSIP ou SIPML5. Selon le cas de figure, la passerelle devra encapsuler les paquets SIP venant de la couche WebSocket dans les protocoles UDP, TCP ou TLS.

### 3.4.6 Transport de Medias

La normalisation du WebRTC spécifie pour le transport des médias l'utilisation du protocole SRTP pour des raisons de sécurité. SRTP est supporté par la plupart des applications de VoIP même si ce n'est pas toujours le cas. Si l'appelé ne supporte pas SRTP, alors la passerelle devra faire le **mapping** entre RTP et SRTP.

### 3.4.7 Translation d'adresses

Pour le WebRTC, les protocoles de traversé de Nat normalisés sont STUN et ICE. Ces mêmes technologies sont implémentées pour les clients SIP. Dans le cas contraire, la passerelle devra se comporter comme un serveur ICE et router les paquets media entre l'appelant et l'appelé.

## 3.5 Les applications WebRTC

Elles sont nombreuses, des applications collaboratives dans les navigateurs aux applications télécoms qui peuvent utiliser le navigateur maintenant comme un terminal comme le téléphone IP, nous pouvons citer entre autres :

- Un système de Web phone
- Une intégration dans les réseaux NGN et IMS.
- Une application Click to call comme un HelpDesk ou un callcenter par exemple pour permettre au client de discuter directement avec un conseiller pour une assistance à travers le Web et toujours sans avoir à installer de plugins.
- Une application de Multi-conférence
- Les réseaux sociaux
- Dans le divertissement dans le navigateur : chat, jeux en ligne ...
- Les applications Web TV

## 3.6 Les codecs

### 3.6.1 Codecs WebRTC

Les appels vidéo et voix sont des fonctionnalités essentielles du WebRTC (Web Realtime communications)[31] tel que défini par l'IETF et le W3C. Cependant, le choix de l'IETF portant sur les codecs audio obligatoires à implémenter (MTI : Mandatory To Implement, MTI Codecs) introduit une notion obligatoire de transcodage lorsqu'il s'agira d'inter opérer avec les terminaux IMS. En plus, l'IETF n'a pas été capable de sélectionner des codecs vidéo obligatoires à implémenter (MTI video codecs).

Plusieurs fournisseurs de services ont lancé divers services basés sur IMS tels que Voice-over-LTE (IR.92) [32] et Video-over-LTE (IR.94) [29]. Il serait souhaitable que ces services existants soient capables d'inter opérer avec les services fournis par le WebRTC. Par exemple, une communication audio-vidéo bidirectionnelle entre des mobiles IMS conformes aux normes 3GPP et des navigateurs supportant le WebRTC.

De nos jours, tous les mobiles, les réseaux mobiles d'opérateur et les systèmes d'exploitation des mobiles supportent le codec voix AMR (et le AMR-WB si, la large bande « voix » est supporté) et le codec vidéo H.264. Si les navigateurs web WebRTC supportent uniquement d'autres codecs, une

entité réseau aura besoin de faire du transcodage entre les différents codecs pour que la communication réussisse.

Les retards (délais) entre l'émission du son à partir de la bouche de l'émetteur jusqu'à la perception de ce son par l'oreille du destinataire (les latences) produisent un impact sur la communication entre humains. Le transcodage augmente de manière assez significative ces retards (délais), ce qui dégrade la qualité de la voix. Le transcodage fait accroître aussi la complexité et les coûts des passerelles « médias » réseau (network media gateway). Se dispenser de ce transcodage devient un idéal, ainsi, sélectionner des codecs communs via une négociation en temps-réel fiable entre les parties en communication serait plus bénéfique.

Pour l'audio et la voix à travers WebRTC, l'IETF a préconisé les codecs MTI : OPUS et G711[33] Pour la vidéo, les candidats sont VP8 [34]et H.264 [35] [36], mais l'IETF n'a pas encore sélectionné des codecs vidéo obligatoires à implémenter.

Par ailleurs, d'aucuns préconisent le remplacement de VP8 par VP9 [34] et H.264 par H.265 qui sont plus performants.

### 3.6.2 Codecs IMS

La GSMA a défini les services IMS suivants utilisant des codecs video/audio communs :

- IR.92 IMS Profile for Voice and SMS [29]
- IR.94 IMS Profile for Conversational Video Service [33]
- IR.39 IMS Profile for High Definition Video Conference (HDVC) Service [37]
- RCC.07 Rich Communication Suite 5.2 Advanced Communications Services and Client Specification [38] for RCS voice and video calling services (Services de communication avancés et spécifications « client » pour les services d'appel RCS voix et vidéo)

Les codecs MTI pour les services susmentionnés sont le AMR (et AMR-WB si la communication voix large bande est supportée) pour la voix et le H.264 pour la vidéo.

Le fait qu'il y ait des différences de codecs audio MTI entre WebRTC et IMS rend obligatoire le transcodage où un codec commun ne peut être négocié entre les terminaux. En effet, les codecs audio OPUS et G.711 proposés [33] sont insuffisants.

Les prochaines instructions sur les codecs à être considérés comme appropriés pour l'interopérabilité entre les clients WebRTC et les services de communication basés sur IMS sont décrit dans un autre brouillon (draft-internet) proposant d'autres codecs pouvant être supportés par WebRTC [39].

### 3.6.3 Interopérabilité WebRTC-IMS

Il faut noter deux éléments essentiels pour l'interopérabilité entre les terminaux IMS et les terminaux WebRTC. Il doit y avoir de l'interopérabilité tant au niveau de la signalisation qu'au niveau du temps-réel audio/vidéo. Pour la signalisation, une passerelle dans le réseau doit faire le lien (map) entre des domaines IMS et WebRTC non-inter opérables pour, par exemple, « **la conversion de protocole de signalisation** ». Ceci est un travail en cours au sein du 3GPP [40].

L'architecture actuelle pour l'accès de WebRTC à IMS est illustrée dans le schéma ci-dessous. Une description plus détaillée des entités de cette architecture est disponible dans 3GPP TS 23.228 [9]. Minimiser le coût de conversion est l'idéal mais, le plus grand impact à l'expérience utilisateur (end-user experience) est attendu dans l'interopérabilité multimédia « temps-réel ». L'un des plus importants

facteurs affectant la qualité dans les communications en temps-réel est la latence aller-retour, et ceci a largement été étudié par l'IUT, 3GPP, ETSI et d'autres groupes. Toutes ces études ont abouti à une même conclusion, consistant à minimiser le délai en gardant entre certains nœuds, un délai donné afin de délivrer une meilleure expérience à l'utilisateur.

La figure I.3.7 ci-dessous montre l'architecture et le modèle de référence d'intégration IMS-WebRTC.

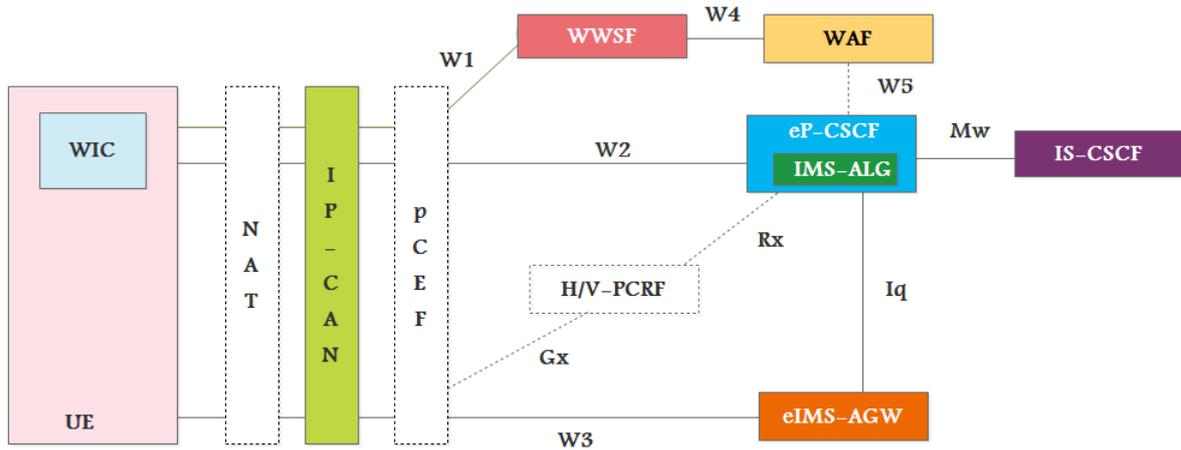


FIGURE I.3.7 – Architecture et modèle de référence WebRTC-IMS [9] [10]

### 3.7 Transcodage

Lorsque deux terminaux impliqués dans une communication sont incapables d'établir un codec commun pour chaque type de média à travers un canal de signalisation, il y a un fort besoin de conversion d'un codec supporté par un terminal vers un codec supporté par l'autre terminal concerné par la communication. Une telle conversion appelée transcodage est généralement faite par le réseau. Le transcodage peut être fourni par diverses fonctions réseau et peut être une opération du réseau d'origine, du réseau de destination ou de terminaison, ou du réseau de transit selon la configuration de chaque opérateur et des accords bilatéraux entre les opérateurs. De plus, le transcodage peut se faire de manière dynamique : par transcodage, les ressources sont insérées dans la session lorsqu'un appelant envoie une requête d'établissement de session contenant une offre de description de session envers l'appelé, dans le cas où le réseau est informé du fait qu'une incompatibilité de codec se produira due au manque de support du codec (proposé par l'appelant) par l'appelé.

Alternativement, le transcodage de ressources peut être introduit durant l'établissement de la session se basant sur le contenu de la réponse à l'offre de description de session retournée par l'appelé, dans le cas d'une incompatibilité de codecs. Un tel transcodage peut avoir lieu dans le cœur du réseau grâce au **Media Control functions**, ou alternativement, à l'extrémité du réseau par la fonctionnalité **Border Control Functions**.

## 3.8 Proposition d'une architecture de test

Pour tester l'intégration de WebRTC dans IMS, nous avons mis en place une plateforme de test ayant les composants suivants :

- Une passerelle WebRTC appelé WebRTC Gateway à base d'Asterisk ;
- Un serveur SIP appelé RNIS Gateway doté d'une carte E1 RNIS permettant de simuler l'accès au réseau fixe de téléphonie ;
- Un équipement permettant d'émuler un réseau RNIS appelé Simulateur RNIS ;
- Un serveur SBC configuré à base d'Asterisk ;
- OpenIMSCore permettant de déployer un cœur de réseau IMS ;
- Des terminaux constitués de PC, de tablettes dotés de navigateur Firefox et Google Chrome.

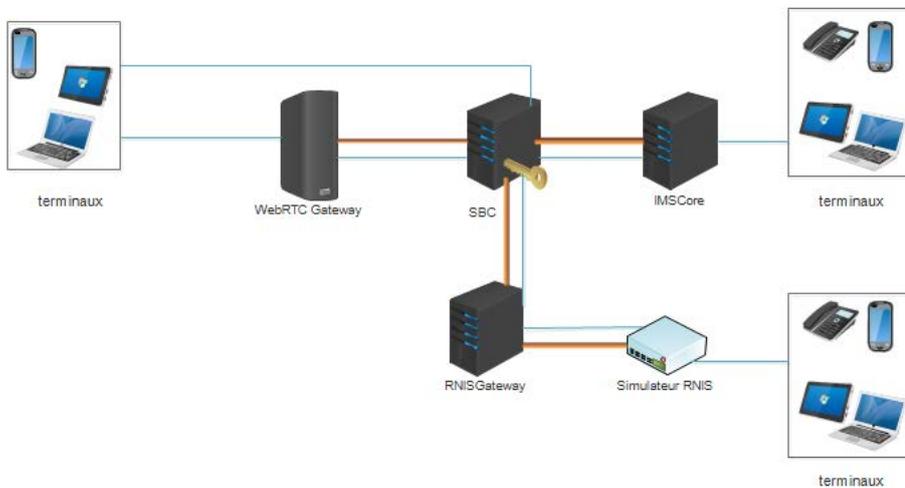


FIGURE I.3.8 – Architecture de test

Il est à noter que le plugin Firefox du codec vidéo OpenH264 fourni par CISCO a été activé sur le navigateur des terminaux.

### 3.8.1 Mise en place de WRTC-GW (WebRTC Gateway)

Nous décrivons ci-après la procédure de la mise en place du WebRTC Gateway de notre plateforme. Pour cela nous allons utiliser une source d'Asterisk version 11 que nous avons compilés avec le support WebSocket et SRTP.

Pour que les flux soient transportés par SRTP sur notre future passerelle WebRTC, il a fallu générer des certificats à utiliser.

#### Activation du protocole WebSocket

On précise les paramètres (adresse, port, ...) de Asterisk pour la signalisation WebSocket.

```

; http://<server>
;
[general]
enabled=yes
bindaddr=0.0.0.0
bindport=8088
;
; Whether HTTP/HTTPS is
; This also affects ma
;
;enabled=yes
"http.conf" 90L. 3159C
/etc/asterisk/http.conf
    
```

FIGURE I.3.9 – Activation du Websocket sous Asterisk

La figure I.3.10 ci-après montre l'activation de la signalisation WebSocket sur notre serveur Asterisk.

```

[general]
context=public
udpbindaddr=0.0.0.0:5060
realm=192.168.1.111
transport=udp,ws
avpf=yes
encryption=yes
; Default
; allowguest=no ; AT
/etc/asterisk/sip.conf
    
```

FIGURE I.3.10 – Activation de la signalisation WebSocket sous Asterisk

### Configuration des clients WebRTC (Sipml5)

La figure I.3.11 ci-après montre la configuration du client WebRTC.

Expert settings

Disable Video:

Enable RTCWeb Breaker:

WebSocket Server URL: ws://192.168.1.111:8088ws

SIP outbound Proxy URL: udp://192.168.1.111:5060

ICE Servers: e.g. [{ urt: 'stun:stun.l.google.com:19302'}, { urt: ...

Max bandwidth (kbps): { audio:64, video:512 }

Video size: { minWidth:640, minHeight:480, maxWidth:640, maxHeight:480 }

Disable 3GPP Early IMS:

Disable debug messages:

Cache the media stream:

Disable Call button options:

Save Revert

FIGURE I.3.11 – Configuration d'un client Sipml5

### 3.8.2 Tests et résultats

Plusieurs scénarios ont été testés et les résultats sont présentés ci-après.

#### Scénario 1 : Appel d'un utilisateur WebRTC vers un utilisateur IMS

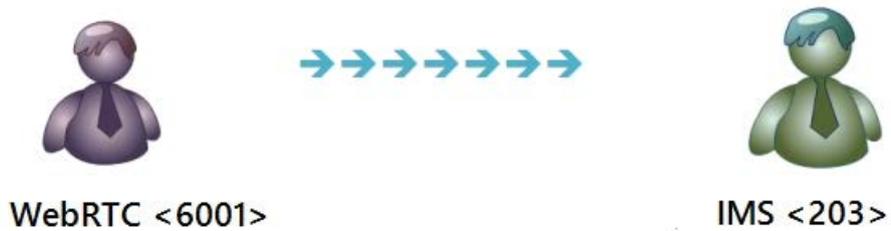


FIGURE I.3.12 – Appel WebRTC - IMS

La figure I.3.13 montre la mise en communication réussie d'un utilisateur WebRTC et d'un utilisateur IMS

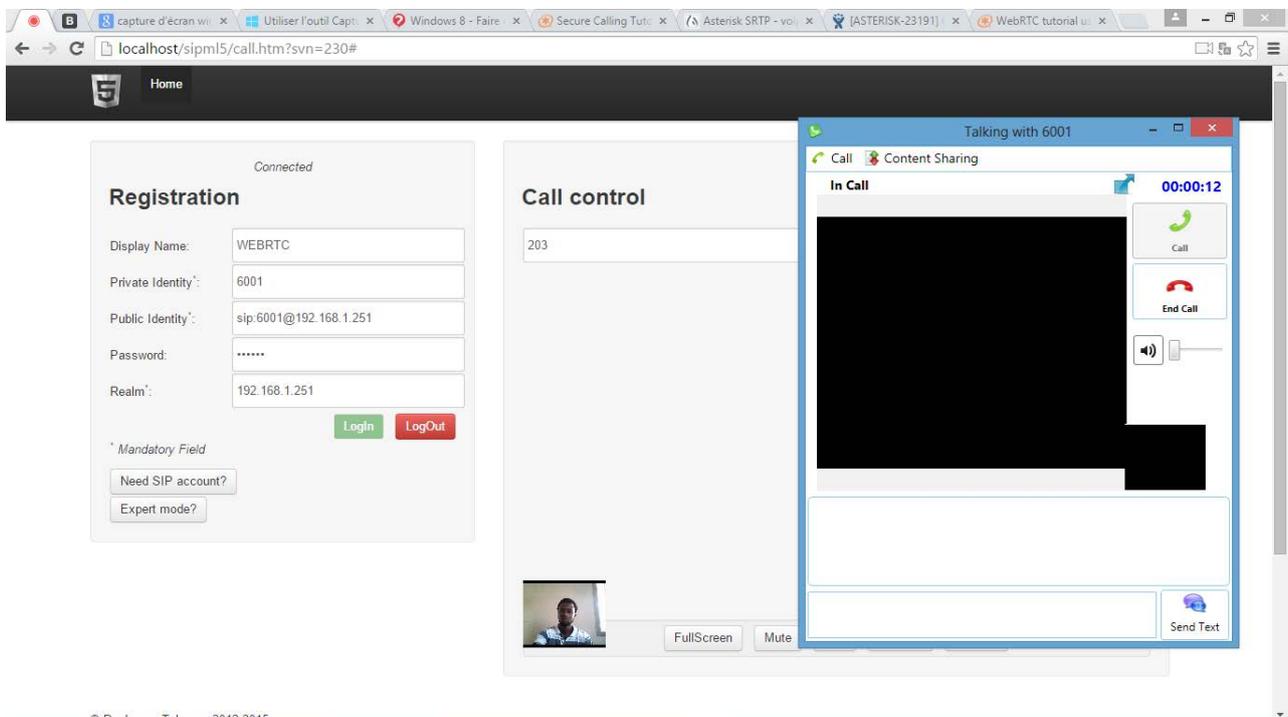


FIGURE I.3.13 – Communication entre un utilisateur WebRTC et un utilisateur IMS

L'interface d'Asterisk montre effectivement la prise en compte des types de flux vidéo entre les utilisateurs.

```

== Using SIP VIDEO CoS mark 6
== Using SIP RTP CoS mark 5
-- Executing [203@from-internal:1] Dial("SIP/6001-00000002", "sip/openims/alice,30,tr") in new stack
== Using SIP VIDEO CoS mark 6
== Using SIP RTP CoS mark 5
-- Called sip/openims/alice
-- SIP/openims-00000003 is ringing
> 0x9273538 -- Probation passed - setting RTP source address to 192.168.1.154:1506
-- SIP/openims-00000003 answered SIP/6001-00000002
-- Channel SIP/6001-00000002 joined 'simple_bridge' basic-bridge <d1a1730f-c1cc-4821-b5c6-80828e420bcf>
-- Channel SIP/openims-00000003 joined 'simple_bridge' basic-bridge <d1a1730f-c1cc-4821-b5c6-80828e420bcf>
> 0x9273538 -- Probation passed - setting RTP source address to 192.168.1.154:1506
-- Channel SIP/openims-00000003 left 'simple_bridge' basic-bridge <d1a1730f-c1cc-4821-b5c6-80828e420bcf>
-- Channel SIP/6001-00000002 left 'simple_bridge' basic-bridge <d1a1730f-c1cc-4821-b5c6-80828e420bcf>
== Spawn extension (from-internal, 203, 1) exited non-zero on 'SIP/6001-00000002'
iallo-Server*CLI> _

```

FIGURE I.3.14 – Interface d’Asterisk montrant la mise en communication WebRTC - IMS

### Scénario 2 : Appel d’un utilisateur IMS vers un utilisateur WebRTC



FIGURE I.3.15 – Appel IMS → WebRTC

La figure I.3.16 montre la mise en communication réussie d’un utilisateur IMS et d’un utilisateur WebRTC.

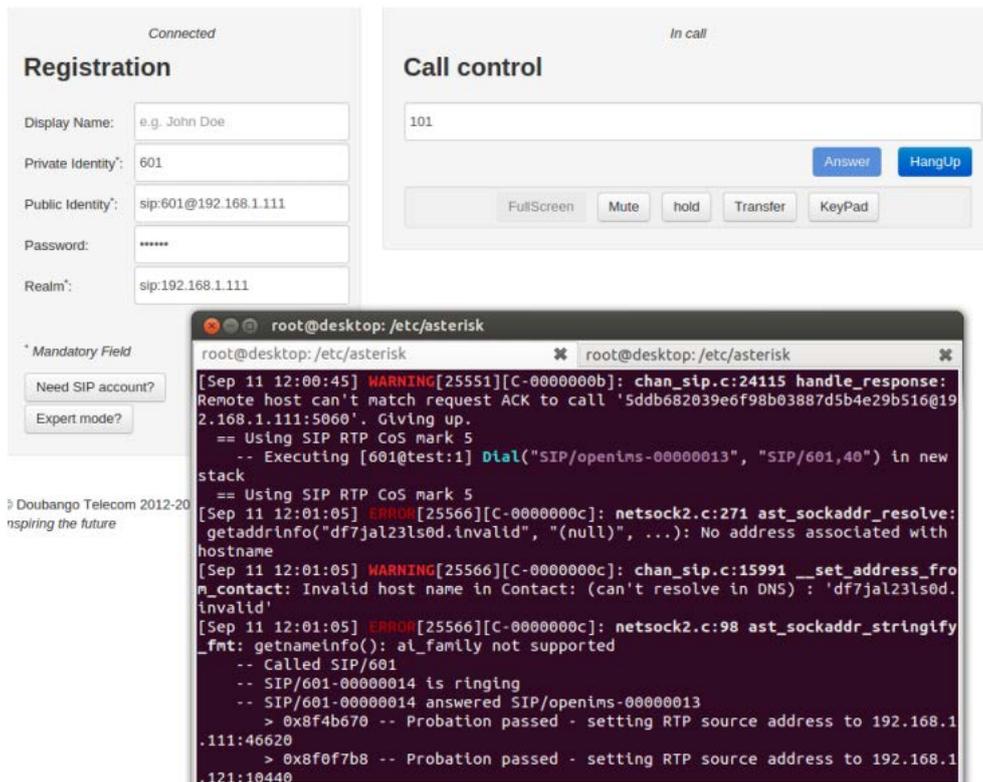


FIGURE I.3.16 – La mise en communication réussie d’un utilisateur IMS et d’un utilisateur WebRTC

**Scénario 3 : Appel d’un utilisateur WebRTC vers un utilisateur fixe**

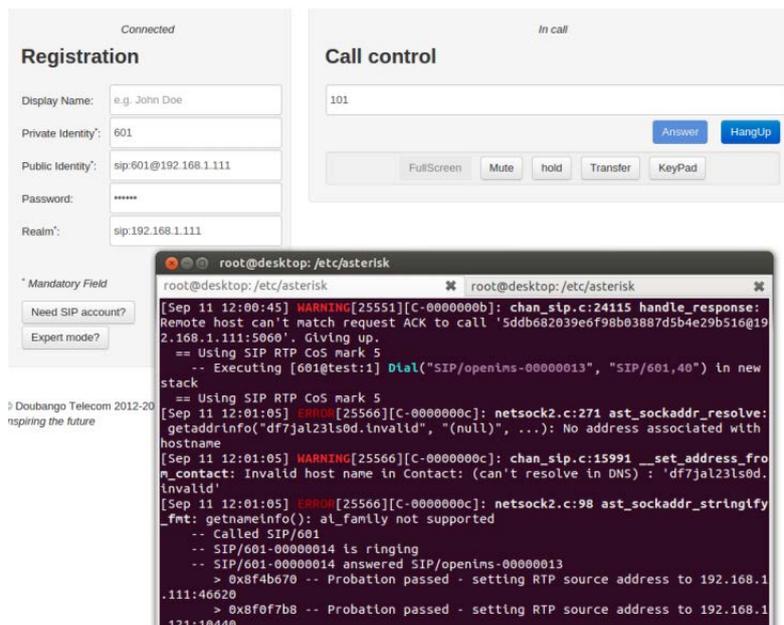


FIGURE I.3.17 – Appel WebRTC → Réseau fixe

Nous avons activé sur le terminal WebRTC Sipml5 et le terminal RNIS le codec audio G711. La

figure I.3.18 montre la mise en communication réussie d'un utilisateur WebRTC et d'un utilisateur d'un réseau fixe RNIS.

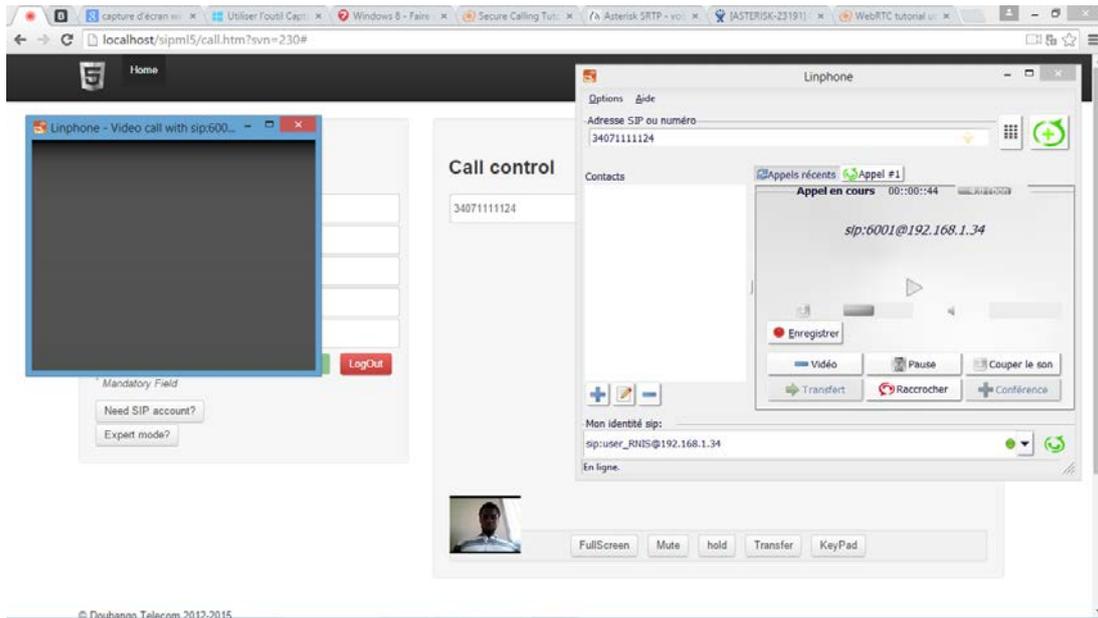


FIGURE I.3.18 – Mise en communication réussie entre un utilisateur WebRTC et un utilisateur fixe

L'interface du SBC montre effectivement la prise en compte des types de flux vidéo entre les utilisateurs. On voit que le SBC « AST\_SBC » redirige l'appel vers la passerelle « Ast\_RNIS ».

```

== WebSocket connection from '192.168.1.154:56277' for protocol 'sip' accepted
using version '13'
-- Registered SIP '6001' at 192.168.1.154:56277
> Saved useragent "IM-client/OMA1.0 sipML5-v1.2015.03.18" for peer 6001
== Using SIP VIDE0 CoS mark 6
== Using SIP RTP CoS mark 5
-- Executing [3407111124@from-internal:1] Dial("SIP/6001-00000000", "sip/A
RNIS/3407111124,30,tr") in new stack
== Using SIP VIDE0 CoS mark 6
== Using SIP RTP CoS mark 5
-- Called sip/Ast_RNIS/3407111124
-- SIP/Ast_RNIS-00000001 is ringing
-- SIP/Ast_RNIS-00000001 is ringing
-- SIP/Ast_RNIS-00000001 answered SIP/6001-00000000
-- Channel SIP/6001-00000000 joined 'simple_bridge' basic-bridge <0eed98b1-
82-4352-848e-8053bb60af82>
-- Channel SIP/Ast_RNIS-00000001 joined 'simple_bridge' basic-bridge <0eed9
1-8982-4352-848e-8053bb60af82>
> 0xb7505a48 -- Probation passed - setting RTP source address to 192.168
.34:16328
-- Channel SIP/Ast_RNIS-00000001 left 'simple_bridge' basic-bridge <0eed98b
8982-4352-848e-8053bb60af82>
-- Channel SIP/6001-00000000 left 'simple_bridge' basic-bridge <0eed98b1-89
-4352-848e-8053bb60af82>
== Spawn extension (from-internal, 3407111124, 1) exited non-zero on 'SIP/60
-00000000'
iallo-Server*CLI>
    
```

FIGURE I.3.19 – Interface du SBC montrant la mise en communication WebRTC - Fixe

## 3.9 Conclusion

Les expériences que nous avons effectuées dans ce chapitre montrent que les technologies WebRTC et IMS offrant des services riches de communications gagneraient à s'intégrer pour offrir aux utilisateurs des opérateurs de télécoms la souplesse quant au choix des terminaux mobiles à partir desquels émettre ou recevoir des communications. Des blocages dus au refus des développeurs WebRTC d'intégrer le codec H264 largement supporté par les terminaux IMS sous prétexte que sa licence n'est pas libre ont été levés grâce au projet open source openH264 de Cisco qui a permis d'implémenter une version libre du codec H264. Ainsi n'importe quel concepteur de navigateur à la possibilité aujourd'hui d'implémenter la prise en charge de WebRTC et le support du codec H264 dans son navigateur. C'est le cas des dernières versions du navigateur firefox.

3GPP a proposé que les proxies de l'IMS soient modifiés pour qu'ils intègrent le support de WebRTC évitant ainsi l'utilisation d'une passerelle externe WebRTC-IMS. Mais une fois encore, les expériences que nous avons effectués en utilisant les plateformes OpenIMScore, Asterisk avec le module WebRTC et l'API SIPML5 montrent la faisabilité de cette proposition. Ainsi à partir des navigateurs, des utilisateurs des opérateurs de télécoms pourraient bénéficier de presque tous les services offerts par IMS et en même temps tirer profit des services riches WebRTC tels que le partage d'écran de leurs terminaux mobiles etc..

# Chapitre 4

## Proposition d'un modèle technologique de réseaux de télécoms dans un contexte E-learning

### 4.1 Introduction

Les trois premiers chapitres de ce travail ont permis de cerner les différentes technologies et les pré-requis pour un déploiement d'un réseau convergent de télécoms. Il ressort de cette étude que les conditions pour un déploiement d'un réseau haut débit de télécoms sont réunies :

- la disponibilité d'un maillage en fibre optique des zones à couvrir
- la disponibilité des bandes de fréquences allouées par l'UIT pour la LTE

Le passage à la TNT (Télévision Numérique Terrestre) a permis de libérer des fréquences radio qui peuvent être utilisées pour déployer la LTE.

C'est pourquoi nous proposons un modèle technologique de déploiement d'un réseau haut débit pouvant favoriser l'émergence des pays en voie de développement notamment les pays africains.

La mise en place d'une université numérique dans les pays en voie de développement nécessite une couverture réseau qui n'est pas toujours de bonne qualité.

Certains pays africains ont tendance à vendre leurs ressources brutes. Ces ressources sont transformées ailleurs, dans les pays du Nord, et revendues aux africains.

En cette période de rareté des moyens financiers, certains pays africains misent sur la vente des fréquences découlant du passage à la TNT pour avoir les moyens d'investir entre autres dans les domaines de la santé et de l'éducation.

Si on prend le cas du Sénégal, l'État à travers l'ADIE dispose d'un réseau de fibre optique, d'un réseau WiMAX et d'un réseau CDMA présents sur une bonne partie du territoire national.

Nous proposons qu'une partie des fréquences libérées grâce au passage à la TNT soit affectée à l'ADIE pour une extension de son réseau ce qui pourrait favoriser l'aménagement numérique du territoire en haut débit.

Ce haut débit permettrait de créer une interactivité forte entre les chaînes de télévision et les téléspectateurs. Une telle télévision interactive connectée peut être utilisée par l'enseignement supérieur ou la santé.

## 4.2 Télévision Numérique Terrestre (TNT)

La TNT (Télévision Numérique Terrestre) est faite, comme son nom l'indique, pour être reçue par voie terrestre, c'est-à-dire grâce aux émetteurs disséminés sur un territoire. Elle s'inscrit dans le cadre de la télévision du futur et va remplacer à terme la télévision habituelle telle que nous la connaissons.

L'objectif étant de fournir une image de bien meilleure qualité. La TNT est transmise sur ondes radio à travers l'espace terrestre de la même façon que la télévision analogique, la principale différence étant l'utilisation d'émetteurs multiplex permettant la transmission de plusieurs programmes sur le même canal. Le dividende numérique [41] correspond à la bande passante dégagée suite au passage à la Télévision Numérique Terrestre.

### 4.2.1 Dividende numérique

L'IUT a alloué au Sénégal dans un premier temps la bande de 790 à 862 Mhz comme premier dividende numérique et ensuite lui a attribué dans un deuxième temps la bande 694 à 790 Mhz. La bande de 470 à 694 Mhz a été réservée à la TNT.

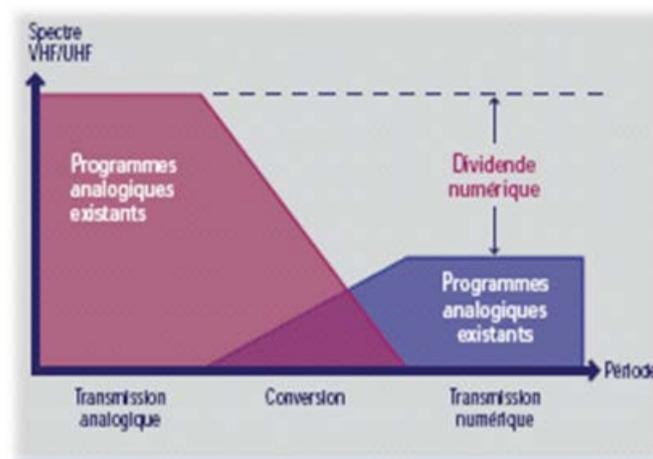


FIGURE I.4.1 – Dividende numérique

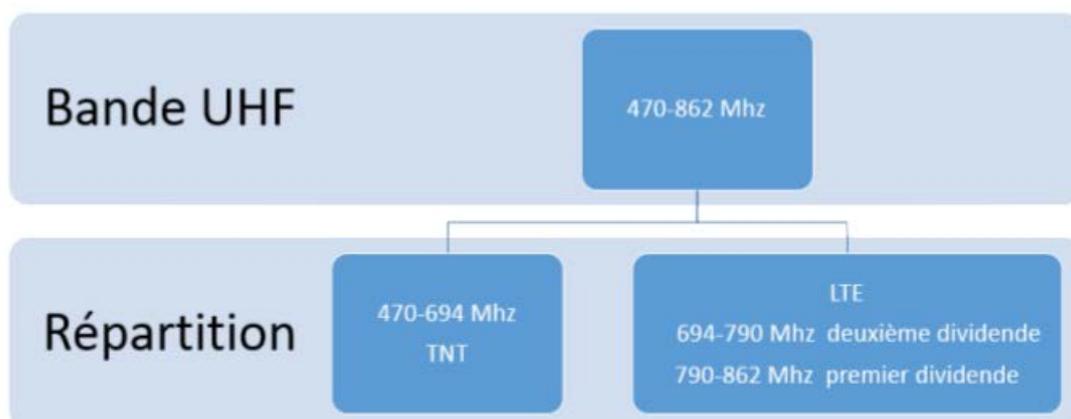


FIGURE I.4.2 – Répartition des fréquences issues du dividende numérique au Sénégal

## 4.2.2 Problème de connectivité dans les zones rurales

Malgré les différentes solutions apportées par les technologies de télécommunications telles que les réseaux 3G, 3G +, et WiMax, il n'existe pas une solution idéale pour résoudre l'enclavement des populations en zone rurale. Cela entraîne un problème quant à l'accès à l'internet dans ces zones, sachant que le développement de ces localités passera nécessairement par le développement technologique.

Pour permettre l'accès à internet dans les zones rurales et renforcer la capacité d'accès à la large bande dans les zones saturées, la technologie Tv White Space se positionne comme la plus intéressante des solutions par l'utilisation des fréquences basses qui possède des caractéristiques très séduisantes.

## 4.3 Principe de la technologie TV White SPACE (TVWS)

Le TVWS a été défini par la norme 802.22 en 2011 par l'IETF. Le principe de cette technologie est de pouvoir utiliser les ondes des fréquences VHF-UHF pour transmettre des données. Or nous savons que chaque antenne a pour particularité de capter un ou plusieurs types de signaux et de travailler dans une bande de fréquences données, C'est pourquoi l'on trouve aussi bien des antennes UHF, des antennes VHF que dans antennes mixtes VHF-UHF pour les ondes TV.

La plupart des appareils des utilisateurs finaux (ordinateurs portables, tablettes, téléphones) ne sont pas faits pour se connecter aux points d'accès opérant dans la bande de fonctionnement du Tv white space. Quelle méthode va donc utiliser la technologie Tv WhiteSpace pour remédier à cela ? Le réseau de Tv WhiteSpace commence par une station de base qui est reliée à un accès fibre ou satellitaire à partir d'un fournisseur de service internet. Cette station de base peut envoyer des signaux aux alentours de 10 km au moins. Des équipements dit appareils "à deux têtes" qui agissent comme les deux émetteurs/récepteurs de données via les Tv white space et les émetteur/récepteurs Wifi constituent les stations clientes. Ces appareils prennent le signal sur les fréquences du white space et les distribuent sur du wifi ou du WIMAX c'est dire sur le 2.4 GHz ou le 5 GHz. Préalablement, la station cliente devra s'authentifier et ensuite envoyer sa position géographique et après se voir être attribuer une fréquence qui n'est pas utilisée dans sa zone. Des constructeurs comme Calson et Motorola sont ceux dont les technologies nous ont inspirées.

### 4.3.1 Les points forts de TVWS

Parmi les points forts de TVWS, on peut noter :

- Abondance en milieu rural : Plus de canaux vacants dans les zones rurales ;
- Les feuilles, les aiguilles et les gouttes de pluie sont beaucoup plus petites par rapport aux longueurs d'onde du TVWS ;
- Faible perte en espace libre compte tenu de la basse fréquence utilisée
- Fréquence idéale pour un débit performances en visibilité indirecte
- Voyage à travers les montagnes
- S'appuie sur l'infrastructure existante

### 4.3.2 Architecture d'un réseau de données avec les TV white space

Plusieurs équipementiers travaillent aujourd'hui sur cette technologie pour proposer une architecture et des équipements adéquats. Parmi ces constructeurs, nous avons retenu principalement deux :

Carlson et Motorola, qui présentent une architecture presque similaire basée sur l'attribution dynamique du spectre fréquentiel d'utilisation à une station cliente. Lorsqu'un client s'authentifie et envoie ses coordonnées GPS, il se voit attribuer une fréquence de travail. La station principale est donc le chef d'orchestre car dialogue avec des bases d'information pour mettre à disposition du client des données correctes dans le temps. Ainsi, il attribuera une fréquence non utilisée au client et mettra la base à jour pour ne plus la réallouer dans cette zone et donc éviter les interférences.

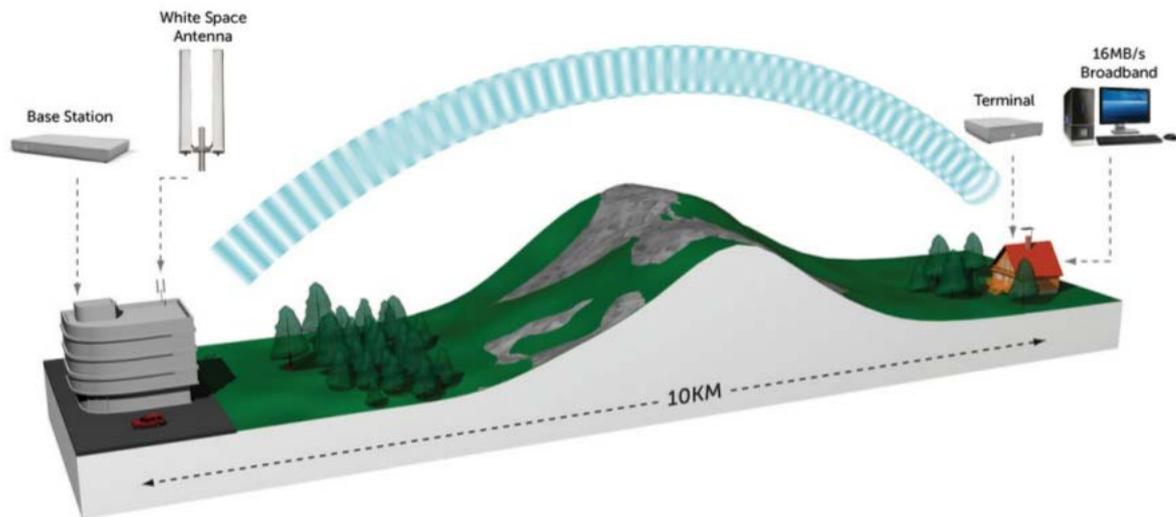


FIGURE I.4.3 – Architecture d'un réseau de données avec TVWS [11]

### 4.3.3 Les différents éléments du réseau

#### Les bases de données

Elles sont des éléments très importants du cœur de ce réseau. Elles recensent toutes les fréquences libres pouvant être utilisées par la technologie TVWS dans la localité de couverture et permette aussi de gérer l'authentification des stations clientes. Ces informations sont généralement :

- La bande de fréquence,
- Les coordonnées GPS de l'émetteur qui les utilisent,
- La portée sur laquelle elle est rayonnée.

Cela permettra d'éviter non seulement les interférences entre les émetteurs de la TVWS, mais également entre les émetteurs de TVWS et les émetteurs des stations de télévision qui ne seront pas encore en numérique. Cela a pour avantage également de permettre une configuration rapide et exacte de nouveaux clients. La mobilité dans la zone de couverture d'un tel réseau sera aussi gérée grâce à ces bases de données.

#### La station de base

La station de base dans un réseau de données avec TVWS est le point de raccordement avec le réseau externe via un fournisseur d'accès. Elle sera connectée à une ou plusieurs stations clientes et relayera les requêtes de ces dernières vers internet. Elles sont connectées aux bases de données pour la gestion de l'authentification et de l'attribution dynamique des fréquences.

Elle est composée d'un ensemble émetteur /récepteur relié au fournisseur d'accès via la fibre ou le satellite et d'autre part aux stations clientes via les ondes du « white space ». Elle dispose d'antenne

omnidirectionnelle ou sectorielle pouvant émettre sur des distances très importantes. Le débit offert peut aller jusqu'à 25Mbps. Dans le cas du Sénégal, on pourrait utiliser les pylônes des émetteurs TNT comme support des stations de base TVWS.

### 4.3.4 Quelques cas d'utilisation de TVWS en Afrique

#### Afrique du sud

Google a annoncé un partenariat avec l'Autorité indépendante des communications d'Afrique du Sud (ICASA) et wireless Carlson pour fournir un accès sans fil à 10 écoles à travers 3 stations de base sur le campus de la Faculté de l'Université de Stellenbosch, de médecine et sciences de la santé dans Tygerberg, Cape Town.

#### Kenya

Un projet pilote par Microsoft et le gouvernement kenyan offrant des vitesses de bande passante allant jusqu'à 16 Mbps à trois communautés rurales qui manquent d'électricité : Homme, Gakawa et Laikipia.

#### Ghana

Il est devenu le premier pays en Afrique à commercialiser les TV White Spaces. L'offre, qui est une joint-venture entre Spectra Wireless et Microsoft 4 Afrika, vise à permettre aux étudiants du pays d'accéder à des forfaits Internet à haut débit à des prix abordables. Le projet TV White Space a été officiellement lancé au Ghana en Février 2013 par Microsoft pour permettre aux personnes vivant en zones rurale ou reculées d'avoir accès à l'Internet à des prix abordables.

TVWS était déjà utilisé bien avant le passage à la TNT et cohabitait avec la télévision analogique. Grâce à la TNT, les fréquences UHF libérées non utilisées vont être dynamiquement utilisées par la TVWS pour offrir un réseau haut-débit. La capacité de TVWS à détecter dynamiquement les fréquences non utilisées fait que son déploiement ne va pas gêner les services qui utilisent de manière primaire les fréquences UHF.

## 4.4 Proposition d'une architecture de télévision interactive

### 4.4.1 Présentation des normes DVB et HBBTV

#### 4.4.1.1 L'adoption de la norme DVB pour la TNT au Sénégal

Depuis le 17 Juin 2005, de nombreux pays Africains tels que le Sénégal, ont commencé progressivement à basculer leurs émissions télévisuelles hertziennes, vers la norme DVB-T. DVB (Digital Video Broadcasting – Terrestrial)[42] est un ensemble de standards pour la transmission de la télévision numérique. Tout comme pour les précédentes émissions analogiques, il permet l'émission de flux audio et vidéo. La principale nouveauté est la possibilité d'intégrer des flux de données à part entière. En effet, DVB est une extension du flux de transport (TS) MPEG-2 contenant plusieurs flux audio et vidéo ainsi que des flux de données. Ces flux peuvent être acheminés sur différents supports, tels que le satellite, l'aérien ou le câble, respectivement DVB-S , DVB-T et DVB-C.

#### 4.4.1.2 Étude de la pile DVB

Les constituants de la pile DVB sont séparés en deux types de couches : les couches hautes et les couches basses.

##### Les couches hautes

Les couches hautes d'une émission télévisuelle suivant la norme DVB sont indépendantes du support de transmission utilisé. Les mécanismes de sécurité s'appliquant à ces couches sont donc valables, quelle que soit la forme de DVB utilisée (DVB-S, DVB-C ou DVB-T).

##### La couche Application

La couche applicative des émissions DVB représente les différents protocoles utilisés par les différents flux élémentaires d'une émission hertzienne. On y retrouve les flux Video et Audio au format MPEG-2 ainsi que des flux de données. Si des mécanismes de sécurité pour MPEG existent, ils sont généralement orientés vers la protection des droits d'auteur (DRM), mais ne permettent en aucun cas au téléspectateur de s'assurer de l'intégrité ou de l'authenticité du flux.

##### La couche Transport

La couche transport des émissions DVB représente le multiplexage des différents flux élémentaires dans un multiplex MPEG-TS. Afin de permettre la diffusion de chaînes payantes, DVB utilise le Common Scrambling Algorithm (CSA). Ce mécanisme permet, à l'aide d'un système de chiffrement symétrique, d'offrir un système de contrôle d'accès aux opérateurs TV pour le contenu payant. La clef de chiffrement est transmise à l'intérieur du flux avec le contenu chiffré, protégé par un système de chiffrement asymétrique. Le téléspectateur possède une carte à puce, capable de déchiffrer la clef de chiffrement, afin de déchiffrer ensuite le contenu télévisuel. Ce mécanisme est uniquement destiné à un contrôle d'accès mais ne permet pas par exemple d'authentifier l'émetteur DVB, ni l'origine des données reçues.

##### Les couches basses

Les couches basses d'une émission télévisuelle suivant la norme DVB dépendent du support de transmission utilisé. Cependant, les trois principaux supports de communication utilisés ont un point commun qui peut être validé sur chaque support avec un équipement analogue.

#### 4.4.1.3 Principe de fonctionnement de la norme DVB

Cette partie porte sur un nouveau type d'utilisation du spectre issu du dividende numérique à destination des terminaux mobiles : le réseau hybride. Un tel réseau peut être constitué d'un réseau TVWS et d'un émetteur DVB-T2 en zone rurale ou d'un réseau 4G et d'un émetteur DVB-T2 en zone urbaine [43]. Les aires de couvertures de ces deux composantes sont confondues et un service peut être transmis indifféremment via l'une ou l'autre des composantes. Ce concept permet de pallier à un point faible du réseau IP c'est à dire la capacité à transmettre efficacement les mêmes données à un grand nombre d'utilisateurs simultanément. Ainsi, le choix de transmettre un service sur l'une ou l'autre des composantes repose principalement sur le nombre d'utilisateurs qui accèdent à ce service. Typiquement, l'utilisation dans le cadre de l'enseignement à distance s'avère intéressante dans la mesure où un cours peut être diffusé en direct en utilisant un émetteur DVB-T2 et les étudiants

suivant ce cours peuvent être autorisés par l'enseignant à poser des questions par le biais du réseau IP. Les différentes interventions des étudiants sont transmises au multiplex DVB-T2 par IP qui la diffuse à l'ensemble des étudiants.

#### 4.4.1.4 Le protocole HBBTV

Dans cette section, nous nous intéressons à la conception d'une interactivité entre les chaînes de télévisions connectées et leurs téléspectateurs sous certaines conditions. Depuis le basculement vers les standards DVB, les flux télévisuels contiennent de nombreux flux de données. Ces flux permettent, par exemple, de transmettre le résumé du programme en cours. Avec l'arrivée des téléviseurs connectés, différents projets de contenu interactif ont vu le jour. Après plusieurs projets indépendants, la norme HBBTV, Hybrid Broadcast Broadband TV, est née en 2010.

#### Présentation de la norme HBBTV

La norme HBBTV [44][45] est destinée à harmoniser les flux télévisuels et Internet. Ce protocole permet aux chaînes de rajouter du contenu interactif à leur programme TV. L'affichage de ce contenu repose sur l'interprétation d'une page HTML sur le téléviseur du téléspectateur. L'affichage du contenu initial de HBBTV ne nécessite aucune action de la part du téléspectateur. Cette page est affichée en superposition de l'image de la chaîne regardée, et est souvent constituée d'un petit encadré incitant le téléspectateur à accéder à la suite du contenu, généralement en appuyant sur le bouton rouge de sa télécommande. Le protocole HBBTV définit 2 méthodes afin de fournir son contenu au téléviseur :

- Soit en spécifiant l'URL de la page à afficher ; cette solution ne marche que si le téléviseur est connecté à Internet et elle permet aussi une gestion plus aisée des mises à jour de la page à afficher.
- Soit en transmettant le contenu de la page à afficher ; cette solution a l'avantage de fonctionner même si le téléviseur n'a pas d'accès à Internet.

## Architecture du protocole HBBTV

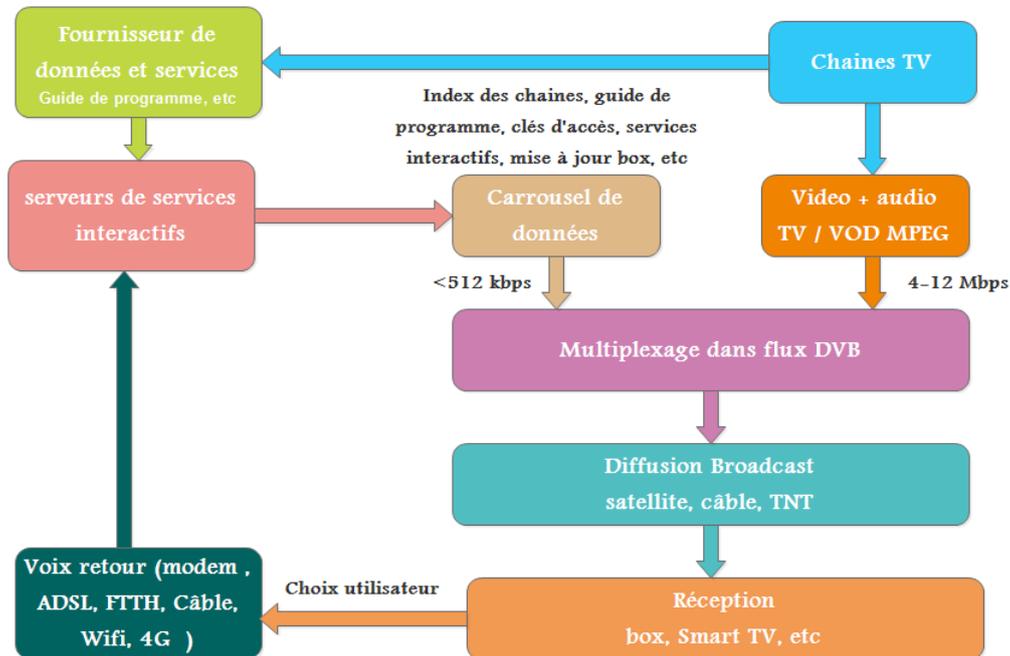


FIGURE I.4.4 – Architecture du protocole HBBTV

### 4.4.2 Télévision interactive

#### 4.4.2.1 Cas d'utilisation 1

Nous proposons d'intégrer une plateforme de communication WebRTC comme décrit dans le chapitre 4 de la partie 2 et désigné par TONUS\_INTERACTIF\_SERVER dans la figure I.4.5. Cette plateforme permet à un enseignant de donner l'autorisation à un étudiant connecté de faire des interventions audiovisuelles. On suppose que l'étudiant dispose d'une clé USB TNT connectée à son ordinateur lui permettant de recevoir les images de son enseignant. Grâce à la connexion Internet fournie par TVWS, l'étudiant pourra se connecter à notre plateforme pour éventuellement demander à prendre la parole. Un étudiant désirant poser des questions, envoie un message appelé TONUS\_NEED\_REQUEST au serveur qui le transmet à l'ordinateur de l'enseignant. L'enseignant peut accepter ou refuser de donner la parole à l'étudiant. Dans le cas d'acceptation, un message de type TONUS\_REQUEST\_ACCEPT est transmis au serveur qui envoie à son tour un message de type TONUS\_AUTHORISED à l'étudiant. Ce dernier peut alors envoyer le flux vidéo de sa webcam matérialisé par TONUS\_MEDIA\_STREAM au multiplex TNT qui le transmet à l'émetteur TNT. En ce moment-là, tous les autres étudiants et l'enseignant pourront voir et écouter l'étudiant poser sa question. Le réseau TVWS peut être remplacé par n'importe quel réseau IP (Wifi, 3G, 4G, etc.), ce qui montre la souplesse de notre solution.

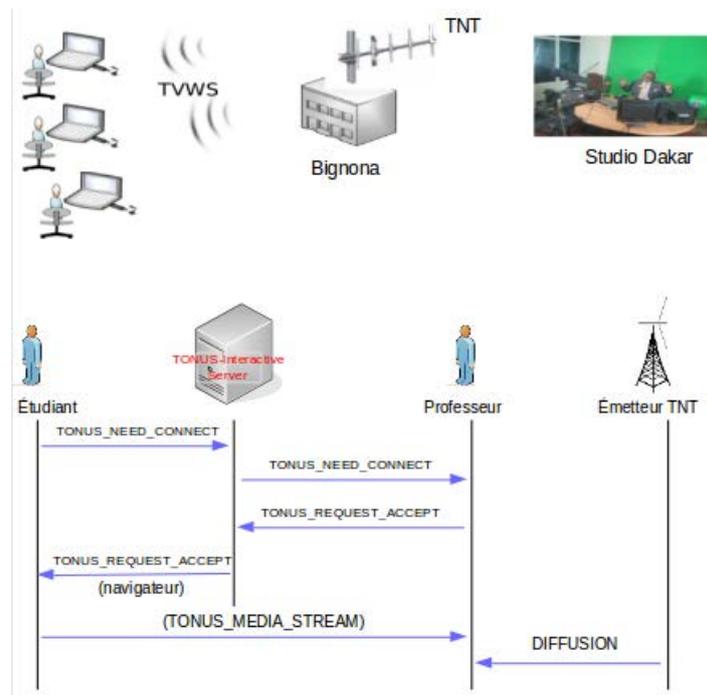


FIGURE I.4.5 – Interactivité enseignant et étudiants

#### 4.4.2.2 Cas d'utilisation 2

Le cas d'utilisation 2 explique comment étudiants et enseignant peuvent collaborer dans un projet de développement logiciel en utilisant la plateforme de collaboration que nous avons conçue dans le chapitre 4 de la partie 2.

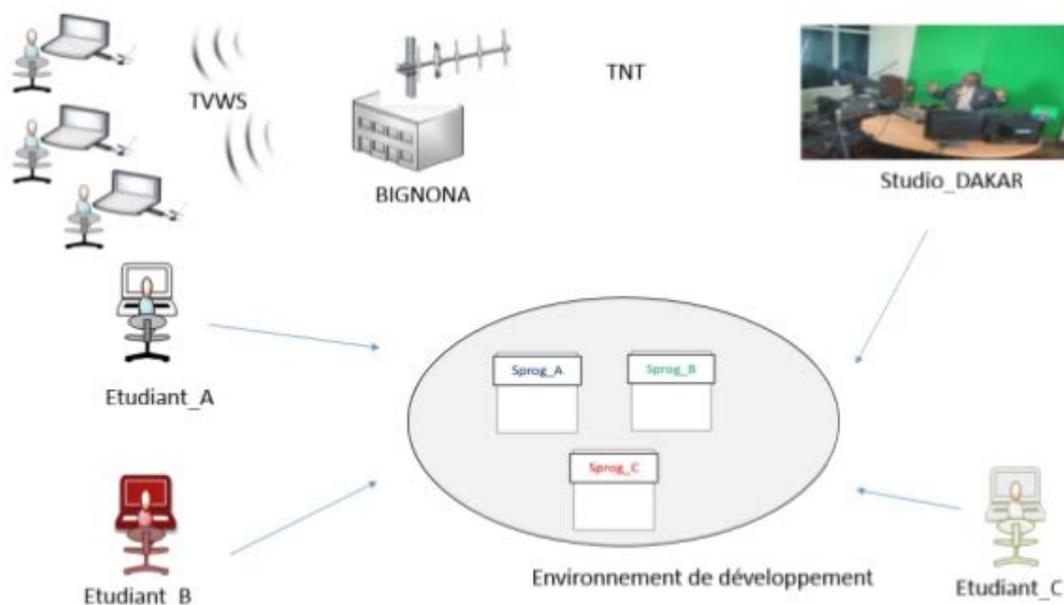


FIGURE I.4.6 – Interactivité enseignant et étudiants développeurs

### 4.4.2.3 Faisabilité du modèle proposé

Les travaux effectués par Yann BACHY dans sa thèse [12], montrent qu'il est possible d'intégrer des flux HBBTV dans les programmes des télévisions. Nous décrivons sa plateforme qui utilise le matériel Ettus N210 et sa carte fille WBX permettant de réaliser de la radio logicielle ou Software Defined Radio (SDR), le logiciel GNU-Radio pour paramétrer le matériel Ettus N210 pour n'importe quel type de modulation radio, le logiciel Opencaster pour générer un flux de transport DVB.



FIGURE I.4.7 – Ettus N210

La figure I.4.8 montre la plateforme de Yann BACHY qui lui a permis d'émettre dans la même bande fréquence qu'une chaîne de télévision TNT. Dans son laboratoire, un téléviseur a pu capter les flux DVB.

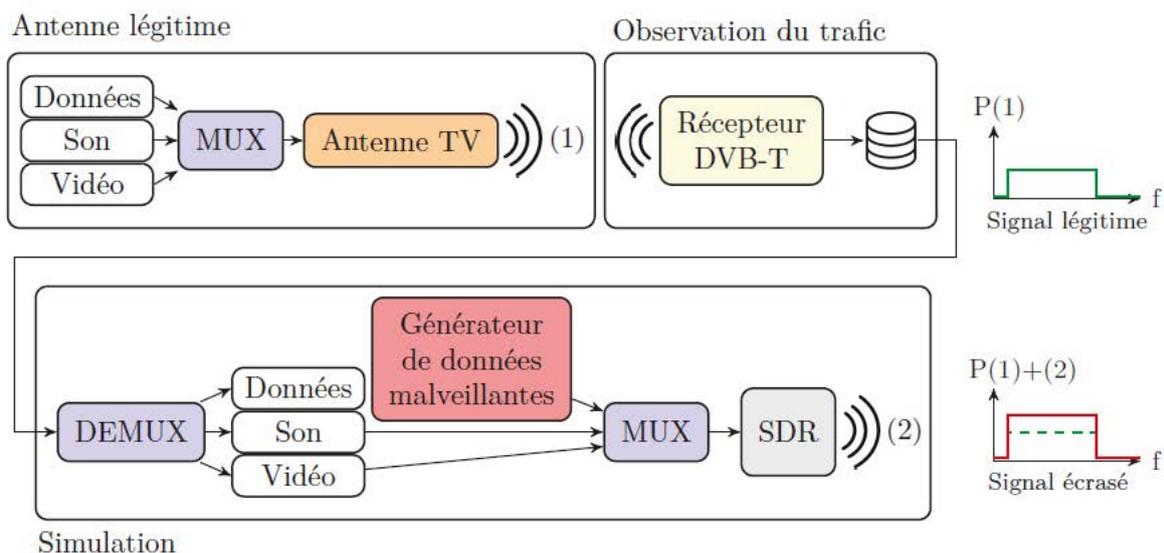


FIGURE I.4.8 – Plateforme d'expérimentations DVB de Yann BACHY

HBBTV permet aux chaînes de rajouter du contenu interactif à leurs programmes TV.

L'affichage de ce contenu repose sur l'interprétation d'une page HTML sur le téléviseur du téléspectateur, comme illustré dans la figure I.4.9 . L'affichage du contenu initial de HbbTV ne nécessite aucune action de la part du téléspectateur. Cette page est affichée en superposition de l'image de la chaîne regardée, et est souvent constituée d'un petit encadré incitant le téléspectateur à accéder à la suite du contenu, généralement en appuyant sur le bouton rouge de sa télécommande.

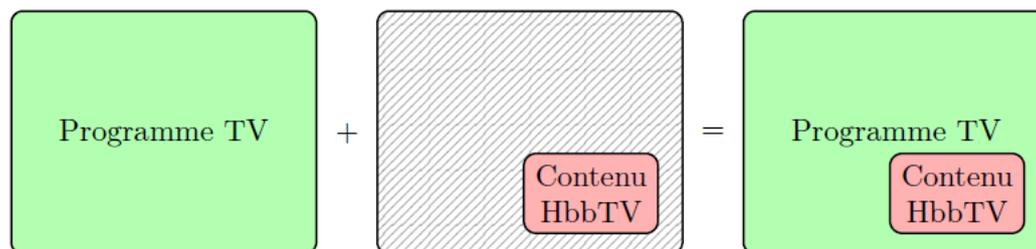


FIGURE I.4.9 – Superposition HbbTV [12]

Le protocole HbbTV définit 2 méthodes afin de fournir son contenu au téléviseur :

- En spécifiant l'URL de la page à afficher ; cette solution a pour désavantage de fonctionner uniquement si le téléviseur est connecté à Internet, mais elle permet une gestion plus aisée des mises à jour de la page à afficher.
- En transmettant le contenu de la page à afficher ; cette solution a l'avantage de fonctionner même si le téléviseur n'a pas d'accès à Internet.

Ces résultats montrent que notre proposition est viable. On va diffuser l'url de notre serveur `Tonus_Interactive_Ser` à l'intérieur d'un contenu HbbTV. Ainsi un étudiant qui désire intervenir pourra utiliser l'url contenu dans le flux HbbTV pour demander l'autorisation auprès de l'enseignant.

## 4.5 Proposition de solution de SMS sur IP et 4G

On se rend compte aujourd'hui que beaucoup d'universités se mettent en réseau (cas du SnRER au Sénégal, WACREN en Afrique du centre et de l'ouest) et peuvent communiquer directement dans un réseau IP. Nous proposons dans ce contexte un centre SMS permettant aux utilisateurs d'envoyer et recevoir des SMS tout en bénéficiant des services à valeur ajoutée. Nous montrerons comment ce centre SMS pourra communiquer avec les SMSC des opérateurs de télécommunications 2G, 3G et 4G.

### 4.5.1 Conception de la solution

La solution que nous proposons utilise et complète le protocole SIMPLE proposé par l'IETF [46]. SIMPLE, Session initiation protocol for Instant Messaging and Presence Leveraging Extensions, est un protocole de messagerie instantanée et de présence basé sur le protocole SIP. Nous rappelons qu'une architecture de téléphonie sur IP basée sur IP est composée des entités suivantes : serveurs d'enregistrement, serveurs de localisation, serveurs de redirection, serveurs proxy [4]. Mais dans les faits, toutes ces entités sont regroupées en une seule que nous appellerons par la suite proxy SIP et nous supposerons qu'il intègre un gestionnaire d'événements.

#### 4.5.1.1 Service de messagerie point-to-point : Cas du protocole SIMPLE

Le protocole SIMPLE permet l'envoi de messages entre clients SIP. Il est basé sur la présence et n'autorise que les échanges en mode connecté. Un message envoyé à un utilisateur déconnecté n'est pas reçu à sa reconnexion.

