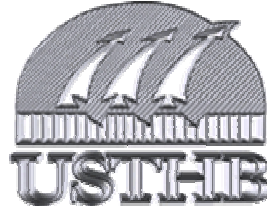


**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE**  
**SCIENTIFIQUE**  
**UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE HOUARI BOUMEDIENE**



*Faculté d'Electronique et Informatique*

**MEMOIRE**  
**Présenté pour l'obtention du Diplôme de Magister**  
**En : Informatique**

**Spécialité : Programmation et Systèmes**

**Par : Djeddou Djamel**

**Thème**

**Adaptation et Transformation de Présentations  
Multimédias**

**Soutenu le 11/04/2007, devant le Jury composé de :**

<b>Mr - Badache Nadjib</b>	<b>Professeur</b>	<b>Président</b>
<b>Mr - Belkhir Abdelkader</b>	<b>Maître de conférence</b>	<b>Directeur de Thèse</b>
<b>Mme - Alimazighi Zaia</b>	<b>Maître de conférence</b>	<b>Examinatrice</b>
<b>Mr - Challal Yacine</b>	<b>Docteur</b>	<b>Examineur</b>

## ***Remerciements***

Mes plus vifs remerciements vont également à mon directeur de thèse, le Dr BELKHIR ABDELKADER, qui m'a proposé le sujet, de m'avoir donné la possibilité de mener à bien ma thèse, et je suis infiniment reconnaissant pour son soutien permanent, ses précieux conseils et sa grande disponibilité et aide tout au long de ce projet.

Un chaleureux remerciement au Professeur BADACHE NADJIB qui m'a fait l'honneur d'être président du jury.

Mes remerciements s'adressent également aux membres du jury :  
Madame ALIMAZIGHI ZAIA, Maître de conférence.  
Monsieur CHALLAL YACINE, Docteur

Ma gratitude s'adresse aussi au Directeur de la Police Judiciaire, le Commissaire Divisionnaire de Police Mr ISSOULI MOHAMED pour leur professionnalisme et compréhension.

Je souhaite inclure dans mes remerciements les collègues, le Commissaire Principal de Police SOUALHI CHAABANE, le Commissaire de Police HASSANI FAYCAL, l'Officier de Police BENRABAH DJAMEL.

J'associe à ces remerciements mes collègues de recherche Mme KHORSI, Mr AGHILES.

Je clos enfin ces remerciements en dédiant cette thèse de magister :  
A l'âme de ma mère, que dieu le Tout-puissant accorde sa sainte miséricorde et l'accueillir en son vaste paradis.

A mon père, que dieu le tout-puissant le protège et le garde.

A sœurs et mes frères.

A mademoiselle Bourdjenne Djazia.

# Table des Matières

## Chapitre 1 : *avancements technologiques*

1- Introduction .....	1
2 Les documents multimédias .....	2
2.1- Les dimensions des documents multimédias .....	2
2.1.1- La dimension temporelle .....	2
2.1.2- La dimension spatiale .....	3
2.1.3- La dimension logique .....	4
2.1.4- La dimension hypermédia .....	4
3- Le World Wide Web Consortium: W3C .....	5
4- Hyper Texte Markup Language .....	5
5- eXtensible Markup Language .....	5
6- Synchronized Multimedia Integration Language .....	6
6.1- Pourquoi SMIL .....	6
6.2- Structure d'un fichier SMIL .....	7
6.3- Nouveauté dans SMIL 2.0 .....	10
6.4- Intégration de SMIL dans une page web .....	14
6.5- La modularisation et le profilage dans SMIL 2.0 .....	14
6.6- Nouveauté dans SMIL 2.1 .....	15
6.7- Conclusion – SMIL .....	16
7- La compression de documents.....	17
7.1– Introduction .....	17
7.2- Principes de base .....	17
7.2.1- La vidéo .....	17
7.2.2- Problème de la vidéo .....	17
7.3- Approches de compression .....	18
7.3.1- Le but de la compression .....	18
7.3.2- Comment compresser les images .....	18
7.3.3- La compression MPEG .....	18
7.3.3.1- Normes et produits de MPEG .....	19

7.4- Conclusion .....	19
-----------------------	----

## **Chapitre 2: Problématique et solutions**

1-Environnements Hétérogènes .....	20
1.1 - Evolution des moyens d'accès .....	20
2- Problématique .....	20
3- La solution : La transformation et l'adaptation .....	20
4- Différentes Approches d'adaptation .....	21
4.1- Approche spécifique. ....	21
4.2- Approche Universelle .....	22
5- Description de Format du document .....	22
6- Contraintes matérielles et Préférences utilisateurs .....	23
6.1- Description du Contexte .....	23
6.1.1- Description du réseau .....	24
6.1.2- Description de l'appareil : .....	24
6.1.2.1- Description .....	24
6.1.2.2- Caractéristiques des machines hétérogènes .....	24
6.1.3- Description du profil utilisateur (client) : .....	25
6.2- Resource Description Framework (RDF) .....	26
6.3- Capacités Combinées/Profils de Préférences (CC/PP) .....	27
6.3.1- Introduction.....	27
6.3.2- Objectifs du CC/PP.....	27
6.3.3- La structure de CC/PP.....	27
6.3.3.1- Les composants.....	28
6.3.3.2- Les attributs .....	28
6.3.3.3- Les valeurs par défaut .....	30
6.3.3.4- L'écrasement d'une valeur par défaut en XML.....	32
6.3.3.5- Conclusion de CC/PP.....	33
7- La négociation : .....	34
7.1- Définition : .....	34
7.2- Classification des stratégies de négociation.....	35
7.3- Les protocoles de négociation .....	36