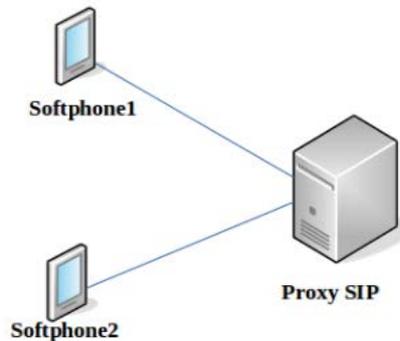


FIGURE I.4.10 – Cas du protocole SIMPLE [13]

#### 4.5.1.2 SMSC-IP

Nous proposons un dispositif permettant à des terminaux d'envoyer et de recevoir des messages même en cas de non connexion du destinataire. Nous appellerons ce dispositif dans la suite SMSC IP.

##### Architecture de la solution

L'architecture proposée décrit la position du SMSC-IP dans un système de VoIP mais aussi prévoit la fourniture de services à valeur ajoutée et l'intégration du SMSC-IP avec les autres opérateurs de télécommunications 2G/3G par l'intermédiaire du protocole SMPP

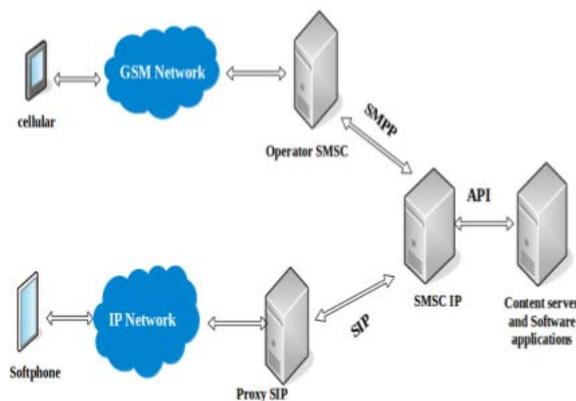


FIGURE I.4.11 – Architecture du SMSC-IP

##### Fonctionnalités du SMC-IP

L'application faisant office du SMSC se doit de se connecter sur socket TCP au gestionnaire des événements intégré au proxy SIP pour recueillir les informations suivantes : état utilisateur (connecté, déconnecté), message non envoyé. Avec ces informations, on pourra garantir les fonctions suivantes au SMSC-IP.

- Gestion de la présence

- Stockage des SMS à destination des utilisateurs non rattachés au réseau pour une retransmission en cas de reconnexion
- Envoi de SMS de diffusion à un groupe ou à l'ensemble des utilisateurs
- Fourniture de services à valeur ajoutée
- Passerelle entre l'environnement IP et les environnements 2G/3G/4G

Nous nous appuyons sur le gestionnaire des événements du serveur proxy SIP pour récupérer et stocker le message d'un utilisateur déconnecté sur un serveur de système de gestion de base de données clé-valeur tel que REDIS [47]. Redis (Remote Dictionary Server) est un système de gestion de base de données clé-valeur, très puissants, écrit avec le langage de programmation C ANSI et distribué sous licence BSD. Une des principales caractéristiques de Redis est de conserver l'intégralité des données en RAM. Cela permet d'obtenir d'excellentes performances en évitant les accès disques, particulièrement coûteux.

Notre dispositif est transparent vis à vis du protocole SIMPLE permettant ainsi à deux utilisateurs connectés de s'envoyer des messages instantanés.

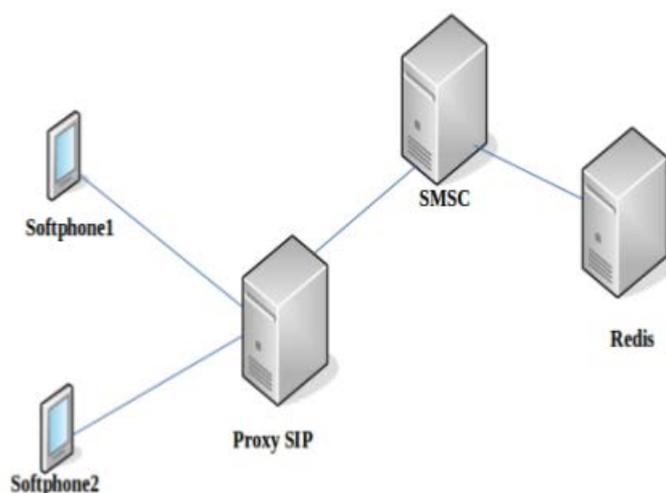


FIGURE I.4.12 – Intégration d'un SMSC

Lorsque le destinataire n'est pas rattaché au réseau, le serveur de localisation indique au SMSC l'absence du destinataire suite à sa requête. A cet effet, le SMSC stocke le message dans une base de données clé-valeur pour une éventuelle retransmission.

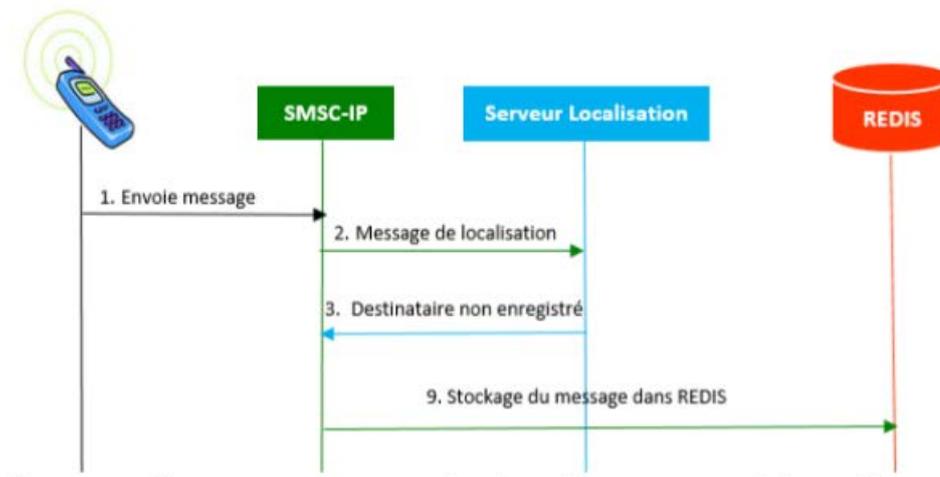


FIGURE I.4.13 – Stockage de message vers un destinataire non rattaché au réseau

Dès qu'un utilisateur est connecté sur le serveur de proxy, le SMSC détecte sa présence et l'enregistre d'abord dans le serveur de localisation. Ensuite, il vérifie dans la base de l'information si l'utilisateur ne dispose pas de message en son absence. Si oui, le SMSC récupère les messages et ordonne au serveur de proxy de les transférer à l'utilisateur, une fois connecté de nouveau. Le terminal, à la réception des messages, délivre un accusé de réception au serveur de proxy qui délivre à son tour un rapport qui indique au SMSC le bon déroulement du processus de livraison du message. A cet effet, le SMSC supprime le message en mémoire.

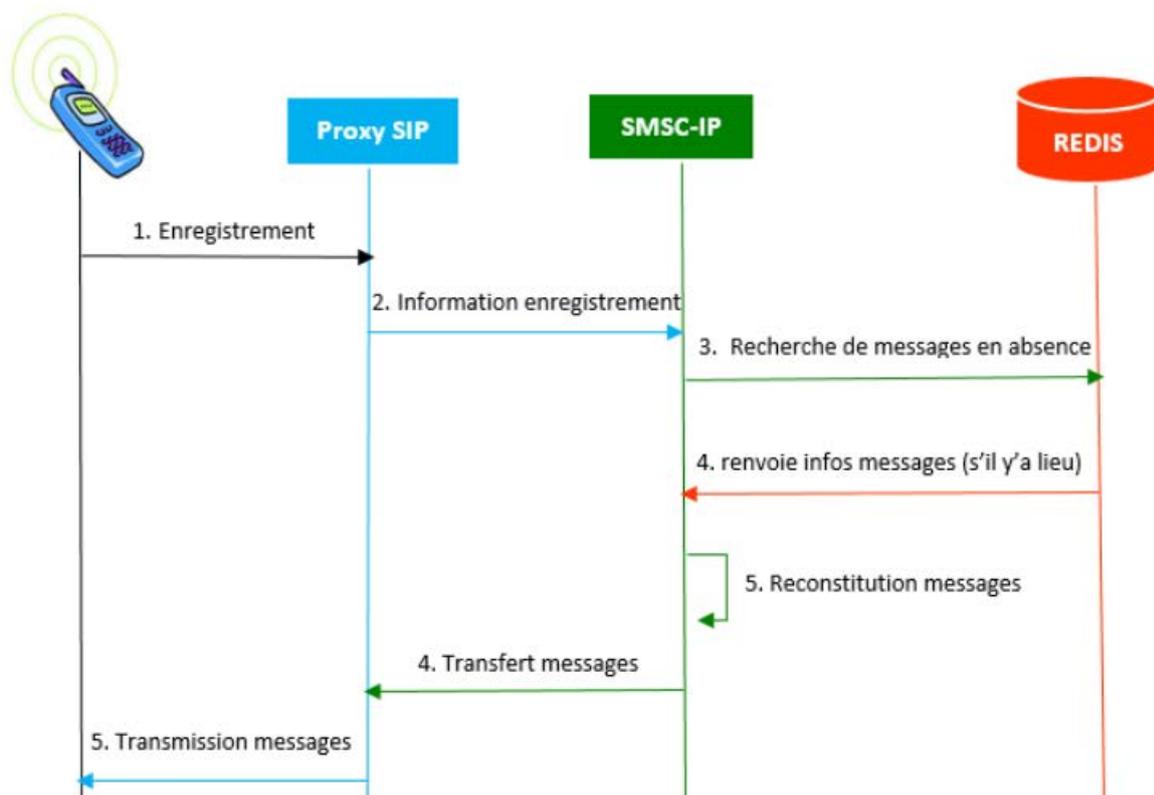


FIGURE I.4.14 – Transmission du message en absence

### Le service SMS multipoint

Le service SMS multipoint permet à un utilisateur d'envoyer un message vers plusieurs destinataires. On parle souvent de messages de diffusion. A cet effet, deux entités ont été définies à savoir : le GBC (Group Broadcast Center) et le GBE (Group Broadcast Entity). Le GBE est responsable d'envoi des messages de diffusion au GBC qui assure la gestion des messages par les fonctions suivantes :

- Il alloue un numéro de série au message de diffusion
- Il modifie ou supprime les messages déjà envoyés
- Il désigne l'ensemble des terminaux à qui le message doit être diffusé
- Il détermine la période de rediffusion du message
- Il détermine la période pendant laquelle le message ne doit pas être diffusé

Il est à noter que lors de la rediffusion du message, les terminaux l'ayant déjà reçu l'ignorent. La figure I.4.15 présente l'architecture du service SMS Group Broadcast.

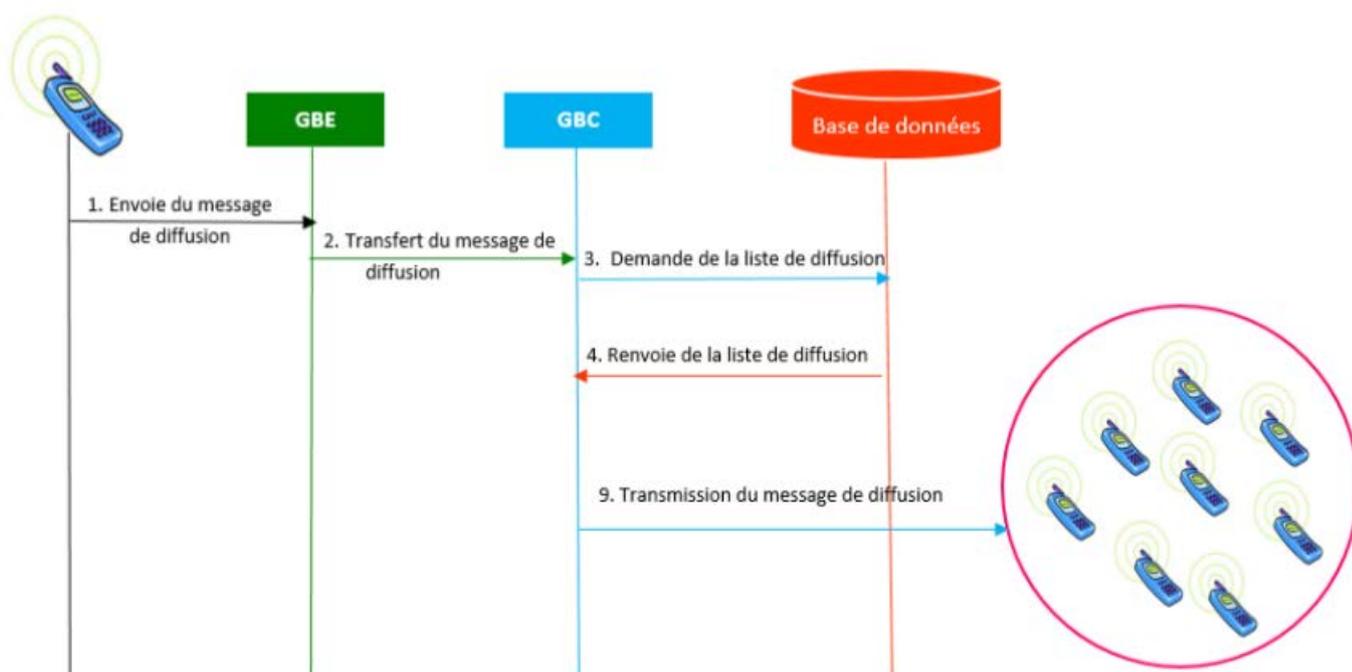


FIGURE I.4.15 – L'architecture du service SMS Group Broadcast

Les fonctions de GBC et GBE seront intégrées au SMSC-IP afin de lui fournir le support de messages de diffusion.

#### 4.5.1.3 Interconnexion avec les opérateurs 2G/3G et Services à valeur ajoutée

Pour tester la faisabilité de l'interconnexion de notre SMSC-IP avec les SMSC classiques des opérateurs de télécoms, nous avons utilisé le logiciel SMPPSim qui utilise le protocole SMPP généralement utilisé par les ESME [48].

Notre SMSC-IP est vu comme étant un ESME par les SMSC et dispose d'une interface SMPP lui permettant de relier le monde IP et le monde 2G/3G.

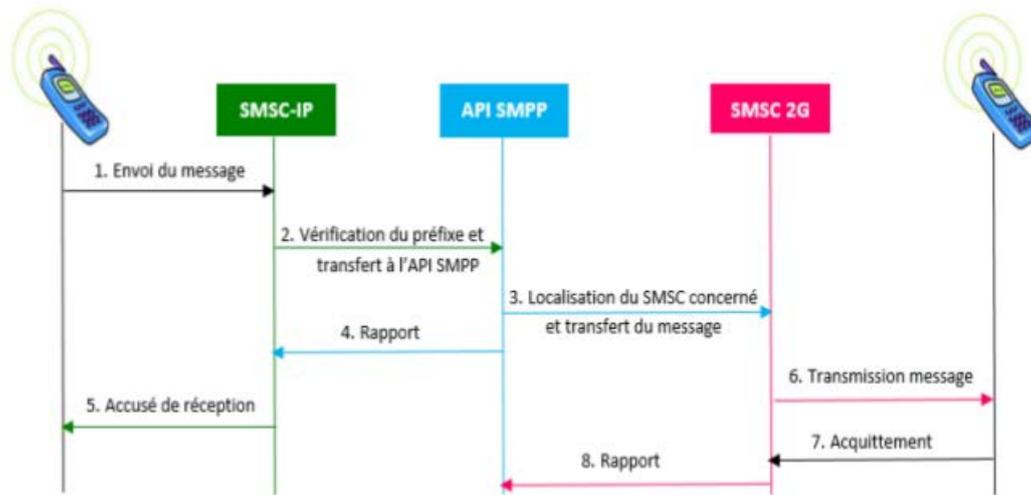


FIGURE I.4.16 – Transfert de message d'un utilisateur SIP vers un utilisateur 2G

Concernant les services à valeur ajoutée, un serveur d'application est prévu pour contenir l'ensemble des services comme le montre la figure I.4.17. Les services sont identifiés à partir d'un mot clé qui doit être la chaîne du corps du message. Ainsi à la réception d'un SMS en destination d'un service, le SMSC-IP effectue un certain nombre de vérification avant d'exécuter le service demandé.

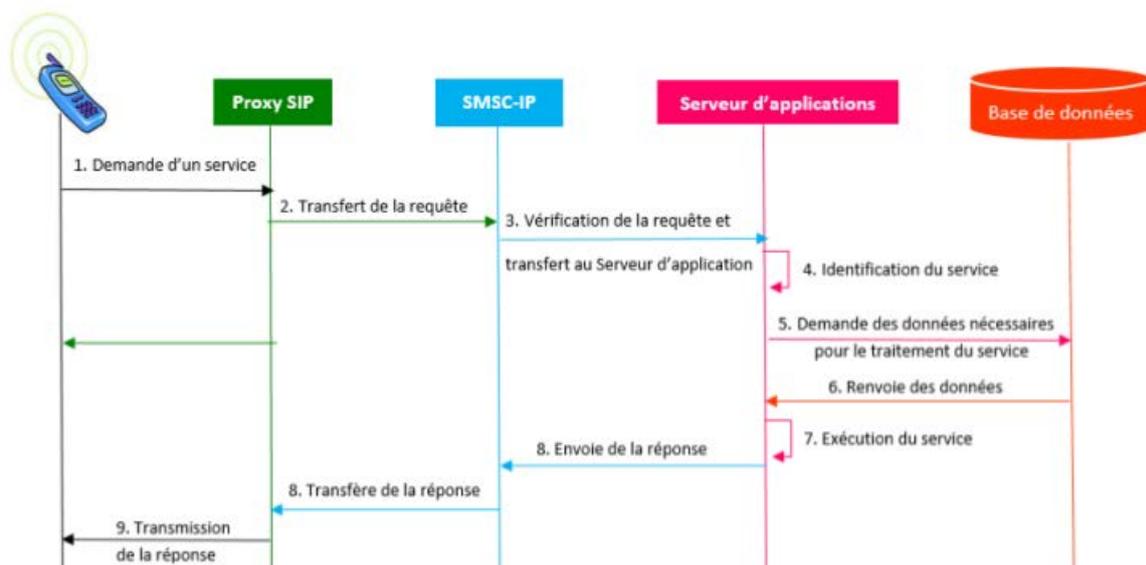


FIGURE I.4.17 – Principe de fonctionnement d'un service à valeur ajoutée

#### 4.5.2 Proposition d'une solution de SMS sur LTE/EPC

Notre solution peut être utilisée dans un réseau LTE/EPC du moment où il est tout IP pour vu que les terminaux des utilisateurs puissent utiliser un client SIP ou intègre un navigateur compatible WebRTC. Néanmoins notre solution est interopérable avec les SMSC des opérateurs 4G.

Selon 3GPP[46], Lorsqu'un mobile est dans un réseau 4G, il n'a pas besoin de basculer vers 2G/3G pour envoyer ou recevoir des SMS. On utilise l'interface SGs utilisé par la signalisation entre le MME et le MSC Server tel que décrite par la figure I.4.18 :

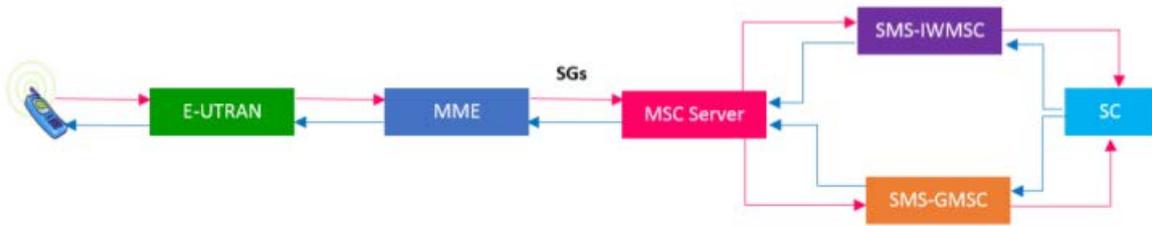


FIGURE I.4.18 – SMS over SGs

Ainsi dans le cas où l'opérateur ne dispose pas d'une infrastructure VoLTE mais dispose plutôt d'un réseau 4G, alors on utilisera le SMS Over SGs décrit ci-dessus pour l'envoi et la réception de messages.

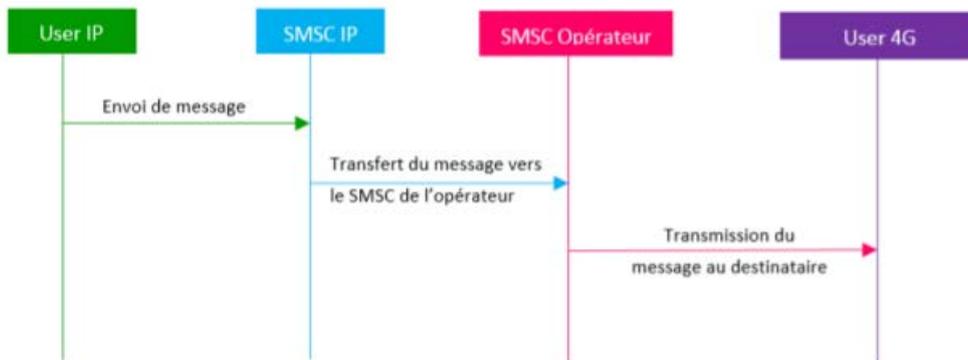


FIGURE I.4.19 – Interopérabilité de notre solution avec le réseau d'un opérateur 4G

Dans le cas d'un opérateur VoLTE disposant d'un SMSC 2G/3G et d'un IP-SM-GW intégré à un réseau IMS, le SMSC de l'opérateur va être l'intermédiaire entre notre SMSC-IP et l'IP-SM-GW de l'opérateur quand un abonné de notre SMSC-IP cherchera à envoyer un message à destination d'un abonné géré par le réseau VoLTE de l'opérateur et inversement.

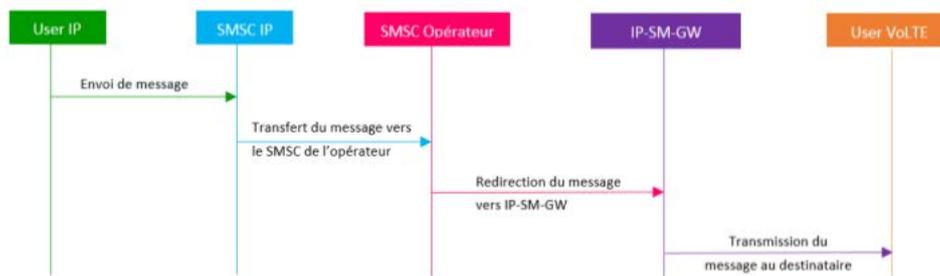


FIGURE I.4.20 – Interopérabilité de notre solution avec le réseau d'un opérateur VoLTE

### 4.5.3 Présentation des tests de validité

#### SMSC

- L'utilisateur Samuel se connecte

```
root@kalidou:~/Bureau/Stage/SMSC1# python smsc.py
The user samuel to the post 1000 has been connected
The user samuel has not received any messages in his ab
```

FIGURE I.4.21 – Connexion de l'utilisateur Samuel

- On vérifie que l'utilisateur est bien dans la liste de présence

```
root@kalidou:~/Bureau/Stage/SMSC1# python smsc.py
The user samuel to the post 1000 has been connected
The user samuel has not received any messages in his ab
```

FIGURE I.4.22 – Vérification de la liste de présence

- L'utilisateur Samuel envoie un SMS à un utilisateur déconnecté nommé Kalidou

```
The user 1000 has send an SMS to the user 10
```

FIGURE I.4.23 – Envoi d'un SMS à un utilisateur déconnecté

- Connexion de l'utilisateur Kalidou et réception du message en absence que lui avait laissé Samuel

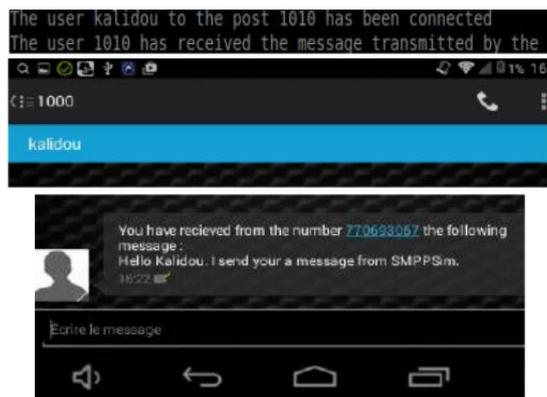


FIGURE I.4.24 – Réception d'un message transmis en absence

- On voit que l'utilisateur Samuel n'est plus dans la liste de présence car étant déconnecté et l'utilisateur Kalidou est bien enregistré

```
127.0.0.1:6379> zrange user 0 -1
1) "kalidou"
127.0.0.1:6379> █
```

- Une fois le message transmis, il est automatiquement supprimé de la base d'informations

```
127.0.0.1:6379> zrange listmessage 0 -1
(empty list or set)
127.0.0.1:6379> █
```

FIGURE I.4.25 – Suppression d'un message transmis de la base de données

## Interconnexion avec les SMSC des opérateurs

Nous avons utilisé SMPPSIM comme simulateur du SMSC des opérateurs

Cas : UA vers UserSIP

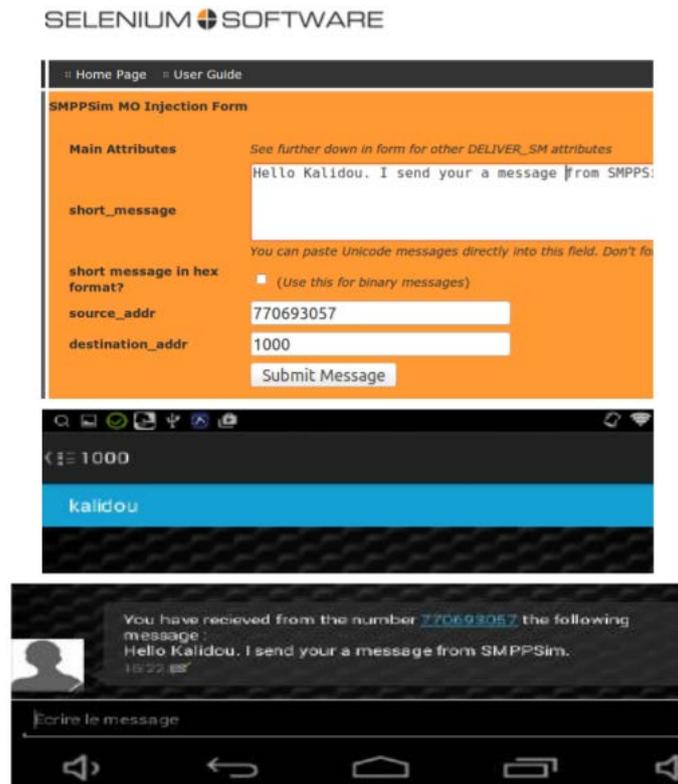


FIGURE I.4.26 – Envoi d'un SMS de UA vers SIP

## Services à valeurs ajoutés : Exemple de demande de bulletin de notes

Ce service permet aux étudiants de l'UVS d'envoyer un SMS avec un mot clé pour obtenir leurs notes. Le message est sous la forme : Notes CodeEtudiant codeSecret.



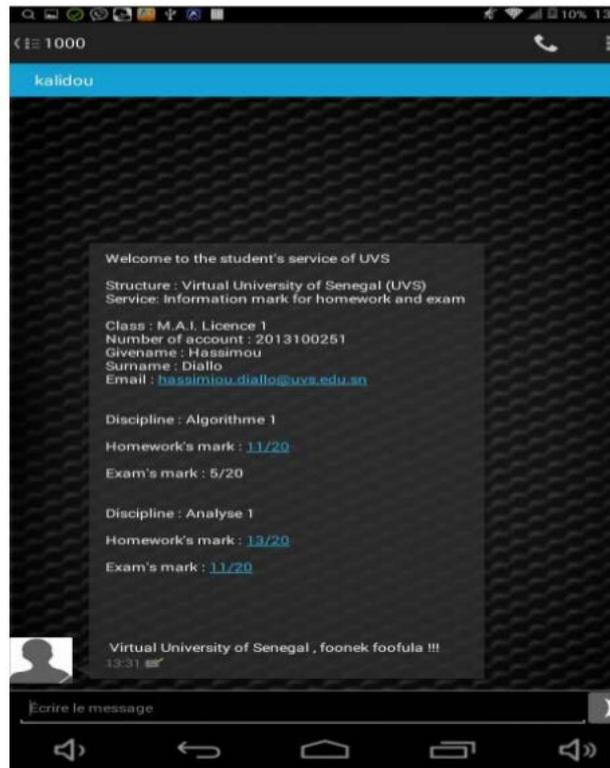


FIGURE I.4.27 – SVA : Demande de bulletin de notes

### Création d'un client SIP Web avec l'API SIPML

Pour permettre à notre système d'être utilisé à partir de terminaux WebRTC, nous avons développé un client SIP sur le Web à partir de L'API SIPMLJS qui permet le développement d'applications VoIP en HTML5, riches et robustes. Ces applications permettent la communication par la voix sur des réseaux compatibles IP, qu'il s'agisse de réseaux privés ou d'internet, filaire (ADSL/câble/optique) ou non (Wifi, GSM, UMTS ou LTE), mais aussi de la messagerie instantanée. Concernant la messagerie instantanée la réception de message n'est pas encore définie dans l'API d'où l'utilisation de notre solution de SMSC-IP. Notons que l'utilisation d'un navigateur pour les services de téléphonie est due à l'avènement du standard WebRTC.

Le client SIP WEB dispose en plus des fonctions de messagerie, les fonctions d'appels audio, vidéo et de présence.

Les différentes figures ci-après montrent les différentes fonctionnalités citées.



FIGURE I.4.28 – L'interface d'accueil du client SIP WEB

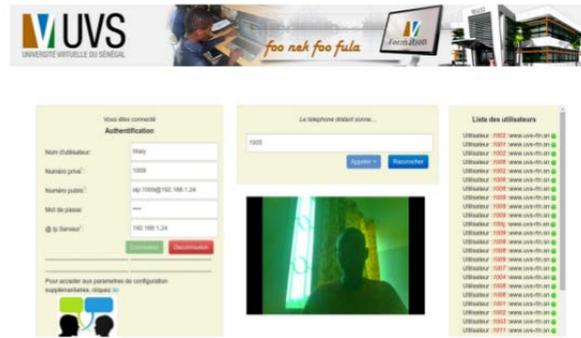


FIGURE I.4.29 – Appel à partir de la plateforme



FIGURE I.4.30 – Envoi d'un message depuis la plateforme vers un softphone

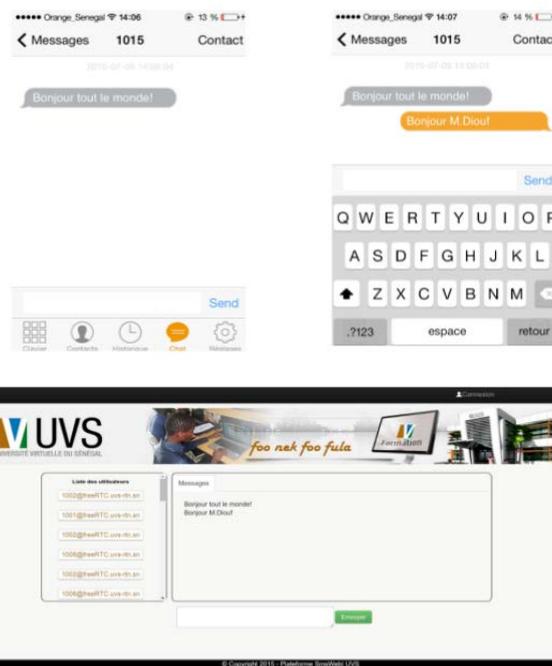


FIGURE I.4.31 – Envoi et réception depuis un softphone

## 4.6 Proposition d'une solution de réseau 4G optimisé

Dans cette section, nous discutons de la pertinence du déploiement d'un réseau 4G utilisant une plage de fréquence issue du dividende numérique. Le coût de déploiement de ce réseau doit être minimisé en vue de proposer des coûts de connexion réduits pour l'éducation et la santé. Le coût d'un système de facturation pouvant gérer à peu près 500.000 abonnés, étant de l'ordre de 1.310.000.000 FCFA d'après un équipementier contacté à Dakar, nous nous sommes dit qu'il serait intéressant de pouvoir proposer un système de facturation à moindre coût dans un environnement convergent de réseau de Télécoms. Or le réseau IMS est aujourd'hui incontournable dans un réseau convergent comme l'a montré le chapitre 1.

### 4.6.1 Présentation du système de billing

Le système de billing que nous proposons est basé sur le logiciel libre a2billing [49] et son principe de fonctionnement est décrit sur la figure I.4.32.

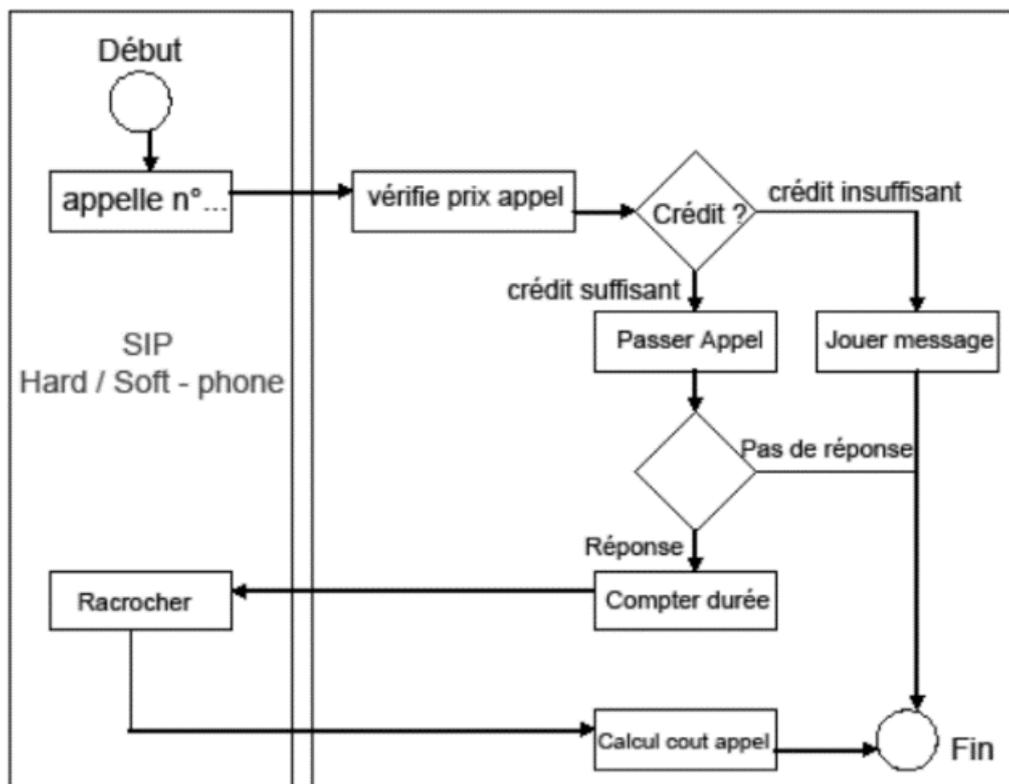


FIGURE I.4.32 – Algorithme du système de Billing

Notre système de facturation présente les fonctionnalités suivantes :

- Un module pour la gestion des clients
- Un module pour la gestion des services
- Un module pour la gestion des revendeurs de crédit appelés agents
- Un module pour la gestion de la facturation proprement dite
- Un module pour l'administration du système proprement dit

#### 4.6.1.1 Architecture

Après avoir décrit les fonctionnalités que peut offrir un système de facturation, il est souhaitable de donner une architecture qui décrit le système.

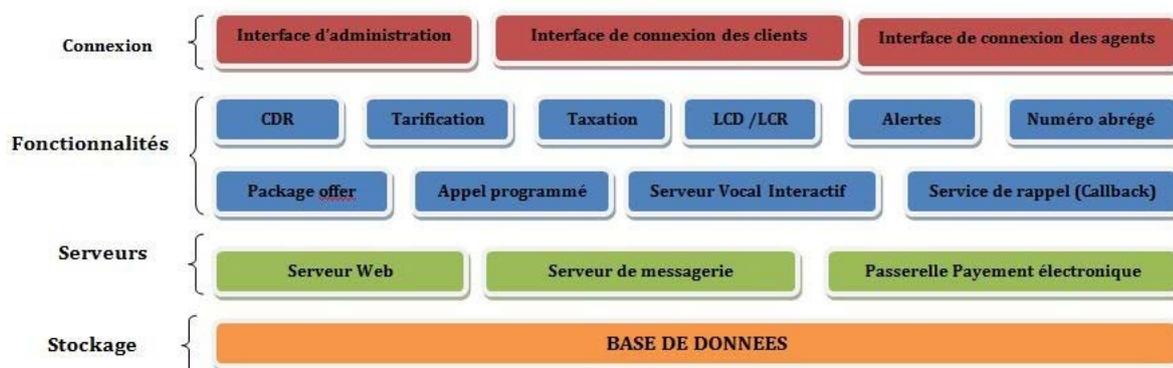


FIGURE I.4.33 – Architecture d'un système de facturation moderne

#### 4.6.1.2 Principe de facturation d'un système de facturation moderne

Pour faire la facturation pour un appel, on a besoin de deux notions importantes : la taxation et la tarification.

##### Définition des notions de facturation

**La tarification** est le fait d'attribuer un prix fixe à une destination donnée. Pour un appel vers une destination, on crée un tarif. Donc à chaque destination correspond un tarif donné.

**La taxation** permet dans un premier temps de comptabiliser la durée de l'appel, et d'estimer le coût de la communication. La taxation est donc le fait d'appliquer la tarification créé pour une communication.

##### Idée de taxation et tarification pour un système de facturation moderne

Pour fixer les taux de taxation et tarification, il faut faire en sorte que les critères de tarification et taxation à mettre soient bénéfiques autant pour l'utilisateur que pour le fournisseur.

##### Proposition de critères de taxation

- taxation des appels en fonction de la destination à atteindre,
- taxation en fonction du niveau d'avantages du client.

##### Proposition de critères de tarification

- On crée des tarifs pour chaque destination,
- On crée l'ensemble des cartes tarifications appliquées pour différentes destinations,
- On crée enfin un plan de tarification qui contient l'ensemble des cartes de tarifications.

## Principe de taxation et de tarification

En se basant sur les idées de taxation et tarification définies, on peut définir trois notions importantes :

**RATE (tarif)** : tarif appliqué vers une destination (pays, client, appel national, international . . .)

**RATE CARD** (carte des tarifs) : ensemble des tarifs (RATES) appliqués vers différentes destinations à travers le même trunk (le même provider, le même fournisseur)

**CALL PLAN** : ensemble des RATE CARDS

Le principe est de définir des «**rates**», c'est-à-dire des tarifications en fonction de la destination des appels (en se basant sur le préfixe). Ainsi, lorsqu'un client passe un appel, on lui affecte une certaine tarification. L'ensemble des tarifs qui peuvent être appliqués à ce client sont regroupés dans une «**rate card**». Les «rate cards » correspondent aux différentes offres tarifaires proposées aux clients. Enfin on a le «**call plan**» qui regroupe toutes les «rate cards » et chaque «call plan» associé à un client.

Comme défini ci-dessus, un callplan est un ensemble de tarifs (rate cards) disponibles à travers différents fournisseurs, que l'on choisit à partir d'une règle de routage prédéfinie. Ces règles de routage font référence à un algorithme pour la sélection de routes. Un bon système de facturation doit être en mesure d'appliquer ces routages. Nous avons deux types de routage : LCD et LCR.

- Le LCR (Least Cost Routing) : revient à chercher le chemin le plus avantageux au fournisseur. Il est appliqué quand on recherche le coût le plus petit pour le fournisseur.
- Le LCD (Least Cost Dialing) : revient à chercher le chemin le plus avantageux au client. Il est appliqué quand on recherche le coût le plus petit pour le client.

### 4.6.1.3 Application

Pour comprendre les notions précitées, nous allons simuler tout le processus de facturation d'une communication IP en montrant comment ces différentes notions interagissent entre elles pour former un système complet de facturation.

On suppose que nous avons deux clients à facturer, ayant des plans de tarifications différentes avec des tarifs différents.

La procédure est la suivante :

#### Les clients effectuent des appels ,

- Pour chaque client, le système effectue d'abord une authentification grâce à : son numéro de compte, ou son caller ID, ou sa carte d'appel, etc....
- Ensuite, on vérifie les informations sur l'utilisateur : le type d'abonnement, le call plan auquel il est lié afin de déterminer comment la facturation sera traitée.

## Création de clients

Les clients représentent les entités de base d'un système de facturation, car sans client pas d'appel et pas de facturation. Pour créer un client, il lui créer un compte. Les informations importantes pour la création d'un compte d'utilisateur sont :

- Un numéro de compte,
- Un mot de passe,
- Un nom d'utilisateur,
- Un solde initial,
- Le type d'abonnement (compte prépayé ou postpayé),

Le numéro de compte et le mot de passe peuvent être utilisé par l'utilisateur pour se connecter sur son portail, afin de modifier à son tour ses informations personnelles, recharger son compte, voir les appels, etc.

## Création des fournisseurs

Les fournisseurs offrent un accès vers des réseaux VoIP ou téléphoniques. Ce sont eux qui se chargent de router (en quelque sorte) les appels. Les appels sont redirigés vers ces fournisseurs suivant les numéros d'appels saisis par le client.

Dans notre application, on crée deux fournisseurs, un fournisseur pour un accès sur le réseau VoIP et un autre fournisseur pour l'accès au réseau téléphonique RTC.

## Création des Trunks

Parallèlement à la création des fournisseurs, il faut penser à créer les trunks associés. Les Trunks sont utilisés pour faire aboutir les appels en se connectant au serveur du fournisseur. En effet, le trunk et la grille tarifaire sont sélectionnés par le module de tarification sur la base des numéros composés pour facturer les appels. Dans notre exemple, nous aurons deux trunks, l'un lié au RTC appelé Trunk RTC et l'autre lié au fournisseur VoIP appelé Trunk IP.

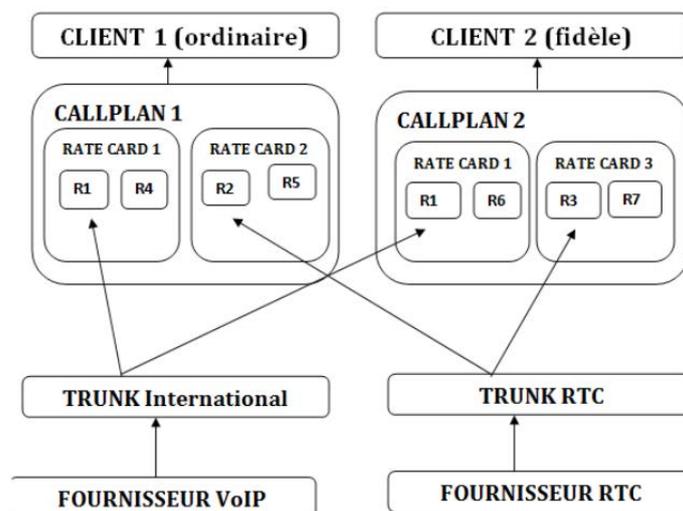


FIGURE I.4.34 – Plan de facturation

## Facturation

Nous voulons facturer les clients en faisant ressortir les aspects clés d'une facturation. On choisit deux clients que nous allons facturer en se basant sur leur statut de fidélité. Dans notre exemple, nous allons utiliser les statuts «fidèle» et «ordinaire». On commence par créer le plan de tarification à associer à chaque client.

### Création d'un plan de tarification (CALL PLAN)

Nous allons créer un callplan pour les clients 1 «fidele» et un autre pour les clients 2«ordinaire». Les clients «fidèle» bénéficient de meilleurs tarifs que les membres «ordinaire».

### Création des cartes de tarifs (RATE CARD)

Ensuite, on choisit de créer trois cartes de tarifs c'est-à-dire les rates cards, un pour les appels sortant vers l'international, et deux pour les appels sortant vers le réseau RTC. Ainsi, nous créons respectivement les rates card RATE CARD 1, RATE CARD 2, RATE CARD 3.

### Création des tarifs

Nous allons maintenant créer trois tarifs différents.

Commençons avec le tarif pour les appels sortant vers un fournisseur VoIP à l'international. Dans ce cas, nous supposons que le fournisseur d'accès VoIP route les appels vers un pays comme l'Europe par exemple, d'où le préfixe 33. Nous allons également supposer que :

- le prix d'achat auprès du fournisseur de VoIP en France est de 100F/minute.
- le temps minimum que le fournisseur facture est de 30 secondes (soit 50F).
- le fournisseur facture chaque communication par bloc de 30 secondes. Ce qui fait qu'une communication de 35 secondes et une autre de 59 secondes coûteront le même prix.

Dans notre exemple, nous allons décider de :

- vendre les appels pour le tarif double (200/ minute).
- le temps minimum à facturer est fixé à 60 secondes.
- on facture la communication par bloc de 60 secondes. Ce qui fait qu'une communication de 1minute 05 secondes et une communication de 2minutes coûteront le même prix.

Pour les appels sortants vers le RTC si on dispose d'une passerelle vers les opérateurs historiques, le principe reste le même.

Le tableau ci-dessous résume la politique de taxation mise en place dans l'exemple :

TABLEAU I.4.1 – Tableau récapitulatif du plan de facturation

Paramètres	Rate1 (International)	Rate 2	Rate 3
Rate Card	Rate Card 1	Rate Card 2	Rate Card 3
Trunk	Trunk VoIP	Trunk RTC	Trunk RTC
Fournisseur	Fournisseur VoIP	Fournisseur RTC	Fournisseur RTC
Préfixe	0033	00221	00221
Prix d'achat	100FCFA	75FCFA	75FCFA
Temps minimum facturé par le fournisseur	30 secondes	30 secondes	30 secondes
Temps de communication par bloc facturé	30 secondes	30 secondes	30 secondes
Prix de vente	200FCFA	150FCFA	120FCFA
Temps minimum facturé par le fournisseur	60 secondes	60 secondes	60 secondes
Temps de communication par bloc facturé	60 secondes	60 secondes	60 secondes

#### 4.6.2 Présentation du processus de communication en IMS : IFC

A chaque usager IMS est associé un profil d'utilisateur dans le HSS. Un profil d'utilisateur consiste en un ensemble de profils de service. Un profil de service contient :

- une ou plusieurs IMPUs (IMS Public User Identities) ayant la forme d'une adresse téléphonique ou d'une URI SIP,
- zéro ou une instance de la classe Core Network Service Authorization indiquant les différents média pouvant être utilisés pour les sessions établies avec ces identités publiques,
- un ensemble (0 à N) de critères de filtrage (iFC, initial Filter Criteria). Un critère de filtrage est une information statique correspondant à une souscription d'un usager à un service du domaine IMS.
- Un ensemble (0 à N) de "Shared iFC set". Un "Shared iFC Set" pointe sur un ensemble d'iFC administrés localement et stockés sur le S-CSCF. Un "Shared iFC Set" peut être partagé par plusieurs profils de service, permettant de minimiser la taille du profil de l'utilisateur.

Le profil de service est obtenu par l'entité S-CSCF auprès du HSS à travers l'interface Cx lorsque l'utilisateur s'enregistre au sous-système IMS. Un critère de filtrage ne peut s'appliquer qu'à une méthode SIP qui est une requête initiale, c'est à dire REGISTER, INVITE, SUBSCRIBE, MESSAGE ;

- AS address : Adresse du serveur d'application à contacter (adresse SIP)
- Priority : Priorité du critère de filtrage indiquant sa position dans la liste.

- Trigger Point : Un trigger Point est composé de une à N instances de SPTs (Service Point Trigger). Des SPTs peuvent être combinés à l'aide d'opérateurs logiques (AND, OR, NOT, etc.).
- Default handling : Prise en charge par défaut si l'entité S-CSCF n'arrive pas à contacter l'AS
- Optional Service Information : Information de service facultative rajoutée au contenu de la méthode SIP avant qu'elle soit acheminée à l'AS (l'IMSI).

La figure I.4.35 décrit les composants d'un IFC.

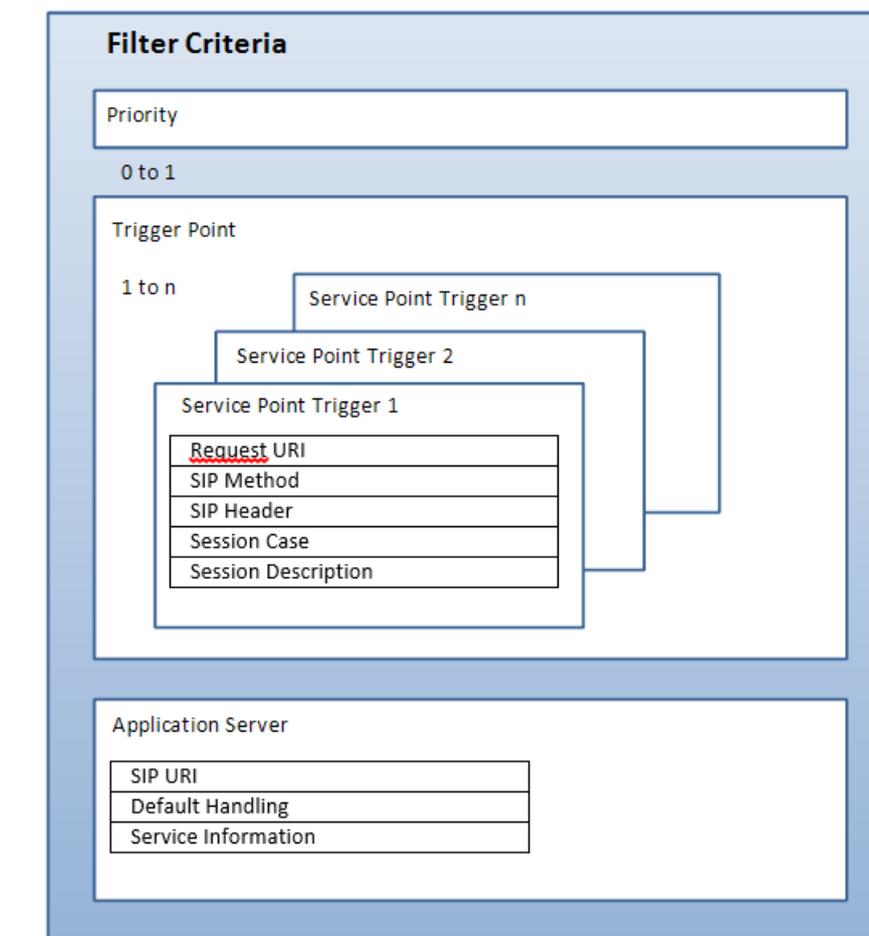


FIGURE I.4.35 – Les composants d'un IFC [14][15]

Cette compréhension des IFC nous a permis de simplifier le processus de facturation dans l'IMS sans utilisation des entités OCS, CDF, OFCS, etc tels que définis par 3GPP et décrit dans le chapitre 2 ni l'utilisation du protocole Diameter.

### 4.6.3 Architecture de déploiement de notre solution

La facturation est gérée par un serveur d'application dans l'architecture IMS. Grâce aux données IFC, un filtre va être effectué pour envoyer tous les appels (Session Case) au serveur d'application de facturation (Billing Application Server – BAS) vers lequel l'entité S-CSCF transfère la requête si les conditions sont réunies. Avec le IMSU c'est-à-dire l'identifiant de l'appelant, le BAS fait le contrôle du crédit du client IMS avant l'établissement d'un appel. Cela conduit aux services de taxation prépayé et postpayé.

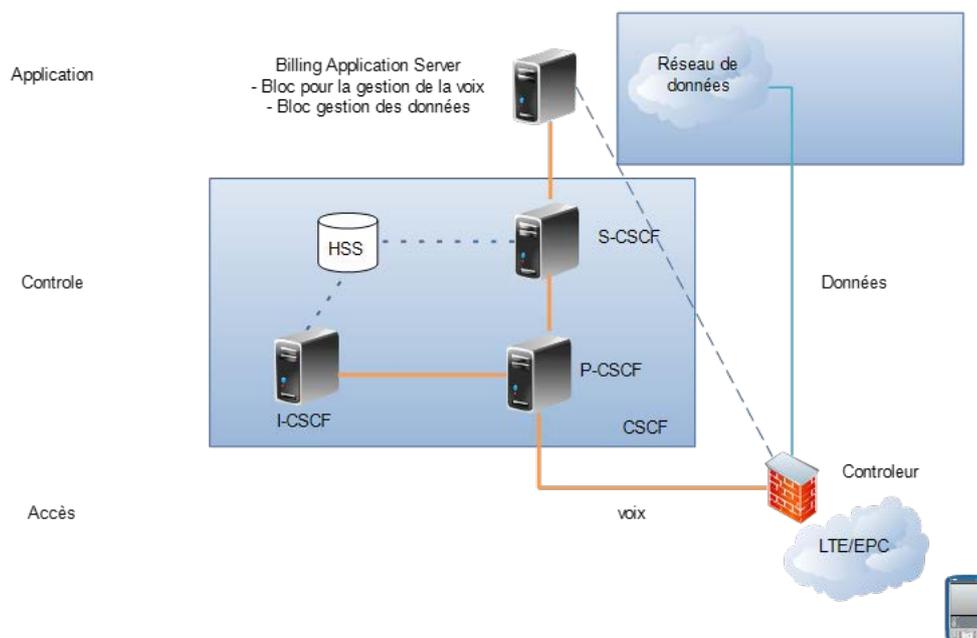


FIGURE I.4.36 – Architecture de notre solution

Notre architecture, bien que tenant compte de la facturation d'accès à Internet, nous ne décrivons dans la suite que la partie facturation voix.

#### 4.6.3.1 Diagramme de fonctionnement de notre solution de facturation

Les détails de fonctionnement de notre système sont illustrés sur la figure I.4.37.

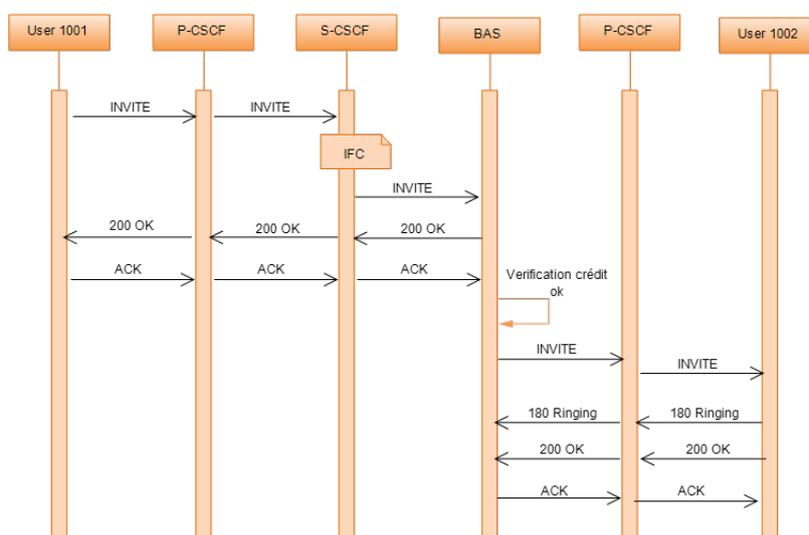


FIGURE I.4.37 – Processus de facturation

Pour faire la preuve de notre système, nous avons utilisé OpenIMSCore.

#### 4.6.3.2 Mise en oeuvre et résultats

Les résultats obtenus sont présentés dans cette section. Nous utilisons les utilisateurs d'OpenIMSCore Claude et MaryTeuw.

## Les étapes d'interconnexion avec le système de facturation

### Création d'un AS

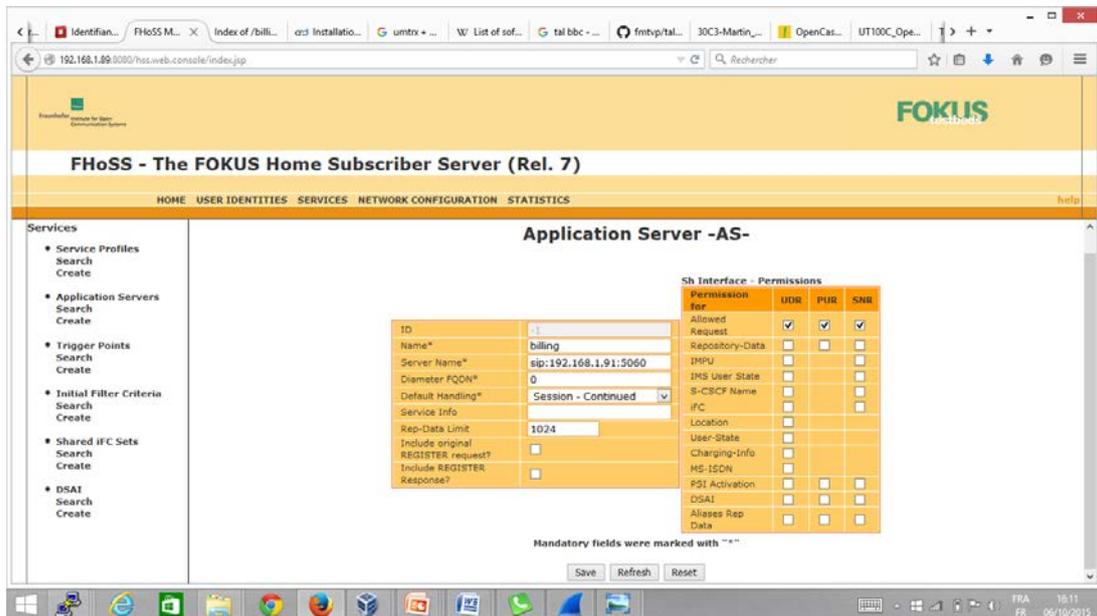


FIGURE I.4.38 – Création d'un AS

### Création du Trigger Point

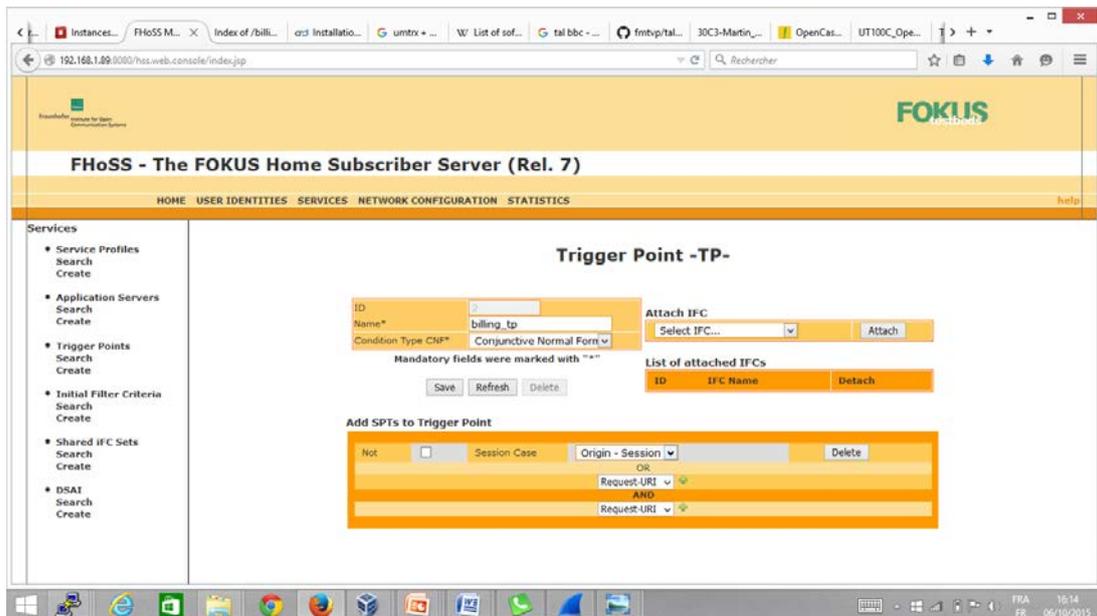


FIGURE I.4.39 – Création du Trigger Point

### Création d'un IFC

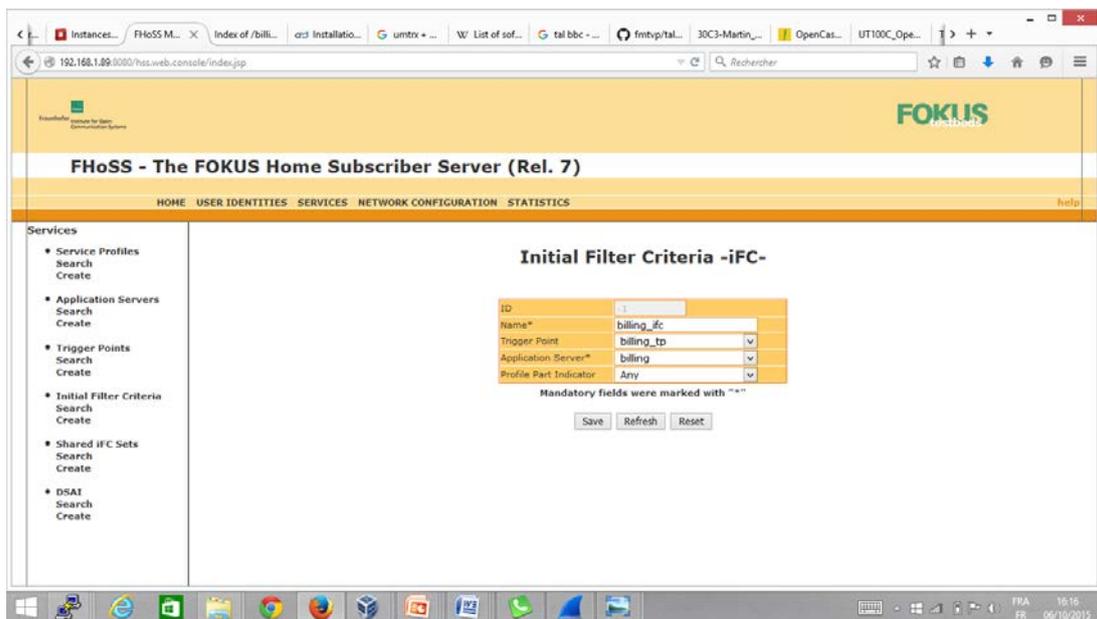


FIGURE I.4.40 – Création d'un IFC

### Création d'un Service Profile - SP

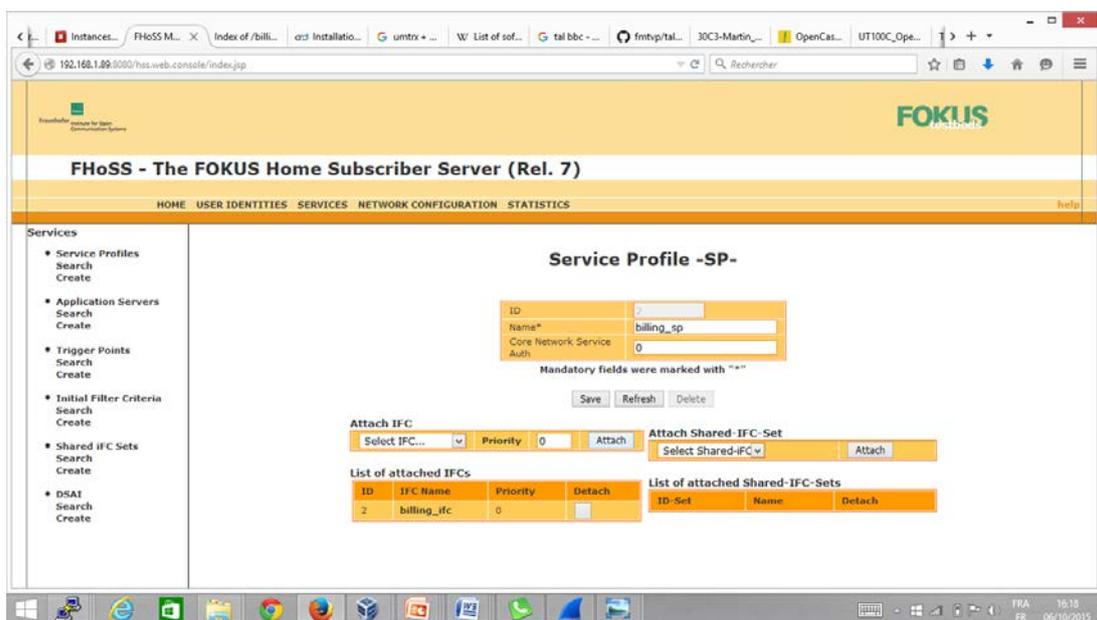


FIGURE I.4.41 – Création d'un SP

### Ajout des clients dans le profile billing -SP

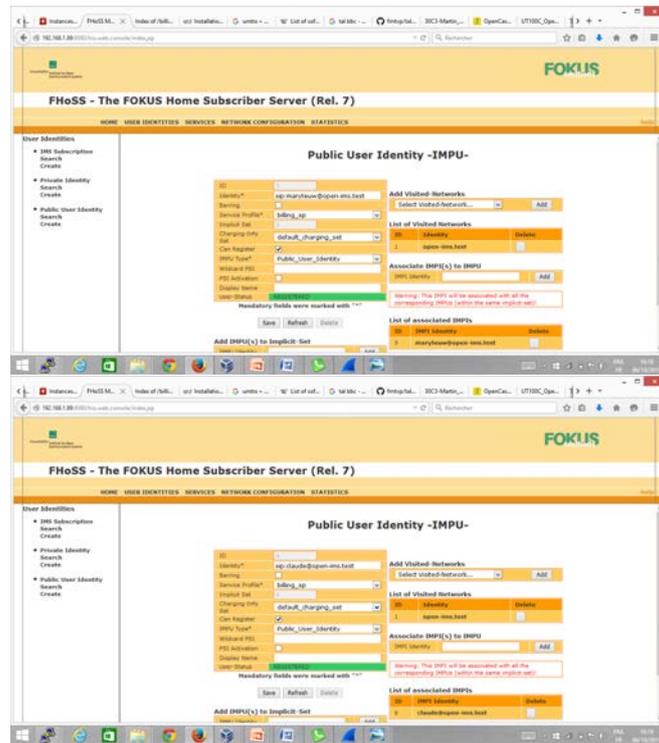


FIGURE I.4.42 – Ajout des clients dans le profile billing

### Modification du IMSU des clients IMS

Le système de billing se basant sur le numéro de l'appelant pour la facturation, le IMSU des clients IMS est modifié en numéro.

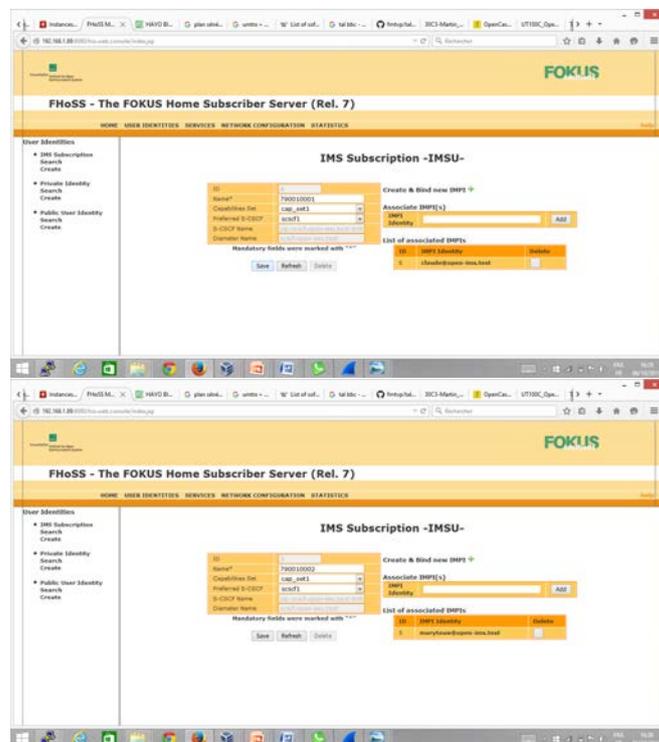


FIGURE I.4.43 – Modification du IMSU des clients

À cette étape, il faut se connecter sur le système de billing pour faire un mapping entre les IMSU

des utilisateurs et les CALLERID dans le système de facturation.

## Page d'authentification du Billing

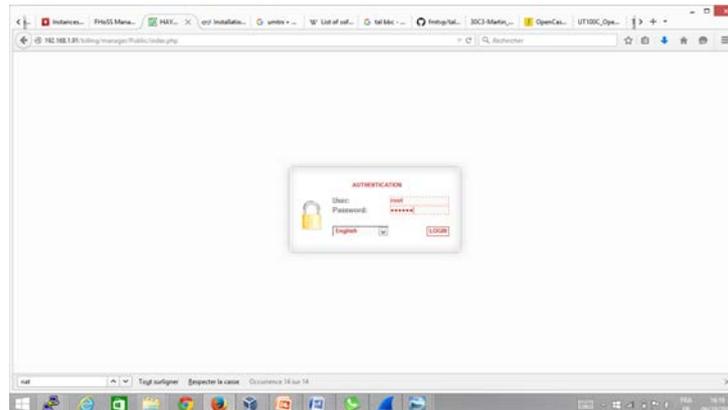


FIGURE I.4.44 – Page d'authentification du Billing

La figure I.4.45 montre la richesse des fonctionnalités offertes par notre système de billing.

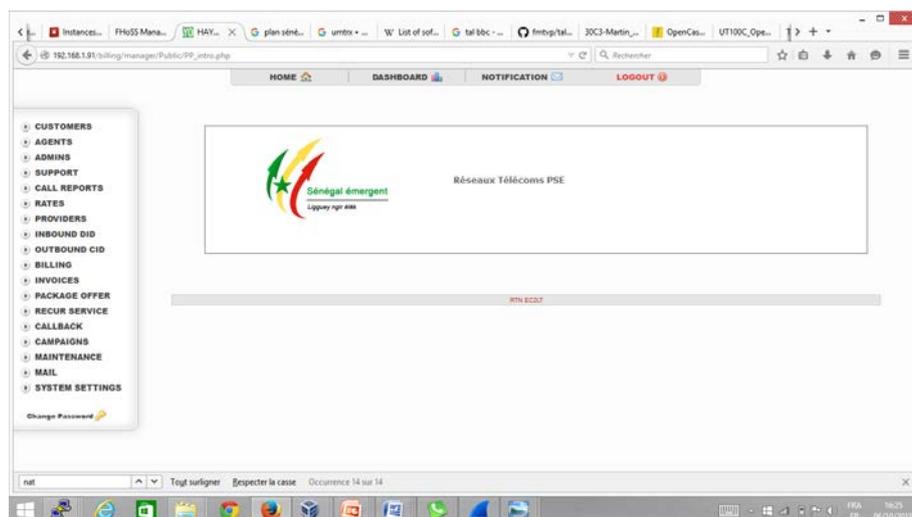


FIGURE I.4.45 – Fonctionnalités du système de billing

A cette étape, il est important de créer des comptes de facturation qui seront associés aux utilisateurs IMS.

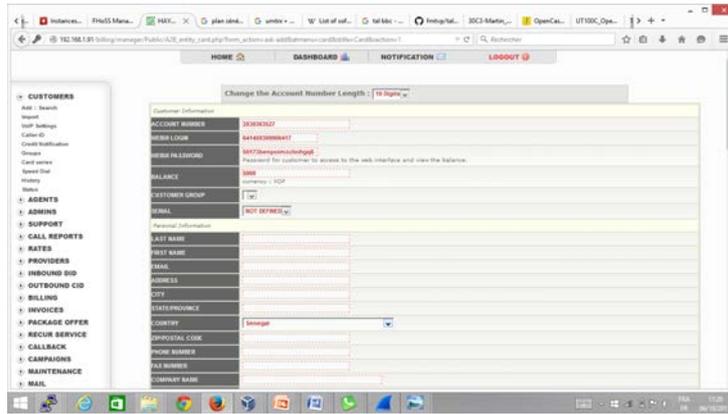


FIGURE I.4.46 – Création de compte pour l'utilisateur Maryteuw sur le billing

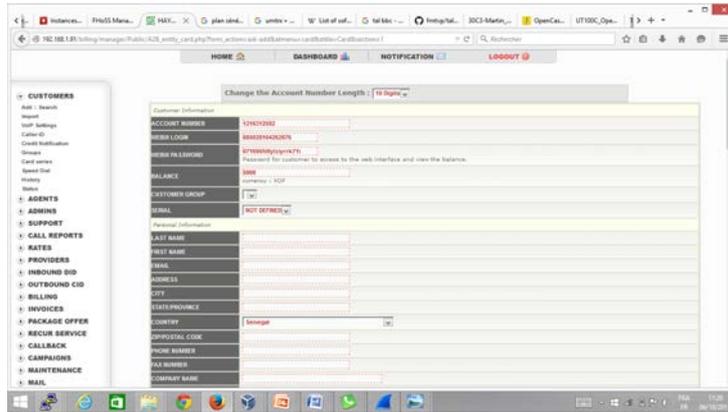


FIGURE I.4.47 – Création de compte pour l'utilisateur Claude sur le billing

Comme nous l'avons dit ci-dessus, une correspondance entre un numéro du client IMS (CALLERID) et compte billing est nécessaire.

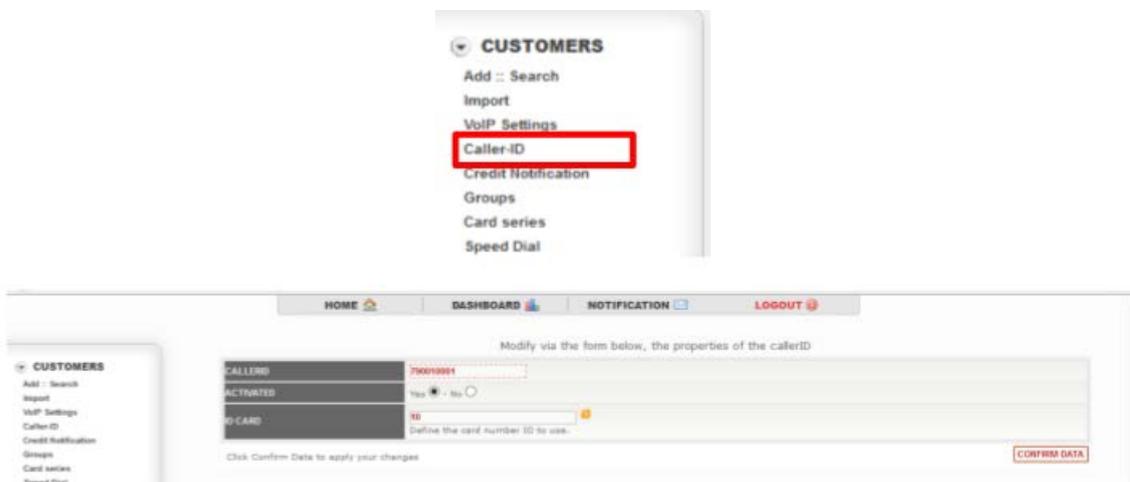


FIGURE I.4.48 – Correspondance entre l'IMSU de Maryteuw et son compte de facturation

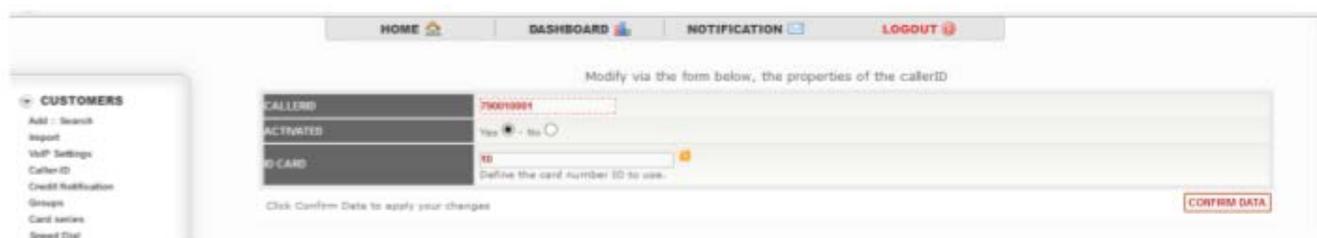


FIGURE I.4.49 – Correspondance entre l'IMSU de Claude et son compte de facturation

Nous pouvons à présent voir la liste des correspondances effectuées précédemment.

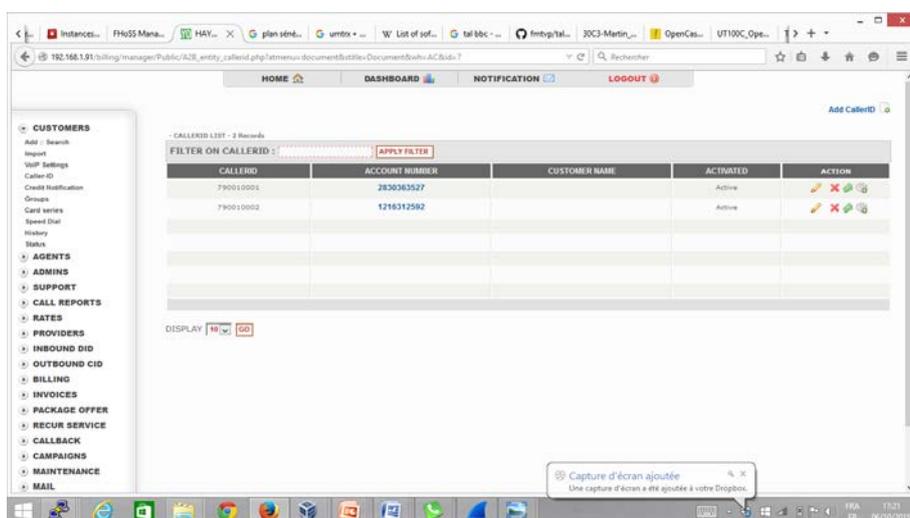


FIGURE I.4.50 – Liste des correspondances entre comptes IMS et CALLERID billing

La figure I.4.51 décrit la définition des tarifs sur notre système de billing.



FIGURE I.4.51 – Définition des tarifs

À présent, nous allons passer à l'ajout des tarifs définis dans le plan de numérotation de nos utilisateurs.

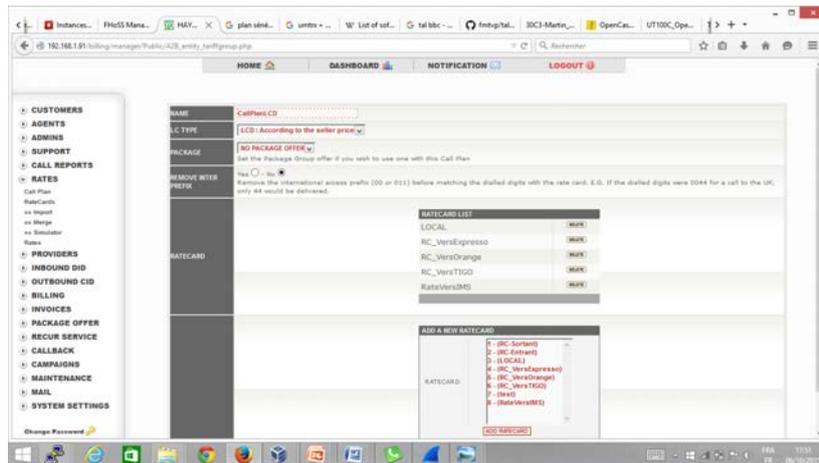


FIGURE I.4.52 – Ajout des tarifs dans le plan de numérotation des utilisateurs

La figure I.4.53 montre l'état du crédit des clients avant les appels.

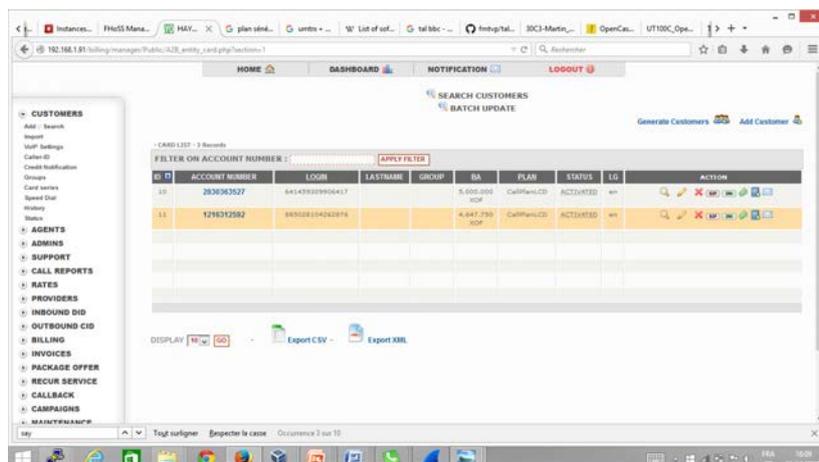


FIGURE I.4.53 – État du crédit des clients

## Tests de notre système de facturation

Notre système de facturation est doté d'une interface permettant de visualiser les détails de communication (CDR)

### Appel du 790010001 vers le 790010002

Les CDRs montrent la durée de l'appel et le coût de l'appel.

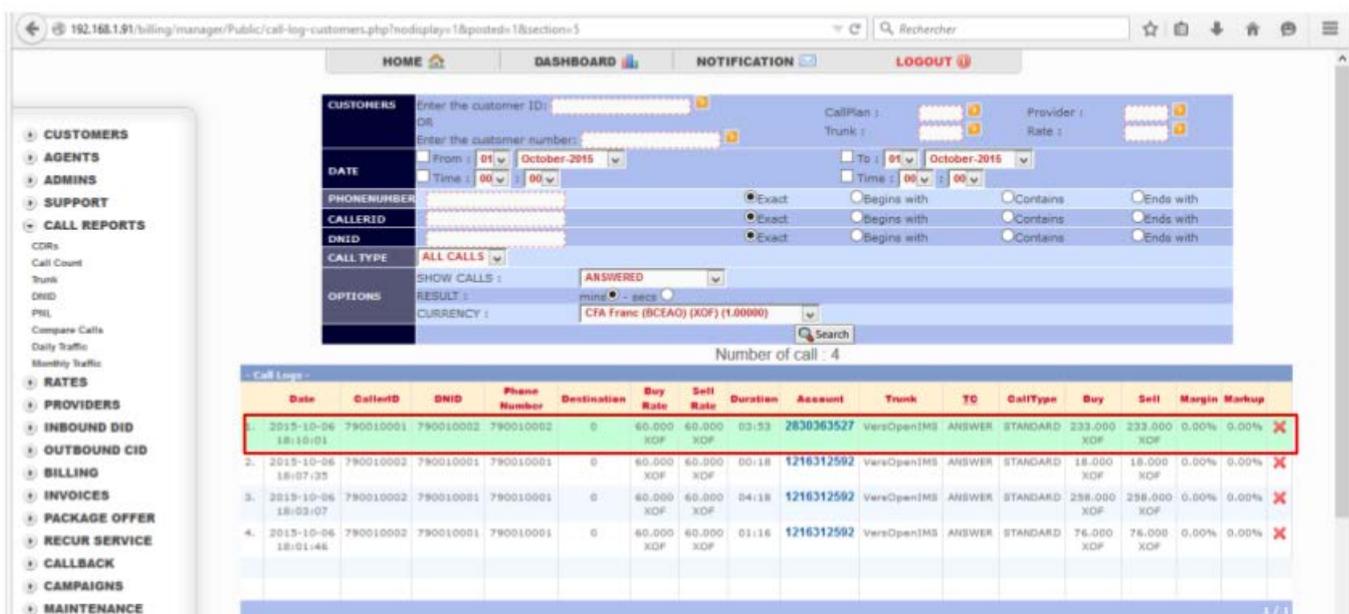


FIGURE I.4.54 – CDR de l’appel du 790010001 vers le 790010002

Nous pouvons vérifier l'état du crédit des clients après l'appel. Le compte du client appelant a été débité conformément à la durée d'appel.

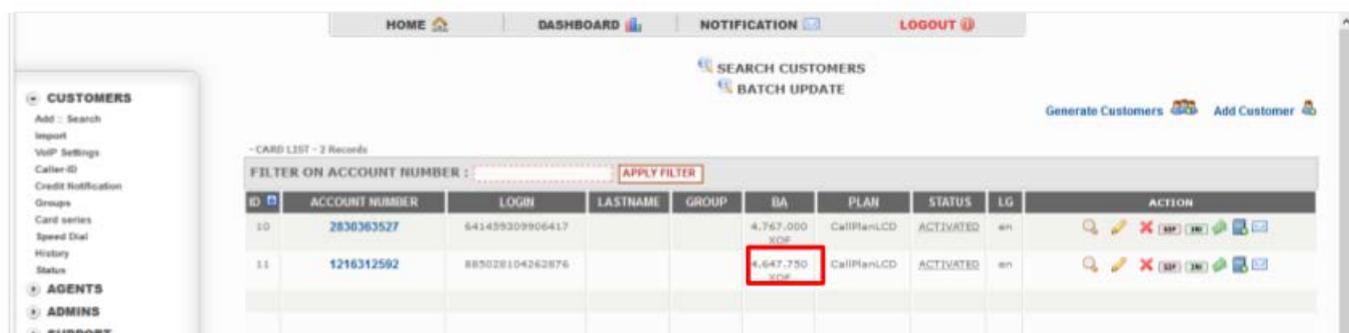


FIGURE I.4.55 – État du crédit des clients après appel

#### 4.6.4 Conclusion

L'architecture que nous avons proposée, permet de facturer les appels dans IMS en se basant sur les critères de routage d'appel iFC. Les entités OSCF, OFCS et le protocole Diameter ne sont pas utiles dans une telle architecture. Notre solution pourrait être adoptée par les pays en voie de développement qui veulent profiter du dividende numérique pour déployer un réseau 4G à moindre coût. Des renseignements auprès des équipementiers télécoms montrent le coût élevé du système de billing.

L'adoption de notre solution par un opérateur 4G de télécoms lui permettrait de réduire le coût du déploiement de son réseau et en conséquence le coût d'accès pour les utilisateurs finaux. Ceci est important dans le contexte de l'enseignement supérieur et la santé qui ont besoin d'un réseau haut débit mobile pour fournir des services de qualités accessibles partout et à tout moment à leurs différents acteurs.

## 4.7 Conclusion

Le dividende numérique provenant du passage à la télévision numérique est une opportunité à saisir par les pays en voie de développement. Son utilisation optimale pourrait permettre la mise en place d'un réseau haut débit.

Ce qui pose problème dans nos pays en matière de services télécoms, c'est plutôt la disponibilité d'un réseau et moins nos capacités à proposer des services adaptés à nos besoins, pour preuve, des services à valeur ajoutée innovants développés par des ingénieurs et techniciens africains.

La solution de télévision interactive connectée que nous avons proposée pourrait rendre certainement service à l'éducation et à la santé. Cette télévision interactive se base sur la technologie WebRTC, la disponibilité d'un réseau IP qu'on peut avoir grâce à la norme TVWS, pour créer la voie retour du téléspectateur vers la chaîne de télévision.

La solution de SMS sur IP que nous avons proposée pourrait servir des réseaux d'enseignement et de recherche qui se créent et qui font partie d'un vaste réseau IP et pourra permettre aux acteurs des universités de bénéficier de services à valeur ajoutée à coût presque nul en utilisant le protocole SIP.

Le réseau 4G étant tout IP, la solution de SMS sur IP, légèrement modifiée, a permis d'avoir un centre SMS sur 4G interopérable avec les SMSC des opérateurs classiques de télécoms.

Au cas où certains États africains seraient tentés de réserver une plage de fréquence sur le dividende numérique pour déployer un réseau 4G, son coût pourrait être optimisé par l'utilisation non seulement du système SMS proposé, mais aussi du système de billing proposé.

# Conclusion générale

La convergence de l'audiovisuel, de l'informatique et des télécommunications crée une nouvelle manière de donner accès à des services aux utilisateurs aussi bien des terminaux mobiles que des télévisions connectées. Le tout IP haut débit mobile est devenu une réalité, en partie grâce à la fibre optique et au dividende numérique. En effet, déployer un vaste réseau IP à l'échelle d'une ville ou d'un pays devient abordable pour beaucoup de pays en quête de l'émergence.

Du coup, l'accès à un réseau IP mobile haut débit pour les populations ne devrait plus poser de problème majeur si les mesures idoines sont prises.

Cette généralisation d'Internet mobile nous a poussés à penser que beaucoup de simplifications pouvaient être opérées pour minimiser les coûts de déploiements des infrastructures et services Télécoms adaptés aux besoins des populations des pays en voie de développement.

En effet, l'utilisation de notre solution de centre SMS sur IP et les variétés de services à valeur ajoutée qui en découlent est de nature à réduire les coûts d'acquisition des centres SMS jusqu'alors proposés par les équipementiers de Télécoms. Un autre argument en faveur de notre solution est d'une part des opérateurs de télécoms tout IP commencent à s'installer en Afrique, c'est le cas de Hayo Sa au Sénégal, et d'autre part, les réseaux de l'enseignement et de la recherche construits autour de l'IP n'ont pas besoin de solutions sophistiquées onéreuses pour proposer des services à valeur ajoutée à leurs utilisateurs.

Il est à noter qu'il serait important que certains pays profitent du dividende numérique pour réserver une plage de fréquences pour le déploiement d'un réseau 4G LTE/EPC dédié à l'éducation et la santé.

Dans ce cas, notre solution de Centre SMS sur 4G pourrait être utilisée dans un tel réseau pour fournir des services à valeur ajoutée. De la même manière, le système de facturation que nous avons proposé pourrait être utilisé pour réduire le coût du déploiement du réseau.

La solution de télévision interactive que nous proposons ouvre des perspectives réelles pour l'enseignement à distance, la télé-médecine et le développement de services à valeur ajoutée pour le grand public. De la même manière que les africains se sont montrés ingénieux dans la conception des services autour du mobile, cette solution peut impulser chez eux, le développement des services innovants autour de la TNT.

# Contributions

Les travaux de recherche menés dans cette partie ont fait l'objet de deux publications IEEE et d'autres sont en phase de soumission à IEEE :

[1] Ouya, S. ; Dahirou Gueye, A. ; Sy, K. ; Niane, M.T. ; Lishou, C., "**Contribution to reducing the effects of geographical separation between actors of virtual universities : Proposal of an IP-SMSC integrating value-added services solutions**" in Interactive Collaborative Learning (ICL), 2015 International Conference on , vol., no., pp.1145-1150, 20-24 Sept. 2015

[2] Yelomé, L. T. ; Ouya ,S. ; Ndiaye S. ; Farssi M.S. ; "**Contribution to SMS management over 4G network in distance education context**" in ACS/IEEE International Conference on Computer Systems and Applications AICCSA 2015, November 17-20, 2015, Marrakech, Morocco

## Deuxième partie

# Applications des services de télécoms aux organisations virtuelles

# Introduction générale

Grâce au développement des Technologies de l'Information et de Communication (TIC), on assiste à une forme d'organisation appelée Organisation Virtuelle (O.V) qui est un projet réunissant plusieurs équipes multifonctionnelles mises temporairement ensemble pour réaliser une mission collective spécifique.

Ainsi, une organisation virtuelle est constituée d'individus ou d'équipes spécialisées mises en relations virtuelles afin de réaliser une finalité commune.

Grâce aux TIC, il est aujourd'hui facile d'attirer les meilleures compétences là où elles se trouvent, et de pouvoir les faire coopérer sur un projet sans être obligé de les réunir dans un même lieu.

A l'heure actuelle, en Afrique et particulièrement au Sénégal, l'enseignement supérieur traverse un certain nombre de difficultés parmi lesquelles on peut citer l'insuffisance des infrastructures et de matériels pédagogiques pour les étudiants malgré les efforts considérables de l'État qui consacre 40% de son budget à l'enseignement.

Malgré un taux de réussite soutenu au baccalauréat, à 0.8% le Sénégal est loin d'atteindre le ratio du besoin d'un pays qui aspire à l'émergence qui est fixé à au moins 2% de la population en âge d'accéder à l'enseignement supérieur.[50]

Pourtant ces dernières années, les universités publiques du Sénégal n'arrivent pas à absorber substantiellement les nouveaux bacheliers.

Ainsi le gouvernement a décidé de faire passer le nombre d'universités classiques publiques de 2 à 5 et de leur faire signer des Contrats De Performance (CDP) [51]. L'État, à travers les CDP, a, entre autres, construit des nouvelles infrastructures équipées de matériels de travaux pratiques dans les 5 universités publiques. Parmi les critères de performance des universités figure le nombre de cours mis en ligne par les enseignants pour les étudiants pour augmenter l'efficacité des enseignements.

Malgré tout cela, le problème d'orientation des nouveaux bacheliers dans les universités publiques n'est pas totalement résolu.

C'est ainsi que l'État du Sénégal a décidé d'organiser la Concertation Nationale Sur l'Avenir de l'Enseignement Supérieur (CNAES) à la suite de laquelle des décisions présidentielles relatives à l'enseignement et la recherche ont été prises[51].

La décision D1 stipulé "**Réorienter le système d'enseignement supérieur vers les sciences, la technologie, les formations professionnelles courtes**" a pour directives :

- Orienter les formations supérieures vers les Sciences, la Technologie, les Sciences de l'Ingénieur et les Mathématiques (STEMS), les filières courtes, techniques et professionnelles
- Mettre en place un nouveau cadre administratif, juridique et institutionnel conforme à la nouvelle orientation de l'Enseignement supérieur et de la Recherche en prenant en compte les préoccupations d'équité (genre et personnes en situation de handicap)
- Appliquer le programme spécial de promotion des sciences et techniques élaboré par l'Académie

nationale des Sciences et Techniques du Sénégal

- Élaborer une politique nationale de développement de la culture et de l'information scientifiques.

Alors que la décision D2 "**Mettre les Technologies de l'Information et de la Communication (TIC) au cœur du développement de l'enseignement supérieur et de la recherche pour améliorer l'accès à l'enseignement supérieur et l'efficacité du système**" a pour directives :

- Mettre en place l'Université virtuelle sénégalaise (UVS) et des Espaces numériques ouverts (ENO) dans chacune des régions du Sénégal et au sein des universités publiques.
- Mettre en place le Système d'Information et de Gestion de l'Enseignement supérieur et de la Recherche (SIGESR).
- Interconnecter tous les établissements d'enseignement supérieur public et privé.
- Créer une bibliothèque nationale virtuelle pour partager les ressources numériques.
- Créer le Centre de Mutualisation et de Partage (CMP) de l'Enseignement supérieur et de la Recherche.
- Développer l'enseignement à distance et encourager le personnel d'enseignement et de recherche à utiliser les TICs.

La première directive de la décision D2 a été exécutée dès 2013 en créant l'Université Virtuelle du Sénégal (UVS) qui a accueilli respectivement en 2013 et 2014, 2000 et 5000 bacheliers.

Il faut noter que les problèmes que rencontre l'enseignement supérieur au Sénégal est partagé en parti par plusieurs pays en voie de développement.

Ainsi dans cette deuxième partie de la thèse, nous nous proposons de faire la synthèse des modèles des organisations virtuelles qui ont fait leur preuve et de proposer des modèles technologiques à mettre en place pour aider les pays qui aspirent à l'émergence, en particulier le Sénégal, à atteindre leurs objectifs en termes de formation et de Santé.

# Chapitre 1

## Organisations virtuelles

### 1.1 Introduction

L'enseignement en ligne semble être le remède aux problèmes posés par les modes d'enseignement traditionnels. En effet, l'enseignement en ligne combine les avantages de l'enseignement présentiel (interaction) et à distance (flexibilité) en évitant les inconvénients de rigidité de temps et de lieu ainsi qu'une interaction très limitée.

Cependant pour un certain nombre d'auteurs, l'enseignement en ligne, devrait rejoindre d'autres médias, comme entre autre la télévision, qui après avoir suscité beaucoup d'espoir et d'engouement n'a pas amélioré l'efficacité de l'enseignement.

Des recherches en science de l'éducation ont montré un taux d'abandon élevé et un taux de réussite bas en enseignement en ligne et ceci pourrait être expliqué par des facteurs externes tels que la motivation et l'isolement des candidats, leur capacité de travail, la maîtrise de l'outil informatique.

Ce qui nous amène à émettre l'hypothèse qu'une interactivité forte entre les différents acteurs de l'enseignement à distance en utilisant des moyens des TIC pourrait permettre d'améliorer l'efficacité de l'enseignement à distance.

Nous émettons aussi l'hypothèse qu'un environnement de formation à distance qui se rapproche de plus de mode de formation en présentiel qui privilégie la pédagogie et qui considère les technologies comme moyens pour atteindre les objectifs pédagogiques pourrait être efficace.

Les logiciels dits libres offrant la possibilité à ses utilisateurs d'accéder à leurs codes, de les étudier et les adapter à leurs besoins contrairement aux logiciels dits propriétaires ne pourraient ils pas jouer un rôle important dans l'amélioration de l'efficacité de l'enseignement à distance ?

Le modèle de coopération utilisé par les logiciels libres et dont beaucoup ont fini de démontrer leur efficacité ne montre t-il pas que l'éloignement des différents acteurs n'est pas aujourd'hui un frein à l'atteinte d'un objectif commun correctement exprimé ?

Ce chapitre se propose

- de faire une synthèse sur organisations virtuelles autour :
  - de l'enseignement et la recherche ;
  - des logiciels libres ;
- d'évaluer les prérequis en termes de connexion réseau favorables à la constitution des organisations virtuelles dans le secteur de l'enseignement et recherche,
- et de proposer une solution technologique de réduction de l'effet de séparation géographique des acteurs de la formation à distance.

## 1.2 Organisations virtuelles dans le domaine de l'enseignement

Il est important de définir des concepts clés rencontrés dans le domaine des sciences de l'éducation.

### 1.2.1 Formations à distance

L'enseignement à distance renvoie habituellement aux formations offertes à des apprenants qui sont physiquement séparés des autres participants et du formateur.

Selon la définition proposée en 1987 par l'UNESCO, l'enseignement à distance serait un « mode d'enseignement, dispensé par une institution, qui n'implique pas la présence physique de l'enseignant chargé de le donner à l'endroit où il est reçu, ou dans lequel l'enseignant n'est présent qu'à certains moments ou pour des tâches spécifiques. Les communications enseignants-enseignés se font principalement par le recours à la correspondance, aux imprimés, aux divers médias audiovisuels, à l'informatique, à certains regroupements ».

Trois caractéristiques essentielles de la formation à distance sont à retenir :

- La non-présence des participants – enseignants et apprenants-, en raison de la distance physique ;
- Le rôle particulier de l'enseignant
- Le recours à divers moyens de communication.

### 1.2.2 Formations Ouvertes et À Distance

Pour un groupe de chercheurs (M'hammed Drissi, Mohamed Talbi, Mohamed Kabbaj), « La formation ouverte à distance fait partie de la famille de la FAD mais se positionne sur l'intégration des technologies de l'information et de la communication, de l'adaptation à l'individu et de la modularité de la formation. Il s'agit d'un dispositif organisé, finalisé, reconnu comme tel par les acteurs, qui prend en compte la singularité des personnes dans leurs dimensions individuelle et collective, et qui repose sur des situations d'apprentissage complémentaires et plurielles en termes de temps, de lieux, de médiations pédagogiques humaines et technologiques, et de ressources »

Quelle que soit la dénomination des différentes formes d'organisation virtuelle autour de l'enseignement supérieur et de la recherche, on note l'usage des TIC et la volonté de collaborer.

### 1.2.3 Université virtuelle

L'université virtuelle peut se définir comme un réseau de sites "virtuels" d'échanges intellectuels dont l'objectif est de rendre obsolètes les anciens bâtiments couverts de lierre, de même que les frontières institutionnelles et politiques.

### 1.2.4 Campus numérique

Dans certains pays francophones essentiellement, on a assisté à la naissance du "campus numérique" qui se distingue des projets "d'université virtuelle", d'une part, sur une affirmation du rôle des institutions universitaires dans le développement de l'enseignement à distance et d'autre part sur une minoration de l'importance de technologies de l'information et de la communication dans l'élaboration de la politique d'enseignement supérieur.

Dans un "campus numérique", les formations doivent être modulaires et diplômantes, elles doivent utiliser les TIC pour offrir des services d'accueil et des ressources pédagogiques numérisées, elles doivent mettre en place du tutorat.

Dans la littérature, nous remarquons une différence nette entre les concepts d'université virtuelle et de campus numérique.

Une université virtuelle fait allusion à un ensemble de campus virtuels utilisant fortement les TIC pour collaborer sans nécessairement avoir l'accord des institutions coiffant ces campus tandis un campus numérique met l'institution au cœur de son système.

Dans un campus numérique, un diplôme est délivré par une institution universitaire.

Dans le cas du Sénégal, l'université Virtuelle du Sénégal (UVS) est une université au même titre que les autres universités physiques du Sénégal sauf sur les modalités de délivrance de contenus de l'enseignement qui utilisent fortement les TIC.

## 1.2.5 Modèles pédagogiques des universités numériques

Une nouvelle architecture du savoir se met en place ; entre la source du savoir et l'apprenant, on retrouve une chaîne de liens qui vont de la conception du produit, de sa fabrication, de sa mise en scène grâce à des programmes intégrant le texte, la voix, la vidéo, le graphisme, la musique, à sa livraison et à son suivi.

Le véritable enseignement en ligne est donc beaucoup plus que l'affichage sur le web de syllabus de cours ou de programmes. C'est tout un nouvel environnement créé par des équipes polyvalentes pour permettre aux apprenants de construire leur propre savoir de façon autonome, critique et créative à partir des ressources très diverses qui leur sont proposées et qu'ils découvrent par eux-mêmes, par un apprentissage collaboratif promouvant le dialogue et l'ouverture d'esprit, des auto-évaluations fréquentes, instantanées et anonymes, une exploitation autonome des ressources illimitées du Web dans une perspective de formation continue.

### 1.2.5.1 Rôles d'un enseignant dans le contexte d'une formation à distance

Il est important que les enseignants qui interviennent dans les formations à distance comprennent les différents rôles qu'ils doivent jouer.

Le tableau ci-après donne une idée de ces principaux rôles.

TABLEAU II.1.1 – Rôles d'un enseignant dans une formation à distance

Enseignement	Action de l'enseignant
<b>Informateur</b>	Rechercher, préparer ou produire des informations ou des documents
<b>Professeur</b>	Produire un discours ou des interactions visant à dire le savoir ou à donner la vision savante ou socialisée de ce qui être appris
<b>Pédagogue</b>	Proposer des situations pédagogiques ou des outils pratiques visant à mettre les apprenants dans la situation de pouvoir construire leurs savoirs
<b>Animateur</b>	Interagir, motiver, susciter des échanges afin de garantir l'intensité de l'activité d'apprentissage et de gérer la progression du groupe
<b>Médiateur</b>	Accompagner et dialoguer avec l'apprenant afin de l'aider à franchir une étape ou résoudre une difficulté d'ordre cognitif
<b>Évaluateur</b>	Mesurer, noter, hiérarchiser les acquisitions des élèves par rapport aux groupes et par rapport aux normes à atteindre. Aider par ces indications l'apprenant à comprendre son erreur

### 1.2.5.2 L'importance des tuteurs dans les formations à distance

Dans le modèle du socioconstructivisme, l'enseignant met en ligne son cours et disparaît du système. Le tuteur intervient pour accompagner les étudiants. Et c'est en collaborant entre eux que les étudiants développent leurs savoirs.

Dans cette perspective l'étude des pratiques de tutorat est essentielle car elle montre une diversification des pratiques en continuum avec l'évolution ou la réorganisation des actions de l'enseignant. Le tuteur prend en compte la fonction d'animation et d'accompagnement qui est une des tâches de l'enseignant.

Le tuteur n'est alors qu'une des facettes de l'acte d'enseigner qui se matérialise soit dans le dispositif médiatique soit dans l'émergence de nouvelles fonctions dans un travail d'équipe de construction.

Ce métier ou cette fonction est aujourd'hui bien connu des acteurs de la formation à distance sans que les compétences nécessaires et les pratiques mise en jeu soient clairement identifiées. La fonction de tuteur pédagogique au sein des dispositifs institutionnels universitaires classique est aujourd'hui reconnu. Au sein des universités le tutorat est avant tout méthodologique. C'est non pas un métier mais bien une fonction temporaire réalisé par un étudiant afin d'accompagner l'apprenant dans son parcours. C'est avant tout une pratique de pairs à pairs qui n'intervient que très rarement dans le processus didactique de l'enseignant.

Les missions du tuteur sont :

- Communiquer avec les apprenants
- favoriser le processus enseignement-apprentissage
- favoriser l'utilisation optimale des plateformes techniques
- favoriser la maîtrise des contenus
- gérer les conditions psycho affectives et sociales de l'apprentissage

Jacques RODET en 2004 définit les champs de l'intervention du tuteur sur quatre niveaux en interaction :

1. **Le cognitif** qui renvoie au contenu disciplinaire, au soutien méthodologique des apprenants qui vise à leur faciliter la mise en œuvre de stratégies d'apprentissage, l'aide technique pour naviguer harmonieusement dans le dispositif et la transmission d'informations administratives, commerciales.

2. **Le socio-affectif** qui regroupe les interventions du tuteur visant à lutter contre le sentiment d'isolement de l'apprenant. D'autres démarches socio-affectives du tuteur visent à l'accroissement de son autonomie.
3. **Le motivationnel** qui vise à encourager la persévérance à la formation et à prévenir l'abandon. Le tuteur, ne doit pas se limiter à intervenir comme aiguillon et relais des stratégies de motivation extrinsèque mais doit également agir de manière à ce que l'apprenant renforce sa motivation intrinsèque.
4. **Le métacognitif** qui renvoie à l'ensemble des activités que l'apprenant devrait réaliser pour avoir un regard distancié sur sa formation.

Devant la massification des effectifs des étudiants dans les universités classiques, certains pays envisagent de mettre en place des universités virtuelles utilisant le modèle Sénégalais. Certains acteurs de l'enseignement qui ont exercé dans les campus numériques sont tentés de transposer le modèle socioconstructivisme dans les universités virtuelles en gestation.

Nous pensons que le modèle socioconstructivisme a lui seul ne saurait permettre la formation efficace des apprenants de plus en plus jeunes.

A notre avis, il faut créer une interactivité forte non seulement entre les apprenants mais aussi entre les enseignants producteurs de contenus et les étudiants.

Nous pensons aussi qu'un modèle d'enseignement à distance doit se créer en tenant compte de l'environnement sociologique des apprenants et doit se rapprocher de plus du modèle présentiel sauf que la distance devient négligeable grâce à l'usage des moyens de communication fournis par les TIC.

### 1.3 Organisations virtuelles autour des logiciels libres

On reconnaît généralement que les technologies de l'information et de la communication contribuent à la réalisation des objectifs du millénaire pour le développement. En ce qui concerne les équipements utilisés dans les processus décisionnels touchant l'architecture des TIC et le choix de plateforme, les pays africains sont très en retard et ne peuvent donc pas exploiter tout le potentiel de ces outils pour éradiquer la pauvreté. Ce décalage dans le savoir entraîne l'utilisation non seulement de systèmes d'informations incompatibles, une maintenance coûteuse et inefficace des structures des TIC, mais aussi le paiement de licences d'utilisation de logiciel qui drainent les ressources. Pour affronter ce problème, il faut créer des compétences et des marchés au niveau local, s'appuyant sur une connaissance des TIC, et développer des industries locales des TIC plus forte, comme cela a été le cas en Asie au cours de la dernière décennie.

Cela ne saurait se réaliser sans l'aide des universités et centres de formations dont nous disposons.

Malgré le coût élevé des équipements, des logiciels et de leurs licences pour les différentes formations, certains centres de formation et universités africaines essaient tant bien que mal d'être à la hauteur des attentes de leurs étudiants. Mais cela n'est pas tout le temps évident.

Comme palliatif aux problèmes d'investissements on voit peu à peu aujourd'hui certaines universités ou centres de formations professionnelles se tourner vers les « logiciels libres ».

Pourquoi cet intérêt soudain pour les logiciels libres dans certaines universités ?

Ces logiciels peuvent-ils répondre aux besoins des universités et centres de formations africaines ?

Les logiciels libres constituent-ils un choix judicieux dans le domaine de l'enseignement supérieur ?

On remarque d'autre part que beaucoup de logiciels libres sont utilisés par des grandes entreprises qui y trouvent satisfaction.

Pourtant la plus part de ces logiciels sont développés par des gens qui ne se connaissent pas, qui sont géographiquement éloignés et qui collaborent sur des projets en utilisant les TIC.

Cela semble être une preuve que la distance n'est pas nécessairement un frein aujourd'hui dans une collaboration autour d'un projet bien bien conçu. Ainsi, cette section présente le libre et examine son apport pour la réussite d'un projet d'université numérique.

### 1.3.1 Les logiciels libres

L'avènement des supports de distribution bon marché comme l'Internet et le CD-ROM a favorisé la diffusion de logiciels développés en dehors des standards industriels. Il s'agit (le plus souvent) de logiciels développés par des personnes physiques par opposition à la personne morale qu'est l'entreprise.

Selon le niveau de contribution demandé à l'utilisateur (gratuité), la disponibilité du code source (transparence) ou encore l'importance accordée à la notion de propriété (droits d'auteur), ces logiciels sont qualifiés de domaine public, *Open Source*, *Freeware*, *Shareware* ou encore *propriétaires*. C'est la nature de la licence associée au logiciel qui détermine l'appartenance à l'une de ces familles de produits. Le concept de logiciel libre est ainsi rattaché à l'ensemble des logiciels couverts par un type de licence particulier : les licences **Open Source**.

#### 1.3.1.1 Qu'est-ce que les logiciels libres

Avant de définir ce que c'est que les logiciels libres, il est important de définir quelques concepts tels que les freewares, les sharewares, les logiciels propriétaires et le domaine public.

##### Le freeware

Souvent développés à titre de hobby par un développeur unique, ces logiciels sont (comme leur nom l'indique) gratuits. Le terme *free* est (ici) associé à la notion de gratuité. En revanche, la disponibilité du code source est laissée à la discrétion du développeur qui demeure propriétaire de son œuvre (notion de **copyright**). Dans la pratique, on constate que le code source d'un *freeware* est rarement disponible.

##### Le shareware

Le concept de *shareware* est très similaire à celui de *freeware*. Il s'en différencie uniquement par le niveau de contribution demandé à l'utilisateur. Un *shareware* n'est pas gratuit, mais son prix reste toutefois modeste. Il existe de nombreux dérivés de l'approche *shareware*, comme le *postware* - où l'auteur vous demandera pour seule rétribution de lui envoyer une carte postale! Là aussi, le code source est rarement disponible.

*Freewares* et *sharewares* sont donc souvent des logiciels à caractère propriétaire. Leur domaine d'application reste principalement axé sur les accessoires de bureau. Le monde du *Macintosh* comme celui de MS-Windows en proposent de nombreux. L'offre logicielle de BeOS est (pour le moment) majoritairement constituée de ce type de logiciels. Il est d'ailleurs notable que *shareware* et *freeware* sont des modes de distribution principalement retenus par les développeurs utilisant des systèmes d'exploitation propriétaires.

Ces deux familles de logiciels ne sont pas directement liées à la notion de *licence* – au sens où nous l’entendons lorsque l’on parle de logiciels libres.

L’appartenance à l’une de ces familles renseigne avant tout sur la gratuité du produit. La licence qui s’y rattache est propre au logiciel et se réduit le plus souvent à quelques lignes rédigées par l’auteur indiquant les restrictions qu’il souhaite imposer à la distribution comme à l’utilisation du produit.

## Le logiciel propriétaire

Tout logiciel distribué sous forme d’exécutable binaire pour lequel il n’existe aucun moyen d’accès légal au code source est un logiciel *propriétaire*. En d’autres termes, tout programme dont le contenu, le mode de fonctionnement interne, est volontairement maintenu "secret" par son auteur, relève de l’approche *propriétaire*. Pour défendre ses intérêts (commerciaux ou autres), le développeur d’une *solution propriétaire* interdira (souvent) la libre redistribution de son produit et veillera (toujours) à ce qu’il constitue une boîte noire pour ses utilisateurs. En conséquence, ces derniers se voient privés de la possibilité d’adapter le logiciel à leurs besoins, d’en corriger les éventuels bogues ou d’y apporter des améliorations.

## Le domaine public

Dans le *domaine public*, la notion de propriété disparaît. Le logiciel n’appartient à personne. Il appartient donc à tout le monde ! Tout y est permis, même de construire une offre 100 % propriétaire sur la base d’un code issu du domaine public.

## Logiciel libre

« Un logiciel libre est un logiciel dont l’utilisation, l’étude, la modification et la duplication en vue de sa diffusion sont permises, techniquement et légalement. Ceci afin de garantir certaines libertés induites, dont le contrôle du programme par l’utilisateur et la possibilité de partage entre individus » [52].

D’autres encore le définissent comme « un logiciel qui est fourni avec l’autorisation pour quiconque de l’utiliser, de le copier et de le distribuer, soit sous une forme conforme à l’original, soit avec des modifications, ou encore gratuitement ou contre un certain montant. Ceci signifie en particulier que son code source doit être disponible » .

Afin de se démarquer de l’offre propriétaire, les développeurs de logiciels libres optèrent pour l’appellation *Free Software* *free* étant cette fois associé à la notion de *liberté* (et non de gratuité).

L’ambiguïté contenue dans le terme *free* pouvait laisser supposer qu’un logiciel de type *Free Software* est forcément gratuit. Or il n’en est rien. Le support physique de distribution, l’assistance technique, la formation ou encore l’adaptation du produit peuvent être soumis à rétribution.

L’adoption du terme *Open Source* tente (entre autre) de résoudre ce problème de libre interprétation. Il se veut le vecteur des notions de transparence et d’ouverture chères au monde des logiciels libres.

### 1.3.1.2 Historique sur les logiciels libres

La tradition du logiciel libre remonte à l’utilisation de l’informatique en milieu universitaire. En effet, les logiciels développés en milieu universitaire étaient fréquemment échangés et améliorés

entre différents laboratoires de recherche. Mais à la faveur de la libéralisation des marchés s'est vite instaurée l'industrie des logiciels commerciaux.

Cependant, si la pensée libérale soutient qu'il n'y a pas de solution de rechange à l'économie de marché, cela n'est pas aussi évident dans le domaine de l'industrie des logiciels. Richard Stallman, professeur à l'Institut de technologie du Massachusetts (MIT) fut sans doute le premier à prendre conscience de cette alternative au début des années 80. En effet, après s'être vu interdit l'accès au code source d'un programme il décida rapidement de créer la Free Software Foundation (FSF), c'est-à-dire la Fondation pour le logiciel libre. Les premiers travaux de cette fondation permirent notamment d'élaborer une licence parfaitement adaptée au logiciel libre afin de lever l'ambiguïté du terme "free" qui en anglais peut signifier libre ou gratuit et d'initialiser en 1984 le projet GNU (prononcer gnou) qui est un acronyme récuratif qui signifie GNU is Not Unix. Richard Stallman a été rejoint plus tard par les étudiants américains Andrew S. Tannenbaum et finlandais Linus Torvalds. En 1989, Andrew S. Tannenbaum ayant trouvé peu convenable le système proposé sur son ordinateur mit au point en 1990 son propre système à vocation éducative qu'il appela Minix.

En 1991, Linus Torvalds alors étudiant à l'université d'Helsinki essaie le système Minix. L'ayant trouvé trop limité pour ses travaux, il décide d'écrire lui-même, en se fondant sur le système d'exploitation Minix, un système d'exploitation qui le satisfasse. La première version autonome du système d'exploitation Linux fut ainsi lancée le 5 octobre 1991.

Linus Torvalds réalise la nécessité de répartir la tâche de développement entre des collaborateurs et d'étendre le projet à un cercle le plus large possible. Il lança alors un appel resté célèbre sur le forum comp.os.minix et décide de mettre le code source sous licence GPL de telle sorte que tout le monde peut alors participer au développement de Linux : la voie venait d'être ouverte pour la mise en œuvre de l'industrie des logiciels libres.

### 1.3.1.3 Le concept de logiciel libre

La majorité des logiciels vendus dans le commerce par les éditeurs comme Microsoft, Apple, ou Oracle, sont distribués en version « exécutable », alors que les logiciels libres sont fournis avec leur « code source ».

Source, exécutable ? Une petite explication de ces termes s'impose pour éclairer un peu plus sur les principales différences entre logiciels propriétaires et logiciels libres quant au fond. On peut considérer le code source d'un logiciel comme la partition de musique de celui-ci, et le code exécutable comme la même musique en version enregistrée. Une partition peut être lue par n'importe qui, modifiée par n'importe qui souhaitant étoffer ou améliorer la mélodie. Mais par contre, avec une version gravée, il n'est possible que d'écouter le résultat obtenu sans pouvoir modifier quoi que ce soit. Le passage d'une version à l'autre s'opère grâce à une « compilation » qui traduit le code source (des lignes écrites en usant des langages de programmation maîtrisés par les programmeurs) en code exécutable (suite binaire uniquement compréhensible par l'ordinateur). Donc, avant d'arriver dans le commerce ou dans l'entreprise, une version en code source d'un logiciel est toujours construite par l'éditeur. Les logiciels libres sont livrés directement sous cette forme (à charge pour les utilisateurs de procéder eux-mêmes à la compilation ou de la faire réaliser par des intermédiaires), tandis que les éditeurs de logiciels propriétaires vendent uniquement le code « exécutable », le reste étant considéré comme secret industriel et non divulgué. Se définissant par opposition au logiciel « propriétaire », le logiciel « libre » a pour but de préserver la liberté des utilisateurs d'exécuter, de distribuer, d'étudier, de modifier et d'améliorer le logiciel au travers de son code source, l'ensemble des lignes de code qui, une

fois compilées en langage machine, donneront le logiciel fini. Ces logiciels sont d'ailleurs généralement distribués sous forme de lignes de code que l'utilisateur configurera selon ses attentes puis compilera au moment de l'installation.

Il existe quatre niveaux de libertés que chaque logiciel qui se dit libre doit respecter :

- la liberté d'exécuter le programme pour tous les usages, ce qui est le degré minimum de liberté que l'on peut attendre d'un logiciel ;
- la liberté d'étudier comment le programme fonctionne et de l'adapter à ses besoins ;
- la liberté de redistribuer des copies sans aucune contrainte ;
- la liberté d'améliorer le programme et de diffuser ses améliorations publiquement de telle sorte que la communauté tout entière en bénéficie.

La liberté de modifier et de distribuer des logiciels libres sans payer de droits d'auteur est fondée sur une licence. Celle-ci représente un contrat de type copyright (comme n'importe quelle licence propriétaire) mais qui confère ses avantages ou plutôt ses droits à l'utilisateur.

Les licences les plus connues sont le contrat GPL et la licence de type BSD (Berkeley).

Pour comprendre la légitimité du logiciel libre, il est nécessaire d'étudier les différentes licences sur lesquelles se base l'économie du logiciel dans son ensemble.

#### 1.3.1.4 Les licences d'utilisation

Dans le cadre juridique actuel, la licence sous laquelle un programme est distribué définit exactement les droits détenus par ses utilisateurs. Par exemple, dans la plupart des programmes propriétaires, la licence retire les droits de copier, de modifier, de prêter, de louer, d'utiliser sur plusieurs machines, etc. En fait, les licences indiquent habituellement que le propriétaire du programme est la compagnie qui l'édite, et que celle-ci ne vend que les droits restrictifs d'utilisation du logiciel.

### Les licences libres

Dans le monde du logiciel libre, la licence sous laquelle un programme est distribué a également une importance primordiale. Il est à noter que même dans le cas des licences libres, l'auteur du programme reste propriétaire. Seule la licence est concédée. Habituellement, les conditions indiquées dans les licences du logiciel libre sont le résultat d'un compromis entre plusieurs objectifs qui sont, dans un certain sens, contradictoires. Parmi ces conditions, on peut citer :

- garantir quelques libertés de base (redistribution, modification, utilisation) aux utilisateurs ;
- assurer quelques conditions imposées par les auteurs (citation de l'auteur dans les travaux dérivés, par exemple) ;
- garantir que les travaux dérivés seront également des logiciels libres.

Les auteurs peuvent choisir de protéger leur logiciel avec différentes licences selon le degré avec lequel ils veulent accomplir ces buts, et les détails qu'ils veulent assurer. En fait, les auteurs peuvent (s'ils le désirent) distribuer leur logiciel sous différentes licences par différents canaux (et différents prix). Donc, l'auteur d'un programme choisit habituellement très soigneusement la licence sous laquelle il sera distribué. Et les utilisateurs, particulièrement ceux qui redistribuent ou modifient le logiciel, doivent soigneusement étudier sa licence.

Heureusement, bien que chaque auteur puisse utiliser une licence différente pour ses programmes, le fait est que presque tous les logiciels libres utilisent une des licences habituelles (GPL, LGPL, artistique, BSD-like, MPL, etc.), parfois avec de légères variations.

Pour encore simplifier les choses, des organismes apparus récemment ont défini quelles étaient les caractéristiques qu'une licence de logiciel devait posséder si elle voulait être qualifiée de licence pour logiciels libres.

Le GNU Project donne une définition du logiciel libre qui fait souvent foi pour départager les vraies et les fausses licences libres. Les différences entre les licences libres se situent habituellement dans l'importance que l'auteur donne aux points suivants :

- protection du caractère ouvert : quelques licences exigent en effet que celui qui redistribue maintienne la même licence que celle d'origine, et par conséquent, les droits du destinataire sont les mêmes (et cela que le logiciel soit reçu directement de l'auteur ou de n'importe quelle partie intermédiaire).
- protection des droits moraux : dans beaucoup de pays, la législation protège certains droits moraux, comme la reconnaissance de l'auteur. Certaines licences assurent également la protection à ce niveau, ce qui permet de contourner certaines législations locales.
- protection de certains droits de propriété : dans certains cas, l'auteur original (celui qui a initialement créé le logiciel) a quelques droits supplémentaires, qui dans un certain sens sont un genre de droits de propriété.
- compatibilité avec des licences propriétaires : quelques licences sont conçues de sorte qu'elles soient complètement incompatibles avec le logiciel propriétaire. Par exemple, elles peuvent interdire de redistribuer n'importe quel logiciel étant le résultat d'un mélange de logiciel couvert par la licence d'un logiciel propriétaire.
- compatibilité avec d'autres licences libres : quelques licences libres ne sont pas compatibles l'une l'autre, parce que les conditions de l'une ne peuvent pas être remplies si les conditions imposées par l'autre sont satisfaites. Dans ce cas-ci, il est habituellement impossible de mélanger des parties de code provenant des logiciels dont les licences sont incompatibles.

Voici les licences les plus connues et utilisées dont s'inspire la majorité des autres licences :

- BSD (Berkeley Software Distribution). La licence BSD couvre, entre autres, les versions des distributions de types BSD (Berkeley Software Distribution). C'est un bon exemple d'une licence « permissive » qui n'impose presque aucune limite à ce que l'utilisateur peut faire avec le logiciel, y compris faire payer les clients pour des distributions binaires, sans l'obligation d'inclure le code source. En résumé, ceux qui redistribuent sont autorisés à faire presque n'importe quoi avec le logiciel, y compris l'utiliser pour des logiciels propriétaires.

Les auteurs veulent seulement que leur travail soit identifié. Dans un certain sens, cette restriction assure aux auteurs une certaine quantité de marketing gratuit (car il ne coûte pas d'argent). Il est important de noter que ce genre de licence n'inclut aucune restriction ayant pour but de garantir que les améliorations futures apportées au logiciel restent en source ouverte. Cette licence se trouve en annexe.

- GPL (GNU General Public License). La GPL a été soigneusement conçu pour favoriser la production d'un logiciel plus libre, et cela grâce à l'interdiction explicite de certaines actions sur le logiciel pouvant mener à l'intégration du logiciel en GPL dans des programmes propriétaires. La licence GPL est basée sur la législation internationale à propos du copyright, qui assure son applicabilité. Les caractéristiques principales de la GPL sont les suivantes :
  - elle permet la redistribution binaire, mais seulement si la disponibilité de code source est également garantie ;
  - elle permet la redistribution des sources (et l'impose en cas de distribution binaire) ;

- elle permet la modification sans restriction (si le travail dérivé est également couvert par GPL);
- elle permet aussi l'intégration complète d'un autre logiciel mais cela uniquement s'il est également couvert par la GPL.

En résumé on a :

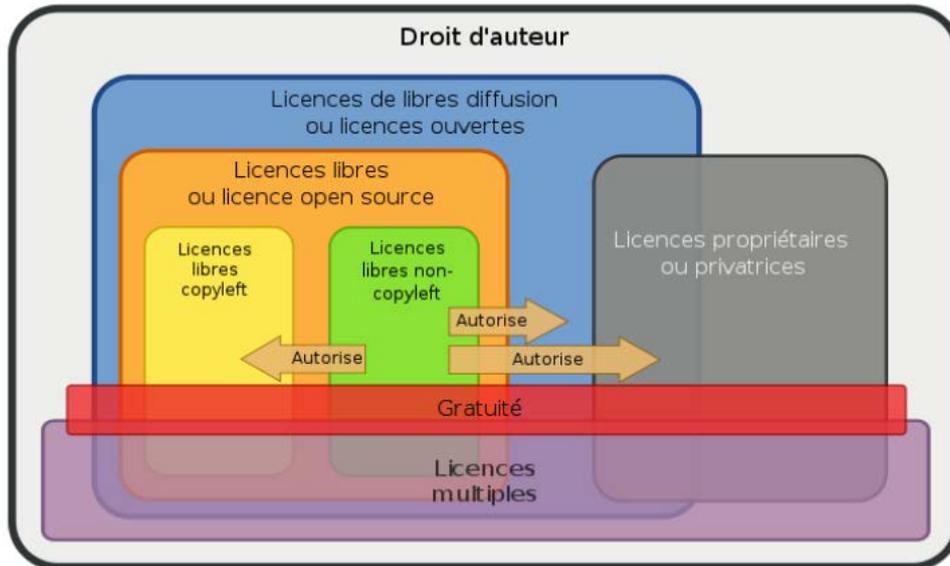


FIGURE II.1.1 – Classification des licences

### 1.3.2 Modèle organisationnel et technologique des logiciels libres

Une des raisons qui expliquent le succès du logiciel libre est relatif à son organisation très structurée. La coordination entre les différents développeurs est le plus souvent assurée par un groupe de personnes, que nous appellerons les « développeurs clefs » et qui supervisent la plus grande partie des développements du logiciel concerné. Les développeurs clefs sont sélectionnés par les autres membres du noyau de développement (c'est le cas d'Apache) ou par le développeur initial du projet (c'est le cas de Linux), mais il y a toujours cooptation à l'instar du milieu scientifique. Ce noyau reste ouvert aux nouvelles idées, aux nouveaux talents : on peut confier à de nouveaux venus la responsabilité de parties nouvelles du logiciel, souvent parce que ce sont eux qui ont commencé à les développer. À la périphérie du noyau de développement, nombreux utilisateurs, souvent eux-mêmes développeurs participent au processus, en donnant des idées, en proposant des améliorations ou en détectant des erreurs.

L'existence de ce groupe élargi est importante pour le succès et l'évolution du logiciel mais il n'a que peu d'impact direct sur le volume de logiciel développé. La deuxième condition nécessaire au bon fonctionnement d'un projet libre est plus technique. Elle porte sur l'architecture du logiciel, qui doit être construit en modules. Les frontières du logiciel de base sont définies, les interfaces de communication entre ce logiciel de base et les logiciels satellites, les « composants », sont rendues publiques. Les développeurs sont alors encouragés à produire des « composants » qui s'ajoutent au cœur du logiciel pour offrir de nouvelles fonctionnalités, indépendamment les uns des autres. La construction modulaire d'un logiciel permet à des groupes de développeurs de travailler relativement

indépendamment. Ainsi les équipes de programmation restent de petite taille, ce qui facilite la coordination et la prise de décisions. Par ailleurs cette organisation modulaire facilite les innovations et l'extension des fonctionnalités des programmes, qui peuvent se faire par des ajouts successifs de modules. Ainsi est-on assuré d'une certaine stabilité de l'évolution du noyau et de l'interopérabilité entre les différents modules. En cela, l'organisation du logiciel libre traite le problème de l'interopérabilité de façon originale : elle est garantie par une organisation, qui contrôle et fait évoluer les standards d'échange de façon publique, en phase avec l'évolution du logiciel lui-même. Elle génère un processus dynamique de standardisation et d'évolution du standard, qui conserve un caractère public pur. Plus largement, ce double caractère d'ouverture et de contrôle permet de construire des normes de définition (David, 1987), référentiel commun de la communauté des développeurs, dont le respect est garant de l'intérêt général de la collectivité (Benezech, 1995). Sur le plan cognitif, on peut interpréter sa dynamique d'interaction comme une communauté épistémique comprise comme le partage par un groupe d'un objectif commun de création de connaissance et d'un environnement d'interaction permettant à chacun des individus de s'insérer en phase avec cette dynamique collective (Cohendet, Creplet et Dupouët, 2001).

Parce qu'elle permet aux utilisateurs d'étudier le code et de tester les logiciels, mais aussi parce que les règles de développement d'un projet libre sont souvent très contraignantes, parce qu'enfin chaque proposition doit être acceptée par le noyau de contrôle, l'organisation du libre produit ainsi des logiciels performants. Leurs qualités technico-économiques (stabilité, respect des standards couplé à une évolutivité des solutions) sont aujourd'hui reconnues et expliquent leur adoption par une population élargie d'utilisateurs, bien au-delà de la seule communauté des individus impliqués directement ou indirectement dans des tâches de développement. Là sont aussi les raisons pour lesquelles plusieurs entreprises du secteur propriétaire ont décidé de basculer dans le monde du libre pour tout ou partie de leurs produits. Ainsi Sun est aujourd'hui le deuxième producteur de logiciel libre, après la FSF, mais avant les universités américaines. Par ailleurs de nombreuses entreprises ont vu le jour, souvent à l'initiative d'individus fortement impliqués dans le monde du libre, en vue d'accompagner le développement d'un logiciel donné par une offre de services associés (RedHatou Mandrake avec Linux, Covalent Technology avec Apache, Helixcode avecGnome, une interface graphique de Linux, etc.).

### 1.3.2.1 Modèle économique

#### **Le logiciel libre, une opportunité pour la création de richesses**

Le logiciel libre favorise la création d'emplois. Ils sont créateurs d'emplois décentralisés de service et peuvent entraîner la constitution de plusieurs sociétés spécialisées dans la maintenance ou le développement personnalisé encourageant ainsi une production complémentaire ou concurrente. Le logiciel libre adapté sur place est créateur de plus d'emplois locaux que le logiciel propriétaire importé qui ne peut être mis à jour que par son éditeur. Il en découle du coup une concurrence plus équitable et une meilleure maîtrise des technologies de l'information et de la communication par les usagers locaux. Le logiciel libre est économique et il favorise la mise en œuvre de projets.

La lutte contre la pauvreté ne réside pas seulement dans le fait d'œuvrer pour disposer de plus de ressources. Mais c'est aussi savoir orienter ses dépenses, les minimiser afin de réinvestir les économies réalisées. L'absence ou le faible niveau des coûts de licences liés à l'utilisation des logiciels libres génère une marge supplémentaire qui peut être réinvestie dans du matériel. Cette économie peut également servir à couvrir les frais d'adaptation du logiciel libre aux besoins de l'entreprise et à former les utilisateurs ou mieux à financer des projets de développements utiles pour la communauté.

### **Le logiciel libre : une qualité technologique sûre découlant de la « loi de Linus »**

Le logiciel libre est rentable, efficace et fiable. Sa rentabilité est la conséquence directe de l'ouverture de son code source. En effet, il y a un énorme gain de temps et de ressources lorsqu'une nouvelle fonctionnalité se prépare à être ajoutée au logiciel liée à la réutilisabilité de tous les codes existants.

L'ouverture des codes sources et la possibilité de les modifier dont bénéficient les logiciels libres permettent la contribution de plusieurs volontaires dispersés dans le monde et qui ne sont pas tous forcément en rapport. Cela permet l'exploration de différentes solutions techniques dont la meilleure est généralement retenue par sélection « naturelle ».

### **Le logiciel libre une garantie de fonctionnement, de liberté et de pérennité**

Alors que la correction d'un bogue au niveau d'un logiciel propriétaire procède souvent par l'achat de sa version suivante ou celui de son correctif, la correction d'un bogue au niveau d'un logiciel libre peut être immédiate, donc plus rapide.

Il est quasiment impossible d'introduire dans un logiciel libre des fonctionnalités cachées dans le but de restreindre les libertés individuelles ou de collecter des informations sur les utilisateurs. Les logiciels libres garantissent donc la libre circulation des informations et la liberté d'expression de chacun quel que soit son équipement.

La disponibilité des codes sources garantit aux utilisateurs la pérennité des logiciels qu'ils utilisent.

L'abandon du support du logiciel ou des mises à jour par la société éditrice ne risque pas d'entraîner la « mort » du logiciel puisqu'il existera toujours une chance de trouver quelqu'un prêt à travailler pour l'évolution du logiciel.

Le logiciel libre est au cœur de l'innovation et des gains de productivité dans la plupart des secteurs économiques dans le monde ces dix dernières années. Les expériences des pays pionniers dans ce domaine (France, Royaume-Uni, et bien d'autres) font état d'économie pouvant aller jusqu'à 90 %.

En somme, les logiciels libres sont pourvus d'énormes qualités intrinsèques qui font d'eux un modèle économique viable notamment pour les pays africains encore à la quête de leur essor technologique. Il importe maintenant d'identifier les stratégies à mettre en place pour favoriser le développement de l'industrie du logiciel libre en Afrique. Et cela ne pourra se faire sans impliquer sérieusement le secteur de l'éducation.

### **1.3.3 Les logiciels libres, une solution pour l'enseignement supérieur en Afrique**

Beaucoup de pays développés aujourd'hui se tournent vers les logiciels libres du fait de leurs souplesses, sécurités, efficacités et surtout du fait qu'ils sont moins coûteux pour les administrations. Pour cela, des lois sont votées en faveur de l'utilisation de ces logiciels. C'est d'ailleurs l'exemple du projet de loi voté récemment en France accordant une priorité aux logiciels libres pour l'ensemble des services et des ressources proposés par et dans l'enseignement public .

Si les pays développés optent pour les logiciels libres dans leurs système éducatifs et dans bien d'autres domaines, pourquoi pas les pays africains ? Devons-nous attendre qu'ils prennent une avance considérable avant de nous décider à suivre le pas ? Quels avantages auront les pays africains à adopter les logiciels libres dans le domaine de l'enseignement supérieur ?

### 1.3.3.1 Les logiciels libres source d'une bonne formation

Le développement d'une solution informatique ou d'un logiciel informatique nécessite souvent une bonne compréhension des principes de la programmation. Pour pouvoir bien développer ou programmer, il faut en plus de connaître le langage de programmation être capable de lire et de comprendre les programmes informatiques. Cela dit, il est tout à fait impossible d'avoir cette chance avec les logiciels propriétaires. Seuls les logiciels libres offrent cette chance aux apprentis programmeurs et à ceux qui s'intéressent au domaine de la production des logiciels. C'est d'ailleurs dans ce sens que Richard Stallman dit « *le code source d'un logiciel libre doit être ouvert et accessible à tout le monde, on offre là la possibilité à tout un chacun en particulier aux élèves et aux étudiants d'apprendre comment les programmes fonctionnent. À l'adolescence, certains d'entre eux veulent tout apprendre au sujet de leur ordinateur et de ses logiciels. Ils sont extrêmement curieux de lire le code source des programmes qu'ils utilisent tous les jours. Pour apprendre à écrire du bon code, ils ont besoin de lire beaucoup de code et d'en écrire beaucoup. Ils ont besoin de lire et de comprendre de vrais programmes que les gens utilisent réellement. Seul le logiciel libre leur en donne la permission.* »

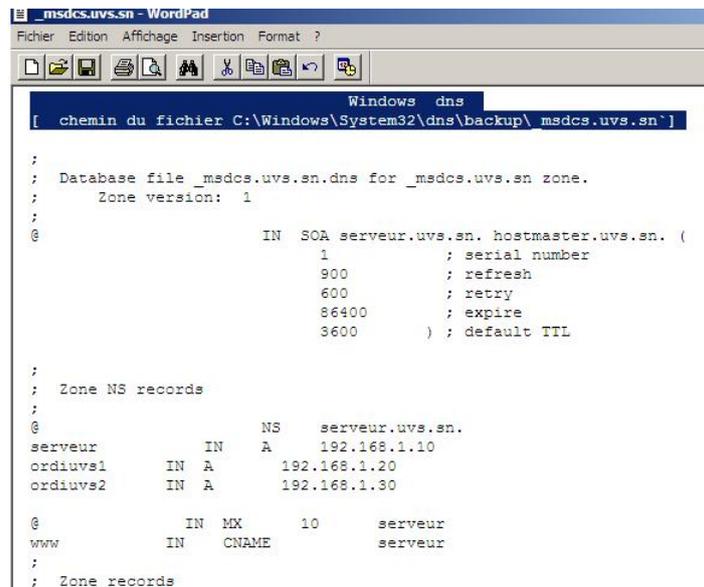
Si nous prenons par exemple les fichiers de configuration du serveur de résolution de nom sous Windows et sous Linux, on remarque qu'ils ont pratiquement la même structure :

```

GNU nano 2.2.6 File: uvs.sn
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      server.uvs.sn. root.server.uvs.sn. (
                        2          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      NS       server.uvs.sn.
@         IN      MX       1      server.uvs.sn.
www      IN      A         192.168.1.169
www      IN      CNAME    server.uvs.sn.

```

FIGURE II.1.2 – Capture des fichiers de résolution directe du serveur DNS sous Debian



```

_msdcs.uvs.sn - WordPad
Fichier Edition Affichage Insertion Format ?
[ chemin du fichier C:\Windows\System32\dns\backup\_msdcs.uvs.sn ]

;
; Database file _msdcs.uvs.sn.dns for _msdcs.uvs.sn zone.
; Zone version: 1
;
@           IN      SOA  serveur.uvs.sn. hostmaster.uvs.sn. (
                        1           ; serial number
                        900          ; refresh
                        600          ; retry
                        86400        ; expire
                        3600         ) ; default TTL

;
; Zone NS records
;
@           IN      NS   serveur.uvs.sn.
serveur    IN      A    192.168.1.10
ordiuvs1   IN      A    192.168.1.20
ordiuvs2   IN      A    192.168.1.30

@           IN      MX   10     serveur
www        IN      CNAME  serveur

;
; Zone records

```

FIGURE II.1.3 – Capture des fichiers de résolution directe du serveur DNS sous Windows Server2008

Sous Windows, le déploiement d'un serveur DNS se fait par l'intermédiaire d'une interface graphique qui cache en quelque sorte ce qui se passe en arrière-plan à l'utilisateur. C'est-à-dire le renseignement qui est fait dans le fichier de configuration. Alors qu'avec Linux, c'est l'apprenant même qui doit introduire les différents paramètres dans les fichiers de configuration. Cela offre deux avantages :

La compréhension des différents paramètres qu'on est en train de renseigner et la connaissance de la structure du fichier dans lequel on renseigne les paramètres. Pour un étudiant qui fait de la programmation système c'est un plus.

Un autre avantage qu'offre les logiciels libres aux programmeurs africains, c'est la possibilité de pouvoir développer des Logiciels par les africains pour les africains ou d'adapter certains logiciels aux réalités africaines. «*Les bons programmeurs savent quoi écrire. Les grands programmeurs savent quoi réécrire (et réutiliser)*» [53] C'est le cas par exemple au Burkina Faso du logiciel comme Gestcarpa une production locale qui sert à la Gestion de la Caisse Autonome des ressources Pécuniaires des Avocats ou encore d'Akwaaba utilisé en Côte d'Ivoire pour la gérance de la connexion de l'administration et des entreprises.

Souvent, on achète des logiciels qui ne sont pas adaptés à nos besoins. C'est-à-dire que parfois on peut se retrouver avec un logiciel qui dispose de plusieurs fonctionnalités dont on utilise que le tiers. C'est par exemple le cas de la suite bureautique Microsoft office qui se compose de Word, Excel, PowerPoint, Outlook, Publisher, OneNote, Access dont on utilise souvent que Word, Excel et PowerPoint alors qu'on a payé pour toutes les licences. Pour un étudiant avisé, plus tard en entreprise il pourrait aider son entreprise à économiser de l'argent pour son entreprise.

Les logiciels libres pourraient aussi aider les universités et centres de formations africains dans leur politique de la promotion de l'enseignement par les TICs. Il existe de nombreux logiciels dans plusieurs domaines qui pourraient y contribuer.

Dans le domaine de l'enseignement à distance : on peut citer les plateformes Moodle, Claroline et Ganesha.

Dans le domaine des télécommunications et de l'informatique : on peut citer les logiciels comme GNS3, Astérisk, Open-BTS, Open-BSC, IMSCore, Kannel, OpenLDAP, FreeRadius, Mobicents, Na-

gios, Wireshark, Apache.

Dans le domaine des mathématiques : Scilab, GNU Octave.

Dans le domaine de traitement d'images et de modélisation 3D : GIMP, Blender. Voici d'ailleurs résumé dans le tableau ci-dessous quelques grands logiciels libres qui ont fait leurs preuves aussi bien dans le domaine de l'éducation que professionnel :

TABLEAU II.1.2 – Quelques logiciels libres dans le domaine de l'éducation

Projet	Description	Indicateurs	Licences et organisations
<b>Linux</b>	Système d'exploitation équivalent à Windows ou Mac	Près de 90% pour les systèmes d'exploitation pour serveurs	Distribué sous licence GPL . Développé sous le leadership de Linus Torvalds par plus de 1000 développeurs
<b>Apache</b>	Serveur Web	Apache a une part importante du marché des serveurs web, essentiels à la gestion de l'infrastructure des utilisateurs d'Internet des entreprises	Licence de type BSD. Développement contrôlé par un groupe de développeurs regroupés dans une association, l'Apache Foundation
<b>Sendmail</b>	Logiciel de gestion des courriers électroniques	Un des premiers programmes pour le routage du courrier électronique. Il est le plus utilisé dans ce domaine	Sous licence BSD, développé par Eric Allman. Eric Allman a créé sa propre compagnie pour vendre une version propriétaire proposant plus de fonctionnalités
<b>Mysql</b>	Système de gestion de bases de données	Il est distribué sous licence GPL et propriétaire. Il fait partie des SGBD les plus utilisés au monde autant par le grand public que par les professionnels, en concurrence avec Oracle, Informix et Microsoft SQL server	Licences de type GPL et propriétaire

Un autre problème que pourrait régler les Logiciels libres, c'est le problème d'augmentation des flux d'étudiants auquel les universités sont confrontées. On peut d'ailleurs lire dans le rapport finale de la CNAES (Concertation Nationale sur l'Avenir de l'Enseignement Supérieur au Sénégal) qui s'est tenu du 06 au 09 avril 2013 : « *Le contexte mondial de l'enseignement supérieur est marqué par un accroissement considérable des effectifs d'étudiants au niveau des pays et des flux internationaux d'étudiants.* » [54]

Les logiciels libres par l'intermédiaire des plateformes de formations à distance comme Moodle combinés avec des outils libres de communication et de conférence libres comme BigBlueButton peuvent aider à désengorger les universités et centres de formations africains.

Avec ces logiciels, l'étudiant n'a besoin que de son ordinateur et d'une bonne connexion internet pour travailler. Il n'est plus limité à une salle de travaux pratiques. En plus de ça, il a l'opportunité de développer des compétences techniques dans plusieurs domaines et surtout dans la résolution des pannes systèmes. Ceci est certainement une plus-value lors de la recherche d'emploi. Les logiciels libres

offrent aux étudiants une certaine ouverture d’esprit pour innover et faire avancer la société.

Cela montre que les logiciels libres occupent une place importante dans le domaine des études et de la recherche. Ils ont une légitimité et ils devraient être reconnus et adoptés par l’enseignement supérieur. Ce sont des logiciels basés sur des normes et des standards internationaux. En plus de ça, ils utilisent les mêmes protocoles que les logiciels propriétaires et ils sont interopérables. Nous estimons que dans le domaine de l’éducation et de la recherche, la manière dont tel logiciel ou tel autre est conçu ne devrait pas être un secret professionnel car le rôle de l’enseignement, c’est quand même de donner la connaissance sans aucune réserve. C’est sur ce point que les logiciels libres sont fascinants, de par l’ouverture de leur code source, ils permettent à tout un chacun s’il est bien outillé de pouvoir lire et comprendre le code source du logiciel qu’il utilise, de le modifier et de l’adapter à ses besoins. *« Le libre permet un transfert spontané de technologies entre le Nord et le Sud. Grâce au libre, n’importe quel développeur africain peut utiliser et adapter le logiciel en fonction de ses besoins. Il permet l’émancipation des utilisateurs, qu’ils soient enseignants, particuliers ou entreprises. »* [52]

Utiliser les logiciels libre dans l’enseignement supérieure en Afrique, c’est préparer aussi la relève de demain. Pour y arriver, les recteurs d’université, les directeurs de centre de formation, les hommes politiques et bien d’autres acteurs doivent être informés.

### 1.3.3.2 Recommandation à l’attention des décideurs

Les universités africaines peuvent désormais arriver avec peu de moyens à bien former. Ceci par l’intermédiaire des logiciels libres. Tout dépendra dorénavant des décideurs, de la solution qu’ils voudront adopter et du rôle qu’ils voudront jouer dans la formation de l’élite africaine de demain. Des recteurs d’universités aux directeurs d’écoles de formation professionnelles, sans oublier les professeurs et les étudiants, une sensibilisation devrait être faite sur l’importance de l’utilisation des logiciels libres. La promotion des logiciels libres en Afrique devrait être une volonté politique bien exprimée.

### 1.3.3.3 Les raisons pour lesquels les décideurs devraient adopter les logiciels libres

Les gouvernements africains doivent s’intéresser aux logiciels libres et formuler en conséquence leurs politiques TIC pour les raisons suivantes :

- La baisse des coûts liés aux technologies de l’information axées sur des logiciels libres et leur haut niveau de sécurité ;
- Le logiciel libre permet de lutter contre la piraterie (les logiciels propriétaires sont très souvent piratés) et d’éviter de faire face à une situation de monopole dans le domaine de l’industrie logicielle ;
- Le logiciel libre “libère” de toute dépendance vis-à-vis du fournisseur et est évolutif.

### 1.3.3.4 Les facteurs qui peuvent bloquer l’utilisation des logiciels libres dans le domaine de l’enseignement en Afrique

Parmi les facteurs contraignants de l’utilisation des logiciels libres il faut désigner du doigt, la prédominance de l’environnement Windows de Microsoft qui est installée sur plus de 90% des ordinateurs personnels, un quasi-monopole qui laisse peu de place pour des solutions alternatives telles que le système d’exploitation Linux dont la présence paraît relativement timide.

Il y a aussi l’appréhension du terme “gratuit” incarné par le logiciel libre qui peut susciter de la part des décideurs quelques réticences quant à l’utilisation des logiciels à code source ouvert.

En effet, la conception selon laquelle ce qui est gratuit, libre n'est pas de meilleure qualité ou est banal est bien ancrée en Afrique. C'est là le premier défi à relever car quoique libres et parfois gratuits, les logiciels à code source ouvert constituent une alternative valable.

Un autre paramètre, c'est l'insuffisance des outils d'aide à la décision : les décideurs ne sont pas assez outillés pour opérer des choix éclairés dans le domaine des nouvelles technologies. Par ailleurs, quand bien même l'option de l'alternative libre aura été adoptée, les décideurs pourraient, craignant les coûts de la migration, hésiter à passer à l'offensive. Mais plus le temps passe et plus la migration sera onéreuse et plus ils seront dépendants des fournisseurs.

Le manque de compétence, les changements dans les habitudes des utilisateurs et la satisfaction du rendement des solutions propriétaires disponibles peuvent être des handicaps au déploiement des Logiciels Libres.

Enfin, il est important qu'une veille technologique soit assurée. En effet, l'un des éléments impressionnants des logiciels libres est que de nombreux produits sont mis à jour quotidiennement et il est parfois difficile de suivre les tout derniers développements. Dans ces conditions, il est donc utile, de s'assurer au préalable que le problème au quel l'on serait confronté n'a pas déjà été résolu par quelqu'un d'autre avant d'engager toute procédure de résolution. Car l'Afrique n'aura aucun intérêt à réinventer la roue. Mais elle aura plutôt du mérite en la perfectionnant davantage et en l'adaptant à ses propres besoins.

### 1.3.4 L'expérience de mise en place d'une université numérique à partir de logiciels libres

Dans le cadre de ce projet, il sera mis en place les technologies suivantes :

- un serveur d'ENT qui jouera le rôle de portail pour l'accès aux différentes ressources de l'université virtuelle : Esup Portal
- une plateforme d'enseignement à distance pour le déroulement des cours : Moodle
- une plateforme de virtualisation afin de pouvoir créer des machines virtuelles à la demande pour les travaux pratiques des étudiants : OpenVZ
- un serveur de fédération d'identités pour gérer les droits d'accès à des ressources mutualisées : Shibboleth
- une plateforme libre de partage et de synchronisation de ressources : Owncloud
- une plateforme de visioconférence sur le Web pouvant être intégrée à Moodle : BigBlueButton

#### 1.3.4.1 Choix des logiciels libres pour la mise en place de l'université

##### ENT (Environnement Numérique de Travail)

Un espace numérique de travail ou environnement numérique de travail ou « bureau virtuel » ou « portail de services » (ENT) est une plateforme de travail collaboratif respectant un cahier des charges réalisé dans le cadre des TICE (Technologies de l'Information et de la Communication pour l'Enseignement) par le ministère français de l'Éducation nationale.

C'est un ensemble intégré de services numériques, organisé, choisi et mis à disposition de la communauté éducative par l'école ou l'établissement scolaire. Il repose sur un dispositif global fournissant à un usager un espace dédié à son activité dans le système éducatif. Il est un point d'entrée unifié pour accéder au système d'information pédagogique de l'école ou de l'établissement. Il travaille s'adresse à

l'ensemble des membres de la communauté éducative d'un établissement (école, collège, lycée, établissement d'enseignement supérieur) ; par exemple élèves, parents d'élèves, étudiants, enseignants, personnels administratifs, techniques et d'encadrement des établissements d'enseignement, et le cas échéant, aux partenaires du système éducatif.

En termes d'architecture, une solution ENT s'organise autour de trois ensembles de services :

- des **services applicatifs** (par exemple : cahier de textes, agenda, média center, notes, bulletins, etc.) ;
- des **services socle** qui servent de support aux services applicatifs (par exemple : annuaire, gestion des identités et des accès, présentation et personnalisation des services offerts, etc.) ;
- des services réseaux qui regroupent les services autour d'infrastructures (réseaux d'écoles ou d'établissements, Internet...) et des fonctions de sécurité (pare-feu, antivirus...).

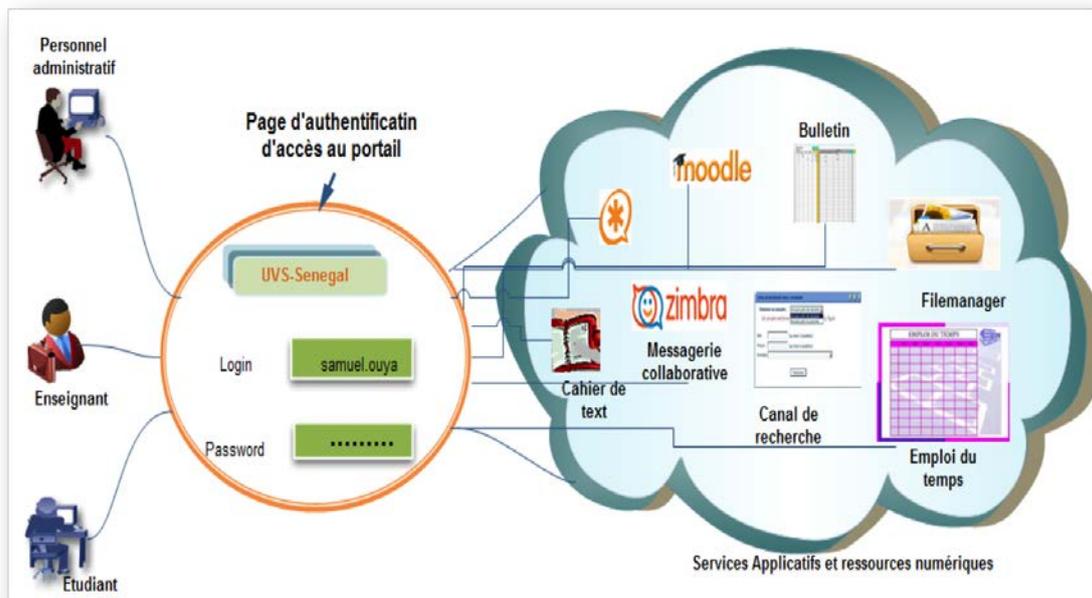


FIGURE II.1.4 – Environnement numérique de travail

## Esup-portal

La solution libre proposée ici est Esup-portal

Le consortium **ESUP-Portail** est porteur du projet "**Espace Numérique de Travail**" (ENT) avec un développement communautaire réalisé en open-source ; ce projet vise à mettre à disposition de la communauté d'enseignement supérieur une solution d'**espace numérique de travail modulaire** permettant l'accès à différents services, sources d'informations et ressources numériques.

Cet environnement est accessible à tous les acteurs universitaires, c'est à dire les étudiants, les enseignants-chercheurs et les personnels administratifs et techniques, ceci quel que soit leur emplacement géographique et le matériel mis à leur disposition.

Ce dispositif facilite en particulier l'activité de tous les usagers de l'université en mettant en œuvre un **point d'accès unique** et **structuré** pour regrouper l'ensemble des services, ressources et informations de l'établissement. Il permet, pour chaque utilisateur, la personnalisation de son environnement de travail, assure un **accès sécurisé** aux services et offre une **authentification unique** s'appuyant entre autres sur les annuaires LDAP et les certificats.

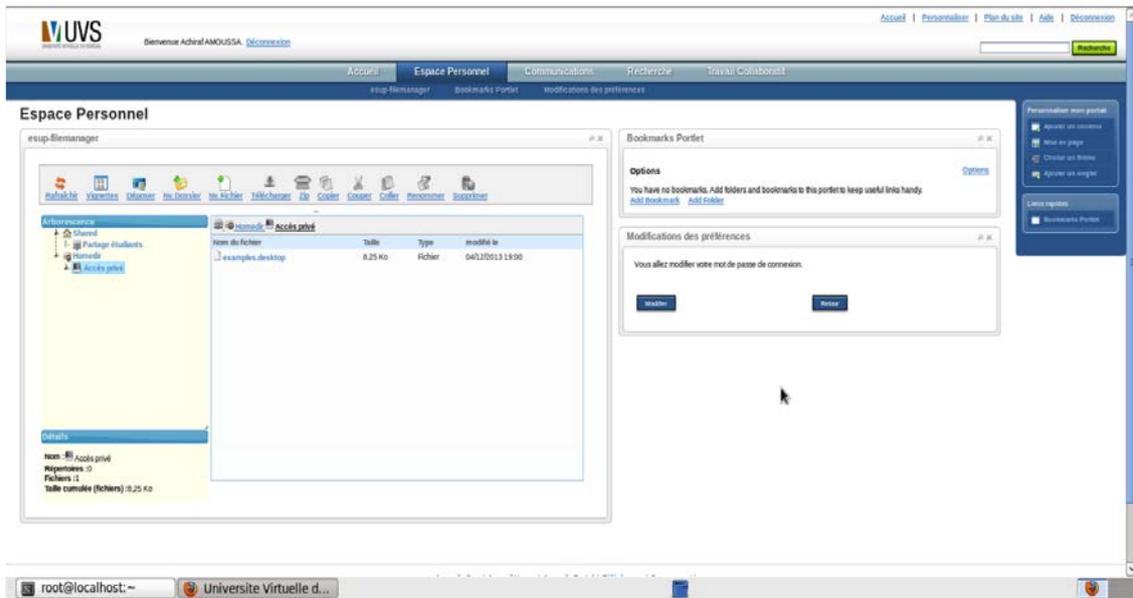


FIGURE II.1.5 – Espace personnel d’un utilisateur connecté à Esup Portal

## Plateforme d’enseignement à distance

Le logiciel libre proposé est Moodle car il réunit toutes les fonctionnalités dont une plateforme d’enseignement à distance doit disposer, en plus c’est l’outil libre le plus utilisé dans le domaine de l’enseignement à distance.



FIGURE II.1.6 – Interface de Moodle montrant l’ensemble des filières de l’UVS en 2014

## Moodle

Moodle est une plateforme d’apprentissage en ligne servant à créer des communautés d’apprenants

autour de contenus et d'activités pédagogiques. Le terme Moodle est l'acronyme de Modular Object-Oriented Dynamic Learning Environment, mais il veut aussi dire « flâner » en anglais. Dotée d'un système de gestion de contenu - (SGC) performant, Moodle a aussi des fonctions pédagogiques ou communicatives lui permettant de créer un environnement d'apprentissage en ligne. Ces fonctionnalités permettent de créer des interactions entre pédagogues, apprenants et ressources pédagogiques formant ainsi un réseau et parfois même une véritable communauté autour d'un thème choisi par les membres de la plateforme (apprentissage d'un logiciel comme utilisation de la plateforme). C'est une particularité que l'on retrouve peu dans les autres plateformes étudiées. Moodle fait partie des systèmes de e-formation qui sont appelés dispositifs de « Formation Ouverte et à Distance » (FOAD), pour favoriser un cadre de formation socioconstructiviste. Ce courant de pensée affirme que les gens construisent activement leurs nouvelles connaissances en interagissant avec leur voisinage. Tout ce que vous lisez, voyez, entendez, ressentez et touchez est comparé à vos connaissances antérieures et si cela est viable dans votre monde mental, cela pourra former une nouvelle connaissance qui vous appartiendra. La connaissance est renforcée si vous pouvez l'utiliser avec succès dans un environnement plus large. Cela met l'accent sur le fait qu'il n'y a pas seulement un transfert d'information d'un cerveau à un autre, mais que tout est conditionné par l'interprétation et l'interaction avec un groupe social.

### **Fédération d'identité**

La fédération d'identités permet de simplifier et sécuriser l'accès à des sites web dont l'accès est contrôlé : plate-forme d'enseignement à distance, portail documentaire, application métier ... L'accès à ces ressources peut être rendu transparent dans le portail de l'organisme, et permet d'étendre la fonction de Single Sign-On interne, pour des ressources extérieures. Cela permettra par exemple à un étudiant d'une université A d'avoir accès à la bibliothèque d'une université B si les deux universités sont en collaboration. Elle répond bien aux besoins de mutualisation entre organismes, aux problématiques de nomadisme et facilite le respect de la loi « Informatique et libertés ».

Le logiciel libre proposé est shibboleth.

### **Shibboleth**

Shibboleth est une suite de logiciels développée à l'origine par le consortium Internet2 et sous la responsabilité du consortium Shibboleth à présent. Cette suite fournit une solution complète de fédération d'identités. Le principe de la fédération d'identités est de déléguer l'authentification web des utilisateurs à un service d'authentification dans l'organisme d'origine de l'utilisateur. Le protocole utilisé est SAML 2, utilisé également par d'autres solutions logicielles de ce type.

Les trois briques Shibboleth sont le fournisseur de services, le fournisseur d'identités et le service de découverte (ou DS/WAYF). Le fournisseur de services est un module d'authentification pour le serveur Web ; il permet de :

- déléguer l'authentification des utilisateurs à un fournisseur d'identités ;
- transmettre le profil utilisateur ;
- gérer le contrôle d'accès de manière optionnelle.

Le fournisseur d'identités est une application Java (servlet) ; il permet de gérer l'authentification des utilisateurs, en réponse à la requête d'un fournisseur de services. L'authentification peut être déléguée à un serveur SSO-CAS (Central Authentication Service) : l'authentification se fait par login/mot de passe ou certificat électronique ou encore propose les deux ; les attributs de l'utilisateur sont extraits

d'un annuaire (AD ou LDAP), d'une base SQL ou bien calculés, puis propagés au fournisseur de services.

Le service de découverte (DS) permet à un utilisateur de sélectionner son organisme de rattachement, c'est-à-dire celui auprès duquel il pourra s'authentifier. Le service de découverte propose un menu déroulant à l'utilisateur avec la liste des fournisseurs d'identités reconnus.

La figure II.1.7 montre l'interface proposant à un utilisateur voulant se connecter à une ressource mutualisée de choisir son université d'origine qui pourra l'authentifier.



FIGURE II.1.7 – Interface d'identification de Shibboleth

## Virtualisation

La virtualisation consiste à faire fonctionner un ou plusieurs systèmes d'exploitation comme un simple logiciel, sur un ou plusieurs ordinateurs- serveurs / système d'exploitation, au lieu de ne pouvoir en installer qu'un seul par machine. Son intérêt dans le projet est de pouvoir créer des machines virtuelles à la demande aux étudiants pour les travaux pratiques. Cela pourra régler les problèmes relatifs aux infections des machines.

La portabilité des machines virtuelles offre la possibilité de mettre en œuvre des processus de restauration à moindres coûts et de réduire considérablement le temps nécessaire à la remise en route d'un serveur. Les nouvelles fonctions offertes par la virtualisation permettent de concevoir des systèmes capables de basculer automatiquement sur un serveur de réplication en cas de défaillance de l'hôte en production. Enfin, en consolidant des infrastructures complexes dans des environnements et des réseaux virtuels séparés, les administrateurs peuvent configurer des règles spécifiques à chaque environnement et maximiser la sécurité (Environnements de tests, de développement, ou d'environnements de production).

la solution open source proposée est OpenVZ.

**OpenVZ** OpenVZ est un projet de virtualisation par cloisonnement. Le principe de fonctionnement d'OpenVZ est très similaire à celui de Linux-VServer, car ils se basent tous les deux sur une modification du noyau Linux pour implémenter un système de cloisonnement au niveau du système d'exploitation. OpenVZ modifie le noyau Linux plus en profondeur que le patch de Linux-VServer. Il y a donc des fonctionnalités spécifiques à OpenVZ. Parmi ces nouvelles fonctionnalités, on peut notamment citer

l'ajout d'un niveau supplémentaire d'indirection pour les ordonnanceurs du noyau. Ce nouveau niveau d'indirection permet de gérer les priorités entre les systèmes invités. Une fonctionnalité intéressante d'OpenVZ est la possibilité de migrer un système invité à chaud. Il offre la possibilité de sauvegarder l'état complet du système invité dans un fichier sur le disque, puis à le transférer sur une autre machine où l'état pourra alors être restauré. Au niveau du contrôle des ressources, OpenVZ propose un ensemble de limites par système invité, que l'administrateur peut configurer. Par exemple, il peut allouer un nombre maximal de processus qu'un système invité aura le droit de créer, etc. Ces limites par système invité permettent un très bon contrôle et garantissent qu'un système ne nuira pas aux autres. La connectivité réseau est très similaire aux autres solutions, chaque système invité peut disposer de sa propre adresse IP, c'est le noyau modifié qui s'occupe de rediriger le trafic réseau vers le bon système.

## Owncloud

Owncloud est un logiciel libre développé en php permettant de partager, de synchroniser des ressources. Il permet de synchroniser des données avec les terminaux sous Linux, Windows ou Mac. L'équivalent propriétaire de Owncloud est Dropbox.

La figure II.1.8 montre l'interface du serveur Owncloud de l'UVS.

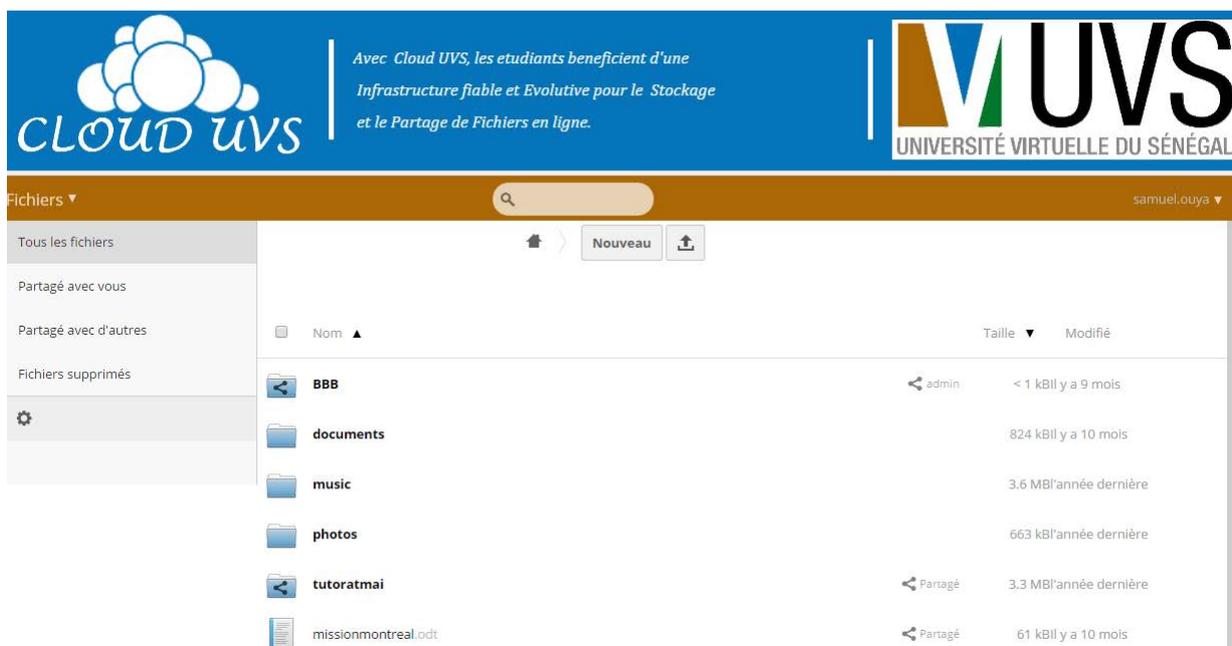


FIGURE II.1.8 – Owncloud

## BigBlueButton

BigBlueButton est une plateforme de visioconférence sur le Web permettant entre autres aux utilisateurs de bénéficier des fonctionnalités suivantes :

- Visioconférence
- Tchat
- Présentation PowerPoint
- Partage d'écran entre participants à une conférence

La figure II.1.9 donne une capture d'écran de conférence entre acteurs de l'UVS se trouvant dans plusieurs villes du Sénégal.



FIGURE II.1.9 – BigBlueButton

## 1.4 Quelques projets d'interconnexion en Afrique

Une organisation virtuelle ne pourra fonctionner correctement sans une bonne connectivité réseau entre ses différents membres. C'est pourquoi, nous présentons les projets d'interconnexion Wacren, UbuntuNet et SnRER.

### 1.4.1 AfricaConnect

#### 1.4.1.1 Présentation du projet

Le projet AfricaConnect vise à établir un réseau internet pour la recherche et l'éducation en Afrique australe et orientale. Il a démarré en mai 2011 mais lancé en novembre 2012 pour une période de quatre ans.

Pour la première année, la conception et le déploiement d'infrastructures ont été réalisés et l'an 2 correspond à la phase opérationnelle du projet.

Grâce à l'interconnexion au réseau européen GÉANT, AfricaConnect permettra aux chercheurs, étudiants et éducateurs de la région non seulement à collaborer entre eux, mais aussi de s'engager dans des projets communs avec leurs pairs en Europe et dans le monde entier.

En plus de UbuntuNet, le projet appuie l'organisation WACREN avec le projet d'assurer la préparation des NREN membres du WACREN pour le projet successeur AfricaConnect 2.

#### 1.4.1.2 Les objectifs du projet

Les objectifs du projet AfricaConnect sont entre autres :

- Bâtir une infrastructure réseau capable de supporter le développement et le déploiement d'applications pour les communautés académiques

- Renforcer le développement des programmes de recherche coopérative entre chercheurs africains et européens
- Promouvoir les activités de recherche et de coopération
- Permettre aux communautés de chercheurs africains de se connecter à leurs paires à travers le monde
- Renforcer les capacités des ingénieurs de la région
- Interconnecter les NREN de la région ESA à travers une alliance régionale

#### 1.4.1.3 UbuntuNet et WACREN

Le projet initial était celui d’AfricaConnet avec UbuntuNet. Mais avec le succès de la phase 1 et à la demande de l’Union Européenne(UE), les experts ont conduit l’étude de faisabilité pour l’extension du projet Africaconnect à la zone de l’Afrique de L’Ouest et du centre (WCA) d’où le projet AfricaConnect2 avec le réseau WACREN.

Cette extension a pour but d’appuyer les NRENs existants et les aider à s’interconnecter entre eux à travers un backbone régional qui sera connecté ensuite aux autres NRENs dans le monde via le RREN GEANT

#### 1.4.1.4 Étendue des réseaux

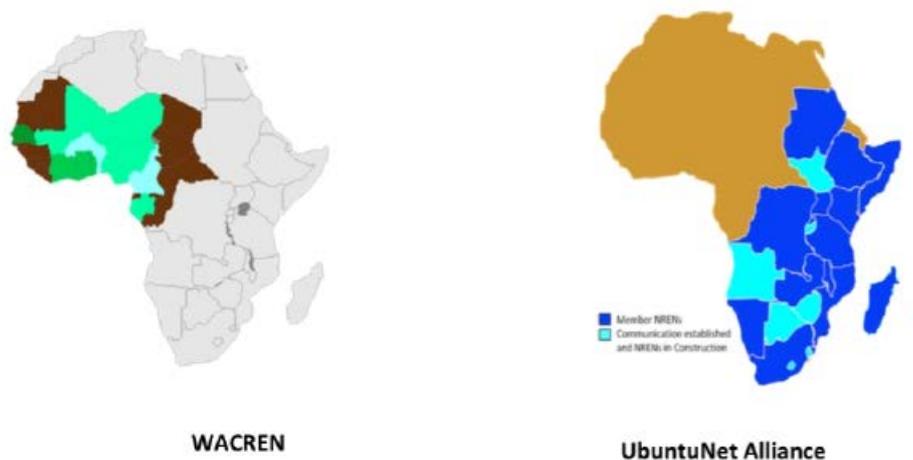


FIGURE II.1.10 – Réseau WACREN

#### 1.4.2 Padtice

Pour renforcer les capacités de mise en œuvre de la réforme LMD dans les institutions de l’enseignement supérieur des pays membres de l’UEMOA, l’UNESCO a initié le Projet d’Appui au Développement des Technologies de l’Information et de la Communication pour l’Enseignement(PADTICE) dont les objectifs spécifiques sont :

- Donner accès à des cours en ligne
- Mettre en place une bibliothèque en ligne accessible à toute la communauté
- Donner la possibilité aux chercheurs d’accéder à des informations sur les activités scientifiques et publications des de la sous-région
- Disposer des outils de communication instantanée

- Favoriser la collaboration et l'échange d'informations entre les différents acteurs de l'enseignement supérieur et de la recherche de la sous-région

Les universités suivantes ont été retenues pour la phase pilote :

- L'Université d'Abomey-Calavi, Bénin ;
- L'Université de Ouagadougou, Burkina Faso ;
- L'Université Félix Houphouët Boigny, Côte d'Ivoire ;
- L'Université Amilcar Cabral, Guinée-Bissau ;
- L'Université de Bamako, Mali : Université des Sciences, des Techniques et des Technologies de Bamako (USTTB) ;
- L'Université Abdou Moumouni, Niger ;
- L'Université Cheikh Anta Diop de Dakar (UCAD), Sénégal ;
- L'Université de Lomé, Togo.

### 1.4.3 SnRER

Le réseau pour l'enseignement supérieur et de la recherche du Sénégal dénommé SnRER a pour objectif spécifique de bâtir une infrastructure de communication numérique pour l'enseignement et la recherche, interconnectant la communauté scientifique nationale aux réseaux d'éducation et de recherche aux niveaux régional et internationale. L'un des aspects les plus fédérateurs, consiste à pourvoir les universités d'une connexion mutualisée à l'internet haut débit, favorisant l'accès aux ressources pédagogiques et aux innovations technologiques à toutes les composantes de l'espace universitaire (enseignants, personnels administratifs et étudiants). Ceci nécessite au préalable un réseau à haut débit. Ainsi, l'État du Sénégal dispose, à travers l'ADIE, une infrastructure fibre optique longue de 1500 km et reliant les 12 des 14 capitales régionales du pays.[55]

Cette infrastructure permettra ainsi à l'État du Sénégal d'interconnecter les eNodes B de son réseau 4G LTE privé couvrant les zones d'implantation des universités classiques et numériques du Sénégal.

L'UVS dispose de sites appelés espaces numériques ouverts (ENO) implantés dans les départements du pays. Ces ENO seront répartis à terme sur 45 départements en 2017. Ils sont un cadre de rencontre, et sont dotés d'infrastructures techniques offrant des connexions à haut débit pour les étudiants et les collectivités locales du département d'implantation. La figure II.1.11 illustre la répartition géographique des ENO en 2015 et le réseau fibre optique de l'ADIE.

En plus des ENO de l'UVS, les six universités publiques du Sénégal sont interconnectées grâce à la fibre optique de l'ADIE ce qui permet aujourd'hui d'envisager la fourniture de services multimédia de toutes sortes pour le bien de l'enseignement et de la recherche au Sénégal.



FIGURE II.1.11 – Cartographie du réseau Fibre optique de l'ADIE

L'infrastructure qui reposait avant 2015 sur la technologie de transmission NG – SDH et qui offrait des capacités :

- STM-64 (10 Gbps) sur la boucle Technopole, Thiès, Diourbel, Kaolack, Fatick, Mbour, Technopole ;
- STM-16 (2.5 Gbps) sur les parties linéaires Thiès, Louga, Saint Louis ;
- STM-64 (10 Gbps) linéaire entre Kaolack, Kaffrine, Kounghoul, Tambacounda, Kolda, Sédhiou, Bignona et Ziguinchor.

a évolué et a une capacité aujourd'hui de 400G pour tenir compte du passage du Sénégal à la Télévision Numérique Terrestre.

## 1.5 Conclusion

Pour relever le défi de développement du continent Africain, il est important de disposer des ressources humaines de qualité.

Avec les TIC, des opportunités sont offertes aux Africains de bien former leur jeunesse pour relever le défi de l'émergence. Aucune université Africaine à elle seule ne peut disposer des compétences pointues dans l'ensemble des domaines pouvant aider à amorcer rapidement l'émergence.

D'où la nécessité pour les universités d'un même pays de se constituer en organisation virtuelle pour mutualiser et partager les ressources. Ensuite, les organisations virtuelles nationales se constituent en une méga organisation virtuelle pour exploiter les points forts de chaque pays.

Étant donné qu'une collaboration efficace ne peut se faire sans utilisation des logiciels fournis par les TIC, il nous faut des logiciels performants adaptés à nos besoins. Sans négliger la place occupent les logiciels propriétaires dans nos sociétés Africaines, nous pensons que les logiciels libres constituent une opportunité importante à saisir. Les pré-requis en terme de réseaux d'interconnexion pour bâtir une organisation virtuelle africaine autour de l'enseignement commencent à se mettre en place avec les projets WACREN, UBUNTUNET et les réseaux d'enseignement et de recherche nationaux.

# Chapitre 2

## Organisations virtuelles et protocoles

### 2.1 Introduction

L'un des objectifs d'une organisation virtuelle est la promotion de la collaboration, de la mutualisation et du partage des ressources. Il s'agit dans ce chapitre de montrer en quoi des protocoles tels que XMPP, RDP et SPICE peuvent contribuer à l'atteinte de cet objectif. En effet, la collaboration dans le domaine de l'éducation peut se faire à travers la mutualisation des machines virtuelles, qui grâce aux protocoles précités, peuvent accéder et gérer des matériels de travaux pratiques connectés à un serveur de virtualisation. Ce qui encourage la promotion des laboratoires en ligne. Les différents services d'une organisation virtuelle peuvent coopérer en échangeant des informations dans des formats ouverts tels que XML. Ainsi, l'extension PubSub de XMPP pourrait permettre la mise en place d'une architecture MOM (Message Oriented Middleware) permettant à des applications d'échanger des informations de manière dynamique.

### 2.2 Étude du protocole XMPP et de PubSub

Le protocole XMPP (Protocole eXtensible de Présence et de Messagerie), issu du projet Open Source Jabber, est basé sur XML et constitué de deux parties bien distinctes. La première est un protocole de base qui regroupe les concepts fondamentaux pour faire fonctionner une infrastructure Jabber ceux-ci sont définis dans différentes RFC (RFC 6120, 6121, 6122, 5122, 3923). La deuxième est constituée des XEP (XMPP Extension Protocol), permettant d'ajouter des fonctionnalités au protocole Jabber. Ces XEP sont continuellement créées, révisées ou améliorées par la fondation XSF (XMPP Standard Foundation) [56].

#### 2.2.1 Le coeur du protocole XMPP

La norme XMPP/Jabber définit toutes les fonctionnalités basiques (connexion, échange de messages, d'informations de présence, etc.) et est constituée principalement de :

- **XMPP-Core** : [RFC 6120] qui définit le cœur du protocole, principalement la communication entre les clients et le serveur, et entre les serveurs ;
- **XMPP-IM** : [RFC 6121] qui définit principalement la manière dont sont échangés les présences et les messages par XMPP ;
- **XMPP-URI** : [RFC 5122] qui définit la manière dont les ressources sont adressées ;

- **XMPP E2E** : [RFC 3923] qui explique comment faire le chiffrement bout à bout avec XMPP ;
- **XMPP ADDR** : [RFC 6122] qui définit la manière dont les clients sont adressés

### 2.2.2 Les extensions du protocole XMPP

Alors que XMPP ne définit que les fonctionnalités essentielles, XSF propose des extensions XEP permettant aux utilisateurs de bénéficier de fonctionnalités additionnelles. Les clients et serveurs XMPP peuvent implémenter certains XEP. C'est le cas du **XEP 0060** permettant à un serveur de proposer le mécanisme PubSub, rendant possible la publication ou la souscription à des informations.

Les fonctionnalités offertes par le protocole XMPP permet à un utilisateur :

- De partager sa propre présence, son statut et son message d'état ;
- De savoir si ses contacts sont en ligne ;
- D'envoyer des messages, même aux contacts hors ligne ;
- De bénéficier d'une authentification sécurisée et du chiffrement de ses communications ;
- De participer à des discussions de groupe ;
- D'échanger des fichiers ;
- D'accéder aux services traditionnels fournis par les serveurs XMPP ;
- De partager une carte de visite avec ses correspondants ;
- D'ouvrir simultanément plusieurs sessions sur le même compte ;
- De se connecter à d'autres réseaux de messagerie ;
- De saisir des informations contextuelles ;
- De découvrir les logiciels qu'utilisent ses correspondants ;
- De montrer à ses contacts son humeur, ce qu'on a fait, la musique qu'on écoute ;
- De régler ses privacy lists ;
- De rechercher des correspondants dans un annuaire ;
- Etc.

### 2.2.3 L'architecture XMPP

XMPP fonctionne sur une architecture client/serveur. De ce fait, toutes les données envoyées d'un client à un autre passent nécessairement par un serveur XMPP au minimum. Le client établit une connexion TCP avec le serveur en utilisant le port 5222. La connexion ainsi établie perdure durant toute la durée de la communication, ce qui évite au client de demander confirmation au serveur de la réception des messages.

- Si le client est toujours connecté : Tous les messages destinés au client en question lui sont directement adressés.
- Si le client n'est plus connecté : Le serveur mémorise les messages qui lui sont adressés afin de pouvoir les lui transmettre lors de la prochaine connexion.

Un serveur XMPP dispose de la capacité de savoir si le client est connecté ou pas. Ce qui lui permet la livraison des messages en temps réel et la capacité du client à publier l'état de présence permet à un serveur XMPP de lui transmettre de manière instantanée un message.

### 2.2.3.1 Un client XMPP

Dans une architecture, un client XMPP n'a que très peu de restrictions. En effet, il est simple à mettre en œuvre et doit être capable :

- D'analyser les messages XML qu'il reçoit afin de pouvoir les interpréter
- Supporter les types de données de bases de XMPP, comme la présence par exemple.

Le fait du report par le protocole XMPP de la complexité sur les serveurs présente 2 avantages :

- D'une part, il est plus facile de mettre à jour les fonctionnalités inhérentes à un tel protocole en évitant de demander au client de mettre à jour lui-même son système ;
- D'autre part, les clients XMPP deviennent très faciles à implémenter dans la mesure où l'objectif est de faciliter l'interopérabilité entre les services de messagerie.

### 2.2.3.2 Un serveur XMPP

Un serveur XMPP de base présente 3 fonctionnalités :

- La gestion des connexions directes avec les clients XMPP ;
- La communication avec les autres serveurs XMPP ;
- La gestion des différents composants ajoutés par un administrateur du serveur.

La modularité d'un serveur XMPP repose sur le fait que chaque fonctionnalité (enregistrement, présence... etc.) est décrite par une portion de code indépendante. Cette flexibilité et cette modularité sont à l'origine même de l'interopérabilité de ce protocole et de sa simplicité de mise en œuvre.

### 2.2.3.3 L'adressage et formats de messages envoyés

L'identifiant d'un utilisateur XMPP est appelé **JID** (Jabber ID) et est composé de deux ou trois parties :

- Un nom d'utilisateur (unique sur un serveur) ;
- Un nom de serveur
- Une ressource qui peut changer, cette partie est optionnelle. La ressource est paramétrée par l'utilisateur qui peut en désigner plusieurs. Généralement, elles permettent de déterminer où l'utilisateur de Jabber est situé. Cette dernière est utilisée lorsque l'utilisateur se connecte à partir de plusieurs endroits ou avec plusieurs clients Jabber différents.

L'identifiant d'un utilisateur, formé de ces 3 informations, est de la forme suivante :

**utilisateur@nomServeur/ressource.**

Lors d'une connexion client/serveur, deux flux XML unidirectionnels sont créés : l'un du client vers le serveur, l'autre du serveur vers le client. Ainsi, la communication entre ce client et ce serveur Jabber repose exclusivement sur ces deux flux. Les données au format XML échangées durant la communication sont appelées « **stanzas** » de XML.

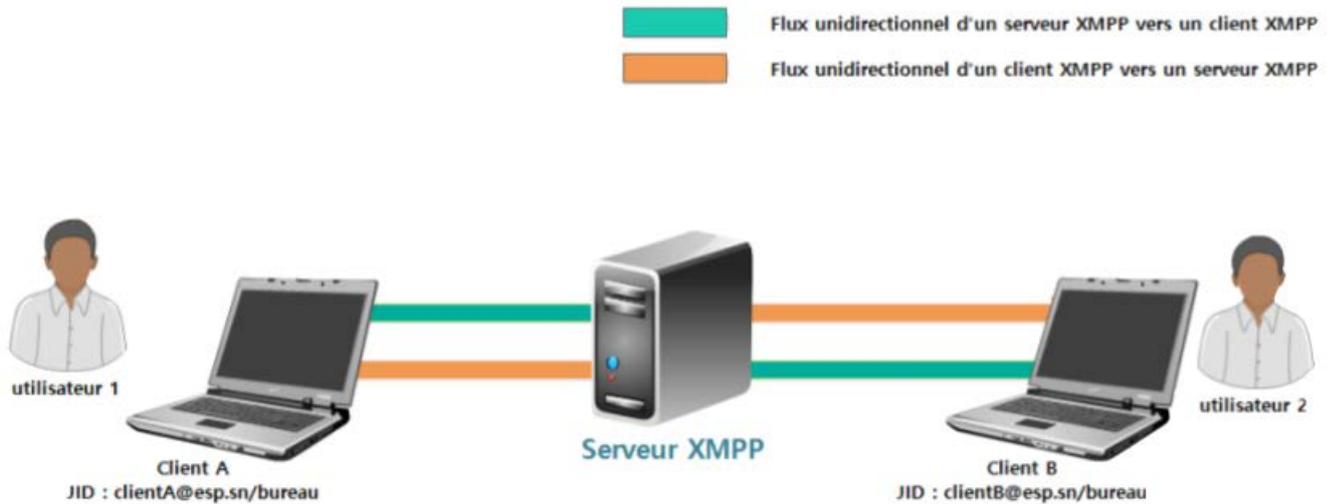


FIGURE II.2.1 – Architecture Client/serveur XMPP

Il existe 3 paquets de base :

- **Présence** : Il indique le statut du client considéré, en l'occurrence du client A. L'envoi de ce paquet est automatiquement effectué par le client, et non par l'utilisateur, vers le serveur.

```
<presence from= "clientA@esp.sn/bureau"
to="clientB@esp.sn/bureau">
  <show>xa </show>
  <status>Absent</status>
  <priority>5</priority>
```

FIGURE II.2.2 – Présence

La ressource qui a la plus haute priorité est considérée comme la ressource par défaut. Lorsque deux ressources ont la même priorité, le message est envoyé à la dernière ressource connectée.

- **Message** : Lorsque le client A communique avec le client B.

```
<message from= "clientA@esp.sn/bureau"
to="clientB@esp.sn/bureau">
  <body>Comment vas-tu?</body>
```

FIGURE II.2.3 – Message

- **Iq (Info/Query)** : Ces paquets sont utilisés pour échanger des données entre les applications XMPP (authentification, récupération d'une liste de contact, ou encore découvrir, accéder et interagir aux services proposés par le serveur). Ces données sont exprimées dans la balise "query", dans l'exemple ci-dessous, il s'agit d'un utilisateur demandant à s'authentifier.

```
<iq type='set' id='auth'>
  <query xmlns='jabber:iq:auth'>
    <username>boris</username>
    <password>passer</password>
    <resource>TelnetClient</resource>
  </query>
</iq>
```

FIGURE II.2.4 – Paquet IP

Le serveur répond alors par :

```
<iq type='result' id='auth' />
```

FIGURE II.2.5 – Réponse du serveur

Si l'authentification a échoué, le serveur renverra un paquet de type "error". En effet, aux types de paquets "set" (pour la mise à jour d'une donnée) et "result" (pour signifier que le message considéré est le résultat d'une requête précédente), s'ajoutent les messages de types "get" pour effectuer une requête et "error" lorsqu'une requête n'a pu aboutir. Dans le cas d'un message de type "error", ce dernier contiendra les éléments de la requête qui a échoué ainsi que le code et la description de l'erreur survenue.

L'avantage de baser un tel protocole sur le XML, réside principalement dans le fait que ce type de flux peut contenir n'importe quelle donnée structurée.

## 2.2.4 Quelques exemples de serveurs XMPP open source

Il existe plusieurs serveurs XMPP open source parmi lesquels on peut citer :

- **Ejabberd** qui fait office de référence à ce jour, écrit en erlang réputé pour ses mécanismes de tolérance aux pannes ;
- **Openfire**, écrit en java, et dont une adaptation interne constitue le moteur du service Google talk.

Le tableau suivant donne quelques plugins d'Openfire intéressant dans un contexte de collaboration :

TABLEAU II.2.1 – Plugins Openfire

Plugins	Fonctionnalités
<b>Kraken IM Gateway</b>	Fournit une passerelle vers d'autres serveurs publics de messagerie instantanée
<b>Openfire Meetings</b>	Fournit la possibilité d'utiliser WebRTC pour des communications audio, video, le partage d'écran etc.
<b>SIP Phone Plugin</b>	Permet de faire un mapping entre un utilisateur XMPP et un compte SIP pour la TOIP
<b>Openfire WebSocket</b>	Fournit un serveur WebSocket permettant de créer un lien bidirectionnel entre deux navigateurs
<b>REST API</b>	Fournit la possibilité de gérer un serveur Openfire en lui envoyant des requêtes HTTP/REST

## 2.2.5 Les bibliothèques clientes de XMPP

La maturité du protocole XMPP permet de disposer à ce jour d'une grande variété de bibliothèques clientes et ceci dans la plupart des langages. Une bibliothèque cliente XMPP est un ensemble de programmes écrits dans un langage donné qui donne accès à des fonctions, méthodes ou objets, selon le paradigme du langage en question, permettant de réaliser simplement les opérations du protocole en masquant ce dernier. Il existe des bibliothèques pour les langages C/C++, Java, C#, Perl, Python, PHP, Ruby, Ada, Lisp, Actionscript, JavaScript, Lua, Erlang, Haskell, objective-C. Parmi les bibliothèques clientes de XMPP, on peut citer :

- Smack API, Tinder API et Wack API qui sont des bibliothèques Java ;
- XIFF API qui est une bibliothèque Flash

Ainsi, on pourra donc écrire un programme se connectant à un serveur XMPP avec un JID donné, qui envoie ou qui reçoit des messages, qui réagit et répond à ces messages, etc. Un tel programme est appelé un **bot XMPP**.

### Exemple

Sur un serveur exécutant une application de gestion métier en Java, il peut être évident de déployer un petit bot tournant sur la même JVM. Il n'est par contre pas judicieux de déployer une JVM sur un serveur Apache ou Squid par exemple, alors qu'un bot en C, voire en perl ou python, s'ils sont nativement installés et utilisés sur la machine, sera peu intrusif. De la même façon, sur un serveur Windows Active Directory, pourquoi ne pas utiliser directement du code natif en C# ? L'idéal est

donc d'utiliser au cas par cas une librairie d'un des langages natifs du système (C/C++, C#, ou langages de script par exemple) ou interagissant nativement avec le service installé que l'on veut rendre communicant (java sur une JVM, PHP sur un web App PHP, par exemple).

Un bot écrit selon ces principes se révèle efficace en termes d'échanges avec le service concerné, peu intrusif en termes de maintenance du code général de la plate-forme et très économe en ressources (CPU, RAM, TCP).

### 2.2.6 Message Oriented Middleware (MOM) et organisations virtuelles

On dispose des serveurs XMPP open source performants tels qu'Ejabberd et Openfire qu'on peut déployer en local ou dans un réseau IP. Donc il est aisé de coder des bots sur la plupart des plateformes de collaboration. L'idée est donc d'utiliser ces bots comme interlogiciel de communication entre les services ou applications de plateformes mutualisées ou de collaboration : les services seront capables d'échanger des messages point-à-point entre eux de façon instantanée. On parle alors d'architecture à base de Message Oriented Middleware ou MOM.

Cependant, si le nombre de plateformes de collaboration devant échanger des informations augmente, il devient fastidieux de gérer le nombre élevé de communication point-to-point. D'où l'intérêt d'avoir un système à canal de publication avec un mécanisme d'abonnement pour les plateformes intéressées par les informations publiées. Ce serait l'objet du paragraphe suivant sur le mécanisme PubSub.

### 2.2.7 PubSub : Principe de fonctionnement et exemples

Un serveur XMPP implémentant la XEP 0060 permet de proposer le mécanisme PubSub ou Publish/Subscribe qui consiste à créer un nœud de publication déclaré sur le dit serveur. Un nœud de publication est un canal identifié par un JID portant sur un sujet donné. Les services ayant des JID intéressés par une information publiée dans un canal peuvent s'y souscrire pour en recevoir des informations.

Un serveur de messages instantanés fonctionnant sur le mode de canaux de distribution ou de queues de messages est appelé message broker dans une architecture à base de MOM. Le principe de fonctionnement de PubSub est décrit par le schéma ci-après :

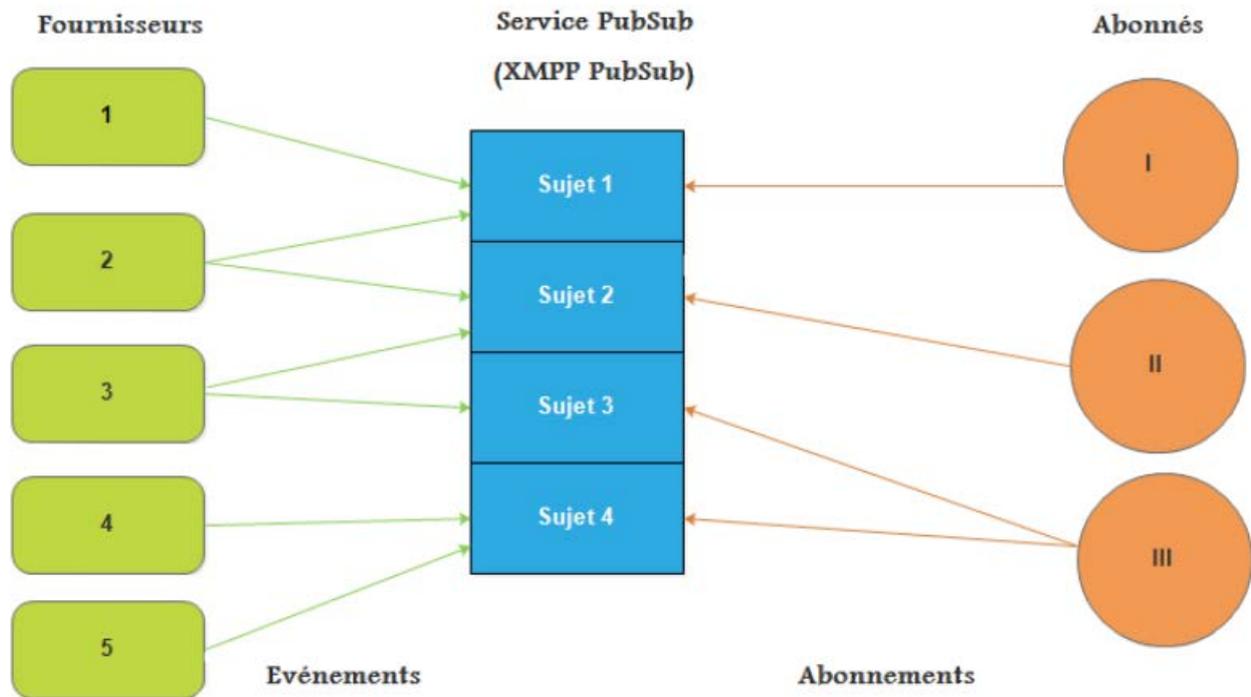


FIGURE II.2.6 – Principe de fonctionnement PubSub

Les fournisseurs créent des canaux d'informations avec des sujets bien définis et les abonnés s'y abonnent.

PubSub fonctionne en échangeant des messages XMPP. Il existe quatre types de messages de base PubSub :

- **Abonnement** : Un abonné envoie un message de type iq pour s'abonner à un sujet. Ainsi l'utilisateur recevra alors les événements du serveur PubSub tant que l'abonnement est actif.
- **Publication** : Le fournisseur envoie un message de type iq pour publier un événement, généralement (mais pas toujours), y compris une charge utile.
- **Événements** : Les événements sont envoyés aux abonnés à travers un message XMPP strophes.
- **Gestion** : XMPP gère le système PubSub ainsi que la découverte des informations sur le serveur. PubSub fournit un contrôle d'accès flexible pour soutenir un certain nombre de grandes approches :
  - Ouvert : tout le monde peut publier ou souscrire à des canaux
  - Géré : le fournisseur de services PubSub décide qui peut faire quoi
  - Contrôlé par l'utilisateur (fournisseurs et / ou contrôler l'accès abonnés)

### Exemple 1 : Mutualisation de ressources entre universités

On pourrait imaginer qu'un bot associé à un serveur de virtualisation de machines envoie sur un canal PubSub nommé SVUVS la notification qu'une nouvelle machine virtuelle vient d'être créée à son niveau. Un bot associé au serveur de virtualisation de l'Ecole Supérieure Polytechnique de Dakar (ESP) et inscrit au canal SVUVS, pourrait par exemple être notifié et à son tour informe ses utilisateurs de la disponibilité de cette machine virtuelle. Un autre bot de l'Université Gaston Berger de Saint-Louis (UGB) aussi inscrit au canal SVUVS et associé au serveur de virtualisation UGB pourrait informer ses utilisateurs. De même, si cette machine virtuelle est occupée ou change d'état, le même processus pourrait permettre d'en informer tous les utilisateurs des universités qui collaborent autour des machines virtuelles.

**Exemple 2 : Les réseaux sociaux**

L'utilisation de PubSub comme principe de base des réseaux sociaux permettraient leurs interopérabilités. Par exemple, les utilisateurs de Facebook pourraient dialoguer directement avec les utilisateurs de Twitter.

**Exemple 3 : Le domaine militaire**

PubSub pourrait être utilisé pour diffuser des informations ciblées de très grande taille.

**Exemple 4 : Le projet open source Archipel**

Archipel est un logiciel de gestion d'hyperviseurs et de machines virtuelles qui, contrairement à d'autres solutions, n'implique pas de remise en question de l'architecture existante. Il utilise la librairie Libvirt pour gérer les machines virtuelles. Cette librairie regroupe plusieurs outils dont le plus connu est la commande Virt qui permet de préparer la configuration réseau de la machine hôte, à créer, modifier, importer ou sauvegarder les états instantanés des machines virtuelles. Les entités d'Archipel utilisent le protocole XMPP pour dialoguer entre elles et noVNC pour l'affichage des machines virtuelles à travers le Web.

Archipel permet, via son système de permissions, de donner accès à des utilisateurs authentifiés, uniquement à certaines ressources de gestion : reboot, accès à la console VNC, modification des paramètres réseaux, redéfinitions de paramètres... Toutes les entités (utilisateurs, machines virtuelles, hyperviseurs) sont identifiées par des comptes XMPP. Un client XMPP appelé agent est installé sur chaque hyperviseur géré par l'Archipel. Ce qui permet d'exposer la présence de l'hyperviseur, ainsi que les machines qu'il gère.

La gestion de la liste de contacts au niveau du serveur XMPP, permet d'avoir une sécurité supplémentaire entre les entités. L'utilisation conjointe de la technologie PubSub et des événements de libvirt permet d'avoir une application asynchrone, donc résistant mieux aux montées en charge. L'Archipel fait appel à PubSub : Les entités qui souhaitent diffuser des informations à plusieurs entités publient via une entrée dans le service PubSub du serveur, et les entités qui souhaitent recevoir les messages pour cette entrée s'y abonnent. Ainsi, c'est encore une fois le serveur XMPP qui va faire le travail de diffuser aux entités abonnées les messages. Cela donne un flux optimal de messages, et simplifie la programmation des agents.

La plateforme de supervision d'Archipel est une application HTML5 qui dialogue avec les autres composants via XMPP. Le déploiement d'Archipel est possible sur une infrastructure libvirt en production, sans interruption de service.

Du point de vue sécuritaire, l'utilisation de XMPP par Archipel permet de rassurer sur l'intégrité des messages échangés. La sécurité qui permet de savoir quel utilisateur peut dialoguer avec une entité repose d'abord sur le roster de celle-ci, car comme le roster est stocké coté serveur XMPP, c'est sans doute la meilleure solution qui puisse être retenue. Il est donc possible de définir clairement quel utilisateur peut dialoguer avec une machine virtuelle ou un hyperviseur donné, et également de savoir qui a accès à une entité donnée. Archipel va encore plus loin dans la gestion de la sécurité. Comme ce sont des stanzas qui définissent les demandes entre les entités, et que cela se traduit par des appels à l'API de libvirt, un filtrage coté agent est en place et permet d'autoriser ou non certaines actions. Ainsi, actuellement, 110 rôles possibles sont définis et il est possible d'attribuer à un utilisateur, l'accès ou non à chacun de ces rôles. Par exemple, on peut donner à un webmaster, le droit d'accéder à la « console VNC » de sa machine virtuelle, ainsi que l'action start et stop, ou à un développeur, le droit

de prendre des instantanés d'une machine virtuelle, et de les restaurer.

Le dernier cas d'utilisation de PubSub dans Archipel nous pousse à utiliser ce mécanisme dans les organisations virtuelles pour la découverte dynamique de la disponibilité des ressources partagées dans les organisations virtuelles.

## 2.3 Étude de quelques protocoles VDI : RDP, SPICE

Aujourd'hui nous assistons à l'émergence de la technologie VDI (Virtual Desktop Infrastructure) permettant de virtualiser les postes de travail pour non seulement réduire les coûts des infrastructures, gérer la sécurité mais aussi faire des économies d'énergies car l'accès aux machines virtuelles peut se faire à partir des clients légers qui consomment peu d'énergie, via des protocoles tels que RDP (Remote Desktop Protocol) et SPICE (Simple Protocol for Independent Computing Environments). Les difficultés rencontrées par les utilisateurs des clients légers tournent essentiellement au tour de l'utilisation des périphériques tels qu'USB, périphériques audio et vidéo, etc. Donc un protocole d'affichage de machine virtuelle à distance se doit de minimiser ces difficultés.

### 2.3.1 Introduction à Virtualbox

Virtualbox est une plateforme open source de virtualisation permettant de lancer de nombreux systèmes d'exploitation au sein de machines virtuelles (différents OS Windows, linux, mac OS X, FreeBSD, OpenBSD, Netware, OS/2,...).

La configuration des machines virtuelles est stockée dans des fichiers XML, ce qui permet leur exportation facilement sur d'autres environnements. Les extensions peuvent être ajoutées au Virtualbox pour lui ajouter des fonctionnalités supplémentaires.

Grâce à l'ajout de l'extension VM Virtualbox Extension Pack, on peut activer un serveur RDP appelé VRDP qui permet d'activer l'accès à distance aux machines virtuelles hébergées.

A chaque machine virtuelle, on attribue un numéro de port et l'accès à cette machine virtuelle se fait à partir d'un client RDP en précisant l'adresse IP du serveur Virtualbox hôte et le numéro de port.

Des outils développés en Python, permettent d'utiliser des lignes de commande pour administrer le serveur RDP de Virtualbox.

### 2.3.2 Quelques fonctionnalités avancées de Virtualbox

#### 2.3.2.1 Connexions multiples au serveur VRDP

Le serveur de VRDP de Virtualbox supporte de multiples connexions simultanées à la même machine virtuelle VM à partir de différents clients. Tous les clients connectés voient la même sortie de l'écran et partagent un pointeur de souris et le focus du clavier. Ceci est similaire à plusieurs personnes utilisant simultanément le même ordinateur, en prenant tour à tour le clavier. C'est une fonctionnalité très intéressante pour des formations e-learning.

#### 2.3.2.2 Moniteurs distants multiples

Le client RDP peut sélectionner le numéro de moniteur virtuel pour se connecter à l'aide du paramètre de connexion de domaine (-d).

### 2.3.2.3 Redirection Vidéo

A partir de Virtualbox 3.2, le serveur de VRDP peut rediriger des flux vidéo du client vers le client RDP. Les images vidéo sont compressées en utilisant l'algorithme JPEG permettant un taux de compression plus élevé que les méthodes de compression d'image bitmap. Il est possible d'augmenter le taux de compression par abaissement de la qualité vidéo.

Le serveur de VRDP détecte automatiquement les flux vidéo dans des zones rectangulaires fréquemment mis à jour. En conséquence, cette méthode fonctionne avec tout système d'exploitation invité sans avoir à installer de logiciel supplémentaire sur le client ; en particulier, les plugins client ne sont pas nécessaires.

### 2.3.2.4 Personnalisation de l'activation des périphériques

Avec Virtualbox 4.0, il est possible de désactiver la sortie d'affichage, la souris et le clavier, le port audio, les ports USB ou presse-papiers individuellement dans le serveur VRD.

### 2.3.2.5 Téléportage des machines virtuelles

Depuis la version 3.1, Virtualbox supporte le " téléportage " - qui est, le déplacement d'une machine virtuelle à chaud sur un réseau d'un hôte Virtualbox à l'autre. Cela fonctionne indépendamment du système d'exploitation abritant Virtualbox : on peut téléporter des machines virtuelles entre les hôtes Linux et Windows, par exemple.

## 2.3.3 Introduction au protocole RDP

Le protocole RDP développé initialement par Microsoft, a été adapté par plusieurs développeurs pour le porter en environnement Unix et autres. Il permet à utilisateur de se connecter sur un serveur RDP qui écoute par défaut sur le port TCP 3389 et possède des clients Windows et GNU Linux. Il a été initialement défini pour permettre aux administrateurs systèmes de prendre la main à distance sur un serveur ou un poste utilisateur. Ce qui a rendu possible la centralisation des équipes systèmes et réseaux sur un même site, qui peuvent administrer des machines se trouvant sur un autre site.

## 2.3.4 Versions et fonctionnalités de RDP

Basé sur le protocole ITU T.share (aussi connu sous le nom T.128), la première version de RDP (version 4.0 à l'époque) a été introduite avec Terminal Services dans Windows NT 4.0 Server, Terminal Server Edition.

- La version 5.0, introduite avec Windows 2000 Server, ajoute la prise en charge de nouvelles fonctionnalités, incluant l'utilisation d'imprimantes locales, un cache local persistant, une reprise de session, un partage de presse papier et l'amélioration de l'utilisation de la bande passante réseau ;
- La version 5.1, introduite avec Windows XP Professionnel, inclut la prise en charge du son, la redirection de disques, les ports COM, la prise en charge des couleurs sur 24 bits ainsi que l'ouverture de sessions par carte à puce ;
- La version 5.2, introduite avec Windows Server 2003, intègre la prise en charge de l'utilisation de ressources locales. C'est la version où apparaît la capacité de protéger les sessions en utilisant TLS 1.0. Il améliore également les mécanismes de sécurité ;

- La version 6.0, introduite avec Windows Vista apporte les fonctionnalités de couleurs sur 32 bits, ainsi qu'un pseudo multi-écrans. Pour la sécurité, il prend en charge Network Location Awareness (NLA) ;
- La version 6.1 : passerelle TS, applications distantes, authentification unique, pilote universel d'impression. Ces nouvelles fonctions nécessitent également que le poste client dispose du protocole 6.1 (Vista SP1, XP SP3) ;
- La version 7.0, introduite avec Windows 7 (Professionnel - Ultimate uniquement]) apporte des fonctionnalités d'audio bidirectionnel, l'accélération des échanges bitmap, la prise en charge de Windows Aero et un vrai mode multi-écrans. Concernant ses fonctions de sécurité, il permet d'utiliser un certificat auto-signé de 2048 bits d'une validité de six mois (inclus dans la configuration par défaut).

RDP présente également de nombreuses fonctionnalités à savoir :

- Prise en charge des couleurs 24-bits, équivalent à 16,7 millions de couleurs. (couleurs 8, 15 et 16 bits également gérées.) ;
- Chiffrement 128 bits, utilisant l'algorithme RC4. (C'est la sécurité par défaut ; des versions plus anciennes peuvent utiliser un chiffrement moins fort.) ;
- Prise en charge du son : les utilisateurs peuvent écouter sur l'ordinateur local le son produit par un programme exécuté sur l'ordinateur distant ;
- Projection des fichiers de l'ordinateur local sur l'ordinateur distant par le biais de la session RDP ;
- Projection d'une imprimante locale sur l'ordinateur distant, pour imprimer sur l'imprimante locale des documents ouverts sur l'ordinateur distant ;
- Projection des ports série et parallèle de l'ordinateur local sur l'ordinateur distant.

Du point de vue sécuritaire, RDP peut implémenter deux niveaux de sécurité :

- Mode Standard : mode utilisé par défaut, il implémente 5 niveaux de sécurité :
  - Aucune (échange en clair)
  - Low
  - Medium
  - Client Compatible
  - High
  - FIPS Compliant
- Mode Enhanced : Mode permettant d'utiliser des mécanismes de sécurité externes (SSP) : TLS ou CredSSP (utilisant au choix une authentification Kerberos ou NTLM).

### 2.3.5 Implémentations libres du protocole RDP

Sous Linux, il existe plusieurs implémentations du protocole RDP dont on peut citer les implémentations

- Côté client
  - FreeRDP
  - Remmina
  - RDesktop
- Côté serveur
  - XRDP qui peut utiliser le serveur Pulse Audio pour gérer le son côté client. Pulse Audio

est un serveur son, un processus d'arrière-plan qui accepte les entrées son d'une ou plusieurs sources (processus ou système de capture) et les redirige vers un ou plusieurs dispositifs (cartes son, serveurs Pulse Audio distants, ou autres processus). Il utilise un modèle dans lequel les applications envoient le son au serveur, contrairement au serveur son JACK qui force quand et dans quel ordre les applications doit lui envoyer le son.

### 2.3.6 Introduction au protocole SPICE

SPICE (Simple Protocol for Independent Computing Environments) est un protocole de communication dédié aux environnements virtuels, développé par l'éditeur de logiciels libres RED HAT. Il permet aux utilisateurs d'afficher la console des machines virtuelles (VM) depuis n'importe où sur Internet. Il s'agit d'un modèle client-serveur qui part du principe que la machine virtualisée est un serveur SPICE et que les utilisateurs peuvent s'y connecter en utilisant un client SPICE. Ce protocole permet de prendre en charge la gestion des périphériques audio, vidéo, USB, etc. Il existe de clients SPICE HTML5 permettant de prendre le contrôle d'une machine virtuelle serveur SPICE à travers un navigateur.

### 2.3.7 Architecture de SPICE

L'architecture SPICE repose sur un modèle à trois niveaux, qui permet aux fonctions intelligentes du protocole de décider du meilleur emplacement pour le rendu des actions de l'utilisateur. Ces trois niveaux sont repartis comme suit :

#### Agent SPICE

Composant logiciel en option, inclus dans le package d'outils Red Hat Entreprise Virtualization qui est installé sur la machine virtuelle invitée. Conçu pour améliorer l'expérience utilisateur, ce pilote exécute des tâches de gestion orientées invité et apporte des améliorations notamment dans le signalement de la position de la souris, les paramètres d'affichage du moniteur, le montage des périphériques USB etc.

#### Serveur SPICE

Composant logiciel faisant partie de l'hyperviseur Red Hat Entreprise Virtualization sur les serveurs hôte. Il joue le rôle d'interface principale entre l'agent qui se trouve sur la machine virtuelle et le client installé sur l'ordinateur personnel de l'utilisateur. Il assure ainsi de manière transparente la communication des interactions de données et de périphériques.

#### Client SPICE

Composant logiciel multiplateforme qui réside sur le périphérique client de l'utilisateur est utilisé pour accéder à chacune des machines virtuelles sur Red Hat Entreprise Virtualization for Desktops. Le client fonctionne à la fois sur des périphériques clients Windows et Linux, notamment sur les clients légers et PC requalifiés.

La figure II.2.7 décrit l'architecture de SPICE.

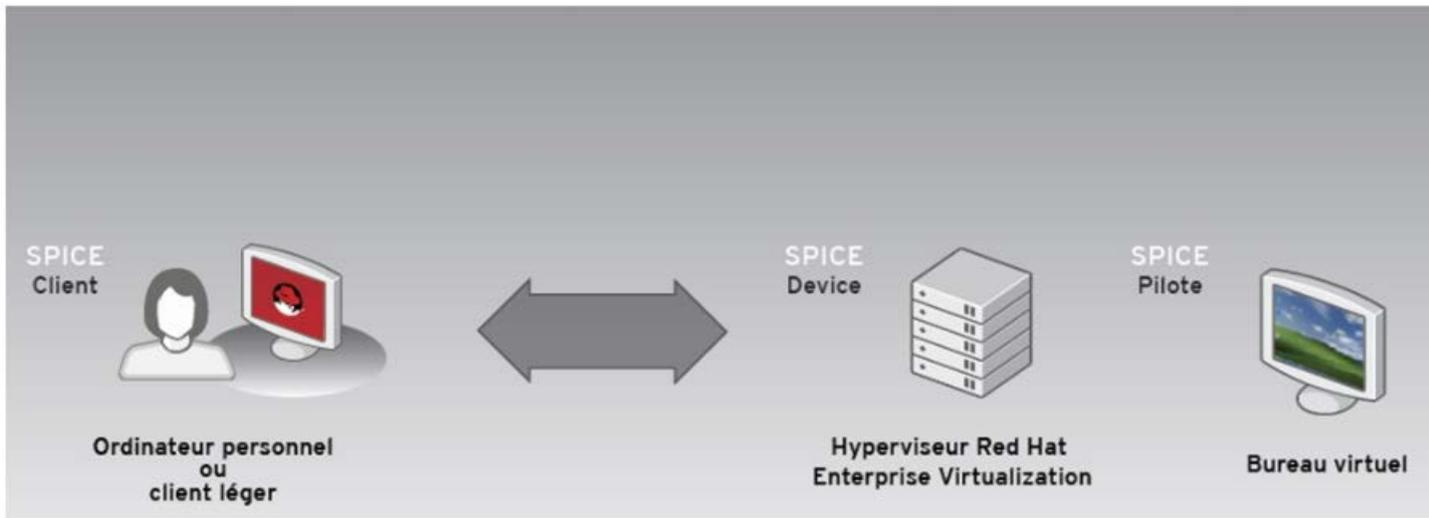


FIGURE II.2.7 – Architecture SPICE

### 2.3.8 Fonctionnement de SPICE et gestion des périphériques

L'accès à la machine virtuelle depuis le poste de l'utilisateur peut être fait en utilisant les fonctionnalités de bureau à distance de XP et plus précisément le protocole RDP. Il est également possible d'utiliser des outils comme VNC ou tout autre produit de contrôle à distance pour obtenir le même résultat.

Pour résumer, l'objectif est de déporter l'affichage de la machine virtuelle sur le terminal de l'utilisateur et de permettre à ce dernier d'utiliser ces périphériques de saisie locaux (claviers, souris) pour contrôler les actions de la machine virtuelle. Au final, l'utilisateur final doit avoir l'impression que la machine virtuelle s'exécute sur son poste.

En complément, les sons générés par la machine virtuelle peuvent aussi être envoyés vers le terminal. Si le terminal est équipé de périphériques de stockage, ceux-ci pourront être utilisés depuis la machine virtuelle pour transférer des fichiers.

#### 2.3.8.1 Détails de fonctionnement

Dans le cas de SPICE, ce ne sont pas les fonctions ou des programmes additionnels du système d'exploitation virtualisé qui sont mis en œuvre. C'est au travers de pilotes installés sur le système hôte que seront gérées les interactions avec le terminal. Le schéma suivant montre les différents composants mis en œuvre.

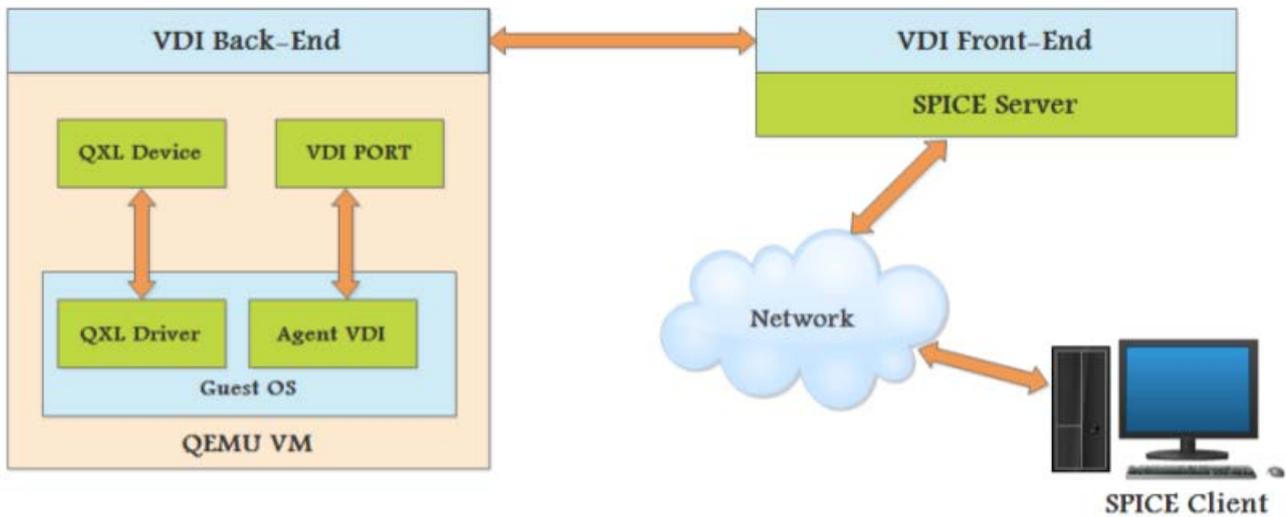


FIGURE II.2.8 – Composants de SPICE

**Le back-end VDI** communique avec le système hôte grâce à un driver QXL qui prend la forme d'un driver vidéo pour les systèmes d'exploitation Windows par exemple.

**L'agent SPICE** est un composant optionnel qui permet d'améliorer l'expérience utilisateur. Le copier/coller devrait ainsi être supporté prochainement vers et depuis l'hôte virtualisé. Une fonctionnalité quasi indispensable en effet.

**Le client SPICE** est résident sur le poste utilisateur, il sert à accéder à chaque bureau virtuel. Il est l'interface pour l'utilisateur final et est multiplateforme c'est-à-dire qu'il fonctionne aussi bien sous LINUX que sous WINDOWS. La communication entre le client et l'hôte s'effectue à partir de différents canaux.

**KVM** : fournit l'infrastructure de base et un module de traitement spécifique.

**QXL/VDI** : QXL est un driver pour le graphisme à installer côté machine virtuelle. Celui-ci permet la communication entre le VDI port et le VDI agent. Il décode, au niveau de KVM, les primitives graphiques qui seront transmises au client SPICE. Celui-ci aura en charge l'affichage via la carte du terminal client. Les environnements virtuels ainsi fournis (2D et 3D) seront fonction des capacités matérielles du client. VDI consiste en l'hébergement d'un système d'exploitation au sein d'une machine virtuelle qui est en cours d'exécution sur un serveur.

L'intégration de SPICE dans les systèmes d'exploitation Linux ou dans OpenStack permet le déploiement de bureaux virtuels en s'appuyant sur les outils de virtualisation QEMU/KVM.

Ces composants fonctionnent ensemble et déterminent l'emplacement idéal afin d'optimiser l'utilisation et de minimiser la charge du système. En effet, avec un client puissant, SPICE lui envoie les commandes graphiques et les traite ensuite directement au niveau client dans le but de diminuer la charge serveur. Dans le cas contraire (client insuffisamment puissant), SPICE traite les images au niveau de l'hôte ce qui diminuera la charge serveur.

### 2.3.8.2 Quelques fonctionnalités de SPICE

La version actuelle de SPICE supporte les fonctionnalités suivantes :

- Traitement et transmission des commandes graphiques 2D ;
- Identification des flux vidéo et transmissions au format M-JPEG ;
- Compression des images à l'aide de plusieurs algorithmes spécifiquement conçus pour SPICE comme QUIC (basé sur SFALIC), LZ, GLZ ;
- Mise en cache des images, palettes et curseur pour réduire la consommation de bande passante ;
- Live migration – évite les interruptions lorsque les serveurs SPICE sont changés d'hôte ;
- Support des moniteurs multiples ;
- Client pour Linux et Windows ;
- Support de l'audio en lecture et enregistrement ;
- Encryption à l'aide d'OpenSSL ;
- Synchronisation de l'audio et de la vidéo à l'aide de l'outil Lip-sync

### 2.3.8.3 Avantages de SPICE

Les avantages de SPICE sont repartis de la façon suivante :

#### Qualité exceptionnelle de l'expérience utilisateur

SPICE tire parti des ressources système du périphérique client de l'utilisateur final pour le rendu des applications gourmandes en ressources, les ordinateurs de bureau distants semblent fonctionner comme s'il s'agissait d'environnements installés localement. Cette approche donne des résultats exceptionnels, notamment avec les applications plus exigeantes comportant son, vidéo et autres formes de contenus multimédias.

#### Réduction des coûts associés

Le fait de pouvoir s'appuyer sur les ressources système du client local permet de libérer parallèlement des ressources précieuses sur le serveur de virtualisation hôte ; cela permet d'optimiser la densité de machines virtuelles sur le serveur hôte. En libérant des ressources système sur le serveur hôte pour gérer d'avantage de machines virtuelles. Il permet à l'entreprise d'acheter et d'entretenir à un coût moins élevé du matériel serveur qu'avec les solutions concurrentes.

#### Garantie de sécurité pour les données

Certaines solutions SPICE offrent une fonction qui peut être exploitée en option pour le cryptage intégral de la connexion SPICE, avec, de bout de bout, un tunnel SSL (Secure Sockets Layer) entre le périphérique client de l'utilisateur et de la machine virtuelle. Ce tunnel sécurisé garantit la protection de la liaison de données entre le client et le serveur hôte, quel que soit l'emplacement ou le périphérique client à partir duquel les utilisateurs accèdent à leur ordinateur de bureau.

#### Connexion de tout périphérique USB

De nos jours, la plupart d'utilisateurs connectent à leur ordinateur divers périphérique USB. Lorsqu'ils passent à une infrastructure de postes de travail virtuels, ils s'attendent à pouvoir faire de même.

A la différence des solutions concurrentes, limitées aux simples périphériques de stockage, certaines solutions SPICE ont recours à la redirection complète des données USB pour les postes de travail virtuels. En conséquence, tous les périphériques d'entrée USB 1.0 et 2.0 sont pris en charge via le protocole SPICE.

## 2.3.9 Étude comparative de systèmes de virtualisation sous Linux : Xen, Qemu, KVM

### 2.3.9.1 Qemu

#### Présentation

QEMU est le projet de virtualisation complète libre le plus abouti actuellement. Il a été fondé par Fabrice BELLARD et diffusé à la communauté en 2003. Fabrice BELLARD est également connu pour avoir fondé le projet FFMPEG, une suite d'utilitaires dédiés au traitement et à la conversion de flux numériques (vidéo et audio). À l'heure actuelle, M. BELLARD est toujours le mainteneur et développeur principal de QEMU, secondé par une communauté active.

#### Technologie

Techniquement, QEMU utilise la virtualisation complète, et il supporte de nombreuses architectures cibles (i.e. émulées), parmi lesquelles les processeurs x86 et l'architecture PC. Il fonctionne sur les plates-formes les plus courantes (Microsoft Windows, GNU/Linux, Mac OS X) et est très simple d'utilisation. QEMU peut en option utiliser un module d'accélération système pour améliorer les performances. Ce module s'appelle **kqemu** (kernel-QEMU) et est disponible pour Windows et GNU/Linux. Il s'intègre au noyau (ou dans les services dans le cas de Windows) et permet à QEMU de contourner certaines couches d'abstraction du système hôte, amenant ainsi un gain de performances

#### Fonctionnalités

Ce qui distingue QEMU de solutions plus intrusives (comme Linux-VServer ou Xen) est sa grande simplicité d'usage. En effet, comme c'est un projet utilisant la virtualisation complète, il y a un simple programme à exécuter sur le système hôte pour obtenir une nouvelle machine virtuelle contenant un système invité. Un des gros points forts de QEMU est la flexibilité qu'il offre au niveau des options pour la configuration de la machine virtuelle. La section suivante détaillera la gestion des interfaces réseau et du trafic réseau en général, car c'est un des points forts de QEMU. L'étendue de ses possibilités dans ce domaine ont été l'un des critères pour sa sélection dans ce comparatif.

#### Communauté

L'activité autour de QEMU est toujours importante et les contributions externes sont relativement régulières, gage que QEMU dispose d'une forte communauté. Cette communauté est rassemblée autour d'une liste de diffusion, d'un forum de discussion et d'un wiki. De plus, il y a plusieurs projets autour de QEMU, qui visent pour la plupart à offrir une interface aux nombreuses options de ligne de commande du programme.

### 2.3.9.2 Xen

#### Présentation

Xen (dont le nom vient du grec *xenos*, étranger) est un projet de virtualisation par hyperviseur géré par la société XenSource. Le projet était à l'origine mené au sein de l'Université de Cambridge, sous le nom de Xenoserver. Le but était alors d'héberger 100 systèmes invités sur une seule machine physique, avec les meilleures performances possibles. En 2003, les initiateurs du projet ont fondé la société XenSource et ont lancé le projet Xen en se basant sur le code source de Xenoserver. Le rachat en août 2007 de la société XenSource par Citrix ne devrait rien changer pour la communauté, la licence du produit libre restant inchangée. Toutefois, les entreprises pourront être rassurées de voir que XenSource est maintenant soutenue par des moyens financiers importants. La solution de virtualisation Xen est séparée en plusieurs produits, ayant tous des finalités différentes. Il y a tout d'abord la version libre, nommée Xen 3.0, qui concentre toute la technologie de virtualisation. Les autres versions (propriétaires) de la gamme se distinguent uniquement par le support proposé, les nombre de machines virtuelles supportées, les systèmes invités supportés et les logiciels annexes. Ainsi, la version Xen Enterprise se base intégralement sur Xen 3.0, mais rajoute des outils de contrôle pour gérer plusieurs dizaines d'instances de Xen. Dans le cas de Xen, la technologie de virtualisation est la même partout, les versions propriétaires ont seulement des outils de gestion à grande échelle et un support technique fourni par XenSource. Les performances sont donc identiques entre les produits.

#### Technologie

Xen est un hyperviseur, c'est à dire qu'il vient s'insérer entre le matériel et le noyau. C'est donc Xen qui a l'accès exclusif au matériel, et les systèmes d'exploitation fonctionnant par-dessus doivent obligatoirement passer par l'hyperviseur pour y accéder. Il peut ainsi répartir précisément les ressources entre les systèmes invités. Xen étant une solution à base d'hyperviseur, les systèmes destinés à s'exécuter au-dessus doivent être portés pour pouvoir cohabiter avec l'hyperviseur. Or, la modification des couches basses d'un système pour améliorer les performances est précisément la définition de la paravirtualisation. C'est pourquoi on considère que Xen est aussi une solution à base de paravirtualisation, car le système doit être modifié pour cohabiter. Au-dessus de l'hyperviseur se trouvent les systèmes invités, qui sont tous contrôlés par Xen. Le premier système d'exploitation démarré par Xen a des privilèges particuliers. Ce système est appelé « domaine zéro » (dom0), par opposition aux systèmes démarrés plus tard, appelés « domaines utilisateurs » (domUs). Le domaine zéro est le seul autorisé à créer et contrôler les domaines utilisateurs. Le projet Xen se compose d'un ensemble de modifications à appliquer au noyau Linux (des patches) ainsi que des utilitaires en espace utilisateur qui permettent d'administrer les systèmes invités (les domaines utilisateurs, dans la terminologie de Xen). Les utilitaires sont dans le domaine zéro et interagissent avec l'hyperviseur par le biais d'appels systèmes spécifiques. Une partie du code source de Xen a finalement été intégrée au noyau Linux durant l'été 2007, après plusieurs années d'attente. La plus grosse partie du code source est cependant encore maintenue séparément, avec les inconvénients que cela comporte.

#### Fonctionnalités

Au niveau des fonctionnalités, Xen est très certainement un des plus complets. La configuration du réseau est à l'image de celle de QEMU : difficile à appréhender, mais néanmoins très puissante une fois maîtrisée. Au démarrage, Xen prend en charge la gestion des interfaces réseau, et crée une interface virtuelle pour chaque système invité, qui seront toutes reliées à une interface physique par un système

de pont (bridging). On peut donc, comme avec QEMU, attribuer une adresse IP distincte à chaque système invité, les rendant ainsi impossibles à différencier de véritables machines physiques sur le réseau. Au niveau des images disques, Xen peut fonctionner de la même manière que les solutions à base de cloisonnement vues précédemment (Linux-VServer et OpenVZ) : le système invité sera un simple ensemble de fichiers dans un répertoire du système hôte. Il y a néanmoins la possibilité de placer les systèmes invités sur des partitions dédiées du disque dur, voire sur un autre disque. De cette manière, les machines virtuelles bénéficient de leur propre espace, séparé du système hôte. Cela permet d'exploiter toutes les fonctionnalités de GNU/Linux pour la gestion des disques, notamment le RAID logiciel et le redimensionnement des partitions sans perte de données. On peut dans tous les cas accéder directement aux fichiers des domaines utilisateurs, il suffit de « monter » la partition dans le système de fichiers du système hôte, de la même façon que l'on monte une clef USB ou un lecteur réseau. Un autre point fort du projet Xen est la gestion des systèmes invités : on peut très facilement, avec les outils d'administration fournis, contrôler l'état d'un système, son activité (consommation du temps processeur et de la mémoire allouée), le mettre en pause ou l'arrêter complètement. Les outils de gestion permettent également de migrer les systèmes invités entre différents systèmes hôtes. La migration « à chaud » est possible dans certaines conditions. En effet, Xen ne peut pas traiter correctement la conservation des connexions réseau si les systèmes d'origine et de destination ne sont pas sur le même domaine de collision (c'est à dire la même portion de réseau physique). La migration « à chaud » d'un système invité sans connexion réseau active est toutefois réalisable assez simplement, et le processus n'interrompt la machine virtuelle que quelques. Pour que la migration s'effectue rapidement, il est néanmoins nécessaire que les deux systèmes hôtes aient accès à l'image disque, par exemple par le biais d'un disque partagé sur le réseau. La rapidité de la migration dépend alors des performances du disque partagé. Une migration avec le système invité arrêté est réalisable sans contraintes, et c'est de toute manière la fonctionnalité la plus pratique dans le cas d'utilisation envisagé.

## Communauté

La communauté de Xen est rassemblée autour de plusieurs listes de diffusion, d'un wiki et de plusieurs forums. Elle est dans l'ensemble assez active, tant en contribution de documentation qu'en projets annexes, permettant de profiter au mieux de Xen. Contrairement à SWsoft avec OpenVZ, la société XenSource est donc très ouverte et soutient activement la communauté utilisateur de Xen. La version propriétaire et payante de Xen apporte juste un contrat de support et des outils de gestion de plus haut niveau, mais pas de gains de performances. Dans l'ensemble, la communauté Xen est très certainement la plus grande communauté parmi celles analysées ici. Le projet a su convaincre des milliers d'utilisateurs, gage de grande qualité et garantie que le projet ne sera pas à court de contributeurs du jour au lendemain.

### 2.3.9.3 KVM

#### Présentation

Le projet KVM (Kernel-based Virtual Machine – machine virtuelle dans le noyau) a été créé en 2006 par la société Qumranet. KVM a su très vite attirer les contributions externes et le code source du projet a été intégré au noyau Linux dès février 2007. Le mainteneur principal de KVM est Avi KIVITY. À l'heure actuelle, le projet KVM est financé principalement par Qumranet, mais IBM participe aussi au développement en salariant un développeur pour travailler sur KVM : Anthony LIGUORI. Il est d'ailleurs intéressant de constater qu'un des fondateurs de la société Qumranet, Moshe BAR, est aussi

un des fondateurs de la société XenSource, la société à l'origine de Xen.

## Technologie

KVM est un projet de virtualisation complète qui utilise les instructions de virtualisation des processeurs x86 récents. Techniquement, KVM se compose :

- d'un module noyau qui utilise les instructions de virtualisation et communique avec le processeur ;
- d'un programme utilisateur, qui utilise le module noyau pour toutes les opérations privilégiées.

La partie utilisateur de KVM est une version légèrement modifiée de QEMU, simplement adaptée pour que les opérations pouvant bénéficier des instructions de virtualisation du processeur fassent appel au module noyau. Il y a actuellement en développement un module de paravirtualisation, qui pourra servir à accélérer encore davantage les performances des Entrées/Sorties, notamment au niveau de la communication réseau. Ce module est toutefois facultatif — en plus d'être instable et toujours en développement — et l'on peut tout à fait s'en passer.

## Fonctionnalités

KVM étant basé sur QEMU, il reprend naturellement toutes les fonctionnalités de ce dernier. Il y a toutefois une différence notable : grâce à l'utilisation des instructions de virtualisation des processeurs récents, les performances des systèmes invités sont bien plus élevées qu'avec QEMU. Une autre fonctionnalité qui se démarque de QEMU est la migration des systèmes invités. KVM propose en effet la migration « à chaud » des hôtes, mais d'un fonctionnement différent de la migration d'images de QEMU. Le processus de migration de systèmes invités de KVM est actuellement plus performant que celui de QEMU, mais QEMU bénéficiera sous peu des améliorations de KVM.

## Communauté

La communauté autour de KVM est à l'image de celle de QEMU : diversifiée et active. Les deux projets partagent d'ailleurs une bonne partie de leur communauté. Les mêmes moyens de communication sont utilisés : liste de diffusion et wiki. Le développement est principalement assuré par les employés de Qumranet (notamment Avi KIVITY, mainteneur officiel), mais il y a des contributeurs externes réguliers. On peut notamment citer Rusty RUSSELL, décidément très actif dans le milieu de la virtualisation, ou encore des employés d'Intel, d'AMD ou d'IBM. Le projet a donc réussi à attirer l'attention des grands acteurs de la virtualisation. La société Qumranet collabore étroitement avec les développeurs de QEMU, et la plupart des modifications apportées à QEMU pour le rendre plus performant sont répercutées dans le projet original. Seules les modifications spécifiques à KVM ne sont pas appliquées, car elles n'ont de sens que pour KVM.

### 2.3.9.4 Bilan de la comparaison

KVM et Xen sont très différents, techniquement parlant. Toutefois, pour l'utilisateur final du service, la différence entre un serveur virtuel — quel qu'il soit — et une machine physique est inexistante, excepté au niveau des performances. Le choix se fait donc essentiellement sur le niveau de performances atteint. Il ne faut cependant pas négliger la facilité d'utilisation et d'administration de la solution.

La complexité de Xen, ainsi que son manque de documentation, sont des facteurs qui pourront se retourner contre le projet, à long terme. En effet, il est difficile dans cette situation d'attirer les contributeurs extérieurs, la période d'apprentissage nécessaire pour maîtriser le code source étant très

longue (de l'ordre de quelques mois). Il a fallu effectuer un véritable travail de recherche et de synthèse pour obtenir une bonne idée du fonctionnement interne de Xen, car les documents disponibles sont rarement à jour, et sont même parfois répartis sur plusieurs sites (site officiel, wiki, site de l'université de Cambridge, etc.). Cette complexité est un gros défaut si l'on souhaite un jour modifier le comportement de Xen. La documentation existante, notamment le manuel d'utilisateur, renseigne plus sur les détails de configuration de Xen que sur comment il fonctionne. Or, il est pour nous indispensable que l'on sache comment se comporte dans une situation donnée.

De plus, du fait même de la technologie utilisée par Xen, un modèle à hyperviseur, le projet est condamné à rester en grande partie hors du noyau Linux, seules les modifications suffisamment génériques et peu intrusives ont une chance d'être intégrée au noyau, comme cela a été le cas au mois d'août 2007.

L'utilisation d'un hyperviseur a un autre effet : Xen doit réimplémenter tous les algorithmes d'ordonnancement entre les systèmes invités. En effet, l'hyperviseur étant le logiciel qui contrôle l'accès aux ressources (processeurs, mémoire vive, disques durs, cartes réseau, etc.), c'est à lui que revient la tâche de répartir les accès de manière équitable.

C'est une tâche très complexe, qui est toujours la source de nombreux bugs. Ce travail d'ordonnancement des tâches est déjà effectué par le noyau Linux au niveau des processus du système d'exploitation, et Xen doit refaire le même travail pour les systèmes d'exploitation. Dans le noyau Linux, les modules d'ordonnancement sont l'objet d'années de développement et d'optimisation, il est à craindre que le projet Xen consacre une grande partie de ses efforts de développement à dupliquer les mêmes fonctionnalités pour simplement pouvoir faire cohabiter plusieurs systèmes d'exploitation sur la même machine. Du côté communautaire, le récent rachat de la société XenSource par Citrix inquiète beaucoup la communauté, qui craint de voir XenSource se tourner de plus en plus vers un modèle propriétaire similaire à celui de Virtuozzo, la version libre devenant un simple produit d'appel, aux performances inférieures à la version propriétaire.

Comparativement à Xen, le projet KVM a l'air plus intéressant, tant au niveau de la simplicité d'utilisation que de la complexité du code source. En effet, étant donné que KVM est un simple processus au sein du système hôte, il n'a pas à réinventer tous les algorithmes d'ordonnancement pour obtenir de bonnes performances. La différence peut se résumer simplement par la citation suivante : « Xen doit travailler dur pour obtenir ce dont nous bénéficions gratuitement. » d'après Dor LAOR, développeur de KVM. Ce dernier met ici l'accent sur la différence primordiale entre Xen et KVM : comme KVM s'exécute au sein du système hôte, il bénéficie des fonctionnalités du noyau Linux en ce qui concerne la gestion des processus et des accès aux ressources, alors que Xen doit tout redévelopper.

On retrouve une différence au niveau de la communauté : les contributions externes sont plus nombreuses pour KVM que pour Xen, en dépit de la jeunesse du projet KVM. La différence peut facilement s'expliquer par la jeunesse du projet, la nouveauté étant toujours très attractive, ainsi que par la simplicité — par rapport à Xen — du code source. De plus, le risque de changement de licence est très faible, et quasiment sans risque pour la pérennité du projet. En effet, maintenant que le code du module KVM est intégré au noyau Linux, il sera toujours diffusé avec, et si Qumranet vient à changer la licence du projet, il restera toujours la partie libre intégrée au noyau, que n'importe qui pourra reprendre pour en faire un nouveau projet libre. Toutefois, les performances sont encore à l'heure actuelle en deçà de ce que Xen offre, même s'il y a récemment eu des améliorations en ce sens. Les performances sont le dernier facteur limitatif de KVM pour l'utilisation en production et l'hébergement de serveurs. Sa grande simplicité lui vaut d'être déjà utilisé par beaucoup de monde

pour les tests d'installation et l'utilisation bureautique.

Nous pouvons retenir à la fin de cette étude comparative que KVM est l'avenir des systèmes de virtualisation.

### 2.3.9.5 La bibliothèque Libvirt

Libvirt est une bibliothèque permettant de s'interfacer avec différentes solutions de virtualisation. Écrit en C, il a notamment des liaisons avec d'autres langages tels que : Python, Perl, Ruby, Java etc. Les différents hyperviseurs pris en charge par ce dernier sont :

- LXC - système léger conteneur Linux
- OpenVZ - système léger conteneur Linux
- Virtual Machine Kernel-based/ QEMU (KVM) - hyperviseur open source pour Linux
- Xen - Bare-Metal
- Virtualbox - hyperviseur par Oracle pour Windows, Linux, Mac OS X et Solaris
- VMware ESX et GSX - hyperviseurs pour le matériel Intel
- VMware Workstation et Player - hyperviseurs pour Windows et Linux
- Hyper-V - hyperviseur pour Windows par Microsoft
- PowerVM - hyperviseur par IBM pour AIX, Linux et IBM
- Parallels Workstation - hyperviseur pour Mac Parallels IP Holdings GmbH

Les différents composants se trouvant à la base de libvirt sont :

**Virsh** : c'est une interface ligne de commande qui permet de gérer les machines virtuelles à travers un terminal. Il comprend un ensemble d'options et d'arguments permettant par exemple de lister l'ensemble des machines virtuelles tournant sur l'hyperviseur, de créer une machine virtuelle, d'en supprimer etc.

**Virt-manager** : c'est une interface graphique utilisateur permettant la gestion des machines virtuelles via libvirt. Il vise principalement KVM comme hyperviseur, mais gère également Xen et LXC (conteneurs Linux). Il présente une vue synthétique des domaines de fonctionnement, leurs statistiques d'utilisation du spectacle vivant et des ressources. Il dispose d'assistant permettant la création de nouveaux domaines, et la configuration et l'ajustement de l'allocation des ressources d'un domaine et le matériel virtuel. Un VNC intégré et spectateur de client SPICE présente une console graphique complète pour le domaine invité. Virt-manager dispose des outils de soutien tel que **virt-install** qui est un outil ligne de commande fournissant un moyen facile pour les systèmes d'exploitation de provisions dans les machines virtuelles. Le **virt-clone** est un outil ligne de commande pour le clonage de clients inactifs existants. Elle copie les images disque, et définit une configuration avec le nouveau nom, UUID et l'adresse MAC en montrant les disques copiés. Le **virt-viewer** qui est une interface légère pour interagir avec l'affichage des systèmes d'exploitations invité. Il peut afficher VNC ou SPICE et utilise libvirt pour rechercher les informations de connexion graphique.

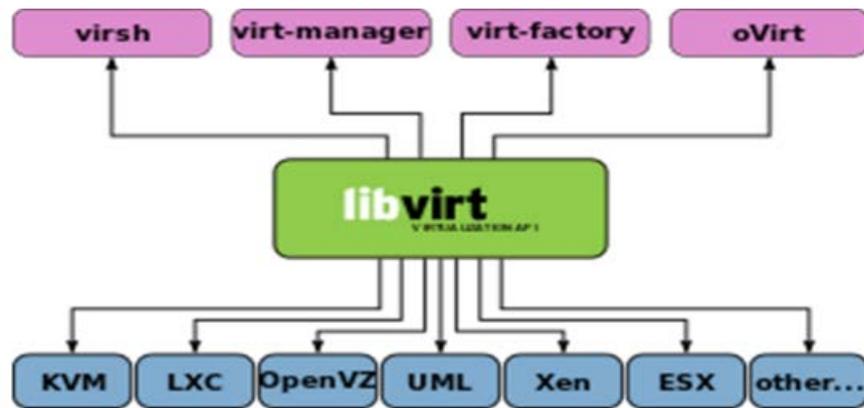


FIGURE II.2.9 – La bibliothèque Libvirt

**Virt-factory** : c'est un ensemble d'applications pour la gestion d'un très grand nombre de machines virtuelles comme cela peut être trouvé dans les centres de données ou fermes de calcul. Le but du projet est de déterminer des moyens nouveaux et efficaces pour lutter contre ces grands environnements. Virt-factory fonctionne par la gestion d'un groupe d'hôtes physiques enregistrés, puis le déploiement de machines virtuelles, qui ont des rôles spécifiques définis par leurs « profils » sur ces hôtes. Ces machines virtuelles peuvent exécuter Xen ou KVM. L'application fournit une interface Web, une API XMLRPC et une interface de ligne de commande. Le soutien à la gestion de base pour l'approvisionnement et le contrôle de la virtualisation (marche / arrêt / pause etc.) est disponible.

**Ovirt** : c'est une application de gestion de la virtualisation. Autrement dit que son interface de gestion peut être utilisée pour gérer les nœuds de matériel, le stockage et les ressources réseau et à déployer et surveiller les machines virtuelles s'exécutant dans un centre de données par exemple. Ovirt est conceptuellement similaire à la plateforme de virtualisation VMWARE vSphere. Ovirt possède plusieurs caractéristiques suivantes :

- Gestion de plusieurs machines virtuelles
- Interface utilisateur sophistiquée permet la gestion de tous les aspects d'un centre de données
- Choix des moyens de répartition des machines virtuelles sur les hôtes
- Migration en direct des machines virtuelles d'un hyperviseur à un autre
- Ajout de nouveaux nœuds de l'hyperviseur facilement et central
- Surveillance de l'utilisation des ressources sur les machines virtuelles
- Gestion des quotas d'utilisation des ressources

L'étude, d'une part, sur les outils libre de virtualisation montre que QEMU/KVM est un outil qui de point de vue technique n'a rien à envier aux outils propriétaires de virtualisation et d'autre part, sur la librairie libvirt permettant de gérer la plus part des outils de virtualisation aussi bien propriétaires que libres montre la maturité et la pertinence de l'utilisation des logiciels dans les organisations virtuelles.

Cet étude a montré que la différence entre les outils propriétaires et libres se fait au niveau de développement des outils d'administration qui ne font qu'utiliser des fonctionnalités existantes qui sont presque les mêmes. En effet, les outils propriétaires utilisent des interfaces conviviales alors que les outils libres mettent beaucoup plus l'accent sur les fonctionnalités que sur les outils de gestion.

Dans le milieu Universitaire, les partisans du libre doivent en tenir compte et inciter les étudiants à s'investir davantage dans le développement des interfaces conviviales d'exploitation des logiciels libres

dans le domaine de virtualisation et cloud Computing.

## 2.4 Conclusion

Dans ce chapitre, un certain nombre de protocoles ont été étudiés car à priori ils ont des propriétés intéressantes pour les organisations virtuelles. Le protocole XMPP à travers le mécanisme PubSub peut être utilisé comme support de collaboration entre les membres d'une organisation virtuelle. Des exemples concrets de la pertinence de ce protocole ont été apportés à travers le projet Archipel et des cas d'utilisation dans le domaine militaire, des réseaux sociaux, et des universités etc. Les protocoles VDI tels que RDP et SPICE, de par leurs fonctionnalités en matière de gestion de multimédia et de la bande passante occupée, nous semblent utiles pour la gestion de la mutualisation et de partage de ressources dans les organisations virtuelles. La richesse de la librairie libvirt et son intégration dans beaucoup de produits libres et propriétaires constitue une piste à explorer pour proposer des solutions de collaboration, de mutualisation et de partage de ressources adaptées.

# Chapitre 3

## Proposition d'un modèle d'organisations virtuelles basé sur le Cloud Computing

### 3.1 Introduction

Plusieurs chercheurs ont prôné l'utilisation de cloud computing pour faciliter le développement des organisations virtuelles. Nous abordons dans ce chapitre, le concept de cloud computing, la présentation de l'outil libre de déploiement de cloud computing openstack ensuite nous allons faire une proposition d'un modèle d'organisation virtuelle permettant aux universités de disposer de plateformes de Travaux pratiques dans le domaine des STEM.

Un modèle décentralisé d'authentification est proposé ainsi qu'un modèle économique à travers un système de facturation.

Enfin, fort des informations acquises à partir du chapitre 2, nous proposons un modèle de découverte dynamique des ressources de l'organisation virtuelle en exploitant les propriétés de l'hyperviseur KVM utilisé dans openstack et celles du protocole XMPP.

### 3.2 Cloud Computing

#### 3.2.1 Historique et définition

La notion de Cloud fait référence à un nuage, tel que l'on a l'habitude de l'utiliser dans des schémas techniques lorsque l'on veut représenter Internet[57][58][59]. Un réseau comme Internet est constitué d'une multitude de systèmes fournissant des services et des informations. Le Cloud Computing est dans cette lignée : un ensemble de services et de données consommables.

Cette notion de consommation a été proposée en 1961, lors d'une conférence au MIT (Massachusetts Institute of Technology), par John McCarthy. John McCarthy suggéra que la technologie informatique partagée (« time-sharing ») pouvait construire un bel avenir dans lequel la puissance de calcul et même les applications spécifiques pouvaient être vendues comme un service public. Cette idée, très populaire dans les années 60, disparu au milieu des années 70 : à l'époque, les technologies matérielles, logicielles et réseaux n'étaient tout simplement pas prêtes. Le Cloud Computing met en

œuvre l'idée d'informatique utilitaire du type service public, proposée par John McCarthy.

L'informatique utilitaire est une version plus commercialisée des grilles de calcul où les fournisseurs de services de haute performance de calcul mettent à la disposition de leurs clients des ressources (exemple : capacités supplémentaires de traitement, de stockage ou les applications à la demande). La définition opérationnelle retenue par le **NIST** (National Institute of Standards and Technology) est la suivante : Le Cloud Computing est un modèle permettant un accès aisé et à la demande, via le réseau, à un ensemble partagé (pool) de ressources informatiques configurables (réseaux, serveurs, stockage, applications et services), qui peuvent être rapidement approvisionnées et mises à disposition avec un minimum d'effort de gestion ou d'interaction avec un fournisseur. Ce modèle favorise la disponibilité. Il est composé de cinq caractéristiques essentielles, repose sur trois modèles de services (SaaS, PaaS et IaaS), et quatre modèles de déploiement (Cloud privé, Cloud communautaire, Cloud public et Cloud hybride).

Le Cloud Computing utilise des technologies telles que la virtualisation du matériel informatique, les grilles, l'architecture orientée services et les services web.

### 3.2.2 Les caractéristiques principales du Cloud Computing

Les caractéristiques fondamentales du Cloud Computing sont les suivantes :

- des services à la demande (ressources en self-service),
- des services mesurés et la facturation à l'usage,
- une bande passante élevée,
- un partage de ressources informatiques (mutualisation des ressources),
- élasticité et une flexibilité des ressources.

**Des services à la demande :** Un utilisateur/client accède depuis une interface web à des ressources informatiques, tels que de la capacité supplémentaire de calcul ou de stockage, sans intervention humaines de la part des fournisseurs de services.

**Des services mesurés et la facturation à l'usage** L'utilisation des ressources du Cloud Computing est mesurée, auditée. Elle fait l'objet de rapports dont les indicateurs de performances sont alignés sur les ressources du client. La règle est qu'un utilisateur est toujours facturé en fonction de sa consommation des services. Par exemple, la taille d'espace de stockage utilisé, le nombre de transactions, le nombre d'entrées et de sorties, la bande passante, la puissance de calcul utilisée, etc.

Les solutions de Cloud Computing offrent un système de contrôle automatique et d'optimisation de l'utilisation des ressources, en fonction du niveau d'abstraction appropriée selon le type de services (stockage, traitement, bande passante, etc.). L'utilisation des ressources peut être surveillée, contrôlée et pilotée en assurant la transparence, aussi bien pour le fournisseur que le consommateur du service.

**Une bande passante élevée** Les capacités actuelles des réseaux rendent accessibles les services Cloud depuis des plateformes clients hétérogènes (PC, téléphones mobiles, ordinateurs portables, PDA, etc.).

**Le partage des ressources** Le fournisseur met en œuvre des ressources qui peuvent être partagées par plusieurs clients et utilisateurs en même temps. Il s'appuie notamment sur le concept de la virtualisation : les systèmes virtuels sont alloués ou réalloués dynamiques selon les besoins du client.

Parallèlement, le concept d'abstraction masque la localisation physique des ressources pour les présenter sous formes de système virtuelle.

**L'élasticité et la flexibilité des ressources** Il est possible d'acquérir rapidement de nouvelles ressources en fonction de l'évolution des besoins du métier. Le client peut soit augmenter la puissance de chaque ordinateur, soit augmenter le nombre d'ordinateurs, soit les deux en même temps. Par ailleurs la gestion des ressources peut être automatisée.

### 3.2.3 Modèles de délivrance et de déploiement de services

L'industrie des services informatiques a pour habitude de casser les services de Cloud Computing selon deux types de modèles :

- **Les modèles de délivrances de services** Ils décrivent les différentes catégories de services accessibles depuis une plateforme de Cloud Computing. Autrement dit, ce sont les types de services que le fournisseur est capable de proposer. Les plus connus sont SaaS, le PaaS et le IaaS, aussi dénommés SPI (Software, Platform, Infrastructure).

- **Les modèles de déploiement de services** Ils se réfèrent à la localisation et à la gestion des infrastructures de Cloud Computing, ils indiquent où la solution de Cloud se situe et pour quel objectif. Les principaux modèles de déploiement sont public, privé, communautaire. Une combinaison des différents modèles sont possibles.

#### 3.2.3.1 Les modèles de délivrance

##### IAAS - INFRASTRUCTURE AS A SERVICE

Il ne faut pas confondre une infrastructure Cloud et un Cloud d'infrastructure :

- une **infrastructure Cloud** correspond à l'ensemble des ressources logicielles et Matérielles qui sont nécessaires à la constitution du Cloud. Cette infrastructure est particulièrement nécessaire lorsqu'une entreprise met en place son propre Cloud en interne.
- un **Cloud d'infrastructure** est le service rendu par le Cloud : une infrastructure virtuelle sur laquelle il est possible de bâtir une solution applicative par exemple. C'est ce qui est fourni à une entreprise dans le cas d'un Cloud d'infrastructure externe.

L'IaaS concerne essentiellement les Clouds d'infrastructure. Ces derniers fournissent à la demande un ensemble de services de niveau « bas », c'est à dire des serveurs, réseaux etc. Cela permet ainsi à une entreprise cliente de pouvoir bénéficier de la puissance d'une infrastructure, ponctuellement, sans devoir investir beaucoup.

Lorsqu'il y a une certitude de variations fortes de charge, ou une incertitude sur la capacité d'une infrastructure à délivrer un service, le Cloud Computing est alors une solution très adaptée : par exemple pour des besoins de type paie, messagerie ou éditique (création massive et ponctuelle de documents). De plus le Cloud peut permettre de tirer parti au maximum du partage du matériel (processeurs, disques etc.).

Dans un Cloud d'infrastructure, l'utilisateur, ou l'entreprise cliente, est maître de son environnement virtuel et peut y installer ce que bon lui semble. De nombreuses offres proposent,

par exemple, l'installation de serveurs virtuels, configurables à la demande, sur lesquels il sera possible d'exécuter ses applications.

### PAAS - PLATFORM AS A SERVICE

Dans ce modèle, le fournisseur met à la disposition des clients des systèmes virtuels, des systèmes d'exploitation, des applications, des services, des plateformes de développement, des transactions et des structures de contrôle. Le fournisseur de service gère l'infrastructure Cloud Computing, les systèmes d'exploitation et les logiciels associés. Le client est responsable de l'installation et de la gestion des applications qu'il déploie. Si l'IaaS concerne essentiellement la production et l'exploitation, le Cloud de niveau PaaS concerne les développeurs et les producteurs d'applications. Donc il donne aux applications un cadre d'exécution qui produira des services SaaS.

### SAAS - SOFTWARE AS A SERVICE

Le Cloud de niveau SaaS représente le plus souvent un catalogue d'applications accessibles en mode service aux utilisateurs ou clients finaux. Dans le mode SaaS, l'usage prime sur la solution : on parle de service de messagerie, de service d'achat dans des boutiques en ligne, de service d'accès à des bibliothèques etc. L'application est déjà construite et opérationnelle, il n'y a pas véritablement de développement mais plutôt du paramétrage ; d'où le succès de ce modèle. La responsabilité du client commence et s'arrête au niveau de la gestion de ses propres données et de l'interaction avec l'utilisateur. Tout ce qui se passe entre l'application et les infrastructures relève de la responsabilité du fournisseur.

Un fournisseur de *software as a service* peut exploiter des services de type *plate-forme as a service*, qui peut lui-même se servir d'*infrastructure as a service*.

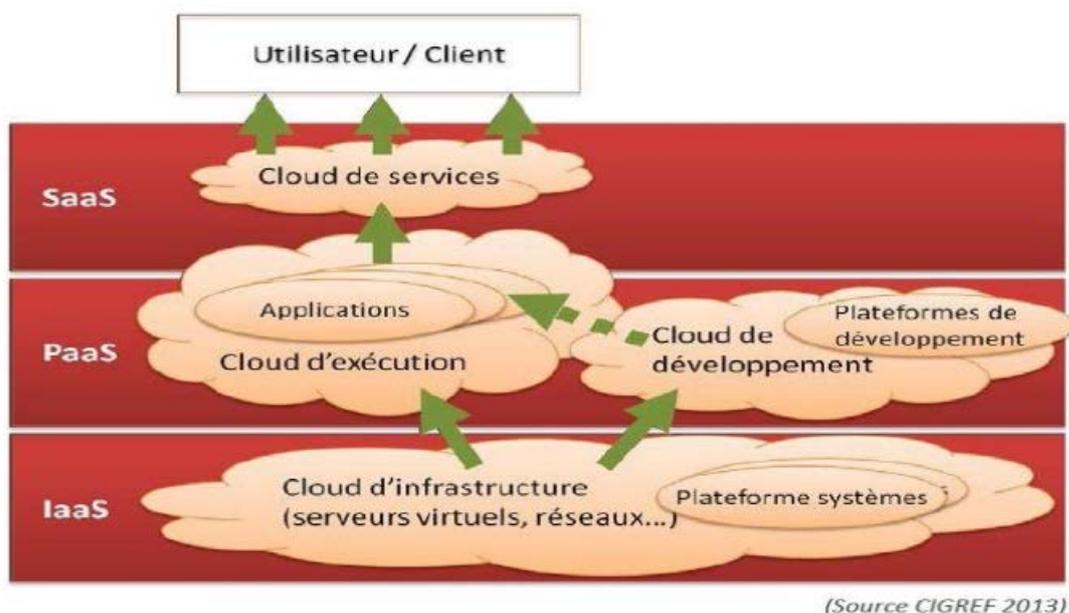


FIGURE II.3.1 – Les principaux modèles de délivrance de services

### 3.2.3.2 Les modèles de déploiement

Il existe quatre modèles de déploiement du Cloud Computing qui correspondent chacun à une architecture particulière :

- Cloud privé
- Cloud public
- Cloud communautaire
- Cloud hybride

**Cloud privé :** Une seule et unique organisation loue ou possède son infrastructure et la gère uniquement pour ses besoins. Elle ne la partage pas avec d'autres organisations. Le service peut être administré par l'organisation elle-même ou par un fournisseur externe.

**Cloud public :** Un fournisseur possède une infrastructure dont il loue les services à plusieurs entreprises ou groupes industriels. Dans ce modèle de déploiement, les fournisseurs gèrent les ressources, comme les applications et le stockage.

Le gros avantage des services publics de Cloud Computing est leur coût raisonnable de mise en œuvre. En effet, les coûts de déploiement du matériel, des applications et du réseau sont pris en charge par le fournisseur, qui leur répartit sur plusieurs clients, donc les fournisseurs réalisent une économie d'échelle.

**Cloud communautaire :** L'infrastructure est partagée par plusieurs organisations et concerne une communauté spécifique qui partage des intérêts communs (une mission, des exigences de sécurité, des obligations légales, etc.)

Il s'agit de partager un environnement afin d'étendre le nombre d'applications et de composants d'infrastructures fournis sous forme de services, ce qui permet de consolider et d'optimiser l'efficacité des opérations, ce que chaque entreprise ne pourrait pas faire individuellement.

Les ONG, le secteur de l'aviation ou de la santé, les laboratoires de recherche et de développement, les universités et les grandes écoles sont friands de nuages communautaires pour partager leurs travaux.

**Cloud hybride :** il s'agit d'une composition de deux ou plusieurs nuages (privé, communautaire ou public). Cette plate-forme constitue une entité unique. Les échanges qui s'y produisent s'effectuent grâce à une technologie standard ou propriétaire qui permet la portabilité des données et des applications.

L'avantage immédiat consiste à fédérer et agréger des données. Des politiques de sécurité ou de gestion des identités. Un autre bénéfice consiste à séparer les ressources, en s'appuyant sur les services de nuages publics pour les applications et des données non critiques, tout en laissant les applications et les données critiques sur un nuage privé.

### 3.2.4 Les principaux composants du Cloud

Un Cloud est essentiellement composé des éléments suivants : les clients, le centre de traitement de données(Datacenter), et de serveurs distribués.

### 3.2.4.1 Les clients

Les clients sont les périphériques que les utilisateurs finaux utilisent pour interagir avec le Cloud. Les clients se répartissent généralement en deux catégories :

- Les appareils mobiles qui sont les tablettes les Smartphones.
- Les clients légers qui sont au sens matériel un ordinateur qui, dans une architecture client-serveur, n'a presque pas de logique d'application. Il dépend donc surtout du serveur central pour le traitement et au sens logiciel sont les navigateurs.

### 3.2.4.2 Centre de traitement de données

Un centre de traitement de données (data center en anglais) est un site physique sur lequel se trouvent regroupés des équipements constituant du système d'information de l'entreprise (ordinateurs centraux, serveurs, baies de stockage, équipements réseaux et de télécommunications, etc.). Il peut être interne et/ou externe à l'entreprise, exploité ou non avec le soutien de prestataires.

C'est un service généralement utilisé pour remplir une mission critique relative à l'informatique et à la télématique. Il comprend en général un contrôle sur l'environnement (climatisation, système de prévention contre l'incendie, etc.), une alimentation d'urgence et redondante, ainsi qu'une sécurité physique élevée.

Un centre de traitement de donnée est toujours constitué de trois composants élémentaires :

**L'infrastructure**, c'est-à-dire l'espace et les équipements nécessaires au support des opérations du centre de traitement de données. Cela comprend les transformateurs électriques, les alimentations sans interruptions (UPS), les générateurs, les armoires de climatisation, les systèmes de distribution électrique, etc.

**Les équipements informatiques** : Ils comprennent les racks, les serveurs, le stockage, le câblage ainsi que les outils de gestion des systèmes et des équipements réseaux.

**Les espaces d'exploitation**, c'est-à-dire le personnel d'exploitation qui pilote, entretient et répare les systèmes IT et non-IT lorsque cela est nécessaire.

### 3.2.4.3 Les serveurs distribués

Les nuages peuvent être configurés pour utiliser des centres de traitements de données repartis géographiquement et des services de réplication entre les centres pour fournir une reprise d'activité pour tous les consommateurs de Cloud. Et Les infrastructures Cloud peuvent être approvisionnées de manière distribuée, en outre si l'infrastructure Cloud a besoin d'être élargie pour ajouter plus serveurs, elle peut être approvisionnée à travers divers emplacements séparés géographiquement.

## 3.2.5 Avantages et inconvénients du Cloud

### 3.2.5.1 Les avantages

Bien que les caractéristiques de base du Cloud Computing soient des avantages indéniables, il faut également tenir compte de ces avantages supplémentaires, même si cette liste n'est exhaustive :

### **Réduction des coûts initiaux**

Développer une infrastructure sur site peut s'avérer une tâche fastidieuse et coûteuse. Il faut commander du matériel onéreux et le payer, puis l'installer et le configurer, le tout devant être effectué largement en amont, avant d'en avoir concrètement besoin. Grâce à l'informatique en nuage, vous ne perdez plus de temps avec toute cette procédure : vous payez uniquement les ressources que vous utilisez de façon variable. Ainsi aucun investissement initial.

### **Réduction des coûts permanents**

La force du Cloud Computing réside dans la possibilité de proposer le même service à un grand nombre d'utilisateurs, et donc de pouvoir amortir les coûts de fonctionnement sur toute la base d'utilisateurs. Au final, le coût d'une solution de Cloud Computing sera donc très raisonnable par rapport à une solution sur mesure.

### **Flexibilité**

Pour faire face une augmentation soudaine des ressources, le Cloud Computing est la solution idéale. Plutôt que d'avoir à acheter, installer et configurer un nouvel équipement, vous pouvez acheter cycles CPU supplémentaires ou le stockage auprès d'un tiers. Étant donné que vos coûts sont basés sur la consommation, vous n'auriez probablement pas à payer autant que si vous devriez à acheter l'équipement. Plus besoin d'anticiper les besoins en termes de capacités d'infrastructure.

### **La fiabilité**

Le caractère évolutif du réseau Cloud et sa capacité à faire du partage de charge, du basculement le rendent extrêmes fiable. En de cela la plupart des composants sont redondés.

### **Vitesse et agilité**

Avec une infrastructure classique, il faut des semaines pour qu'un serveur soit commandé, livré et mis en service. Or, une telle lenteur réprime les innovations. Avec l'informatique en nuage, vous mettez en service de nouvelles ressources dès que vous en avez besoin. Vous pouvez déployer des dizaines, voire des centaines de serveurs en quelques minutes, sans avoir à demander l'aide de qui que ce soit. Cet environnement en libre-service vous permet d'accélérer le développement et le déploiement de vos applications, tout en permettant à vos équipes de tester des nouveautés plus rapidement et plus fréquemment.

### **Accent sur les applications et non sur les opérations**

L'informatique en nuage vous permet de réorganiser vos ressources en redirigeant les investissements depuis les centres de données et les opérations vers les projets novateurs. En particulier, vous pouvez concentrer vos rares ressources en informatique et ingénierie sur des projets qui favoriseront l'expansion de vos activités, au lieu qu'elles soient accaparées par l'infrastructure informatique qui, bien qu'importante, ne permet quasiment jamais de se distinguer de ses concurrents.

### **Maintenance simplifiée et mise à jour**

Le Cloud permet de réduire considérablement à la fois la maintenance matérielle et logicielle pour les organisations de toutes tailles. Avec moins de matériel (moins de serveurs) nécessaires à l'organisation, les coûts de maintenance sont immédiatement réduits. Le système étant centralisé, vous pouvez

facilement appliquer des correctifs et des mises à niveau. Cela signifie que vos utilisateurs ont toujours accès aux dernières versions des logiciels.

### **Augmentation de la sécurité des données**

Et toutes les données que vous stockez dans le nuage Il reste dans le nuage, quelque part. Contrairement à l'informatique de bureau, où un crash de disque dur peut détruire ou rendre indisponible toutes vos données, un ordinateur en panne dans le nuage n'affecte pas le stockage de vos données. Comme les systèmes sont virtuels, on facilement lancer une nouvelle instance et lui octroyer l'espace de stockage.

### **Collaboration accrue**

Les données peuvent être partagées, puisque tout utilisateur du Cloud peut aisément rendre disponibles ses données à un ou plusieurs autres utilisateurs du Cloud. Il est donc possible de créer une plateforme virtuelle collaborative en un temps record.

### **Mobilité**

À partir du moment où les utilisateurs disposent d'une connectivité, ils peuvent travailler de n'importe quel endroit. Cette flexibilité a une influence positive sur l'équilibre entre vie professionnelle et vie privée des employés, ainsi que sur la productivité.

#### **3.2.5.2 Les inconvénients**

##### **Dépendance à la connectivité constante**

En absence de connexion, on n'a plus accès aux ressources.

##### **Nécessite une bonne vitesse de connexion**

Une connexion Internet à faible vitesse, telle que celle qu'on trouve sur le modem analogique, rend difficile l'accès aux services et souvent impossible. Les applications basées sur le Web ont souvent besoin de beaucoup de bande passante pour télécharger les pages, tout comme les documents de grande taille du service Cloud.

**Confidentialité des données** La confidentialité est remise en cause, le gestionnaire de Cloud ayant un accès total à toutes les informations stockées.

On présentera ensuite OpenStack en détails

## **3.3 Présentation de l'outil OpenStack**

OpenStack est une initiative IaaS open source qui permet de créer et de gérer de grands groupes de serveurs privés virtuels dans un environnement de Cloud Computing. Les objectifs de l'initiative OpenStack est de soutenir l'interopérabilité entre les services de Cloud Computing et de permettre aux entreprises de construire des services de Cloud Computing comme ceux d'Amazon dans leurs propres centres de données. OpenStack, qui est disponible gratuitement sous la licence Apache 2.0 et plus de 150 sociétés contribuent au projet : AMD, Intel, Canonical, SUSE Linux, Red Hat, Cisco, Dell, HP, IBM, Yahoo!, Ericsson, VMware, la NASA etc.

## Composants

OpenStack possède une architecture modulaire qui comprend de nombreux composants, les composants utilisent les API RESTful pour communiquer. Les composants majeurs sont les suivants :

- OpenStack Compute, nom de code : Nova
- OpenStack Imaging Service, nom de code : Glance
- OpenStack Identity, nom de code : Keystone
- OpenStack Dashboard, nom de code : Horizon
- OpenStack Network, nom de code : Neutron
- OpenStack Block Storage, nom de code : Cinder
- OpenStack Telemetry, nom de code : Ceilometer
- OpenStack Storage, nom de code : Swift
- OpenStack Orchestration, nom de code : Heat
- OpenStack DataBase, nom de code : Trove

La figure II.3.2 donne l'architecture globale d'OpenStack.

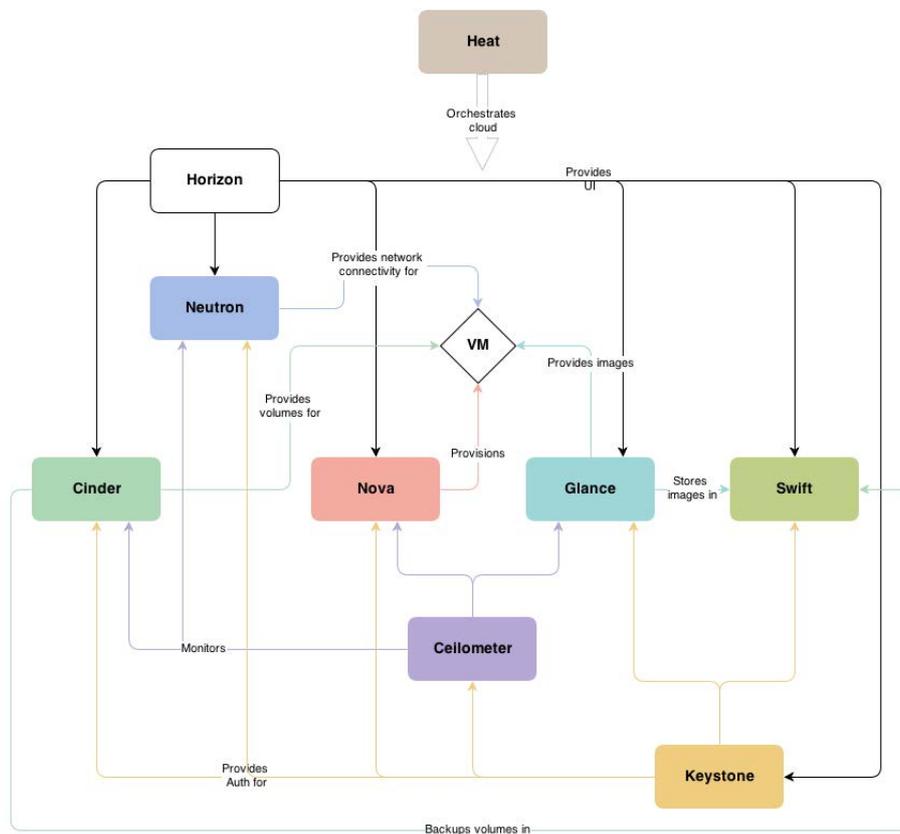


FIGURE II.3.2 – Architecture d'OpenStack

**OpenStack Compute (Nova)** Ce module permet la gestion de larges réseaux de machines virtuelles et d'une architecture redondante et évolutive. Elle fournit une interface d'administration et l'API nécessaire à l'orchestration du Cloud. Elle inclue les gestions des instances serveurs, et les contrôle d'accès.

**OpenStack Imaging Service (Glance)** Ce module fournit les services de stockages, de découvertes, d'enregistrements et de distributions pour les images disques de machines virtuelles. Il supporte

de multiple format d'images (RAW, AMI, VHD, VDI, QCOW2, VMDK, OVF,...). Il fournit également une API compatible REST permettant d'effectuer des requêtes pour obtenir des informations sur les images hébergées par les différents magasins de stockage.

**OpenStack Object Storage(Swift)** Ce module permet la création d'espace de stockage redondant et évolutif pour le stockage de plusieurs pétabits de données. Il ne s'agit pas réellement d'un système de fichier mais est surtout conçu pour le stockage à long terme de gros volumes. Il utilise d'une architecture distribuée offrant plusieurs points d'accès pour éviter les SPOF (Single Point Of Failure).

**OpenStack Block Storage (Cinder)** Ce module fournit un stockage persistant instances Nova. Historiquement ce module faisait partie de Nova et se nommait nova-volume, depuis Folsom nova-volume a disparu pour laisser sa place à Cinder.

**OpenStack Networking (Neutron)** Ce module permet la mise en place de réseaux virtuels. Historiquement connu sous le nom de Quantum il fut renommé en Neutron dans la version d'Havana. Permet la connectivité réseau en tant que service pour d'autres OpenStack services, tels que OpenStack Compute. Le service fonctionne en permettant aux utilisateurs de créer leurs propres réseaux et de leurs attacher des interfaces. Neutron a été conçu à partir d'une architecture de type enfichable permettant de supporter de nombreuses technologies réseaux en occurrence VSwitch, VLAN, etc. Il propose de nouvelles technologies comme VXLAN, Open vSwitch tout en conservant la possibilité d'utiliser les bridges Linux.

**OpenStack Identity (Keystone)** Keystone est un annuaire qui centralise toutes les authentifications et autorisations nécessaires aux multiples services d'OpenStack C'est le service d'identité utilisée par OpenStack pour l'authentification et les autorisations des utilisateurs. Chaque service de Keystone est traité par un backend enfichable qui permet d'utiliser le service de manière spécifique à celui-ci. La plupart de ces services supportent des backends standard comme LDAP ou SQL.

**OpenStack Dashboard (Horizon)** Afin de rendre l'utilisation d'OpenStack plus user friendly, la communauté a développé une interface web permettant de gérer facilement la création d'instance, de volume, d'image, etc. Horizon est un tableau de bord permettant la gestion du Cloud OpenStack via une interface web. Ce Dashboard est développé en Python, basé sur le Framework Django, il utilise le module WSGI (en) d'Apache.

**OpenStack Telemetry (Ceilometer)** Ce module permet de mesurer la consommation d'un projet sur une plate-forme OpenStack. Ceilometer peut-être utilisé comme outil de facturation. Ceilometer fournit un point de contact unique pour les systèmes de facturation, en fournissant tous les compteurs dont ils ont besoin pour établir la facturation des clients, pour tous les composants actuels et futurs OpenStack.

**Openstack orchestration (Heat)** C'est le projet principal dans le programme OpenStack orchestration. Il met en œuvre un moteur d'orchestration qui peut lancer plusieurs applications composites de Cloud basés sur des modèles sous la forme de fichiers texte qui peuvent être traités comme code.

**OpenStack DataBase (Trove)** C'est la base de données en tant que service pour OpenStack. Il est conçu pour fonctionner exclusivement sur OpenStack, avec l'objectif de permettre aux utilisateurs d'utiliser rapidement et facilement les caractéristiques d'une base de données relationnelle sans la charge de traitement des tâches administratives complexes. Utilisateurs de Cloud et les administrateurs de base de données peut fournir et gérer des instances de bases de données multiples en fonction des besoins. Initialement, le service mettra l'accent sur la fourniture de l'isolement des ressources à haute performance tout en automatisant les tâches administratives complexes, y compris le déploiement, la configuration, la correction, les sauvegardes, les restaurations, et le suivi.

Il est à noter qu'OpenStack intègre aujourd'hui le protocole SPICE/Html5 permettant d'accéder à des machines virtuelles à travers un navigateur.

Ce point est essentiel car il donne la possibilité d'utiliser le système de fédération d'identités pour gérer une infrastructure de cloud autour d'OpenStack.

## 3.4 Proposition d'un modèle

### 3.4.1 Les motivations du modèle proposé

Les études en STEM nécessitent que des travaux pratiques et des expériences soient effectués pour permettre aux étudiants de mieux comprendre les concepts enseignés.

Dans beaucoup de pays en voie de développement, des universités sont créées et beaucoup offrent les mêmes types de formation.

Sachant que les matériels de travaux pratiques coûtent cher, et à défaut de ces matériels, la formation se fait au rabais.

On peut avoir dans chaque université une compétence humaine pointue qu'on ne peut utiliser dans une autre université. Il est important pour amorcer le développement de disposer pour chaque domaine des STEM, des expériences et des plateformes de travaux pratique conçus par les meilleurs enseignants du domaine.

Les expériences et les plateformes dans chaque université devraient être mutualisées et partagées sur le plan national.

Pour fixer les idées, nous partons de l'hypothèse que chaque université dans un pays dispose des compétences pointues dans un domaine donné.

Prenons l'exemple du Sénégal. Chaque université choisit son domaine d'excellence et propose des travaux pratiques à partager et à mutualiser.

TABLEAU II.3.1 – Choix des universités dans le modèle proposé

Université	Domaine de compétences choisi
UCAD	Réseaux Télécoms
UGB	Électronique
UADB	Développement logiciels
UT	Médecine
UASZ	Agriculture

D'une part, grâce au développement des TIC, beaucoup d'équipements de travaux pratiques et de recueils des résultats d'expériences ont des interfaces de gestion accessibles via un ordinateur.

D'autre part, des nouveaux protocoles d'accès à des machines virtuelles tels que SPICE offrent la possibilité de pouvoir manipuler des périphériques d'ordinateurs à distance et ceci à travers des interfaces web.

C'est pourquoi le cloud computing nous semble pertinent pour héberger les plateformes de travaux pratiques à partager et à mutualiser.

Dans le cas du Sénégal, les universités sont interconnectées par fibre optique dans le réseau SnRER ce qui permet aux universités de communiquer avec une bonne bande passante.

### 3.4.2 Proposition du modèle

Le modèle que nous proposons repose sur l'outil OpenStack permettant de bâtir une infrastructure de cloud computing.

#### 3.4.2.1 Architecture du modèle

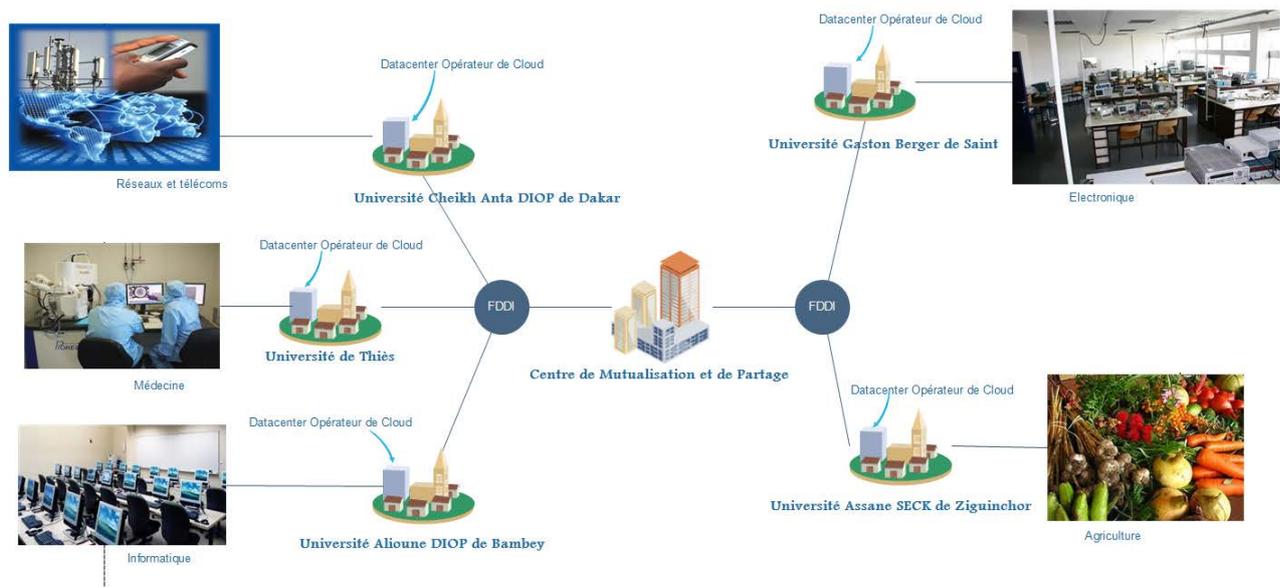


FIGURE II.3.3 – Architecture du modèle proposé

Chaque université dispose d'un cloud computing de même que le centre de mutualisation et de partage du ministère de l'enseignement supérieur et de la recherche.

Pour une efficacité organisationnelle, nous proposons que chaque cloud computing soit géré par un opérateur privé de cloud.

Chaque opérateur de cloud dans une université contractualise avec les meilleurs formateurs de cette université pour la conception et le développement de la plateforme de travaux pratiques.

#### 3.4.2.2 Les avantages du choix des opérateurs de cloud

Les avantages tirés des opérateurs de cloud sont les suivants :

- Motivation des enseignants participant à la conception de la plateforme

- Motivation des étudiants de niveau supérieur d'études cooptés par leur enseignant pour participer à la conception de la plateforme
- Disponibilité des ressources et pérennité du projet

Nous proposons que tous les opérateurs de cloud aient l'obligation de se constituer en organisation virtuelle.

Chaque opérateur de cloud a accès aux ressources d'un autre opérateur en cas de besoin.

Ainsi, il est important qu'un système de facturation soit mis en place fonctionnant de la même manière que celui utilisé par les opérateurs de télécoms interconnectés à travers un pays : chaque opérateur pourra estimer les coûts de ses ressources utilisées par d'autres opérateurs de cloud et l'organisation virtuelle.

### 3.5 Preuve de la faisabilité et de la pertinence techniques du modèle

Pour montrer la faisabilité de notre modèle, nous avons installé cinq (5) infrastructures de cloud computing avec l'outil OpenStack.

Dans le cadre de ce travail, nous allons décrire les cas UCAD et UADB.

#### 3.5.1 Plateforme de travaux pratiques de l'UCAD

Elle est composée entre autres d'une plateforme de travaux pratiques sur les réseaux 4G intégrant OpenIMScore, Asterisk et un système de billing voix et données.

La figure II.3.4 montre représente le réseau virtuel de la plateforme de TP dans le cloud.

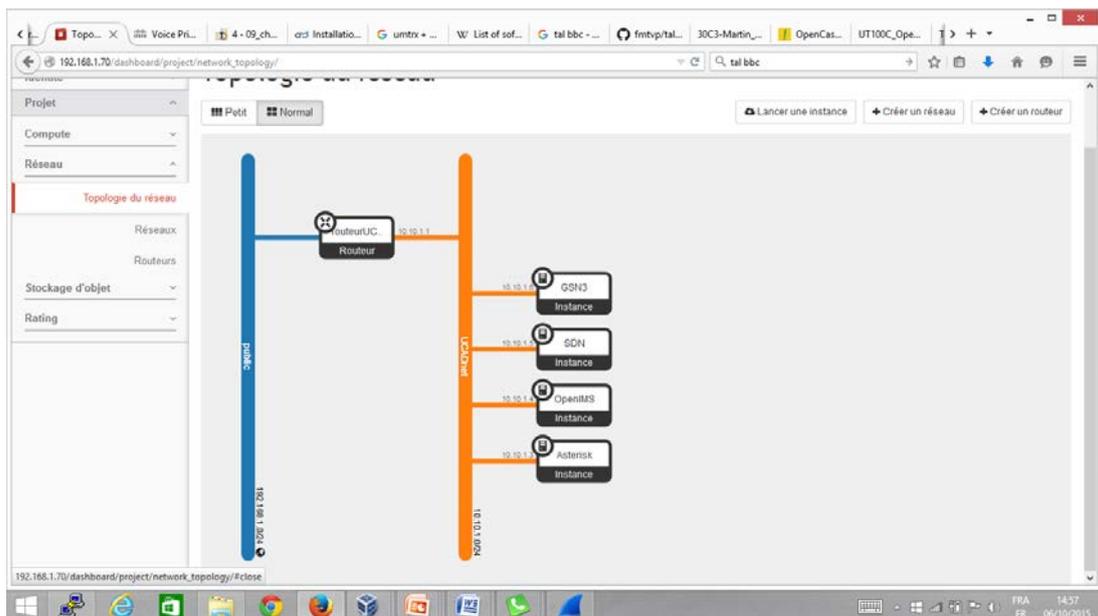


FIGURE II.3.4 – Réseau virtuel de l'UCAD

La figure II.3.5 montre les instances de machines virtuelles utilisées dans la plateforme de TP.

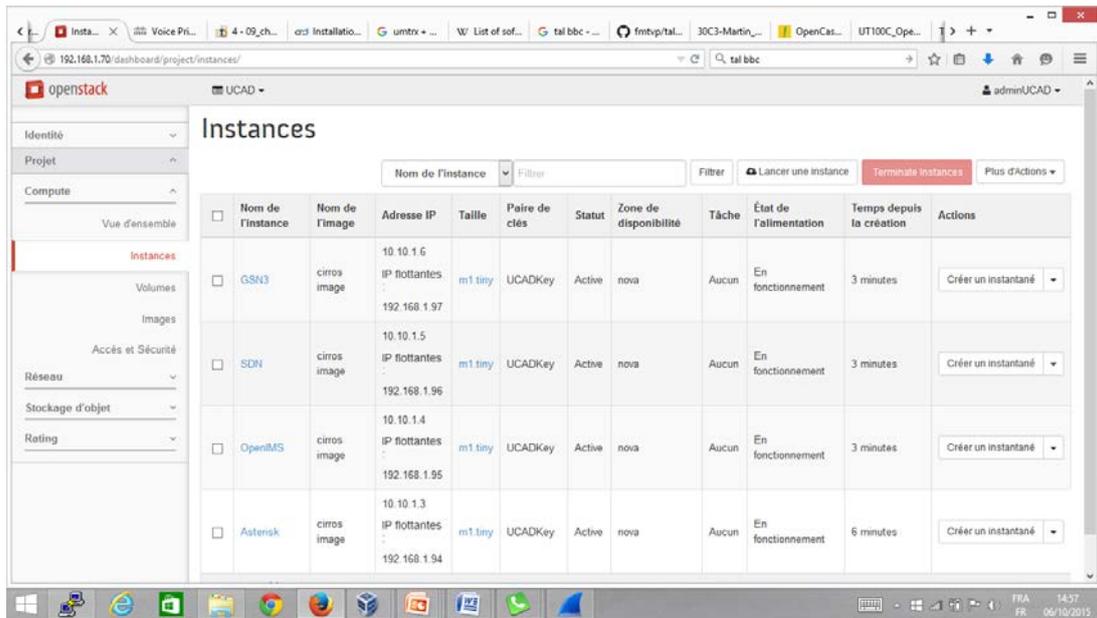


FIGURE II.3.5 – Les instances des machines virtuelles de l'UCAD

La figure II.3.6 présente l'interface d'accès web à la plateforme.

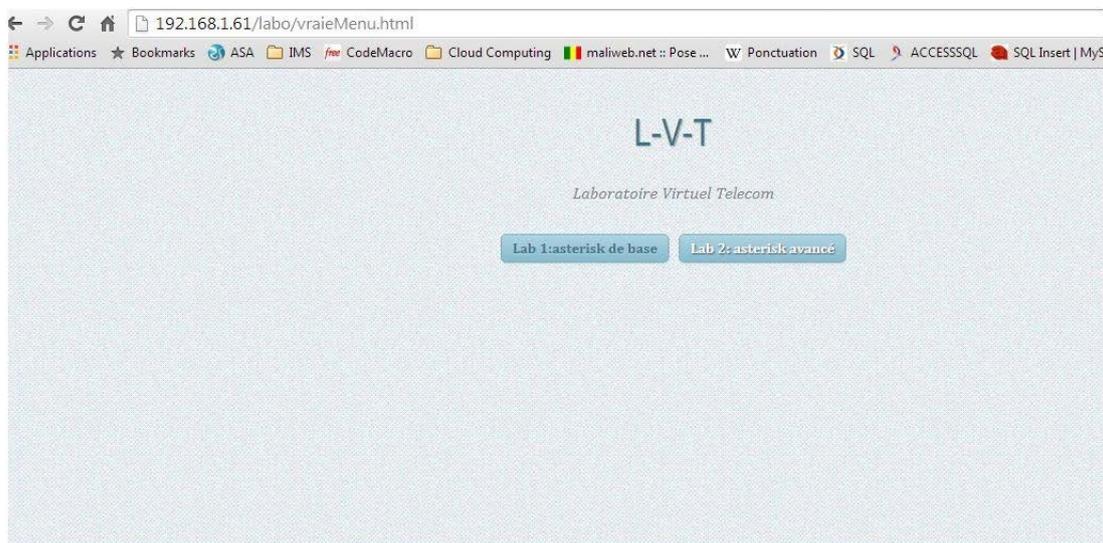


FIGURE II.3.6 – Interface Web d'accès à la plateforme

### Quelques résultats obtenus

On s'est mis à la place de deux étudiants de télécommunications et réseaux qui veulent faire des TP de réseau 4G.

L'accès leur est donné à travers l'interface web comme le montre la figure II.3.7.

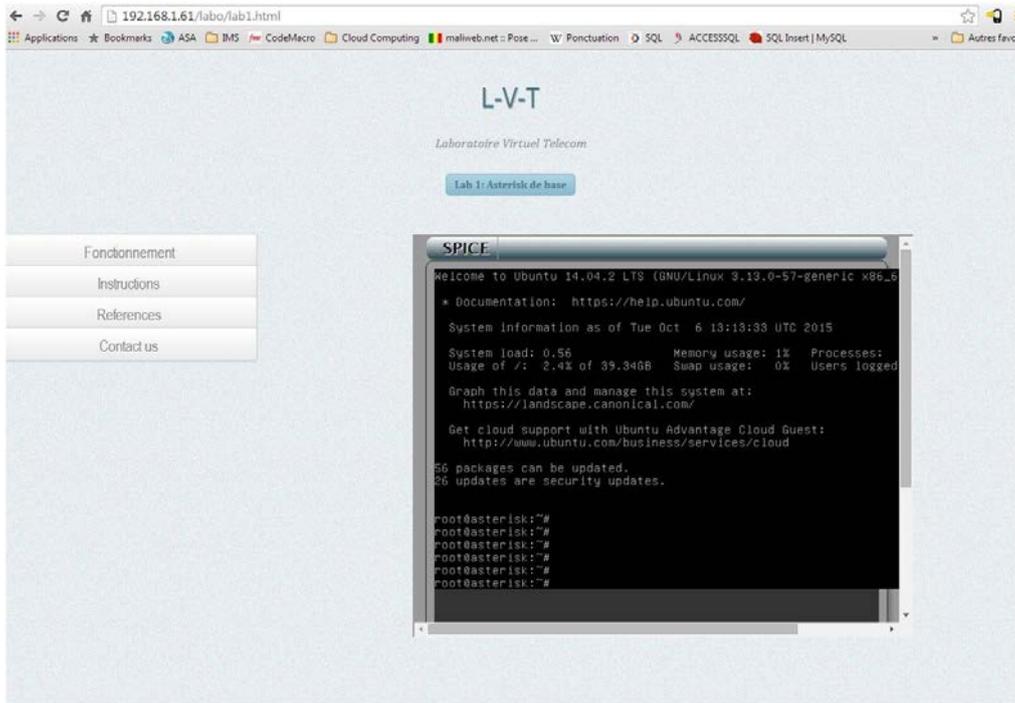


FIGURE II.3.7 – Accès d'un étudiant à la plateforme de TP

Les étudiants ont paramétré la plateforme IMS (OpenIMSCore) et ont créé des comptes utilisateur.

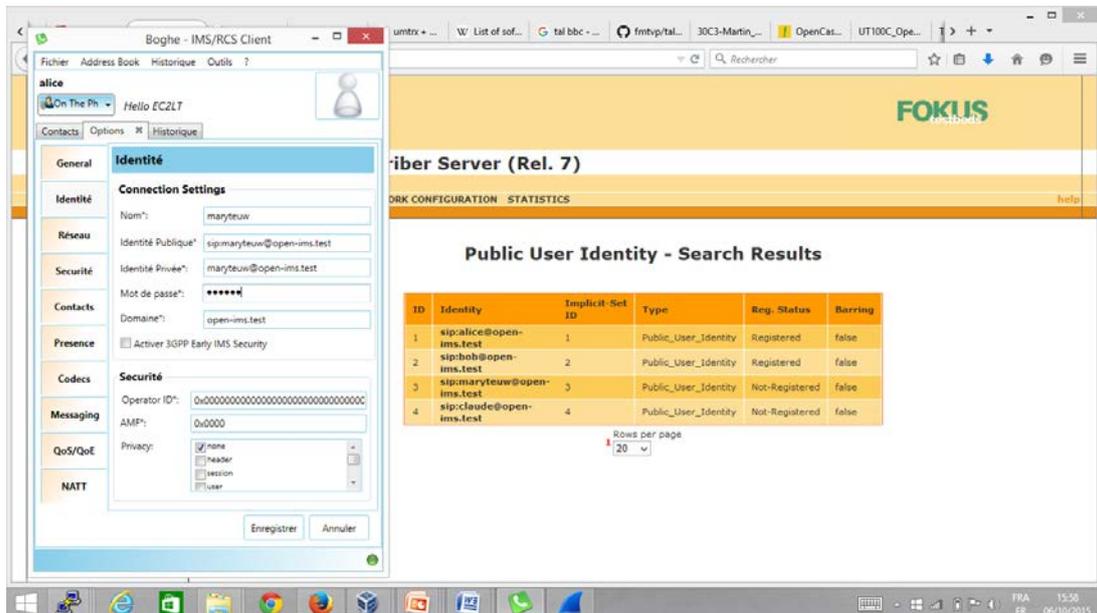


FIGURE II.3.8 – Interface montrant les utilisateurs connectés dans OpenIMSCore

Les appels vidéo IMS entre Mary Teuw et Claude passent comme le montre l'interface du terminal de l'étudiant jouant le rôle de Claude.

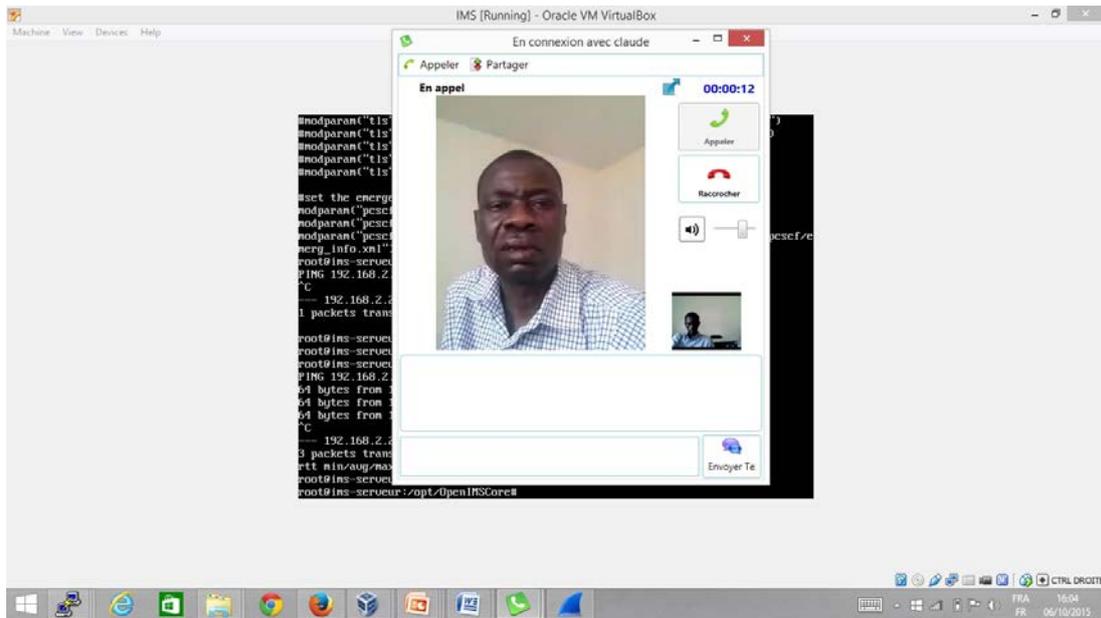


FIGURE II.3.9 – Appel vidéo entre deux utilisateurs

### 3.5.2 Plateforme de travaux pratiques de l'UADB

Elle est composée d'un serveur de développement collaboratif à base de Saros tel que nous l'avons conçu dans le chapitre 4 . et un serveur ZPanel permettant de mettre à disposition des développeurs des plateformes pré-configurées en termes de réseaux, de logiciels nécessaires.

La figure II.3.10 montre représente le réseau virtuel de la plateforme de TP dans le cloud.

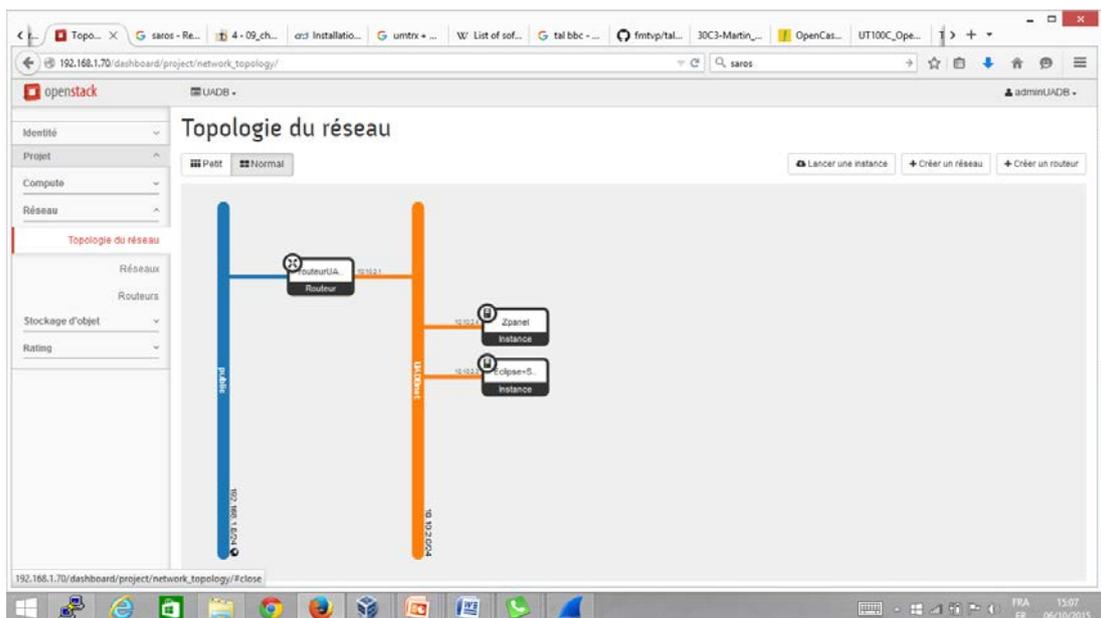


FIGURE II.3.10 – Réseau virtuel de l'UADB

La figure II.3.11 montre les instances de machines virtuelles utilisées dans la plateforme de TP.

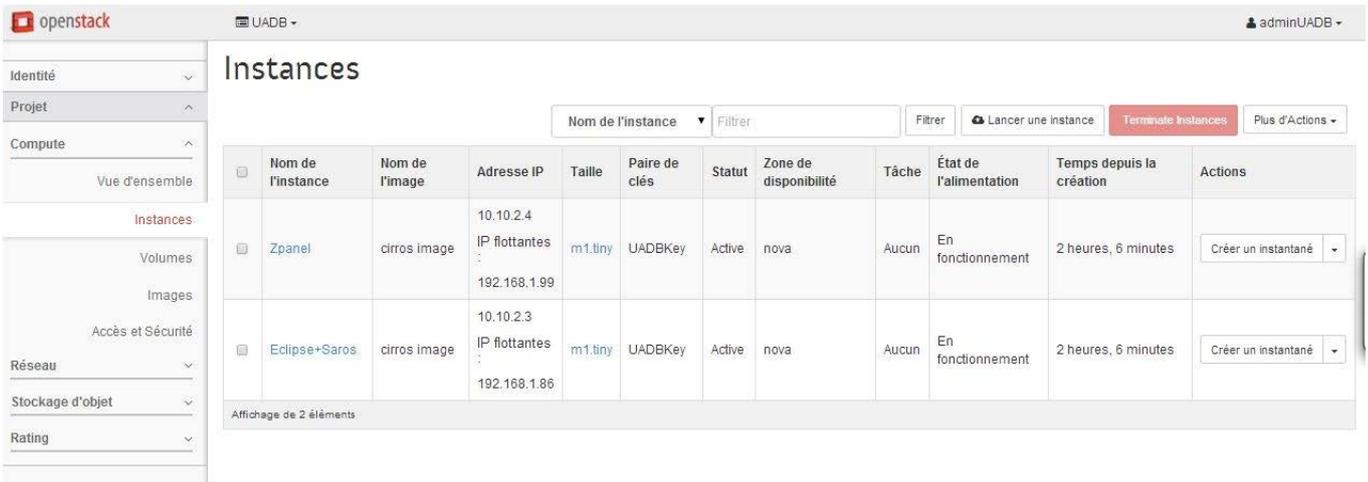


FIGURE II.3.11 – Les instances des machines virtuelles de l'UADB

### 3.5.3 Pérennité du modèle : Proposition d'un système de facturation

Pour permettre aux différents opérateurs de cloud implantés dans les universités de pouvoir facturer l'utilisation de leurs ressources à des acteurs de l'université sénégalaise, nous proposons un système de facturation basé sur l'outil CloudKitty qui est développé en Python.

En effet, cloudKitty exploite les données qui sont recensées par le composant Ceilometer d'OpenStack.

La figure II.3.12 montre l'interaction entre OpenStack et cloudKitty en vue de la facturation.

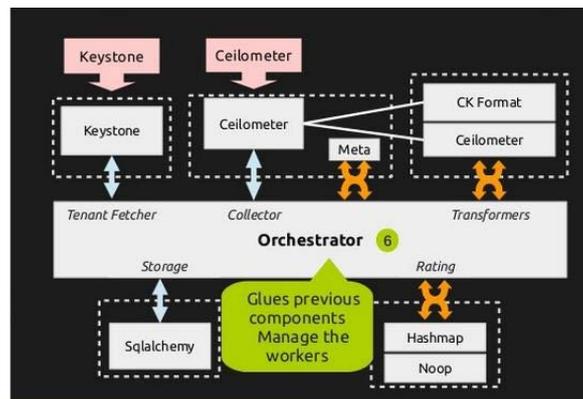


FIGURE II.3.12 – Interaction entre CloudKitty et OpenStack

Dans la figure II.3.13, nous montrons le positionnement du système de facturation global entre opérateurs de cloud de différentes universités. Nous proposons que ce système global de facturation soit hébergé au niveau du centre de mutualisation et de partage du Sénégal. Il permettra aux opérateurs de faire de la conciliation c'est à dire ce que l'un doit à l'autre après avoir utilisé ses ressources.

Chaque opérateur de cloud pourra aussi implémenter son propre système de facturation à base de cloudKitty ou d'un autre outil.

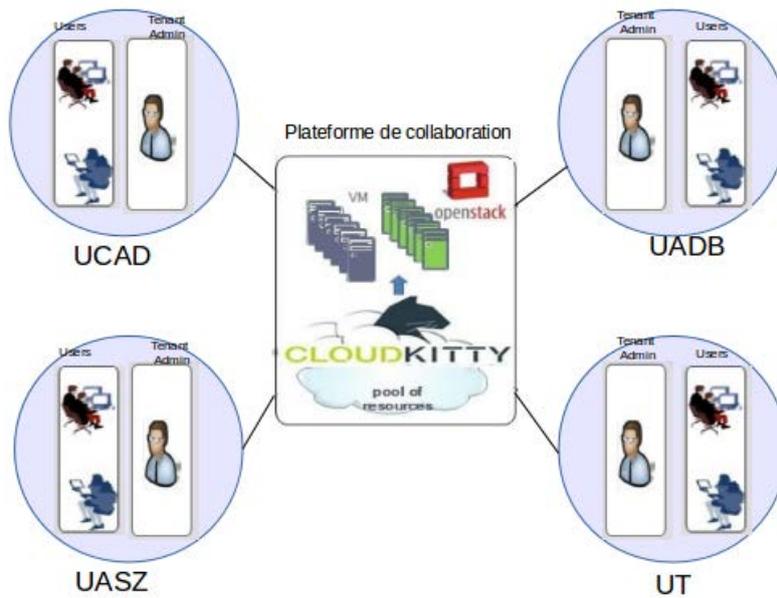


FIGURE II.3.13 – Architecture d'intégration d'une solution de facturation dans une OV basée sur le cloud

La figure II.3.14 montre l'interface de facturation des services avec CloudKitty.

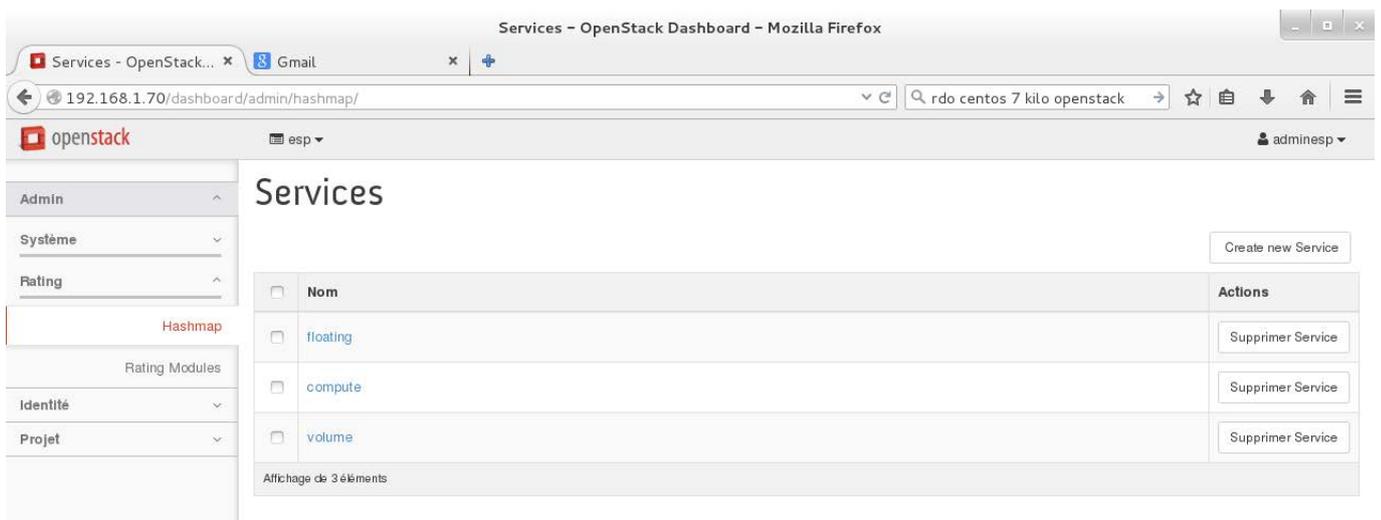


FIGURE II.3.14 – Les services facturés

La figure II.3.15 montre un exemple de facture à l'intention d'un utilisateur.

The screenshot shows a web interface for launching a cloud instance. On the left, there are several form fields: 'Zone de disponibilité' (nova), 'Nom de l'instance' (ubuntu), 'Gabarit' (m1.small), 'Nombre d'instances' (1), 'Source de démarrage de l'instance' (Démarrage depuis une image), and 'Nom de l'image' (ubuntu (244,1 Mo)). On the right, there is a section titled 'Spécifiez les détails de démarrage d'une instance.' with a table 'Détails du gabarit' showing specifications for 'm1.small': 1 VCPUs, 20 Go Disque racine, 0 Go Disque éphémère, 20 Go Disque total, and 2,048 Mo RAM. Below this is a 'Limites du projet' section with three progress bars: 'Nombre d'instances' (0 sur 10 utilisés), 'Nombre de VCPUs' (0 sur 20 utilisés), and 'RAM totale' (0 sur 51 200 Mo utilisés). At the bottom right, there is a 'Price' field showing '0.0000240000000000 \$' and two buttons: 'Annuler' and 'Démarrer'.

FIGURE II.3.15 – Facturation lors du lancement de l'instance

L'outil cloudKitty pour le moment a des fonctionnalités limitées. On pourra étendre ces fonctionnalités à volonté car il est libre et développé dans un langage facile Python.

### 3.6 Méthode d'authentification décentralisée pour l'accès aux ressources de l'organisation virtuelle

Étant entendu que tous les opérateurs de cloud des universités se sont constitués en organisation virtuelle autour du cloud computing OpenStack, la mise en place d'une fédération d'identités permettra à un utilisateur d'une université d'accéder à une plateforme de travaux pratiques d'une autre université et se faire authentifier par son université d'origine. C'est ce que nous appelons une méthode d'authentification décentralisée.

Tout ceci doit être transparent pour l'utilisateur final qui n'a pas besoin de savoir nécessairement dans quelle université se situe la ressource sollicitée.

#### 3.6.1 Conception de la solution

Dans ce modèle, nous commençons d'abord par définir les concepts d'organisation d'origine et d'organisation détentriche de la ressource au sein d'une organisation virtuelle.

L'organisation d'origine détient et gère les informations d'identification concernant un utilisateur. Dans notre approche, nous intégrons le concept de la délégation de l'authentification qui consiste à authentifier l'utilisateur depuis son organisation d'origine. Une organisation membre de l'organisation virtuelle qui détient la ressource sollicitée par un utilisateur est appelée organisation détentriche de la ressource ou fournisseur de service. Une organisation d'origine peut être une organisation détentriche au cas où elle détient la ressource sollicitée. Dans notre approche, nous intégrons les clouds dans l'organisation

virtuelle. Ainsi, une organisation membre souhaitant partager des ressources (machines virtuelles, espaces de stockage, applications, base de données d'expérimentation etc.) va implémenter une solution de cloud computing. Pour permettre l'accès à ces ressources fournies au sein du cloud, nous proposons une méthode d'authentification décentralisée permettant à tout utilisateur de s'authentifier d'abord auprès de son organisation d'origine pour accéder à une infrastructure cloud partagée. La figure II.3.16 illustre le principe d'authentification d'un utilisateur voulant accéder à une ressource d'une organisation membre de l'organisation virtuelle. L'organisation détentrice sera matérialisée pour le cloud qui héberge les ressources partagées.

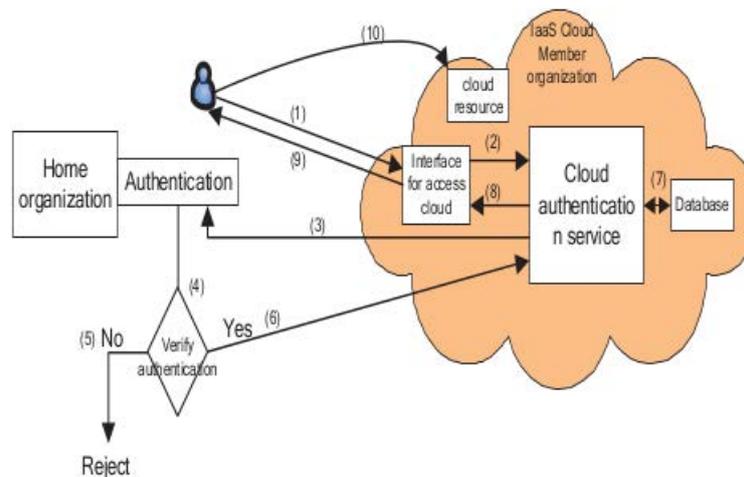


FIGURE II.3.16 – Principe d'authentification

La ressource principale ici c'est l'infrastructure de cloud computing IaaS avec les composants suivants : une interface d'accès au cloud, un service d'authentification interne du cloud et une base de données contenant les identités et attributs des utilisateurs du cloud. La gestion de l'authentification au sein de l'organisation virtuelle obéit à la démarche suivante :

- (1) : l'utilisateur effectue une requête d'accès à une ressource du cloud d'une organisation membre de l'OV.
- (2) : L'interface d'accès au cloud reçoit la requête de l'utilisateur et la transmet au service d'authentification du cloud.
- (3) : Le service d'authentification contacte l'organisation d'origine de l'utilisateur pour authentification.
- (4) : L'organisation d'origine détentrice des identités et attributs de l'utilisateur authentifie ce dernier.
- (5) : Si l'authentification n'est pas réussie, la requête est rejetée.
- (6) : Si l'authentification est de succès, l'organisation d'origine envoie au service d'authentification la preuve d'authentification de l'utilisateur et quelques attributs si nécessaire.
- (7) : Le service d'authentification crée un profil temporaire de l'utilisateur avec une clé de session.
- (8), (9) : Le service d'authentification envoie à l'utilisateur via l'interface d'accès au cloud toutes les informations d'accès à la ressource. Ainsi, l'utilisateur pourra s'approvisionner en ressources.

### 3.6.2 Utilisation de shibboleth dans la gestion

Shibboleth est un projet Internet2 conçu pour mettre en œuvre un protocole ouvert basé sur des normes pour un transfert en toute sécurité entre sites collaboratifs [60]. Il désigne à la fois une norme et un produit (open source). C'est une extension de SAML qui enrichit ses fonctionnalités de fédération d'identités en facilitant pour un ensemble de partenaires la mise en place de deux fonctionnalités importantes, la délégation d'authentification et la propagation d'attributs. Shibboleth a été conçu pour répondre aux besoins des communautés de l'enseignement supérieur. Pour gérer le problème d'authentification, nous intégrons, Shibboleth pour permettre d'une part à chaque administrateur membre de gérer l'identité et l'authentification de ses propres utilisateurs ; et d'autre part de shibboliser l'interface de gestion d'accès du système cloud. Le processus d'authentification de base de shibboleth, basé sur le langage SAML est illustré dans les travaux [60] [61] à travers la figure II.3.17.

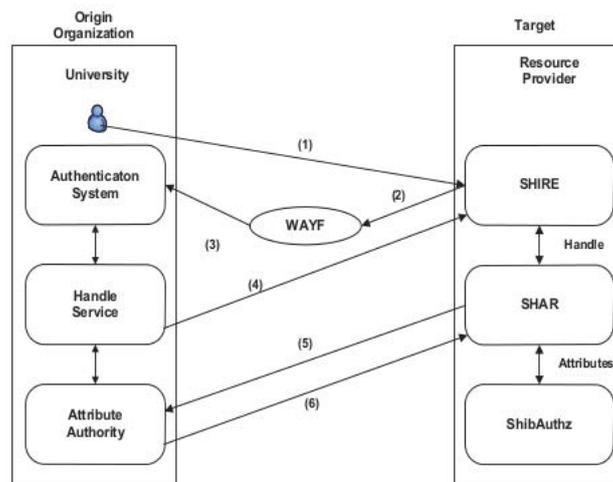


FIGURE II.3.17 – The basic shibboleth authentication process

Nous allons nous inspirer des travaux de ces auteurs pour intégrer shibboleth dans le cloud afin d'apporter une réponse à la problématique de la gestion décentralisée.

Un enseignant d'une université peut se connecter à une infrastructure cloud IaaS d'une autre université partenaire pour s'auto-provisionner lui-même en ressources virtuelles. Pour cela, nous implémentons un service de découverte WAYF pour rediriger un utilisateur vers son organisation d'origine pour authentification. La figure II.3.18 illustre un tel scénario.

### 3.6.2.1 Proposition de modèles de plateformes technologiques pour les universités numériques

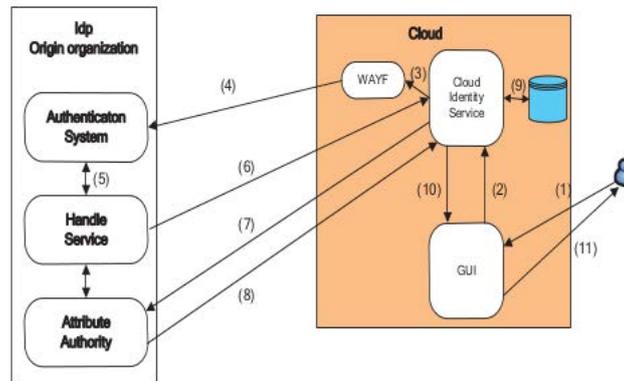


FIGURE II.3.18 – Architecture d'intégration de Shibboleth dans le cloud

### 3.6.2.2 Service keystone

Parmi ces composants, nous nous intéressons au keystone, responsable de l'authentification et de la gestion des utilisateurs d'openstack.

Le composant keystone (Identity) est le service d'identité utilisé par OpenStack pour l'authentification et les autorisations des utilisateurs. Chaque service de Keystone est traité par un backend enfichable qui permet d'utiliser le service de manière spécifique à celui-ci. La plupart de ces services supportent des backends standard comme LDAP ou SQL. Keystone n'est pas une application web mais intègre par défaut un serveur web nommé Eventlet. Cependant, Eventlet ne prend pas en compte les systèmes d'authentification autres que base-auth alors qu'idéalement les organisations partenaires devraient pouvoir être en mesure d'utiliser leurs méthodes d'authentification traditionnelles comme kerberos ou infrastructure à clé publique (PKI) pour échanger avec le keystone d'openstack. De plus Eventlet ne supporte pas IPV6 alors que pour les déploiements basés sur le cloud, les gens ont besoin d'un grand nombre d'adresses IP routables. Par conséquent, il est souhaitable de remplacer Eventlet par un serveur apache qui prend en considération les limitations de Eventlet.

### 3.6.2.3 Shibbolisation de keystone

Pour accéder aux ressources partagées dans le cloud, les utilisateurs externes vont s'auto-provisionner à travers le dashboard. Mais avant d'accéder au dashboard, les utilisateurs doivent s'authentifier auprès du module keystone, responsable de la gestion de l'authentification et de l'autorisation. Après authentification, ce dernier octroie un jeton qui sera utilisé pour les requêtes ultérieures vers les autres services (Nova, Neutron, Glance, Cinder, etc.). Avant de proposer une solution permettant de shibboliser keystone, nous allons décrire le processus d'authentification de keystone.

### Processus d'authentification de keystone

Il est primordial de rappeler le processus d'authentification de keystone qui selon le contenu de la variable `REMOTE_USER` offre ou non un jeton pour l'accès aux services.

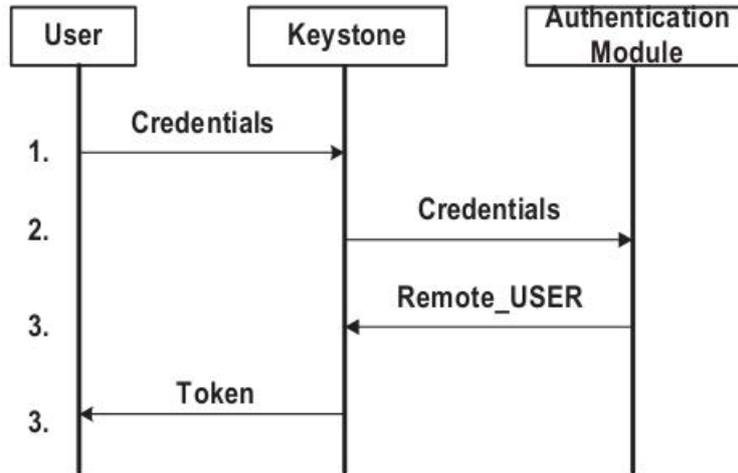


FIGURE II.3.19 – Processus d'authentification keystone

Keystone récupère les credentials de l'utilisateur pour les envoyer au module d'authentification qui peut interne ou externe au keystone. Le module d'authentification authentifie l'utilisateur et envoie au keystone une variable d'environnement REMOTE\_USER. Si la variable est renseignée, l'authentification est de succès sinon si le contenu de la variable est vide l'authentification n'a pas réussie.

### HTTPD comme moteur d'authentification

Dans ce cas de figure, le moteur d'authentification est remplacé par HTTPD comme l'illustre la figure II.3.20.

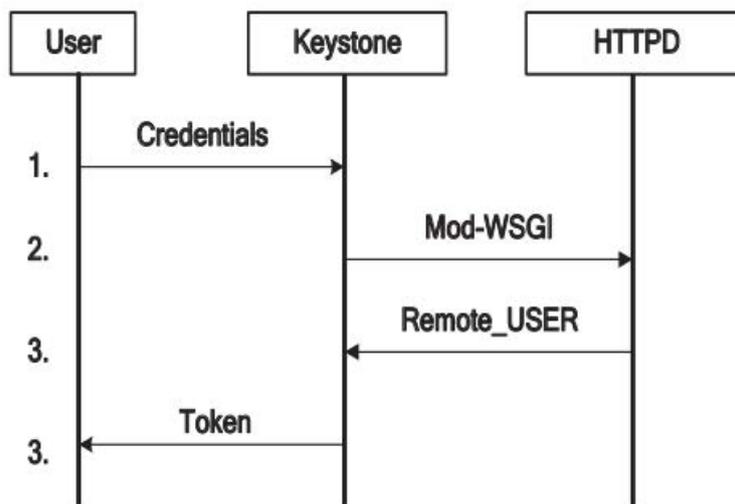


FIGURE II.3.20 – Keystone avec HTTPD comme moteur d'authentification

Keystone intègre un module python SGI (service gateway interface) qui lui permet de communiquer avec le module WSGI d'apache pour renseigner la variable REMOTE\_USER Apache intè-

gre plusieurs modules d'authentification lui permettant d'utiliser n'importe quelle base d'information comme MySQL, LDAP, Kerberos, etc.

## Hébergement de keystone dans HTTPD

Le service keystone gère l'authentification des utilisateurs de manière centralisé, ce qui constitue un obstacle un obstacle majeure dans les organisations virtuelles. Pour gérer le problème de l'authentification centralisé, nous proposons d'intégrer shibboleth dans openstack pour mettre en place une fédération d'identités inter-clouds. Dans cette partie, nous comptons shibboliser l'interface de gestion d'openstack à travers le service keystone pour que chaque administrateur puisse créer librement des instances sans avoir à centraliser l'authentification. Ainsi en shibbolisant keystone, ce dernier devient un keystone SP (Service Provider) intégrant le module WAYF pour permettre à tout utilisateur de choisir son organisation d'origine pour authentification. Après authentification d'un utilisateur de l'organisation virtuelle auprès de son organisation d'origine, keystone devra être en mesure d'accepter le système d'authentification externe et créer dans sa base locale un projet pour cet utilisateur (pour une toute première tentative).

Pour cela, nous proposons d'externaliser l'authentification de keystone en intégrant keystone dans apache. Un site virtuel est créé dans apache avec comme méthode d'authentification shibboleth. L'utilisateur entre l'URL du site virtuel. Ce dernier intégrant le WAYF le redirige vers son IDP d'origine pour authentification. Si l'authentification est de succès, la variable REMOTE\_USER sera renseignée par l'ID. Keystone accepte l'authentification externe et attribue un jeton à l'utilisateur pour l'accès aux ressources.

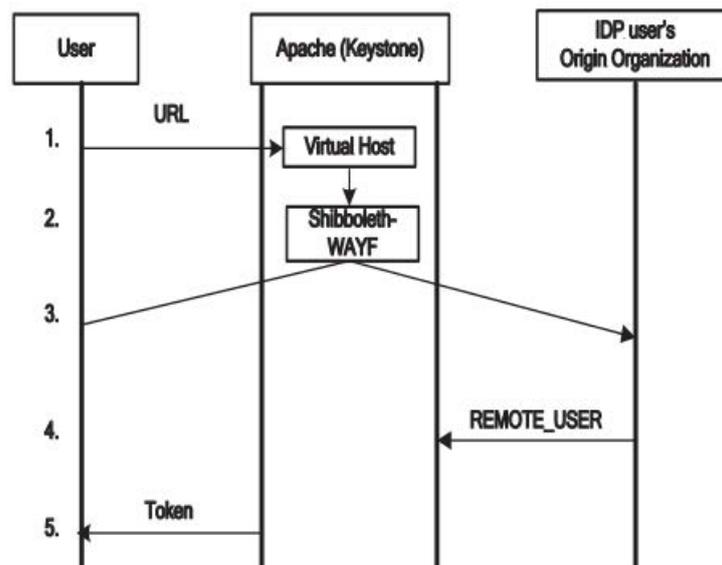


FIGURE II.3.21 – Hébergement de keystone dans apache

### 3.6.2.4 Shibbolisation du dashboard

La nouvelle version API v3, permet à keystone de recevoir les attributs shibboleth et les mapper avec ses propres attributs. Cette possibilité nous a amené à utiliser apache en tant que frontal de dashboard. Nous avons utilisé le module d'authentification shibboleth de apache lib-apache2-shib2 pour rediriger un utilisateur vers son IDP. Une fois authentifié, les attributs shibboleth reçu par apache sont redirigés via le module mod-wsgi vers Dashboard qui tourne sous Django qui est un framework web python. Le processus est illustré à travers la figure II.3.22.

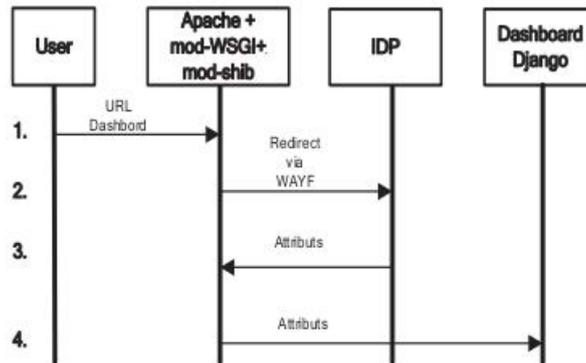


FIGURE II.3.22 – Shibbolisation dashboard (1)

Après la phase d'authentification illustrée par la figure II.3.22, grâce à la version 3 de l'API openstack, keystone reçoit les attributs shibbo et les mappe avec ses propres attributs, ce qui permet à keystone d'attribuer un jeton à l'utilisateur via dashboard.

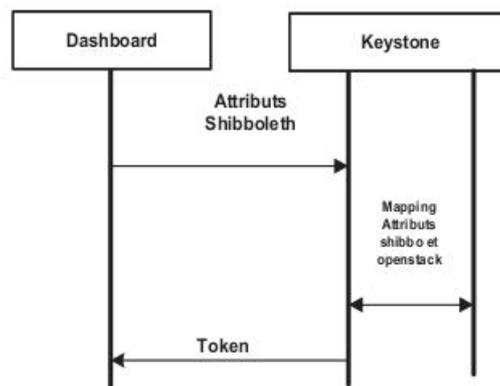


FIGURE II.3.23 – Shibbolisation dashboard (2)

### 3.6.2.5 Résultats obtenus

Les étapes suivantes ont permis la shibbolisation du dashboard.

Etape1 : Installation de shibb sur la machine hébergeant apache

Etape2 : configuration de Shibboleth

Etape 3 : configuration du serveur apache comme SP

Voici un extrait du fichier de configuration de apache qui est hébergé sous ubuntu 12.04  
**/etc/apache2/conf.d/shib.conf**

```
<Location /secure>  
AuthType shibboleth  
ShibRequestSetting requireSession 1  
require validuser  
</Location>
```

```
<Location /dashboardshib>  
AuthType shibboleth  
ShibRequestSetting requireSession 1  
require validuser  
</Location>
```

Etape 4 : Configuration de la communication apache et dashboard via le module wsgi.  
**/etc/apache2/conf.d/openstackdashboard.conf**  
**WSGIScriptAlias**  
**/dashboardshib/usr/share/openstackdashboard/openstack\_dashboard/wsgi/django.wsgi.**

Pour illustrer le principe de l'authentification décentralisée, nous avons mis en place notre propre fédération d'identités Shibboleth avec ses composants : fournisseur d'identité (IdP), fournisseur de services (SP) et le service WAYF. Ainsi, pour tester le bon fonctionnement de nos différentes briques Shibboleth, on a testé l'accès à l'interface d'accès du cloud pour un utilisateur de l'OV

**Étape 1** : l'utilisateur va utiliser un navigateur web de son choix et saisir l'url d'accès du dashboard.

**Exemple** :<https://servpro.sn/uvscloud/dashboard>

**Étape 2** : un service de découverte (WAYF) est affiché à l'utilisateur où ce dernier va choisir son université de rattachement, correspondant à l'IdP chargé de l'authentification de l'utilisateur.

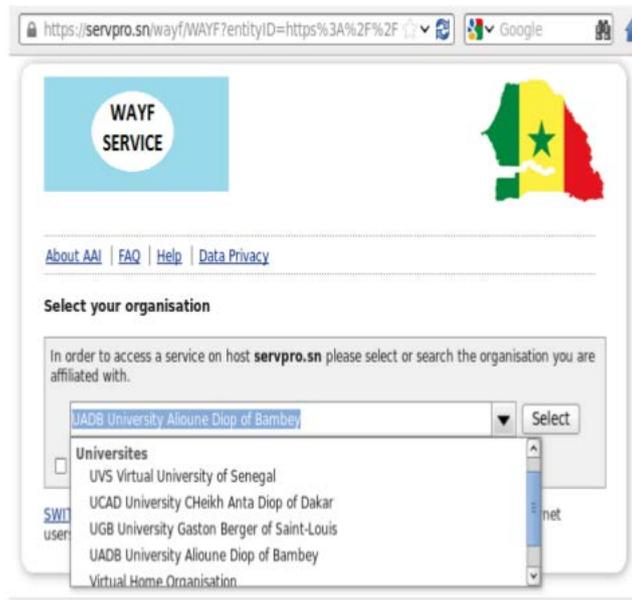


FIGURE II.3.24 – Service de découverte

L'utilisateur choisit son établissement de rattachement sur la liste.

**Étape 3 :** Une fois le choix effectué, qui correspond au fournisseur d'identités (IDP), l'utilisateur est redirigé vers la page d'authentification fournie par l'IDP.

Lorsque l'authentification est validée, on redirige alors l'utilisateur vers l'interface d'accès du dashboard où il pourra s'auto-provisionner.

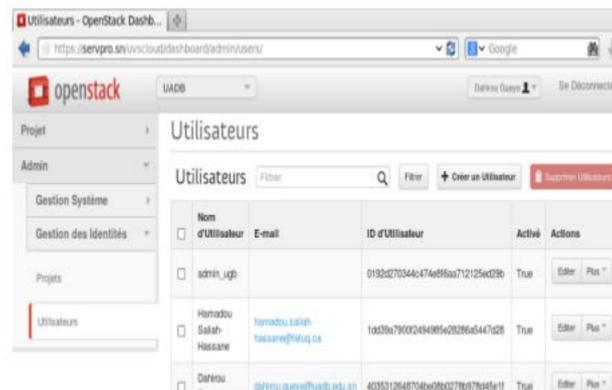


FIGURE II.3.25 – Accès à l'infrastructure Cloud

### 3.7 Découverte dynamique des ressources disponibles

Pour permettre à chaque opérateur de Cloud Computing des universités de bien régler l'accès à ses ressources, nous proposons un modèle de publication et de découverte dynamique des ressources

en nous basant sur le mécanisme PubSub offert par le protocole XMPP tel que décrit dans le chapitre 2. Dans notre modèle, la notion de priorité entre un utilisateur d'un opérateur de cloud Computing détenteur de ressources sollicitées et un autre utilisateur provenant d'un autre opérateur de cloud Computing demandant les mêmes ressources dans une même période est gérée.

Cette gestion de priorité d'accès est importante car elle est de nature à rassurer les opérateurs de cloud Computing quant à la maîtrise des ressources propres mutualisées. La non prise en compte de cette priorité aurait pour effet de dissuader des opérateurs de cloud Computing à publier leurs ressources disponibles. Or l'objectif cherché en mettant les opérateurs de cloud Computing dans une organisation virtuelle est l'utilisation optimale des ressources mutualisées ; qu'une ressource soit disponible dans une université alors qu'une autre en a besoin serait contraire à l'idée de mutualisation et de partage. Dans beaucoup de modèles de partage de ressources, la déclaration des disponibilités des ressources se fait manuellement par un administrateur. Les limites de tel modèle résident dans le fait qu'une information sur la disponibilité d'une ressource peut être juste à un moment donné et ne plus l'être à un autre moment car tombée en panne ou défectueuse.

La figure II.3.26 montre l'architecture du système de découverte dynamique des ressources.

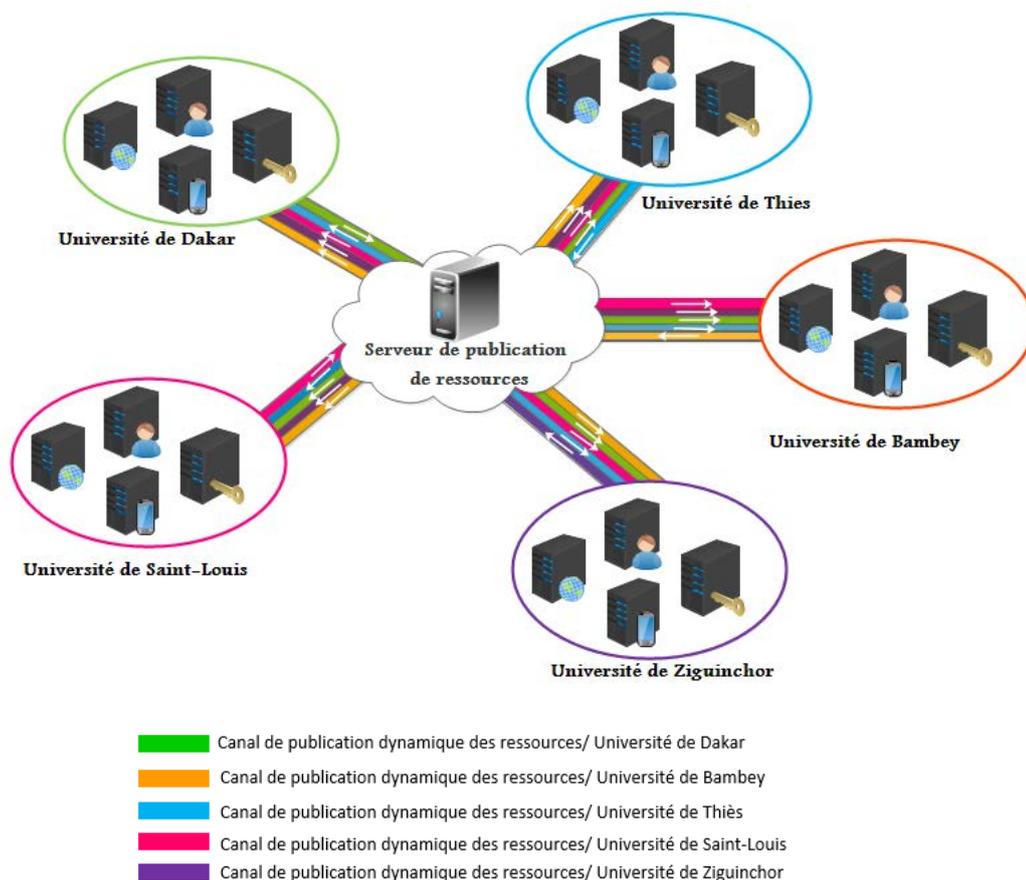


FIGURE II.3.26 – Architecture du système de publication dynamique des ressources

Le système de publication dynamique des ressources s'appuie sur le mécanisme PubSub. Chaque université dispose d'un canal de publication à travers lequel il pourra publier les informations relatives à la disponibilité de ses ressources. Ainsi toute université désirant s'enquérir de l'état des ressources d'une université partenaire se doit de s'abonner à son canal par l'intermédiaire du serveur de publication de ressources. Ce dernier est un élément central qui constitue le socle du système proposé et apporte ainsi

des solutions aux limitations de la seule utilisation du protocole XMPP. Ces limitations proviennent du fait que XMPP ne gère que la présence et non la disponibilité des ressources. Le serveur de publication de ressources offre aussi des fonctionnalités : abonnement/désabonnement d'une université partenaire, déclaration, partage, restriction d'accès à une ressource particulière. Comme le montre les résultats que nous allons présenter dans les sections ci-après, on a pu ajouter, à partir du serveur de publication, l'abonnement de l'UVS. Le serveur de publication de ressources, s'appuyant sur les droits d'accès inhérents au mécanisme PubSub, délègue la priorité d'accès aux ressources aux administrateurs des opérateurs de Cloud de chaque université. De manière précise, la décision d'autorisation d'accès à une ressource sera donnée par l'Opérateur de Cloud détenteur de la ressource.

Les plateformes de Cloud Computing utilisées par les opérateurs de notre système étant Open-Stack qui utilise l'hyperviseur KVM, il est aisé pour nous d'utiliser la librairie libvirt décrite dans le chapitre 2. En effet, sur chaque hyperviseur est installé un programme client XMPP dont le rôle est de transmettre l'état des machines virtuelles gérées par un hyperviseur au serveur de publication de ressources et de transmettre des instructions du serveur de publications aux hyperviseurs.

### 3.7.1 Cas d'utilisation du système de publication dynamique des ressources

Après une étude des besoins les acteurs suivants ont été déterminés :

- **Utilisateur simple** : c'est celui qui utilise la plateforme pour se doter d'une ressource.
- **Administrateur** : administre les ressources locales d'un site (ou organisation partenaire).
- **Super Administrateur** : Il est chargé d'ajouter les organisations faisant partie de l'organisation virtuelle.

Partant de ces acteurs, les cas d'utilisations suivants ont été jugé utiles :

- **Authentification** : Tout utilisateur doit s'authentifier afin d'utiliser le système.
- **Réservation ressource** : Permet à un utilisateur de faire une réservation de ressource pour une durée.
- **Controler ressource** : permet à un utilisateur d'avoir le contrôle total de la ressource.
- **Gestions sites** : permet au super-administrateur de pouvoir ajouter, modifier et supprimer un site.
- **Gestions Utilisateurs** : Permet au administrateur d'un site de gérer les utilisateurs de leur sites.
- **Gestion administrateur** : Permet au Super Administrateur de gérer les administrateurs des sites.
- **Gestion des groupes** : Permet aux Administrateurs de gérer les groupes d'utilisateurs de leurs sites.
- **Gestion des parcs** : Permet aux administrateurs de gérer les parcs de ressources de leur site.
- **Gestion de permission** : permet aux administrateurs de définir des permissions sur les groupes et les parcs.

Ainsi, le diagramme des cas d'utilisation est illustré dans la figure II.3.27.

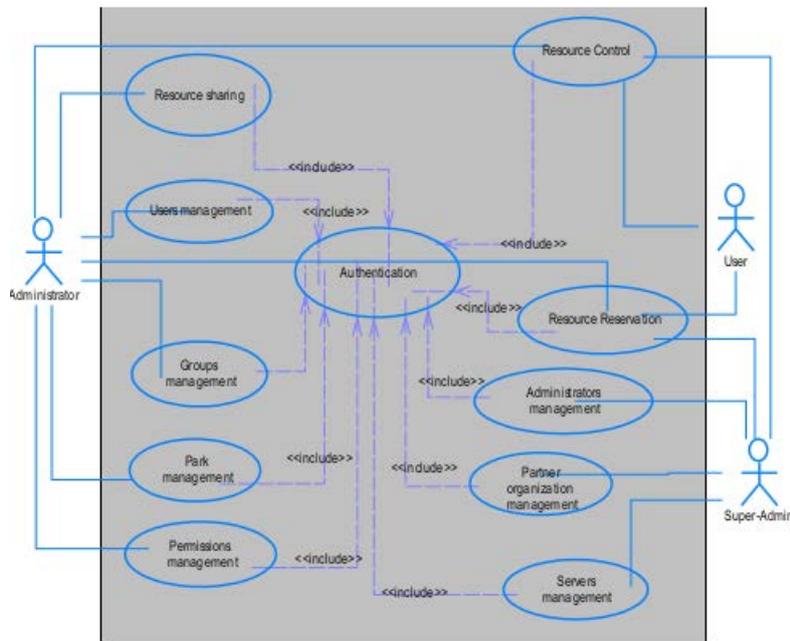


FIGURE II.3.27 – Diagramme des cas d'utilisation

### 3.7.2 Description de quelques cas d'utilisation de l'application

**Partage de ressource** : L'administrateur d'une organisation partenaire peut partager des ressources avec d'autres organisations. Pour partager une ressource, l'administrateur local effectue les opérations suivantes :

- Envoi des informations de la ressource à l'élément central.
- L'élément central vérifie si la ressource existe dans sa base de données.
  - Si oui, l'élément central initialise son champ « partage » à 1.
  - Sinon, il la sauvegarde puis initialise le champ « partage » à 1.

**Restriction de ressource** : L'administrateur d'une organisation partenaire peut restreindre des ressources déjà partagées avec d'autres entités. Pour restreindre une ressource, les opérations suivantes sont effectuées.

- Envoi des informations de la ressource à l'élément central.
- L'élément central recherche la ressource dans la base de données par son JID puis initialise le champ « partage » à 0.

**Réservation de ressource** : Avant d'avoir le contrôle total d'une ressource, l'utilisateur doit faire la réservation de la ressource pour une durée déterminée. Cela permet d'éviter que deux utilisateurs accèdent en même temps à une ressource. Lors de réservation d'une ressource, les actions suivantes sont effectuées.

- Envoi des informations de la ressource à l'élément central
- L'élément central vérifie si la ressource est disponible :
  - Si oui, il crée un enregistrement de réservation reliant l'utilisateur et la ressource dans sa base de donnée puis envoie à l'utilisateur un message d'acceptation de la réservation.

- Sinon, il envoie un message de non disponibilité de la ressource.

**Recherche de ressource par rapport à une priorité** : un utilisateur de l'organisation virtuelle peut rechercher une ressource suivant plusieurs critères (RAM, DISQUE, etc.). Pour rechercher une ressource, les étapes suivantes sont effectuées.

- Envoi des critères de recherche à l'élément central
- L'élément central effectue les tâches suivantes selon les critères envoyés :
  - Si c'est par rapport à la RAM, l'élément central parcourt l'ensemble des serveurs XMPP des différentes organisations partenaires et compare la capacité de RAM des ressources avec la capacité reçue comme critère. Puis, il les classe par ordre de priorité avant d'envoyer la réponse.
  - Si c'est par rapport à la disponibilité, l'élément central parcourt l'ensemble de serveurs XMPP des différents sites et vérifie si les ressources disponibles sur la machine physique permettront de lancer la machine virtuelle sollicitée. Puis l'élément central les classe par ordre de priorité avant d'envoyer la réponse.

### 3.7.3 Format d'échange de message entre XMPP et l'application

Pour communiquer avec le serveur, l'application envoie les messages sous forme texte au serveur XMPP, puis le serveur les transmet à l'agent archipel qui gère la machine virtuelle. Une fois que le message est reçu par l'agent, ce dernier le traduit en commande libvirt compréhensible par la machine virtuelle. Pour le serveur XMPP les messages reçus sont juste des messages textes, alors que c'est des commandes permettant de contrôler une machine virtuelle. Cela constitue un des avantages d'utiliser un serveur XMPP car il permet de faciliter le dialogue entre l'application et la machine virtuelle.

Les messages suivants sont échangés entre l'application et le serveur XMPP.

- ['roster', 'users'] : donne l'ensemble des contacts d'un utilisateur
- ['start', 'create', 'boot', 'play', 'run'] : permet de démarrer une machine
- ['shutdown', 'stop'] : permet d'arrêter une machine
- ['destroy'] : destruction de l'instance d'une machine
- ['pause', 'suspend'] : mettre en pause une machine
- ['resume', 'unpause'] : démarrer une machine en pause
- ['info', 'how are you', 'and you'] : donne les informations sur la machine
- ['desc', 'xml'] : donne la description de la machine sous forme XML
- ['vnc', 'screen'] : donne l'adresse et le port permettant d'accéder à l'écran VNC de la machine

### 3.7.4 Réalisation

L'application gérant le serveur de publication a été développée en J2EE. Cette solution se base sur la virtualisation de ressources de chaque université partenaire afin qu'elles soient accessibles à distance par les autres universités via web.

Le super-administrateur de l'application peut effectuer les opérations suivantes :

- Lister les ressources,
- Réserve,
- Contrôle de ressource,
- Gestion des administrateurs,

- Gestion des Serveurs,
- Gestion des Sites,
- Partage d'une ressource,
- Restriction de ressource

La figure II.3.28 illustre un exemple de profil du super admin avec ses fonctionnalités.



FIGURE II.3.28 – Exemple de profil super admin

Le profil de l'administrateur d'une organisation partenaire permet d'effectuer les opérations suivantes : Le super-administrateur de l'application peut effectuer les opérations suivantes :

- Lister les ressources,
- Réservation,
- Contrôle de ressource,
- Gestion des groupes,
- Gestion des utilisateurs,
- Gestion des accès,
- Gestion des parcs,
- Partage d'une ressource,
- Restriction de ressource



FIGURE II.3.29 – Profil d'un admin local

La figure II.3.30 illustre la prise de contrôle à distance d'une machine virtuelle avec l'outil vnc. La machine virtuelle est d'abord partagée par un partenaire pour être ensuite réservée par un utilisateur de l'OV avant que ce dernier ne puisse l'utiliser comme le montre la figure II.3.31.



FIGURE II.3.30 – Statut des machines virtuelles



FIGURE II.3.31 – Prise de contrôle d'une machine

### 3.8 Conclusion

Dans la littérature, beaucoup de recherches ont traité de la pertinence de l'utilisation de Cloud Computing dans les organisations virtuelles. Il s'est agi dans ce chapitre de proposer des modèles d'organisations virtuelles autour de Cloud Computing en vue de régler des problèmes de l'enseignement supérieur dans les pays en voie de développement. En effet, nous constatons qu'à cause de la massification des effectifs des étudiants, les établissements d'enseignement supérieur ont des difficultés à organiser des travaux pratiques dans le domaine des STEM. La solution d'opérateurs de Cloud spécialisés dans une discipline donnée qui mettent en place des plateformes d'expérimentation et de travaux pratiques facilement mutualisables car gérées dans le Cloud, va contribuer à encourager les établissements d'enseignement supérieur à renouer avec les travaux pratiques qui sont importants dans l'assimilation par les étudiants des modules de formation dans le domaine des STEM. Pour sa part, la solution fournissant une méthode d'authentification décentralisée, utilisant une fédération d'identité, apporte une flexibilité d'utilisation des ressources mutualisées et garantit donc l'autonomie des universités quant au choix de leur méthode d'authentification. Enfin, la solution de publication dynamique

des ressources, basée sur le mécanisme PubSub du protocole XMPP et intégrant un système de priorité aux utilisateurs de l'opérateur de Cloud détenteur de la ressource sollicitée, est de nature à dissiper toute opposition à la mutualisation des ressources de peur d'en perdre le contrôle.

# Chapitre 4

## Proposition de plateformes WebRTC pour l'amélioration du système d'enseignement des universités

### 4.1 Introduction

La technologie WebRTC offre des fonctionnalités intéressantes de communication à travers le web sans besoin d'installer des logiciels compliqués pour l'utilisateur final.

Nous proposons dans ce chapitre des solutions permettant d'améliorer l'enseignement supérieur notamment une solution de collaboration permettant entre autres de pallier les problèmes de travaux pratiques, une solution permettant d'avoir tout un ensemble d'outils de communication dans l'environnement de formation à distance et enfin une solution permettant à des acteurs des universités de développer de manière collaborative des logiciels tout en ayant dans leur environnement de développement un ensemble d'outils de communication temps réel.

### 4.2 Proposition d'une plateforme WebRTC d'appui au système d'enseignement des universités dans un contexte de connexion Internet limitée

Nous proposons dans cette partie une plateforme Web basée sur la technologie WebRTC permettant aux universités d'améliorer leur système d'enseignement en y intégrant l'e-Learning. La plateforme proposée permet non seulement, aux enseignants et à leurs étudiants, à travers un simple navigateur de communiquer par tchat, audio, video mais aussi de transférer des fichiers ou partager leur écran d'ordinateur interfacé avec des équipements de travaux pratiques et tout ceci dans un environnement IP sans nécessairement disposer d'Internet. Pour cela, il a fallu implémenter un serveur de signalisation WebRTC pour gérer les applications temps réel, utiliser les trois API de WebRTC qui sont l'API MédiaStream pour l'acquisition et la synchronisation des flux audio et video, l'API PeerConnection pour la communication entre les navigateurs des utilisateurs, l'API RTCDataChannel pour le transfert de fichiers, le Tchat et le partage d'écran.

Notre plateforme a été testée par les enseignants et étudiants de l'université Cheikh Anta Diop de

Dakar comme un système de classe virtuelle.

L'université Cheikh Anta Diop de Dakar est la plus grande université du Sénégal avec plus de 100000 étudiants et recevant chaque année plus de 20000 nouveaux bacheliers. Malgré les efforts fournis par l'état pour construire des infrastructures réseau, des salles de classes et de travaux pratiques, le problème d'engorgement et d'accès à internet au sein du campus pédagogique, qui est vaste, n'est pas pour autant réglé et pose un problème de l'efficacité du système d'enseignement. Certes, des efforts ont été faits ces dernières années pour mettre en place sur le campus pédagogique un réseau de fibre optique interconnectant les différents bâtiments du campus et un réseau sans fil pour la connectivité des étudiants mais la bande passante vers Internet n'est pas toujours suffisante. Profitant de la bonne bande passante du réseau local, l'état du Sénégal encourage les enseignants à mettre certains de leurs cours en ligne pour non seulement désengorger les salles de classe mais aussi d'augmenter l'efficacité du système d'enseignement. Dans ce contexte, nous pensons que la mise en place d'un système de collaboration en temps réel ne nécessitant pas l'accès à internet et s'appuyant sur le réseau IP local pourrait contribuer à améliorer l'efficacité du système d'enseignement de l'UCAD.

Compte tenu du niveau de certains utilisateurs en informatique, le système proposé doit être simple d'utilisation et ne doit pas nécessiter l'installation des logiciels coté utilisateurs.

Dans la littérature, plusieurs auteurs [62] [63] ont montré l'intérêt pour les universités classiques d'intégrer un système de cours en ligne dans leur modèle d'enseignement pour augmenter son efficacité : on parle alors d'un modèle hybride.

Certains auteurs [64] [65] [66] ont proposé des plateformes de collaboration ou de classes virtuelles nécessitant l'installation des plugins coté utilisateurs dans leur système d'enseignement en ligne. Certaines de ces plateformes ne peuvent pas être utilisées à partir des smartphones et des tablettes, Ce qui constitue une limite.

Des recherches sur les outils permettant de développer des applications temps réel utilisables à partir des terminaux aussi variés que les ordinateurs portables, les tablettes et les smartphones nous ont conduits vers la technologie WebRTC qui nous a permis de développer notre système de collaboration que nous proposons.

Dans la suite, il s'agira tout d'abord de discuter de l'état de l'art de la technologie WebRTC du point de vue du développement logiciel, ensuite de décrire les fonctionnalités de notre système, puis de spécifier l'architecture de la solution proposée et de présenter les cas d'utilisation de notre système, enfin de présenter les résultats de la plateforme.

### 4.2.1 État de l'art sur le WebRTC du point de vue du développement logiciel

Le WebRTC est un projet open source présenté par Google en 2011[31] qui permet d'assurer des communications en temps réel via une API JavaScript. Le projet vise à mettre en place une technologie permettant aux navigateurs web de supporter des communications interactives point à point et d'offrir des échanges de données en mode synchrone [30][67]. Le WebRTC a pour but de donner aux navigateurs la capacité de proposer des communications audio, vidéo ou écrites, le transfert de fichiers, le partage d'écran, et le contrôle à distance des ordinateurs.

Les principaux composants du WebRTC API définis par les groupes de travail W3C (World Wide Web Consortium) et IETF sont [31] :

- MediaStream : Permet à un navigateur d'accéder à la camera et au microphone ;

- RTCPeerConnection : d'activer les appels audio et vidéo ;
- RTCDataChannel : Permet aux navigateurs d'envoyer des données dans une connexion peer-to-peer.

La figure II.4.1 met en évidence, d'une part l'API WebRTC C++ implémentée dans certains navigateurs et d'autre part l'API WebRTC web permettant aux développeurs web d'intégrer des services offerts par le WebRTC dans leurs applications.

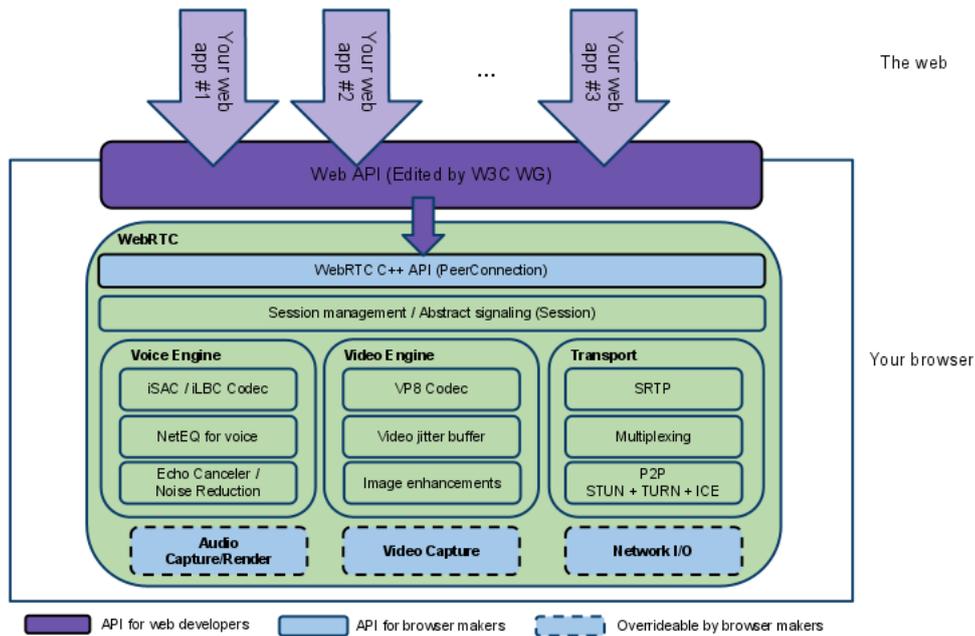


FIGURE II.4.1 – Architecture générale de la technologie WebRTC

Contrairement aux technologies basées sur XML permettant l'échange entre ressources, la technologie WebRTC offre un mode d'échange synchrone entre entités web en rompant avec l'architecture familière client-serveur de communication. Une telle architecture a l'avantage de réduire de façon notable la latence en s'attachant à établir des communications directes entre clients, "peer-to-peer", ou uniquement en effectuant des rebonds intermédiaires, via des tunnels, dans le cas où la communication directe n'est pas possible à cause de translations NAT ou de firewalls [68]

Les organismes de normalisation W3C et IETF n'ont pas imposé un protocole de signalisation particulier dans WebRTC dans le but de laisser libre champs aux développeurs de choisir l'un des protocoles existants (SIP / Jingle) [69] ou de personnaliser leur propre protocole de signalisation en se basant sur les websockets. Cette stratégie de signalisation permet d'éviter la redondance et de maximiser la compatibilité avec les technologies déjà établies. Le processus de communication est décrit dans la figure II.4.2.

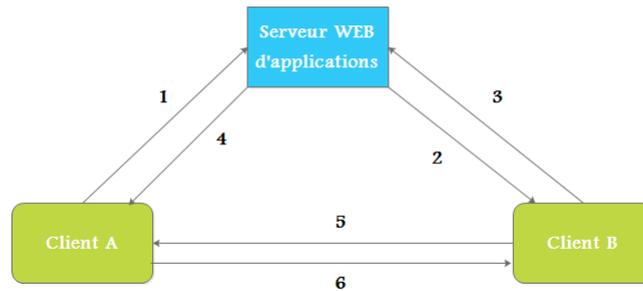


FIGURE II.4.2 – Processus de communication WebRTC

Le serveur WebRTC ou serveur de signalisation permet de transmettre aux utilisateurs des messages de signalisation sous format SDP (Session Description Protocol) [70] qui décrit le contenu multimédia de la connexion comprenant la résolution, les formats, les codecs, le cryptage, etc., afin que les deux pairs puissent se comprendre.

Une fois les utilisateurs connectés au serveur de signalisation, l'échange direct entre eux par l'instanciation d'un objet `RTCPeerConnection`. Cet objet prend en premier paramètre la tâche à exécuter (visioconférence, partage d'écran, transfert de fichier et texte) et en deuxième paramètre les serveurs STUN/TURN pour éventuellement contourner le problème de traversée de NAT lorsque les utilisateurs WebRTC en communication NAT se trouvent dans deux réseaux différents derrière un serveur NAT.

Le processus de mise en communication WebRTC permet aussi de gérer :

- Des messages d'erreur.
- Des métadonnées de médias tels que les paramètres de codec, la bande passante et les types de médias.
- Des Identifiants WebRTC des utilisateurs utilisés pour établir des connexions sécurisées.
- Des données réseau, comme l'adresse IP et le port d'un hôte tel que vu par le monde extérieur

L'identifiant d'un utilisateur généré par un serveur WebRTC est assez complexe et ressemble à ceci :

I am Sh46K2a-P2O9crxzAAAA

FIGURE II.4.3 – Identifiant WebRTC d'un utilisateur

Nous pensons que WebRTC est adapté à une collaboration puisqu'il permet la mise en œuvre d'un système de communication face à face via le web. En vue d'échanger des données entre utilisateurs avec WebRTC, il faut utiliser un système et un protocole de signalisation.

Donc notre approche consiste à mettre en place un serveur de signalisation WebRTC et trouver un algorithme de mapping entre l'identifiant WebRTC d'un utilisateur et son identifiant unique dans le système que nous allons proposer pour obtenir une solution technologique utilisable par les universités des pays en voie de développement qui font face à une massification de leur effectif d'étudiants et qui ont une bande passante limitée vers Internet.

### 4.2.2 Fonctionnalités du système

Le système proposé offre une interface web permettant aux utilisateurs de s'authentifier avant de bénéficier des services suivants selon leur profil :

- Gestion d'utilisateurs, de classes et de cours.
- Planification de cours à une date et heure données.
- Projection et suivi du cours à distance via la visioconférence.
- Transmission des fichiers (cours, exercices) aux étudiants en temps réel
- Partage de l'environnement du bureau de l'enseignant aux étudiants.
- Système d'échange entre l'enseignant et les étudiants via la messagerie instantanée
- Demande de désactivation de la webcam d'un utilisateur
- Réactivation au milieu d'une conversation de sa webcam préalablement désactivée
- Éjection du système d'un étudiant par un enseignant
- Etc..

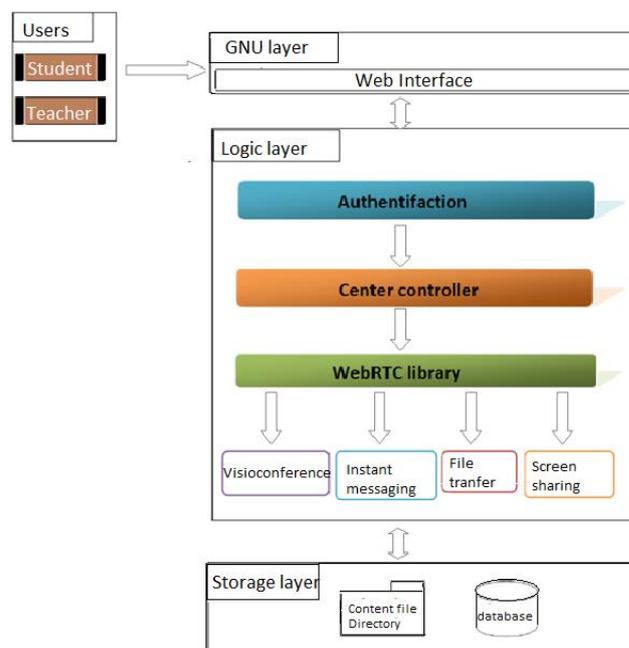


FIGURE II.4.4 – Fonctionnalité du système E-learning

### 4.2.3 Présentation de l'architecture de la solution

L'architecture proposée est constituée de plusieurs composants représentés sur la figure II.4.5 :

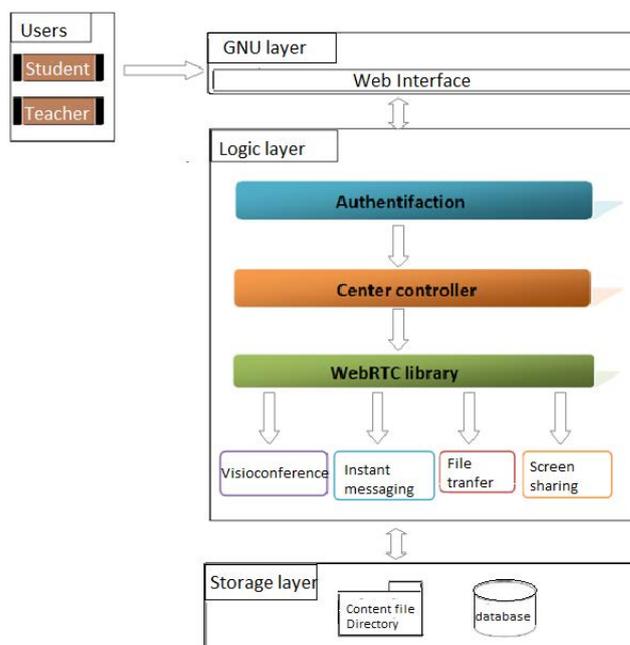


FIGURE II.4.5 – Architecture du système

Le composant **plateform layer** est une application web hébergée sur un serveur web qui est l'intermédiaire entre les utilisateurs et le **serveur de signalisation**.

La couche de stockage est aussi bien utilisée par la couche applicative et le serveur de signalisation pour l'authentification, l'identification et la gestion des droits des utilisateurs.

La figure II.4.6 illustre les composants logiciels de l'architecture proposée pour le système e-Learning.

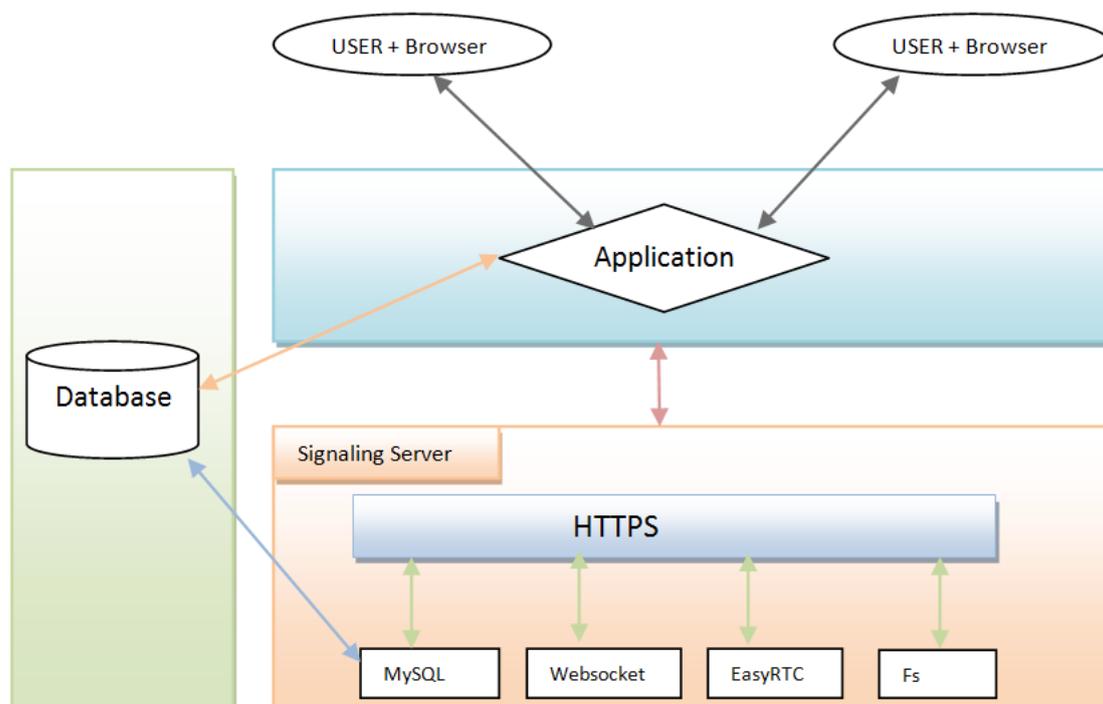


FIGURE II.4.6 – Structure logique du serveur de signalisation

Plusieurs développeurs se sont basés sur l'API WebRTC pour proposer des bibliothèques masquant

la complexité de l'API et facilitant ainsi le développement des applications WebRTC. C'est le cas de la bibliothèque EasyRTC que nous avons utilisée dans ce travail [71].

Le fonctionnement de notre système dépend essentiellement du serveur de signalisation mis en place à partir de modules suivants :

- HTTPS : permet de créer un serveur web sécurisé coté signalisation ;
- Websocket : permet de créer des flux bidirectionnels permettant l'échange en temps réel dans les deux sens de communication [72] ;
- EasyRTC : fournit une bibliothèque simplifiant le développement des applications WebRTC ;
- MySQL : permet de connecter et de faire des requêtes sur la base de données MySQL.
- Fs : permettant l'import des fichiers de configuration du serveur HTTPS avec les clés SSL.

Les deux couches applicatives et signalisation communiquent via le protocole sécurisé HTTPS.

#### 4.2.4 Description de quelques cas d'utilisation de la plateforme

Dans cette section, nous décrivons certains cas d'utilisation de notre système.

##### Créer une nouvelle salle de conférence

Cette fonctionnalité est sollicitée lorsqu'un enseignant veut lancer un cours à distance pour une classe donnée. L'enseignant saisit les informations de la salle de cours et les passe au serveur de signalisation pour créer la salle et ensuite insère la salle dans la base de données.

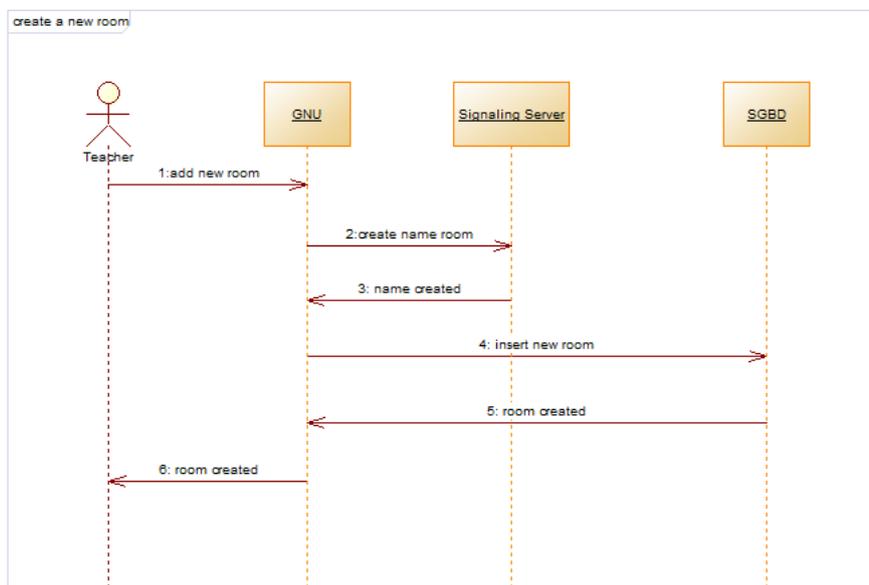


FIGURE II.4.7 – Création d'une nouvelle salle de réunion

##### Ajouter un étudiant à une salle de conférence

Une classe nouvellement créée ne contient que l'enseignant qui l'a créée et celui-ci a la possibilité d'ajouter une liste d'étudiants à cette classe. Ce service se déroule comme suit :

- La récupération de l'étudiant auprès de la base de données
- de donner un id à l'étudiant en se connectant ce dernier au serveur de signalisation
- de modifier l'étudiant dans la base de données en lui affectant l'id de signalisation

- Affecter l'étudiant à la salle en passant au serveur de signalisation le nom de la salle et l'id de signalisation de l'étudiant

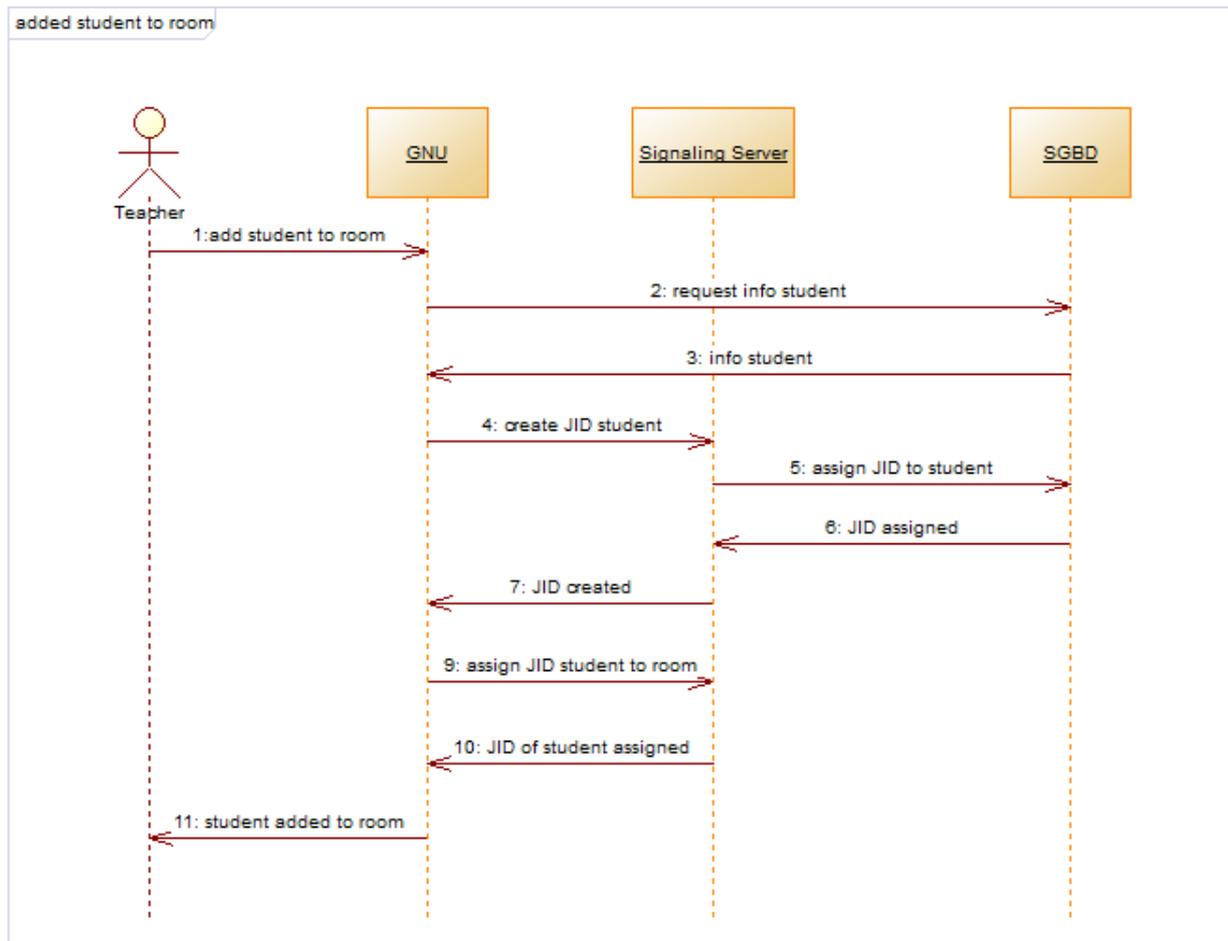


FIGURE II.4.8 – Ajout d'un utilisateur dans une chambre de conférence

#### 4.2.4.1 Partage d'écran entre étudiants et enseignants

La figure II.4.11 décrit le processus de partage de l'environnement du bureau de l'enseignant aux étudiants. . Le processus commence par l'obtention de l'id de signalisation pour les différents interlocuteurs auprès du serveur de signalisation, mettre à jour les infos des interlocuteurs dans la base de données en leur affectant les id de signalisation, l'obtention du screen par le biais du l'extension

WebRTC screen sharing et enfin l'enseignant appelle le groupe d'étudiants pour leur partager son écran.

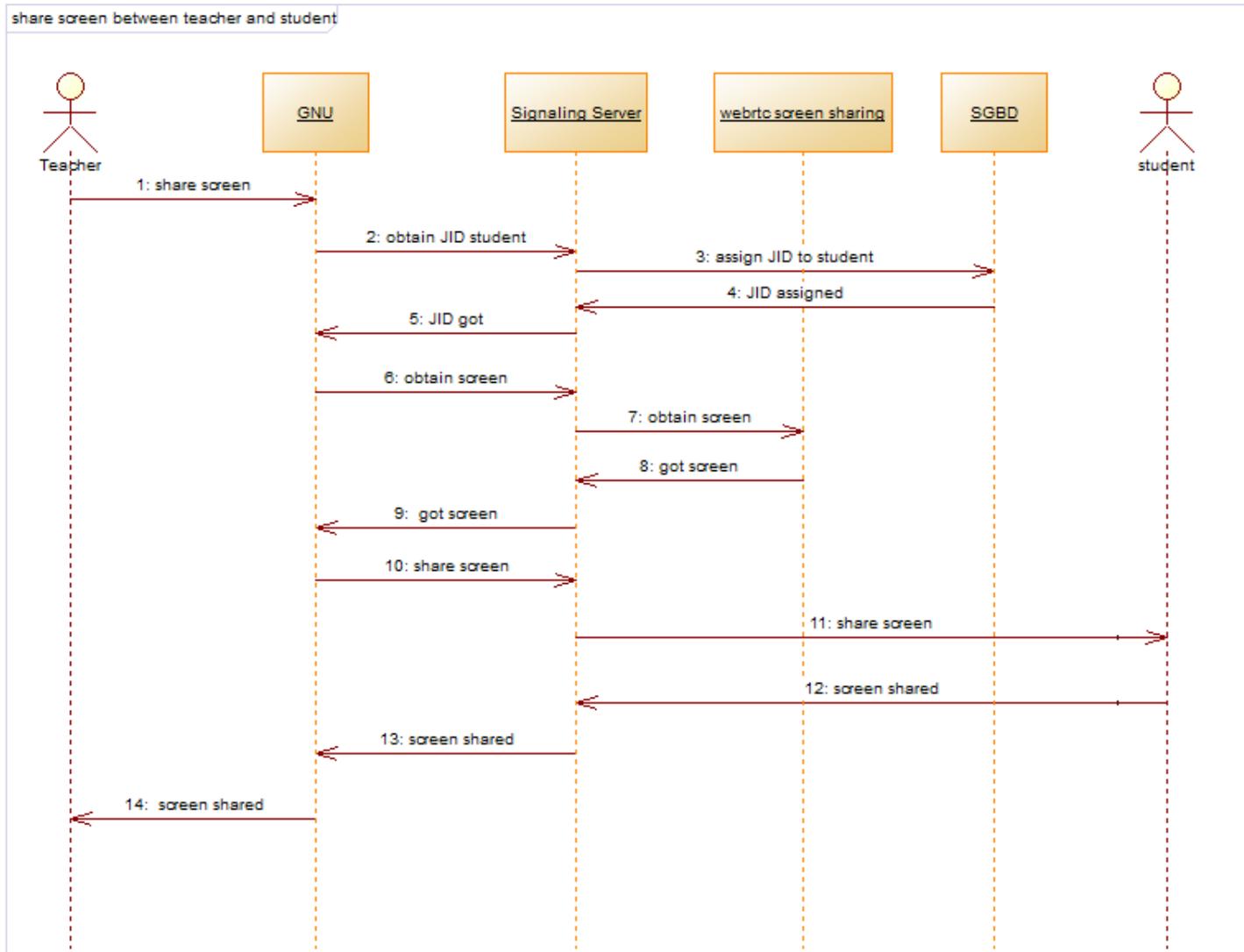


FIGURE II.4.9 – Partage d'écran entre étudiant et enseignant

### 4.2.5 Résultats

Nous avons implémenté une plateforme web pour montrer la faisabilité de notre approche. Nous avons utilisé pour l'élaboration de cette plateforme les langages de programmation suivantes PHP, NodeJS, JQuery, HTML5, CSS et une base de données MYSQL.

La figure II.4.10 montre une interface de visioconférence où l'enseignant soumet un cours aux étudiants d'une classe donnée. L'enseignant peut partager uniquement l'audio, la vidéo ou les deux ensemble avec ses interlocuteurs. En plus il peut déconnecter un étudiant du cours s'il juge que ce dernier ne se comporte pas correctement.

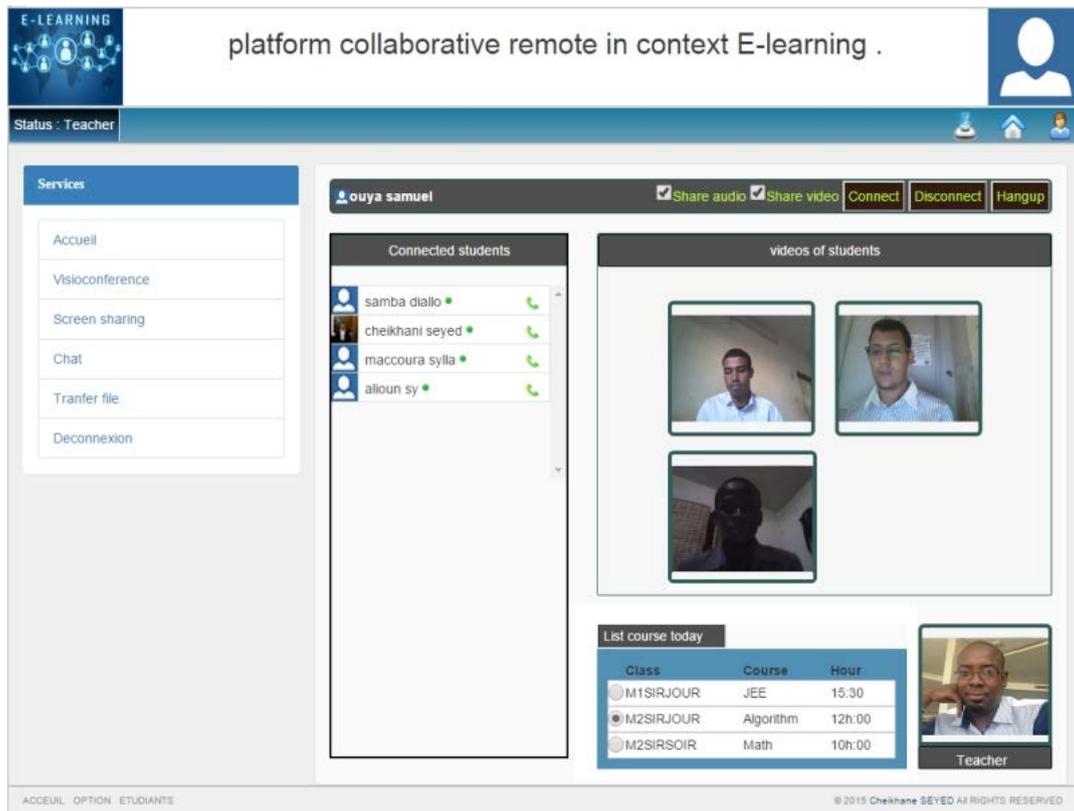


FIGURE II.4.10 – Visioconférence entre l'enseignant et les étudiants

La figure II.4.11 montre une interface de partage de l'environnement du bureau de l'enseignant. L'interface indique que l'étudiant reçoit le document partagé ainsi que la vidéo permettant l'explication en direct du cours.

Ainsi L'étudiant reçoit toutes les opérations exécutées sur le bureau de l'enseignant en temps réel. Dans cette phase, l'enseignant pourra selon les circonstances partager uniquement l'audio, la vidéo ou les deux.

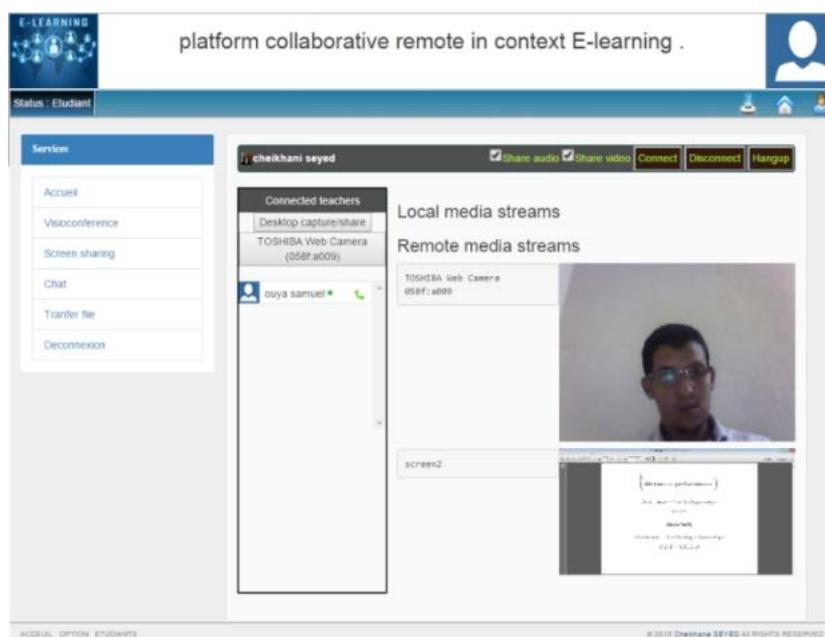


FIGURE II.4.11 – Partage d'écran de l'enseignant

Cette fonctionnalité pourrait être utilisée par les enseignants de la Faculté de Médecine et de Pharmacie où ceux du département de chimie biologique pour permettre aux étudiants de manipuler à distance des équipements de TP qui sont interfacés à un ordinateur. La plupart des équipements de TP en médecine et en Biologie peuvent aujourd'hui être interfacés à un ordinateur, ce qui implique que l'accès à l'écran d'un ordinateur équivaut à l'accès d'un équipement de TP. C'est le cas de la figure II.4.12 qui montre un microscope optique Microscope confocal - fv1000 [73] interfacé à un ordinateur.



FIGURE II.4.12 – Microscope optique interfacé à un ordinateur

La Figure II.4.13 montre une interface de transfert de fichiers entre l'enseignant et l'étudiant. Sur cette interface, l'enseignant doit accepter pour recevoir le fichier de la part de l'un des étudiants comme qu'il envoie un fichier à un autre étudiant.

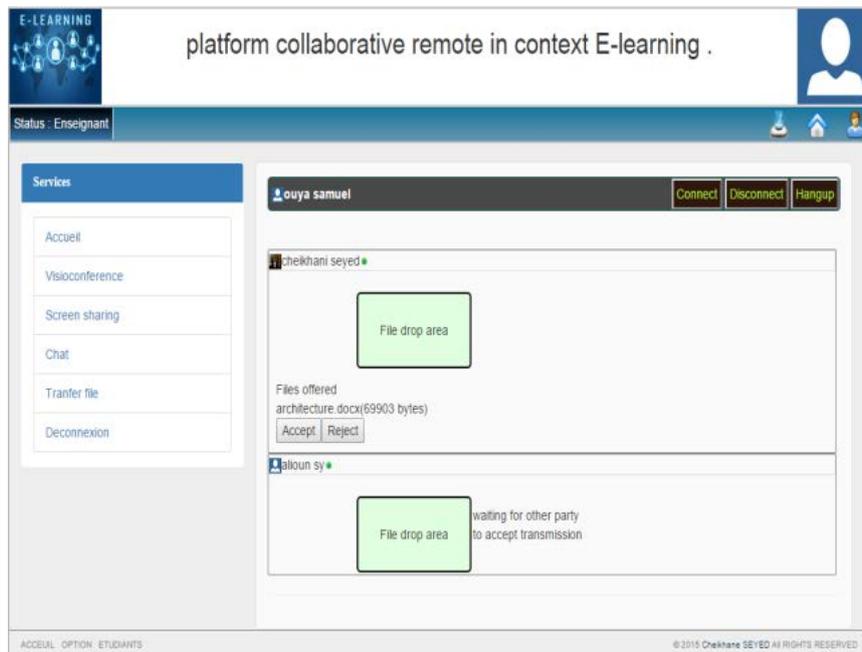


FIGURE II.4.13 – Transfert de fichiers

Notre solution basée sur WebRTC est un dispositif d'appui à l'enseignement supérieur dans le contexte de massification des effectifs des universités et d'une connexion Internet inexistante ou à bande passante limitée.

Un serveur de signalisation étant fondamental pour mettre en place un système de communication et d'échange de donnée en temps réel entre utilisateurs, nous avons intégré des solutions qui nous ont

permis d'avoir un serveur de signalisation WebRTC fonctionnel. Ce serveur WebRTC a été la base de notre système qui nous a permis de le tester avec les étudiants qui sont géographiquement séparés sur le campus de l'université Cheikh Anta de Dakar. Ce qui est très important dans le contexte de cette recherche.

La fonctionnalité de partage d'écran a été appréciable du moment où il permet aux enseignants de montrer à leurs étudiants des interfaces des équipements de travaux pratiques connectés à leurs ordinateurs et de pouvoir commenter en temps réel les résultats.

La fonctionnalité de transfert de fichiers en temps réel peut être utilisée par les étudiants pour s'échanger en temps réel les fiches et rapports de travaux pratiques.

En perspective, nous comptons faire utiliser notre solution dans le réseau des universités interconnectées par Fibre optique pour permettre aux universités sénégalaises de mutualiser et partager des compétences des enseignants car actuellement il est difficile de trouver dans une seule université toutes les compétences.

## 4.3 Solutions de collaboration en environnement e-learning

### 4.3.1 Impact de l'intégration du WebRTC dans les plateformes e-learning des universités

Cette partie traite d'une solution permettant aux universités numériques d'étendre les fonctionnalités de leur plateforme d'enseignement à distance pour améliorer, non seulement leur modèle de livraison de contenus pédagogiques mais aussi leur mode d'évaluation de connaissances des apprenants. En effet, cette section propose une solution pour la plateforme de FOAD moodle, permettant de créer des classes virtuelles, intégrant la vidéo, l'audio, le tchat, l'enregistrement audio des différents acteurs. Cette solution intègre un outil d'évaluation des connaissances en audio pour les modules de cours de langues et permettant aux apprenants d'apprendre à rédiger ou à faire des devoirs en ligne nécessitant l'utilisation massive des symboles et outils mathématiques.

Notre solution est construite autour de la technologie WebRTC, s'utilise directement avec des navigateurs courants sans ajout de plugins aussi bien qu'à partir des ordinateurs que des smartphones et tablettes.

Notre solution a été testée par les acteurs de l'université virtuelle du Sénégal et a permis, entre autres, aux enseignants de langues et de mathématiques d'évaluer les étudiants en ligne comme il se doit sans nécessairement faire recours aux QCM ou être obligés de modifier les épreuves pour tenir comptes des difficultés liées à l'usage des plateformes d'évaluations en ligne.

Aujourd'hui, avec le développement des moyens de communication moderne (internet, téléphonie mobile, ordinateurs de poche), les universités classiques intégrant la Formation Ouverte et à Distance (FOAD) et celles virtuelles doivent relever les défis de la distance, de l'isolement ou de l'éloignement géographique séparant les différents acteurs. On se rend compte aujourd'hui que la Formation Ouverte et à Distance (FOAD) qui, initialement, intéressait les professionnels qui voulaient compléter leur formation, accueille des jeunes dès leur obtention de baccalauréat entre 17 et 18 ans, c'est le cas de l'Université Virtuelle du Sénégal (UVS).

Ces jeunes ont besoin d'être bien formés dans les domaines, entre autres, de la communication, des STEM. Pour se faire, Beaucoup de plateformes de formation à distance existent et intègrent des outils d'autoévaluation et d'évaluation en ligne [74].

C'est le cas de Moodle qui est utilisé par beaucoup d'universités [75]. Par ailleurs, on se rend aussi compte que Google et openfire utilisent l'API WebRTC pour offrir des outils permettant de communiquer en temps réel à partir des navigateurs. Ces outils de Google pourraient être utilisés, pour permettre aux apprenants et aux enseignants de communiquer en temps réel.

Nous pensons que ce serait mieux que tous ces outils de communication synchrone soient intégrés dans l'environnement de travail que les utilisateurs connaissent le mieux, entre autres pour des raisons de sécurité. Alors on se demande « comment proposer une solution basée sur WebRTC pour améliorer l'efficacité et le confort d'utilisation des plateformes e-learning des universités ? »

Un autre problème rencontré par les universités virtuelles, telle que l'UVS est l'évaluation des connaissances des apprenants, dans les filières de langues ou STEM et donc notre solution se doit de prendre en charge cette question. Ainsi la suite de ce travail est organisée comme suit :

D'abord, nous décrivons l'architecture de notre solution, ensuite nous présenterons nos résultats et les perspectives.

#### 4.3.1.1 Description de Moodle

Le terme MOODLE est l'acronyme de Modular Object-Oriented Dynamic Learning Environment. Son développement est amorcé en Australie dans les années 1990 par Martin Dougiamas. C'est une plateforme e-learning, sous licence GPL (GNU Public Licence), accessible via un navigateur web et permettant de gérer des cours en ligne (espace de dépôt de documents, mise en ligne d'activités et de ressources avec les étudiants, gestion des inscriptions et des droits d'accès, tutorat etc.) [76]. Cette plateforme e-learning a été adoptée par plusieurs universités et organismes à travers le monde, car elle offre un ensemble d'outils intégrés conçus, initialement, sur la base du modèle pédagogique dit socioconstructivisme [77]. Moodle est modulaire et constitué d'un noyau auquel des plugins viennent se greffer selon les besoins pour apporter des fonctionnalités supplémentaires. Les plugins bénéficient d'un grand nombre de services de la part du noyau tels que l'accès générique à la base de données, la prise en charge des mises à jour, l'inclusion des informations dans les rapports etc. La figure II.4.14 ci-après montre l'architecture de Moodle.

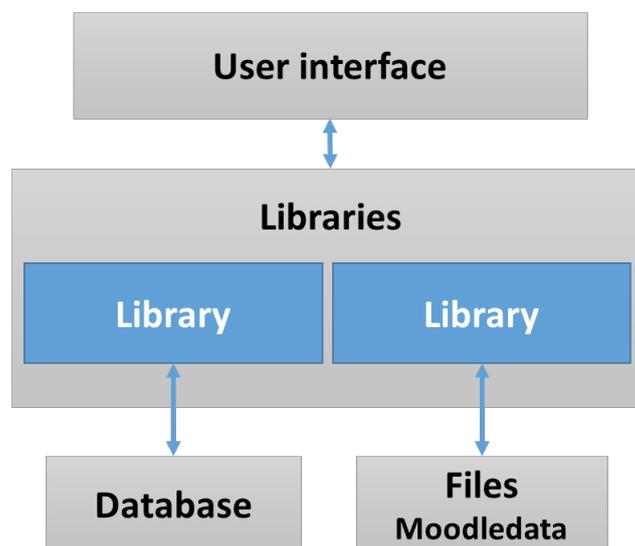


FIGURE II.4.14 – Architecture Moodle

#### 4.3.1.2 Développement de plugin Moodle

Un plugin ou module d'extension est un paquet qui complète un logiciel hôte pour lui apporter de nouvelles fonctionnalités. Il peut être positionné dans une séquence pédagogique d'un cours. Le tableau suivant montre quelques fichiers de bases d'un plugin moodle.

TABLEAU II.4.1 – Fichiers Moodle

Fichiers et dossiers	Description
version.php	Il contient les informations suivantes : la version déclarée du module ; la version minimale requise de Moodle pour que le plugin fonctionne et les paramètres du cron(tâche régulière)
index.html	Permet d'atteindre la liste des instances des objets pédagogiques dans un cours
mod.html	Permet à Moodle d'atteindre l'écran de modification des objets
icon.gif	L'icône qui apparaîtra dans le cours pour symboliser l'activité
lib.php	Librairie des fonctions où Moodle va chercher les divers callbacks qui lui apportent des informations "standard" sur le module
Le fichier "view.php"	Permet de réaliser tous les écrans du module
Backup	Dossier contenant les informations de sauvegarde et de restauration

#### 4.3.1.3 Architecture de la solution

La figure II.4.15 ci-après montre l'intégration de notre plugin dans Moodle. Ce plugin utilise un serveur externe WebRTC dont l'adresse IP sera paramétrée dans le fichier de configuration du module et est matérialisé sur la même figure par :

- Présentation Lib : représente l'ensemble des bibliothèques de notre plugin qui interagissent entre autre avec WebRTC API.
- Présentation UI : représente les vues de notre plugin

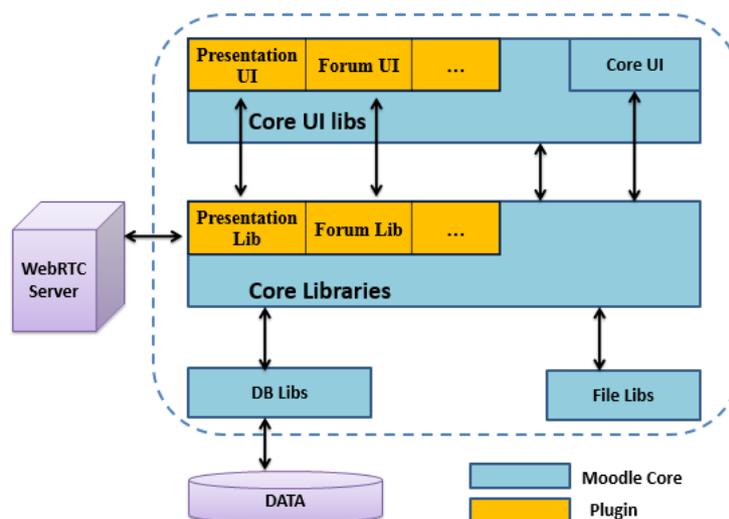


FIGURE II.4.15 – Architecture de la solution

#### 4.3.1.4 Fonctionnement de l'application

La figure II.4.18 montre le processus de mise en communication d'un enseignant avec les étudiants dans une classe virtuelle qui intègre, la voix, la vidéo, le tchat et une présentation de slides. Il faut noter que lorsqu'un utilisateur se connecte sur un serveur WebRTC un identifiant unique est généré et lui est attribué. Exemple : **65-75eCC2jY8jPniEqNk**

Il est donc important de créer un lien entre l'identifiant WebRTC d'un utilisateur et son identifiant Moodle.

Ce mapping est important car il évite d'afficher dans notre application les identifiants WebRTC des utilisateurs qui sont incompréhensibles. La figure II.4.16 ci-après explique le processus de mapping de l'identifiant WebRTC d'un utilisateur et son identifiant Moodle que nous avons implémenté dans notre application.

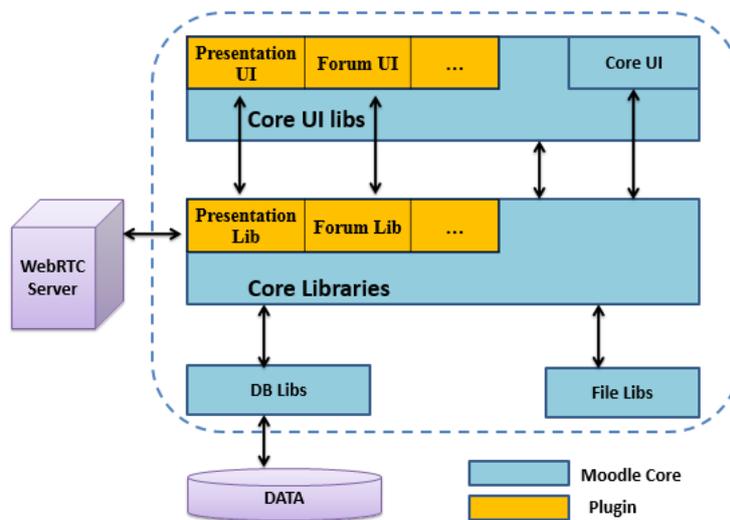


FIGURE II.4.16 – Processus de mapping

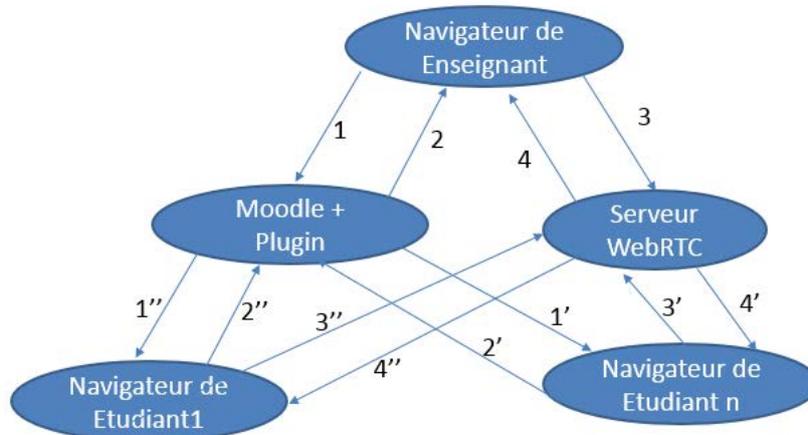


FIGURE II.4.17 – Fonctionnement du plugin

- 1, 1', 1'' : Demande de chargement de la présentation
- 2, 2', 2'' : Chargement effectué

- 3, 3', 3'' : Demande connexion au serveur WebRTC
- 4, 4', 4'' : Etablissement de la connexion au serveur WebRTC

#### 4.3.1.5 Résultats

##### Présentation de la classe virtuelle

La figure II.4.18 montre l'interface de notre solution de classe virtuelle dans Moodle. Elle montre un enseignant présentant un cours en ligne suivi par plusieurs étudiants.



FIGURE II.4.18 – Classe virtuelle

##### Tableau blanc

La figure II.4.19 montre l'affichage d'un tableau blanc pouvant servir de support de dessin aux différents acteurs connectés et qui peuvent en même temps communiquer en vidéo.

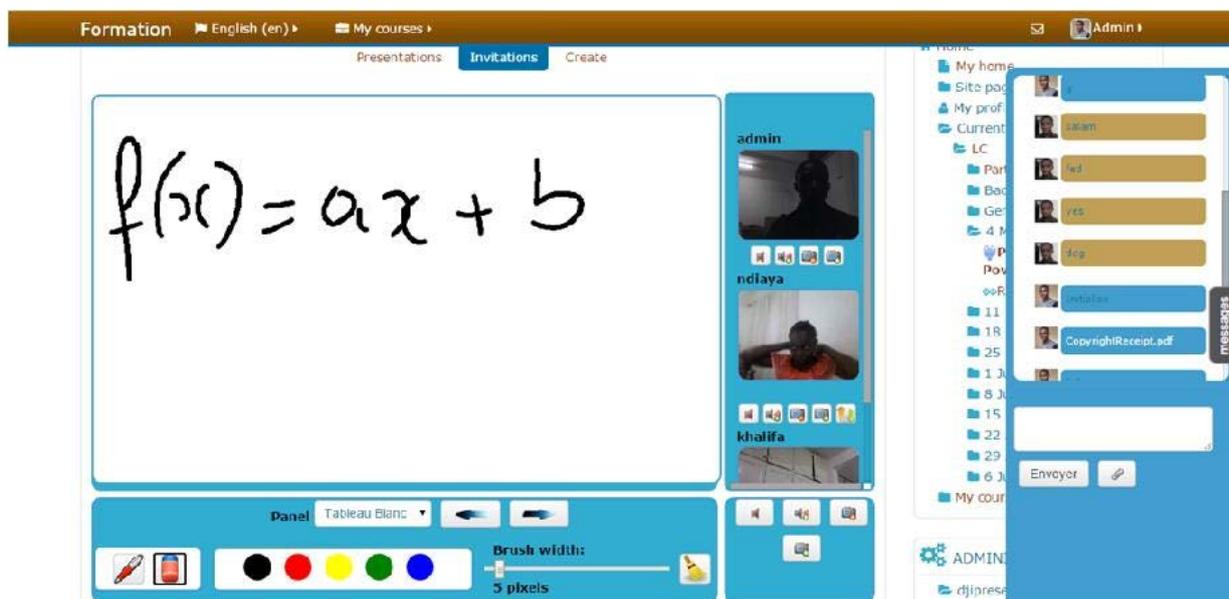


FIGURE II.4.19 – Tableau blanc

## Éditeur Mathématiques

La figure II.4.20 montre l'affichage d'un environnement permettant de pouvoir traiter des exercices utilisant des symboles mathématiques tout en ayant la possibilité de communiquer en vidéo avec les autres acteurs connectés ; ce qui permet à tout le monde de participer, d'aider ou d'orienter le raisonnement de l'acteur désigné devant résoudre le problème.

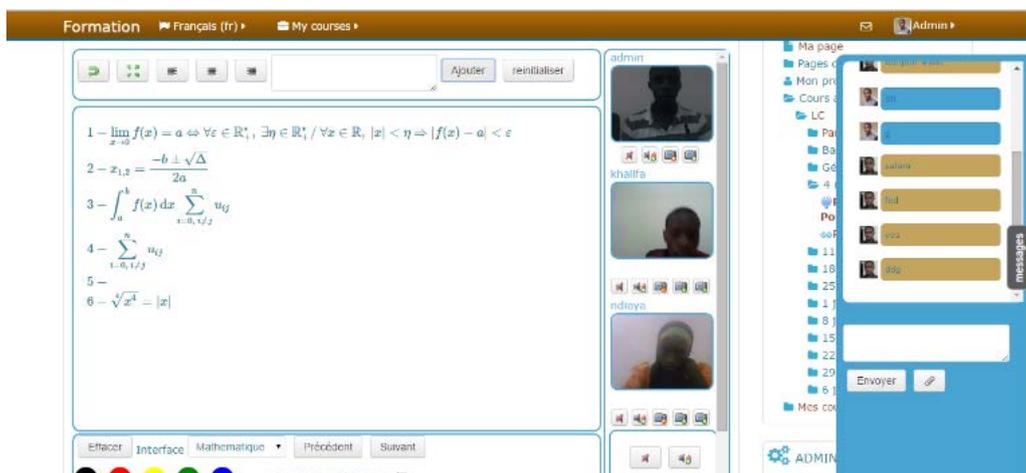


FIGURE II.4.20 – Éditeur Mathématiques

## Transfert de fichiers

La figure II.4.21 montre comment envoyer des fichiers en glisser déposer dans notre application.

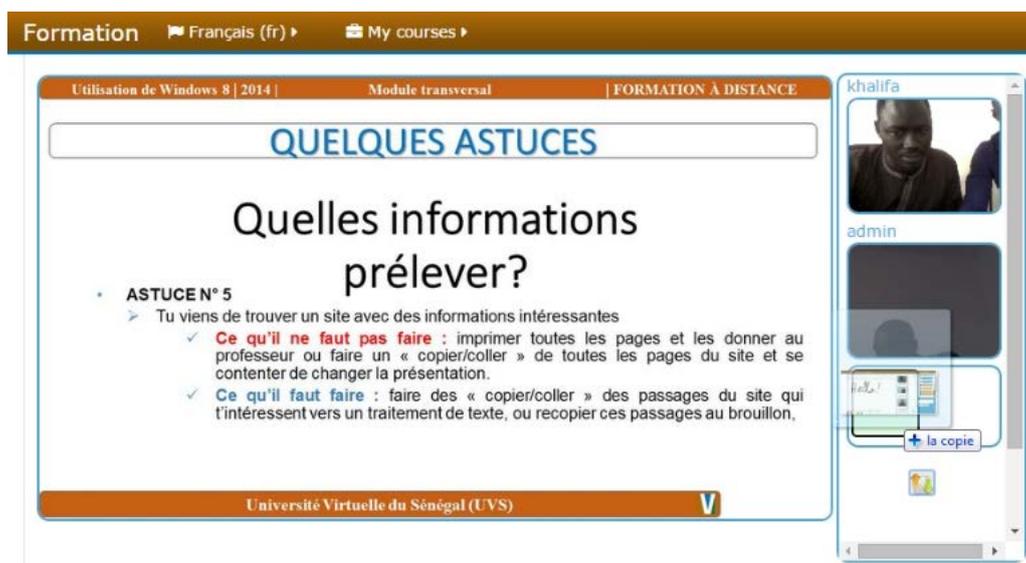


FIGURE II.4.21 – Transfert de fichiers

La figure II.4.22 montre la notification au destinataire. Ce dernier est libre d'accepter ou de refuser le fichier envoyé. Dans le cas où le destinataire accepte alors le fichier est téléchargé sur son terminal.

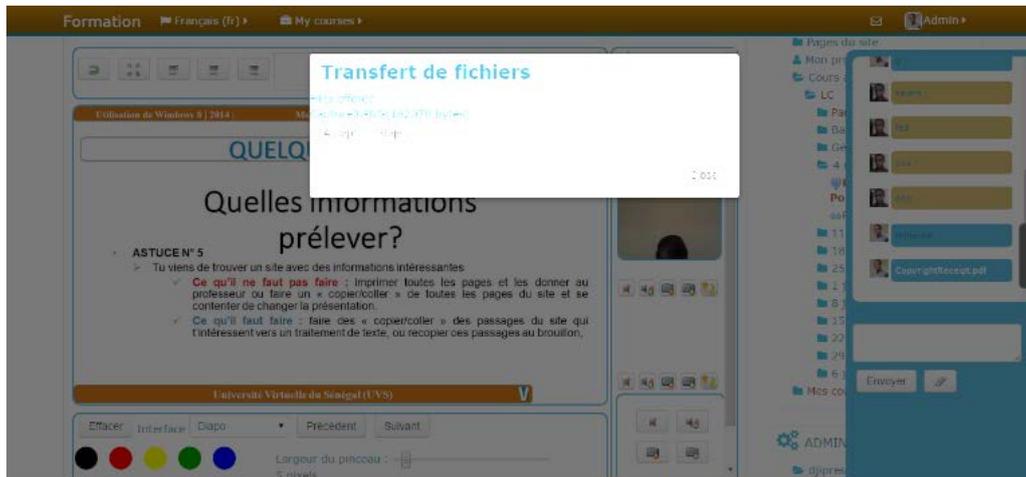


FIGURE II.4.22 – Réception de fichiers

Dans cette partie, nous avons proposé un plugin permettant d'étendre les fonctionnalités de Moodle. Les tests que nous avons effectués à l'UVS permettent aux acteurs d'e-learning qui sont déjà habitués à la plateforme Moodle de pouvoir communiquer en temps réel en audio, vidéo, tchat entre eux tout en restant dans l'environnement Moodle. La solution permet aussi à ces différents acteurs d'échanger des fichiers de grande taille grâce à la souplesse de la technologie WebRTC. Cette plateforme a été aussi expérimentée avec des étudiants de Mathématiques pour les évaluations en ligne nécessitant l'utilisation massive des symboles mathématiques. Quelques séances d'utilisation de notre plugin a permis à ces étudiants d'être à l'aise dans la formulation de leurs réponses en mathématiques en ligne. La fonctionnalité d'enregistrement d'audio de notre solution a permis aux enseignants de langue de tester le niveau de compréhension et restitution orale des étudiants. Vu que les échanges audio, vidéo, et le transfert de fichiers entre deux utilisateurs utilisant WebRTC se font d'utilisateur à utilisateur sans passer par le serveur de signalisation WebRTC, nous poussent à penser que WebRTC peut permettre à des plateformes e-learning de supporter un nombre important d'échange de données en mode synchrone. L'utilisation de notre plugin Moodle basé sur WebRTC à partir des terminaux aussi variés que des smartphones, des tablettes, sans compter les ordinateurs dotés de navigateurs, est un atout de taille vis-à-vis d'autres solutions utilisant des technologies tel que flash. Nos prochains travaux devraient s'orienter vers les tests de robustesse et de montée en charge des solutions basées sur WebRTC comparées à d'autres solutions.

### 4.3.2 Proposition d'une plateforme de développement logiciel collaboratif pour les universités numériques

Cette partie présente une plateforme de développement collaboratif de logiciel permettant aux acteurs des universités traditionnelles et virtuelles qui sont géographiquement éloignés de pouvoir travailler sur un projet de développement logiciel utilisant des langages de programmation usuels.

Dans le contexte des universités virtuelles opérant dans le domaine des STEM plus particulièrement du génie logiciel, les travaux pratiques sont nécessaires pour acquérir des compétences. Cette solution fournit aux différents acteurs de ces universités virtuelles et traditionnelles qui font du e-learning un environnement préconfiguré de développement collaboratif de logiciel incluant la synchronisation des fichiers d'un projet, la messagerie instantanée, l'audio et la vidéo.

Notre plateforme dispose d'une interface d'administration permettant de gérer les droits et l'allocation des machines virtuelles préconfigurées aux différents acteurs.

Pour atteindre nos objectifs, nous avons utilisé le protocole XMPP pour la messagerie instantanée et la synchronisation des fichiers puis le WebRTC pour l'audio et la vidéo entre les différents acteurs et enfin le protocole RDP pour l'accès à distance aux machines virtuelles.

Nous montrons dans ce travail comment notre plateforme a été utilisée par les enseignants et étudiants de Mathématiques Appliquées et Informatique pour des travaux pratiques en C/C++, Java, Python, PHP.

La république du Sénégal a organisé en 2013 une concertation nationale sur l'avenir de l'enseignement supérieur. A l'issue de cette concertation, deux décisions importantes ont été prises. La première décision est relative à la réorientation de l'enseignement supérieur vers les sciences technologie ingénierie et mathématique (STEM).

La seconde décision est l'utilisation des TIC au service de l'enseignement supérieur en vue d'améliorer son efficacité. Cette dernière décision a été traduite en action par la création de l'université virtuelle du Sénégal (UVS) et ses espaces numériques ouverts (ENO).

La question est comment former efficacement des étudiants en ligne dans le domaine des STEM ?

Plusieurs auteurs ont fait part de leurs expériences en termes de laboratoires en ligne dans le domaine des STEM [78][79].

Dans cette partie, l'objectif principal est de savoir comment fournir une plateforme de développement collaboratif pour les universités virtuelles.

Cette partie propose un environnement de collaboration préconfiguré et adapté à chaque projet de développement logiciel incluant des outils de communication temps-réel. Pour ce faire nous sommes parti d'un plugin d'Eclipse open source Saros [80] qui offre une partie des fonctionnalités que nous voulons et nous l'avons étendu pour y intégrer l'audio et la vidéo via le WebRTC [81].

Pour permettre aux différents acteurs d'avoir un environnement préconfiguré facile à dupliquer nous avons fait appel à la virtualisation en utilisant Virtualbox [82]. Le choix de Virtualbox est guidé entre autre par la possibilité d'utilisation du plugin Virtualbox Extension Pack qui active un serveur RDP [83] appelé serveur Virtualbox RDP. Ce serveur VRDE permettra aux acteurs d'utiliser un client RDP à partir de Windows ou de Linux ou de Mac ou d'un client léger pour accéder à des machines que l'administrateur de notre plateforme leur a allouées.

La suite de cette partie est organisée comme suit : Nous décrirons tout d'abord l'état de l'art avant de décrire notre environnement de développement collaboratif, ensuite nous présenterons les principaux cas d'utilisation de notre solution, et enfin nous présenterons les résultats obtenus.

#### 4.3.2.1 État de l'art

Dans la littérature, des chercheurs tels que [84] ont montré que la distance séparant les acteurs d'un projet de développement collaboratif n'est plus un problème compte tenu des moyens technologiques actuels. Des modèles efficaces d'organisation d'équipes de développement collaboratif ont été éprouvés par des chercheurs tels que [84].

Les études menées par les chercheurs [85][86][87] ont montré l'intérêt d'avoir un environnement de développement social qui unifie les concepts de réseau social et environnement collaboratif appliquant le modèle 3C (coopération, communication et coordination) pour maintenir la motivation des différents membres de l'équipe de développement.

En environnement e-learning les études menées par [88] montrent l'importance du modèle socioconstructivisme qui prône la collaboration entre pairs pour construire un savoir solide. Dans [89], l'auteur pose le problème de suivi et de mesure de l'apport de chaque étudiant en environnement e-learning.

Notre approche consiste à avoir un environnement de développement coopératif de logiciels intégrant des outils de communication, de coopération et permettant à un enseignant de suivre et de mesurer l'apport de chaque étudiant sur un projet donné.

#### 4.3.2.2 Description de l'environnement de développement collaboratif proposé

##### Architecture réseau

Dans l'environnement proposé, il est mis à disposition des acteurs des machines virtuelles sur lesquelles est installé et préconfiguré le logiciel Eclipse avec une version du plugin Saros que nous avons adapté à notre besoin de communication audio et vidéo intégré.

Les fonctionnalités de l'environnement proposé sont :

- l'accès à distance aux machines virtuelles
- la synchronisation des fichiers d'un projet Eclipse entre les différents acteurs
- une conférence de tchat entre les acteurs
- une conférence audio et vidéo entre les acteurs
- la génération, la gestion des droits d'accès et l'allocation des machines virtuelles aux acteurs

La figure II.4.23 montre l'architecture réseau de l'environnement.

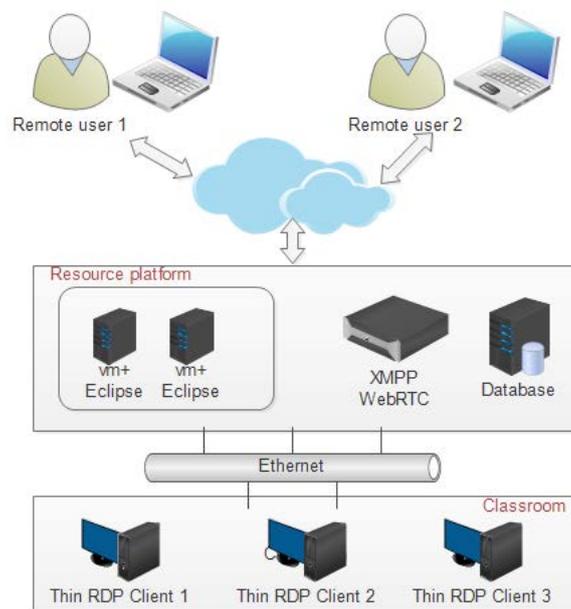


FIGURE II.4.23 – Architecture réseau de l'environnement

Dans la proposition, la conférence de tchat est gérée avec un serveur XMPP, la conférence audio et vidéo est gérée avec le WebRTC et la gestion et allocation des machines virtuelles se fait via une application web que nous avons développée.

Les acteurs accèdent aux machines virtuelles en utilisant des clients RDP.

## Architecture logicielle

Le plugin Eclipse Saros est basé sur le protocole XMPP, il utilise une architecture client/serveur. La communication entre les clients se fait par des messages XMPP. L'échange des actions du client (les modifications, la sélection, les opérations sur les fichiers du projet, ...) se fait en enveloppant la description de l'activité dans les messages XMPP et en les envoyant aux autres clients. Le transfert de fichiers quant à lui se fait en peer-to-peer.

Lors de la collaboration sur un projet, une session Saros est initiée pour gérer l'invitation des membres de la session et la synchronisation des actions effectuées par tous les participants de la session.

Pour ajouter le support de l'audio et de la vidéo à Saros, nous avons modifié sa classe SarosSessionManager qui gère le démarrage, l'ajout de contact et l'arrêt des sessions afin de récupérer les paramètres (login, password, adresse du serveur XMPP) de chaque utilisateur en vue de les ajouter dans une conférence WebRTC grâce au navigateur JxBrowser [90].

La figure II.4.24 montre une partie des modifications apportées au code de Saros.

```
// Add WebRTC audio/video support
final XMPPAccount account = xmppAccountStore.getActiveAccount();
String login = account.getUsername();
String passwd = account.getPassword();
String host = account.getDomain();
String confid = sessionId;

OfmeetConference.startConference(login, passwd, host, confid);
```

FIGURE II.4.24 – Ajout de la conférence audio/vidéo dans Saros

### 4.3.2.3 Description de quelques cas d'utilisation de la plateforme

#### Assignment d'une machine virtuelle à un utilisateur

La figure II.4.25 montre la création et l'assignation d'une machine virtuelle à un utilisateur.

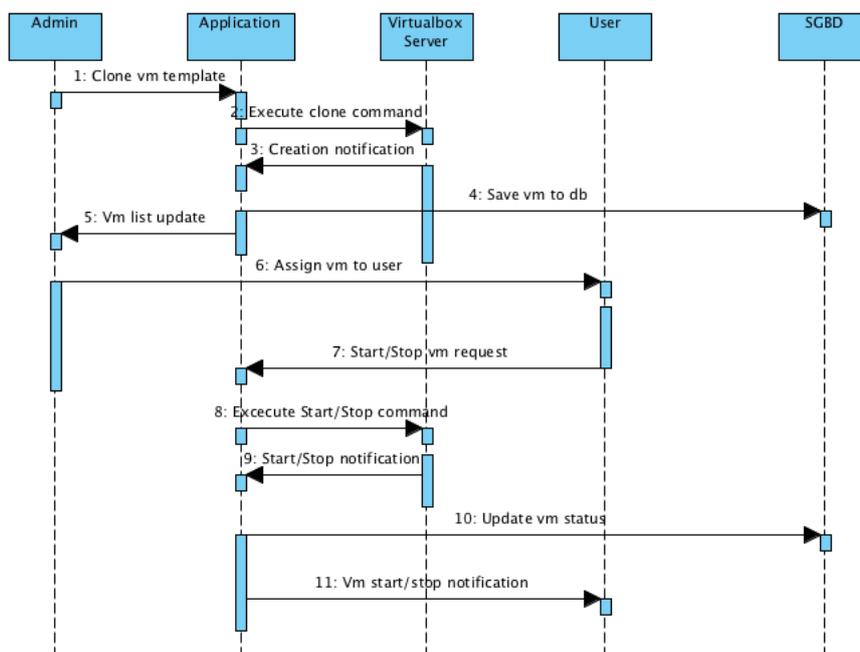


FIGURE II.4.25 – Diagramme décrivant l'assignation d'une VM

## Le travail collaboratif dans le cadre d'un TP

Un enseignant crée un projet et invite un groupe d'étudiants qui sont géographiquement dispersés ou non à y travailler ensemble. Ce projet peut utiliser les langages de programmation tels que Java, C/C++, PHP, Python, HTML5/CSS3 ...

La figure II.4.26 décrit le processus de collaboration sur le projet créé.

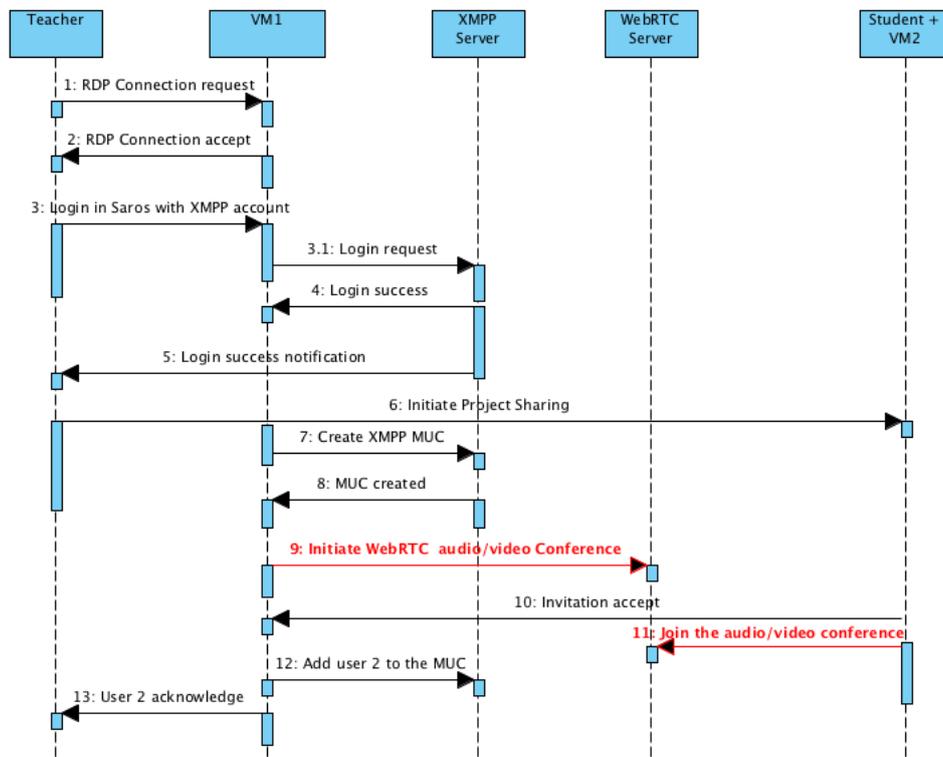


FIGURE II.4.26 – Diagramme décrivant le travail collaboratif

### 4.3.2.4 Résultats

La figure II.4.27 montre le suivi du développement collaboratif par l'enseignant.

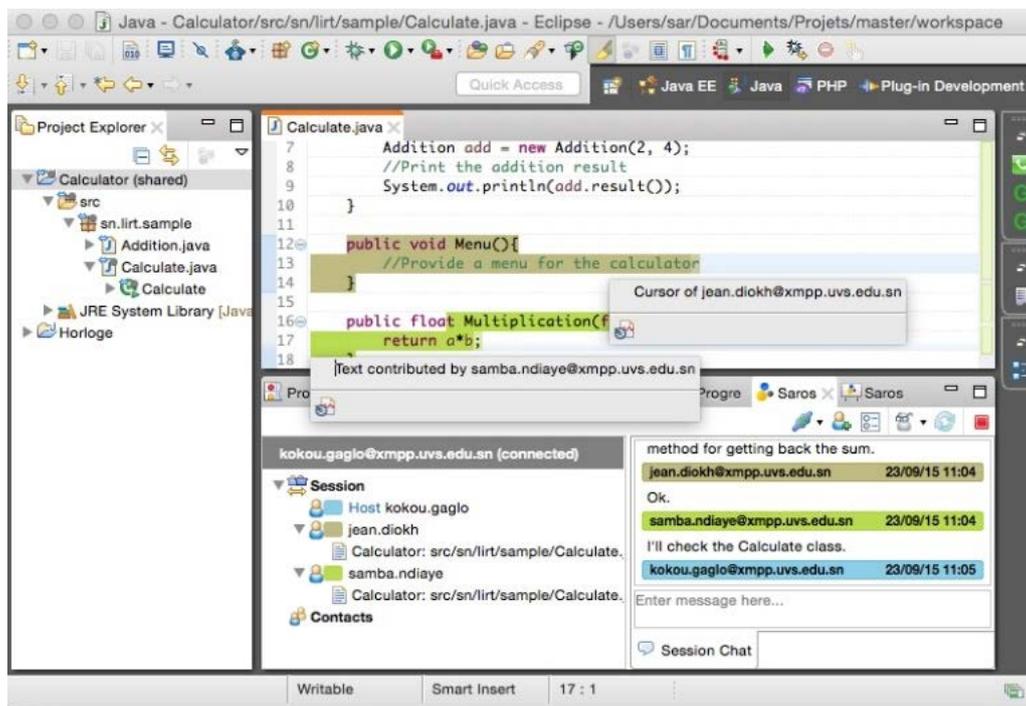


FIGURE II.4.27 – Développement de logiciel collaboratif

La figure II.4.28 montre les différents membres du groupe de travail. Chacun est identifié par une couleur et on peut voir en temps-réel le fichier sur lequel il travaille.

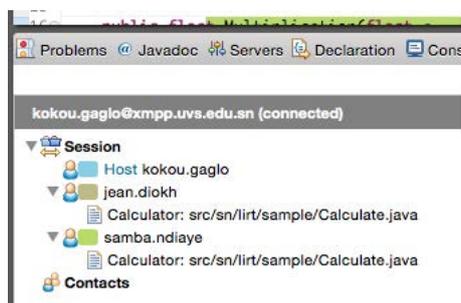


FIGURE II.4.28 – Les membres de session de la TP

La figure II.4.29 montre l'enseignant (kokou.gaglo) en train de donner des instructions aux étudiants grâce à la conférence XMPP.

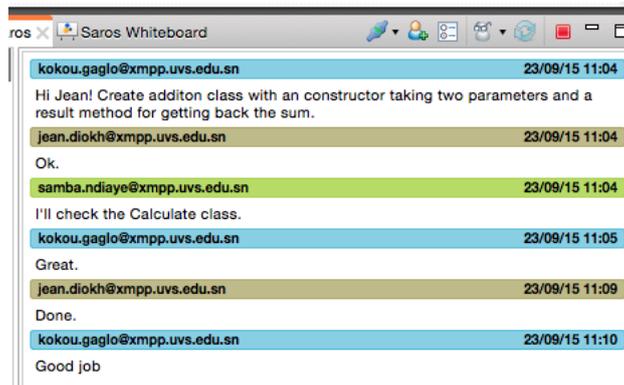


FIGURE II.4.29 – Discussion dans la conférence XMPP

La figure II.4.30 montre la fonctionnalité de conférence audio/vidéo WebRTC initiée lors des travaux pratiques entre un enseignant et ses étudiants.



FIGURE II.4.30 – Session audio/vidéo

Les figures II.4.31 et II.4.32 présentent la plateforme de gestion des ressources et des utilisateurs.



The screenshot shows the UVS user management interface. At the top, there is a navigation bar with 'Home', 'Hosts', 'Role', and 'User' tabs. Below the navigation bar, there is a table with 6 columns: 'List (6)', 'Create', 'With selected', 'First Name', 'Last Name', 'Email', 'Xmpp User', and 'Xmpp Pass'. The table contains 6 rows of user data.

		First Name	Last Name	Email	Xmpp User	Xmpp Pass
<input type="checkbox"/>		Samba	Ndiaye	samba.ndiaye@uvs.edu.sn	samba.ndiaye@xmpp.uvs.edu.sn	3s4as1
<input type="checkbox"/>		Noella	Faye	noella.faye@uvs.edu.sn	noella.faye@xmpp.uvs.edu.sn	c610e3
<input type="checkbox"/>		Jean	Diokh	jean.diokh@uvs.edu.sn	jean.diokh@xmpp.uvs.edu.sn	66ct61
<input type="checkbox"/>		Ibrahima	Sanogo	ibrahima.sanogo@uvs.edu.sn	ibrahima.sanogo@xmpp.uvs.edu.sn	r43142
<input type="checkbox"/>		Kokou	Gaglo	kokou.gaglo@uvs.edu.sn	kokou.gaglo@xmpp.uvs.edu.sn	13c043

FIGURE II.4.31 – Gestion des utilisateurs



The screenshot shows the UVS interface for assigning a virtual machine to a user. The user 'Noella Faye' is selected. The interface displays details for the 'Windo8' VM and the 'VRDE' service. The 'Assign to user' dropdown is set to 'Faye'.

**Windo8**  
 OS : Windows 8.1 (64 bit)  
 Status : poweroff  
 Memory : 2048 Mo

**Assign to :**  
 Noella Faye  
 XMPP user : noella.faye@xmpp.uvs.edu.sn  
 XMPP pass : c610e3

**VRDE**  
 Status : on  
 Port : 3300  
 Host IP : virt.uvs.edu.sn  
 Start Stop

Assign to user :  
 Faye Go

FIGURE II.4.32 – Assignation de machine virtuelle

Une fois l'étudiant connecté, il peut démarrer/arrêter la machine virtuelle qui lui est attribuée, s'y connecter avec un client RDP et commencer à collaborer avec son enseignant grâce au logiciel Eclipse installé et préconfiguré. La figure II.4.33 montre l'interface de connexion d'un étudiant avec tous les paramètres dont il aura besoin.



FIGURE II.4.33 – Interface utilisateur

Dans cette partie, nous avons proposé une plateforme de développement collaboratif de logiciel permettant à des enseignants d'organiser des séances de travaux pratiques avec leurs étudiants géographiquement dispersés.

L'environnement proposé a été testé à l'Université Virtuelle du Sénégal par les étudiants et enseignants de la filière mathématiques appliquées et informatique. Cela leur a permis de collaborer sur des projets utilisant les langages de programmation C/C++, Java, Python, PHP, HTML5/CSS3.

Les nouvelles fonctionnalités de création automatique de conférence vidéo associé à un projet et utilisable directement dans l'environnement de développement ont contribué à augmenter le niveau de motivation des étudiants et constituent un apport appréciable aux fonctionnalités classiques de Saros.

L'interface d'administration développée s'appuie sur la virtualisation pour générer de manière automatique des machines virtuelles préconfigurées avec notre solution et utilisable à distance par les acteurs grâce au protocole d'accès à distance RDP. Cela a permis d'éviter de reproduire manuellement notre solution sur les machines des différents acteurs.

En perspective, nous comptons intégrer notre plateforme dans un Cloud Computing pour mieux gérer la montée en charge et le désir des enseignants d'avoir un nombre assez conséquent d'étudiants par groupe de collaboration.

## 4.4 Conclusion

L'ensemble des outils WebRTC que nous avons proposés pourront servir aussi bien à des universités classiques pour gérer la massification qu'aux universités virtuelles.

Ces solutions pourront encourager des enseignants à réintégrer dans leur enseignement les travaux pratiques.

En effet, dans beaucoup d'établissements de nos universités, certains TP ne sont pas faits compte tenu du nombre important des étudiants.

Le fait que les solutions proposées ne dépendent pas d'Internet pourra favoriser la collaboration de plusieurs groupes d'étudiants géographiquement éparpillés dans un campus pédagogique sur des séances de travaux pratiques.

# Conclusion générale

A l'ère de l'économie numérique, il devient plus pressant d'avoir une ressource humaine de qualité pour prétendre à l'émergence de son pays ou de maintenir son niveau de développement.

Des politiques de l'accompagnement de l'enseignement primaire et secondaire mis en place ces dernières années en Afrique commencent à donner des résultats positifs et ont permis d'accroître d'année à année le nombre d'élèves obtenant leur baccalauréat et qui frappent à la porte de l'enseignement supérieur.

Beaucoup de pays Africains n'ont pas encore atteint le seuil minimal de jeunes à l'âge d'être dans l'enseignement et qui y soient. Pourtant, on assiste à une massification des amphithéâtres des universités classiques faute d'infrastructures physiques et d'enseignants en nombre suffisant.

De l'autre côté, nous assistons à l'émergence de moyens de communication de plus en plus performants et des jeunes qui utilisent massivement des réseaux sociaux pour communiquer. Quand nous examinons les modèles d'organisation des acteurs du monde des logiciels libres, on se rend compte des résultats positifs que ces modèles ont fournis dans beaucoup de secteurs.

Ces acteurs de logiciels libres bien que géographiquement éloignés ont su profiter de l'Internet et ont mis en place une forme d'organisation virtuelle qui leur a permis d'obtenir des résultats remarquables.

Nous avons donc émis l'hypothèse que l'évolution positive des services des télécommunications et une organisation appropriée pourraient être une solution pour contribuer d'une part, au règlement de problème de massification des effectifs des étudiants dans l'enseignement supérieur et d'autre part, au développement d'un modèle hybride d'enseignement qui tire profit des technologies de l'information et de la communication pour améliorer significativement le service rendu aux étudiants .

Nous avons proposé dans cette deuxième partie de thèse plusieurs modèles de solutions pour non seulement favoriser les travaux pratiques dans le domaine des STEM dans les universités classiques mais aussi faire en sorte que les formations dans les universités numériques puissent véritablement permettre aux Etats Africains de former efficacement les acteurs de l'économie numérique qui ne soient pas déconnectés des réalités sociologiques de leur environnement car ne collaborant pas seulement derrière les ordinateurs.

En effet, les solutions de collaboration permettant à des différents acteurs des universités géographiquement éloignés utilisant des technologies WebRTC pour rendre négligeables les distances, pourraient accompagner la réorientation du système éducatif vers les STEM dans les pays Africains qui le souhaitent.

Des propositions de modèles de mutualisation et de partage autour de cloud computing ont été faits et expérimentés avec succès dans nos travaux et leur adoption par les pays aspirant à l'émergence pourrait contribuer à créer dans les établissements d'enseignement supérieur de ces pays des pôles d'excellence spécialisés qui vont bénéficier de la généralisation de haut débit pour se constituer en organisation virtuelle dans le but d'augmenter l'efficacité de l'enseignement supérieur.

## Contributions

Des résultats obtenus dans nos travaux ont fait l'objet de 7 publications IEEE.

- [1] Samuel Ouya, Ahmath Bamba Mbacke, Gervais Mendy, Pape Waly Diouf and Kalidou Sy. "**Social network integration to e-learning environments : a solution for dropout rate education and relationships recovery factor between different actors in distance education**" in ACS/IEEE International Conference on Computer Systems and Applications AICCSA 2015, November 17-20, 2015 Marrakech, Morocco
- [2] Moussavou, D.E; Ouya, S.; Faye P.D.; Niane M.T.; Lishou C, "**Remote Laboratory experience for STEM education :the case of Senegal Virtual University**", in International Conference on Interactive Collaborative Learning (ICL), 20-24 September 2015, Florence, Italy
- [3] Dahirou Gueye, A.; Ouya, S.; Lishou, C.; Sanogo, I.; Saliah-Hassane, H., "**Decentralized authentication method for accessing pedagogical resources in a cloud computing based virtual organization**" in Interactive Collaborative Learning (ICL) , 2014 International Conference on , vol., no., pp.656-662, 3-6 Dec. 2014
- [4] Gueye, A.D.; Ouya, S.; Moussavou, D.E.; Sanogo, I.; Saliah-Hassane, H.; Lishou, C., "**Dynamic publishing and availability management of virtual machines in Virtual Organization**" ,in Global Engineering Education Conference (EDUCON), 2015 IEEE , vol., no., pp.793-798, 18-20 March 2015
- [5] Samuel Ouya; Khalifa Sylla; Pape Mamadou Djidiack Faye; Claude Lishou , "**Impact of integrating WebRTC in universities e-learning platforms**", 2015 5th World Congress on Information and Communication Technologies (WICT), 978-1-4673-8712-5/15/©2015 , IEEE
- [6] Samuel Ouya; Kokou Gaglo; Gervais Mendy; Ahmath Bamba Mbacké and Claude Lishou, "**Proposal of a collaborative software development platform for the virtual universities : the Virtual University of the Senegal (UVS) experience**" 2015 5th World Congress on Information and Communication Technologies (WICT), 978-1-4673-8712-5/15/©2015 IEEE
- [7] Samuel Ouya; Cheikhane Seyed; Ahmath Bamba Mbacké , "**WebRTC platform proposition as a support to the educational system of universities in a limited Internet connection context**", 2015 5th World Congress on Information and Communication Technologies (WICT), 978-1-4673-8712-5/15/ ©2015 IEEE

# Conclusion générale de la thèse

L'étude que nous avons menée dans cette thèse interpelle nos pays en voie de développement sur leur responsabilité dans l'exploitation du dividende numérique qui joue aujourd'hui le même rôle que les matières premières que ces pays exportaient de manière brute pour être transformées ailleurs. Les produits issus de cette transformation sont par la suite vendus à prix élevé à ces pays.

Certains pays seraient tentés de vendre toutes les plages des fréquences issues du dividende numérique aux opérateurs de Télécommunications qui vont mettre en place des services de télécommunications qui seront revendus à l'Etat pour les secteurs de la santé et l'enseignement.

Alors qu'une plage de fréquences allouée à l'enseignement et la santé aurait permis de déployer un réseau IP haut débit mobile dont les bénéfices seraient considérables.

La mise en place d'un tel réseau haut débit full IP, dont le coût de déploiement peut être optimisé comme le montre le chapitre 4 de la partie I de notre thèse, permettrait la promotion des services innovants à valeur ajoutée ou des solutions d'interactivité forte téléspectateurs et chaînes de télévisions numériques. Ces solutions pourraient véritablement être utilisées pour non seulement former en masse des jeunes Africains acteurs de l'émergence du continent mais aussi favoriser le développement de la télé-médecine.

En effet, beaucoup d'enseignants des universités devant la massification des effectifs d'étudiants ont tendance à se contenter des cours théoriques dans les domaines des STEM et des langues. Or ces travaux pratiques sont très importants dans l'acquisition des compétences.

Ainsi l'existence d'un réseau haut débit full IP, appartenant à l'Etat, accessible aux différents acteurs de l'enseignement supérieur, même non connecté à Internet, peut être le support des services innovants, que nous avons proposés dans la deuxième partie de cette thèse, pour redonner le goût des travaux pratiques aux enseignants opérant dans le domaine des STEM et de langues.

# Bibliographie

- [1] André Pérez. *Architecture des réseaux mobiles : GSM/GPRS, UMTS/HSPA, EPS, NGN, IMS*, lavoisier edition, oct 2010.
- [2] Future Computer and Communication. Ims network architecture. *ICFCC 2009*, pages 329–333, Apr 2009. International Conference.
- [3] André Pérez. *La voix sur LTE : Réseau 4G et architecture IMS*, lavoisier edition, October 2010.
- [4] Laurent Ouakil and Pujolle Guy. *Téléphonie sur IP*, seconde edition, 2008. ISBN :978-2-212-12359-3.
- [5] Rosenberg, J. AND Schulzrinne, H. AND Camarillo, G. AND Johnston, A. AND Peterson, J. AND Sparks, R. AND Handley, M. AND E. Schooler. *SIP : Session Initialtion Protocol*, jun 2002. RFC 3261.
- [6] S. Lataste and B Toussou. *From network layer to IMS service continuity*, 2008. In proceedings on 12th International Conference of Intelligence in Service Delivery Networks.
- [7] 3rd Generation Partnership Project. *Universal Mobile Telecommunications System (UMTS); LTE; Open Service Access (OSA)*, mar 2010. 3GPP TS 23.198 version 8.0.0 Stage 2 (Release 8).
- [8] A.H. Darvishan, H. Yeganeh, K. Bamasian, and H. Ahmadian. Osa parlayx gateway architecture for 3rdparty operators participation in next generation networks. *Advanced Communication Technology (ICACT)*, pages 75–80, feb 2010. 2010 The 12th International Conference.
- [9] *3GPP TS 23.228 : IP Multimedia Subsystem (IMS); Stage 2*. <http://www.3gpp.org/DynaReport/23228.htm>.
- [10] 3rd Generation Partnership Project. *Technical Specification Group Services and System Aspects; Study on Web Real Time Communication (WebRTC) access to IP Multimedia Subsystem (IMS)*, dec 2013. 3GPP TR 23.701 V12.0.0 , (Release 12).
- [11] S. Kawade and M. Nekovee. Broadband wireless delivery using an inside-out tv white space network architecture. *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–6, dec 2011.
- [12] Yann BACHY. *Sécurité des équipements grand public connectés à Internet : évaluation des liens de communication*. PhD thesis, Institut National des Sciences Appliquées de Toulouse (INSA de Toulouse), France, 2015.
- [13] *Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS) Point-to-Point (PP)*. ETSI, ETS TS 300 901, GSM 03.40 version 5.8.1 (Release 1996).
- [14] Xun Zhaoyong, Wang Wenbo, and Qi Qi. A new ifc-sfc-based dynamic service triggering algorithm for ims network. *Broadband Network and Multimedia Technology (IC-BNMT), 2013 5th IEEE International Conference*, pages 39–44, nov 2013.

- [15] R. Spiers, N. Ventura, and T. Magedanz. Improvements on the application triggering architecture for the ip multimedia subsystem. *AFRICON, 2011*, pages 1–6, sept 2011.
- [16] 3rd Generation Partnership Project. "digital cellular telecommunications system (phase 2+); universal mobile telecommunications system (umts); general packet radio service (gprs); service description", dec 2006. 3GPP TS 23.060 V6.15.0 Stage 2 (Release 6).
- [17] 3rd Generation Partnership Project. "technical specification group gsm/edge radio access network; mobile switching centre - base station system (msc-bss) interface; layer 3 specification", dec 2009. 3GPP TS 48.008 V8.8.0 (Release 8).
- [18] 3rd Generation Partnership Project. "digital cellular telecommunications system (phase2+); universal mobile telecommunications system (umts), technical realization of the short message service (sms)", oct 2014. 3GPP TS 23.040 V12.2.0 (Release 12).
- [19] 3rd Generation Partnership Project. Universal mobile telecommunications system (umts); lte; architecture enhancements for non-3gpp accesses. 3GPP TS 23.402 version 12.6.0 (Release 12).
- [20] Telecommunications and internet converged services and protocols for advanced networking (tispan); ngn functional architecture; network attachment sub-system (nass), mar 2010. ETSI ES 282 004 V3.4.1.
- [21] Yang, himin, AND Wen, Xiangming AND Zheng, Wei AND Lu, Zhaoming . Convergence architecture of internet of things and 3gpp lte-a network based on ims. *Mobile Congress (GMC)*, pages 1–7, oct 2011.
- [22] A. Tekovic, I. Pesut, and Z. Moric. Voice service in an lte network-csfb. *ELMAR*, pages 251–254, sep 2013. 55th International Symposium.
- [23] L. Bhebhe and M.M Heiskanen. Performance of circuit switched fall back and single radio voice call continuity from td-lte to umts. *Software, Telecommunications and Computer Networks (Soft-COM)*, pages 84–88, sept 2014. 22nd International Conference.
- [24] 3rd Generation Partnership Project. Technical specification group services and system aspects; single radio voice call continuity (srvc), dec 2012. 3GPP TS 23.216 V11.7.0; Stage 2, (Release 11).
- [25] Koshimizu, T. AND Tanaka, I. AND Nishida, K. Improvement on the volte (voice over lte) domain handover with operator's vision. *World Telecommunications Congress (WTC)*, pages 1–5, march 2012.
- [26] 3rd Generation Partnership Project. *IP Multimedia Core Network Subsystem (IMS) Multimedia Telephony Service and supplementary services*, dec 2010. 3GPP TS 22.173 V9.5.0 (Release 9).
- [27] V.F. Tsema and O.I. Ladik. Ip multimedia subsystem billing. *Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET), 2010 International Conference*, page 150, feb 2010.
- [28] R. Spiers, R. Marston, R. Good, and N. Ventura. The uct ims iptv initiative. *Next Generation Mobile Applications, Services and Technologies, 2009. NGMAST '09 . Third International Conference*, pages 503–508, sep 2009.
- [29] GSM Association. Gsma prd ir.92 ims profile for voice and sms. April 2015.
- [30] J. Deruelle. Jslee and sip-servlets interoperability with mobicents communication platform. *Next Generation Mobile Applications, Services and Technologies, 2008. NGMAST 08. The Second International Conference*, pages 634–639, sep 2008.

- [31] W. Elleuch. Models for multimedia conference between browsers based on webrtc. *Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference*, pages 279–284, oct 2013.
- [32] Webrtc 1.0 : Real-time communication between browsers. <http://www.w3.org/TR/webrtc/>.
- [33] GSM Association. *GSMA PRD IR.94 IMS Profile for Conversational Video Service*, Nov 2014.
- [34] Jean-Marc Valin and Cary Bran. *WebRTC Audio Codec and Processing Requirements.*, Feb 2014. <http://tools.ietf.org/html/draft-ietf-rtcweb-audio-05>.
- [35] *VP8 and VP9 video encode and decode – part of WebM : an open web media project.* <http://www.webmproject.org/>.
- [36] *ITU-T Recommendation H.264 : Advanced video coding for generic audiovisual services.* <https://www.itu.int/rec/T-REC-H.264>.
- [37] GSM Association. *IMS Profile for High Definition Video Conference (HDVC) Service*, Jul 2014.
- [38] GSM Association. *Rich Communication Suite 5.2 Advanced Communications Services and Client Specification*, Sep 2014.
- [39] Stephane Proust, Espen Berger, Bernhard Feiten, Bo Burman, Kalyani Bogineni, Miao Lei, and Enrico Marocco. *Additional WebRTC audio codecs for interoperability with legacy networks*, Aug 2104. <http://tools.ietf.org/html/draft-proust-rtcweb-audio-codecs-for-interop-01>.
- [40] *3GPP Work item : Web Real Time Communication access to IMS.* <http://www.3gpp.org/DynaReport/FeatureOrStudyItemFile-580062.htm>.
- [41] B. Modlic, G. Sisul, and M. Cvitkovic. Digital dividend — opportunities for new mobile services. *ELMAR, 2009. ELMAR '09. International Symposium*, pages 28–30, sep 2009.
- [42] P. Siebert. Dvb : Developing global television standards for today and tomorrow. *Telecom World (ITU WT), 2011 Technical Symposium at ITU*, pages 171–175, oct 2011.
- [43] M.M. Anis, X. Lagrange, and R. Pyndiah. A simple model for dvb and lte cooperation. *Broadband Multimedia Systems and Broadcasting (BMSB), 2013 IEEE International Symposium*, pages 1–6, jun 2013.
- [44] Z. Lukac, M. Radonjic, B. Veris, T. Maruna, and N. Kuzmanovic. The experience of implementing a hybrid broadcast broadband television on network enabled tv set. *MIPRO, 2011 Proceedings of the 34th International Convention*, pages 840–844, May 2011.
- [45] K. Merkel. Hybrid broadcast broadband tv, the new way to a comprehensive tv experience. *Electronic Media Technology (CEMT), 2011 14th ITG Conference*, pages 1–4, mar 2011.
- [46] 3rd Generation Partnership Project. Technical specification group services and system aspects;circuit switched (cs) fallback involved packet system (eps);stage 2, mars 2010. 3GPP TS 23.272 V8.7.0 ; (Release 8).
- [47] Redis. [www.redis.io](http://www.redis.io).
- [48] *ETSI, ETS TS 300 901, Digital cellular telecommunications system (Phase 2+); Technical realization of the Short Message Service (SMS) Point-to-Point (PP) (GSM 03.40 version 5.8.1 Release 1996).*
- [49] A2billing. <http://www.a2billing.net/>.
- [50] UNESCO. *UNESCO*.

- [51] Ministère de l'Enseignement Supérieur et de la Recherche. *MESR*. MESR.
- [52] Richard Stallman and Joshua Gay. *Free Software, Free Society*, 2009. ISBN 978-1441436856.
- [53] Eric S. Raymond. *La cathédrale et le bazar*, 1998.
- [54] Concertation nationale sur l'avenir de l'enseignement supérieur au sénégal, April 2013.
- [55] Réseau pour l'enseignement supérieure et la recherche du sénégal. <http://snrer.edu.sn>.
- [56] Xmpp software foundation. <http://xmpp.org/about-xmpp/xsf/>.
- [57] K. Keahey, M. Tsugawa, A. Matsunaga, and J. Fortes. *Sky Computing*, 2009.
- [58] Dan C. Marinescu. Cloud computing- theory and practice. *Elsevier Sciences Technology Books*, 2013. ISBN : 978-0-12-404627-6.
- [59] Barrie Sosinsky. Cloud computing bible. *Wiley Publishing, inc.*, 2011. ISBN : 978-0-470-90356-8.
- [60] 2013. Online.
- [61] Ying Wang. Research on multi-level security of shibboleth authentication on mechanism. *Third International Symposium on Information Processing*, 2010. 978-0-7695-4261-4/10.
- [62] L.N. Anaraki and A. Heidari. Knowledge management process in digital age : Proposing a model for implementing e-learning through digital libraries. *Application of Information and Communication Technologies (AICT), 2011 5th International Conference*, pages 1–5, oct 2011.
- [63] Yi-Wen Luo. The effect of using javascript and html to implement an e-learning web sites as a remedial teaching aid on english underachievers at universities of science and technology. *Computers and Industrial Engineering (CIE), 2010 40th International Conference*, pages 1–7, jul 2008.
- [64] Darryl and Y. Bandung. Designing evaluation system for virtual class service in limited network capacity using servqual methodology. *Cloud Computing and Social Networking (ICCCSN), 2012 International Conference*, pages 1–5, apr 2012.
- [65] E. Cahyadi, H.A. Santoso, Y. Bandung, and A.Z.R. Langi. Augmented reality virtual class box for digital learning services. *Cloud Computing and Social Networking (ICCCSN), 2012 International Conference*, pages 1–4, apr 2012.
- [66] J. Blichar and P. Podhradsky. Liferay as a collaboration and communication platform for m-learning subsystem. *ELMAR, 2012 Proceedings*, pages 175–178, sep 2012.
- [67] Zeidan, Adham AND Lehmann, Armin AND Trick, Ulrich. Webrtc enabled multimedia conferencing and collaboration solution. *WTC 2014; World Telecommunications Congress 2014*, pages 1–6, jun 2014.
- [68] M. Phankokkruad and P. Jaturawat. An evaluation of technical study and performance for real-time face detection using web realtime communication. *Computer, Communications, and Control Technology (I4CT), 2015 International Conference*, pages 162–166, apr 2015.
- [69] W3c editor'sdraft. <http://w3c.github.io/mediacapturemain/getusermedia>.
- [70] Branislav Sredojev, Dragan Samardzija, and Dragan Posarac. Webrtc technology overview and signaling solution design and implementation. *Information and Communication Technology, electronics and Microelectronics (MIPRO), 2015 38th International Convention*, pages 1006–1009, may 2015.
- [71] Easyrtc. <https://easyrtc.com/>.

- [72] Puranik, D.G. AND Feiock, D.C. AND Hill, J.H. Real-time monitoring using ajax and websockets. *Engineering of Computer Based Systems (ECBS), 2013 20th IEEE International Conference and Workshops*, pages 110–118, apr 2013.
- [73] Microscope optique. [http://www.hellopro.fr/documentation/pdf\\_prod/7/7/0/3622\\_2f756d40b5a08f39a0bde](http://www.hellopro.fr/documentation/pdf_prod/7/7/0/3622_2f756d40b5a08f39a0bde)
- [74] C.C. Aydin and G. Tirkes. Open source learning management systems in elearning and moodle. *Education Engineering (EDUCON), 2010 IEEE*, pages 593–600, apr 2010.
- [75] M. Blas. Teresa and serrano-fernández, the role of new technologies in the learning process : Moodle as a teaching tool in physics. *Ana Computers and Education*, 3 :35–44, mar 2009.
- [76] William Rice. *Moodle 2.0 E-Learning Course Development*. Packt, 2011.
- [77] A. Al-Ajlan and H. Zedan. Why moodle. *Future Trends of Distributed Computing Systems, 2008. FTDCS '08. 12th IEEE International Workshop*, pages 58–64, oct 2008.
- [78] Y. H. Elawady and a. S. Talba. A general framework for remote laboratory access : A standarization point of view. *IEEE International Symposium Signal Process. Inf. Technol. ISSPIT 2010*, pages 485–490, 2011.
- [79] R. Hashemian and J. Riddley. Fpga e-lab, a technique to remote access a laboratory to design and test. *Proc. - MSE 2007 2007 IEEE Int. Conf. Microelectron. Syst. Educ. Educ. Syst. Des. Glob. Econ. A Secur. World*, pages 139–140, 2007.
- [80] Saros. <http://www.saros-project.org/specoverview>.
- [81] H. Cola, C. ; Valean. On multi-user web conference using webrtc. *System Theory, Control and Computing (ICSTCC), 2014 18th International Conference*, pages 430–433, 2014.
- [82] Virtualbox. <https://www.virtualbox.org/>.
- [83] Rdp. <https://www.virtualbox.org/manual/ch07.html>.
- [84] M. Mirzakhani, H. Ashrafzadeh, and a. Ashrafzadeh. The virtual university : Advantages and disadvantages. *Distance Learn. Educ. (ICDLE), 2010 4th Int. Conf.*, page 32–36, 2010.
- [85] Chaczko, Z. ; Aslanzadeh, S. ; Klempous, R. Development of software with cloud computing in 3tz collaborative team environment. *Broadband and Biomedical Communications (IB2Com), 2011 6th International Conference on*, pages 318–323, 2011.
- [86] Reis, R. ; Escudeiro, P. ; Fonseca, B. High-level model for educational collaborative virtual environments development. *Advanced Learning Technologies (ICALT), 2012 IEEE 12th International Conference on*, pages 356–358, 2012.
- [87] Bani-Salameh, H. ; Jeffery, C. ; Al-Gharaibeh, J. A social collaborative virtual environment for software development. *Collaborative Technologies and Systems (CTS), 2010 International Symposium on*, pages 46–55, 2010.
- [88] Leow Fui Theng ; Neo Mai. Students' perceptions of a constructivist classroom : A collaborative learning approach. *Educational Media (ICEM), 2013 IEEE 63rd Annual Conference International Council for*, pages 1–11, 2013.
- [89] Alanis-Funes, G.J. ; Neri, L. ; Noguez, J. Virtual collaborative space to support active learning. *Frontiers in Education Conference (FIE)*, pages 12–15, 2011.
- [90] Jxbrowser. <https://www.teamdev.com/jxbrowser>.