7.3.1-	Le protocole de transfert hypertexte.....	36
7.3.2-	Le Protocole d'Initiation de Session .....	36
7.3.2.1-	Définition .....	36
7.3.2.2-	Fonctionnalités de SIP: .....	37
7.3.2.3-	Architecture de SIP : .....	37
7.4-	But de la négociation .....	38
8-	La transformation.....	39
8.1-	Définition .....	39
8.2-	Présentation des feuilles de style.....	39
8.3-	Définition d'un style.....	40
8.4-	CSS 2.0 .....	40
8.5-	Le langage XSL .....	41
8.5.1-	Définition .....	41
8.5.2-	Les différents langages de XSL .....	41
8.5.2.1-	Le XSLT.....	41
8.5.2.2-	Structure générale d'une feuille de style XSLT.....	42
8.5.2.3-	Le XSL/FO .....	43
8.6-	Transformation des objets média.....	43
8.7-	Transformation au niveau de la structure logique .....	43
8.8-	Adaptation au niveau sémantique.....	44
8.9-	Logiciels de transformation.....	44
8.9.1-	Ixslt : .....	44
8.9.2-	VXT : .....	44
9-	Solutions proposées.....	45
9.1.-	Adaptation <i>a priori</i> d'un document SMIL.....	45
9.2-	Adaptation <i>a posteriori</i> d'un document SMIL.....	45
9.3-	NAC (Negotiation Adaptation Core) .....	46
9.4 -	STAMP (Synchronized Templates for Adaptable Multimedia .....	47
10-	Conclusion .....	49

## Chapitre 3 Contribution

1-	problématique.....	50
2-	La solution : L'adaptation et la transformation : .....	50

3- La description de contexte .....	51
3.1- Notion de profil.....	51
3.2 Les métaphores .....	51
3.3- Représentation du profil par la métaphore.....	51
3.4- Formulation du Profil.....	55
4- La négociation .....	55
5- La transformation .....	56
5.1- L'adaptation au Matériel .....	57
5.1.1- Module de Conversion de pixel en pourcentage .....	57
5.1.2- L'adaptation à l'affichage .....	59
5.1.3- L'adaptation à la couleur et à la résolution.....	59
5.1.4- L'adaptation à la mémoire et à la vitesse d'exécution.....	59
5.2- L'adaptation au réseau.....	61
5.2.1- Adaptation au débit de connexion : .....	61
5.3- L'adaptation au logiciel.....	62
5.4- L'adaptation aux préférences de l'utilisateur .....	62
5.4.1- L'adaptation pour un handicapé visuel : malvoyant .....	63
5.4.2- L'adaptation pour un handicapé auditif.....	64
5.4.3- L'adaptation à la langue .....	64
5.5- L'adaptation au contenu .....	64
5.5.1- L'adaptation au niveau Spatial .....	64
5.5.2- L'adaptation au niveau temporel .....	65
5.5.3- L'adaptation au niveau sémantique et structurel.....	65
5.5.4- L'adaptation au niveau de la structure logique .....	66
5.5.5- Adaptation au niveau de l'hypermédia .....	66
5.5.6- Inconvénients .....	66
5.5.7- La représentation quantitative et qualitative d'un document multimédia .....	67
5.5.7.1- L'adaptation Quantitative.....	68
5.5.7.2- L'adaptation Qualitative .....	68
5.5.7.3- Application de l'adaptation quantitative et qualitative .....	68
5.5.8- L'adaptation globale.....	71
6 - Conclusion .....	71
7- L'architecture proposée.....	71
7.1 - Principe de l'architecture ATPM.....	72

7.2- Objectifs de l'architecture ATPM.....	72
7.3 - Schéma général de l'architecture ATPM.....	74
7.4 - Rôles des différentes entités de l'architecture ATPM.....	76
7.4.1 - L'application Cliente.....	76
7.4.2 - Le Serveur de Contenu : SC.....	76
7.4.3 - Le Serveur SIP : .....	76
7.4.4 - Le serveur de Profils et de Négociation et de Génération : SPNG.....	77
7.4.4.1- Algorithme de fonctionnement de SPNG.....	78
7.5 - Le Serveur de Transformation et de Validation : STV.....	79
7.5-1 Optimisation dans l'envoi du Contenu multimédia.....	80
8- Algorithme de parcours de l'architecture ATPM.....	80
9- Exemple : d'application.....	83
10 - Conclusion .....	88
- Conclusion et perspectives.....	89

## Table des Figures

<b>Figure 1 : Dimension temporelle d'un document multimédia.....</b>	<b>2</b>
<b>Figure 2 : Dimension spatiale d'un document multimédia.....</b>	<b>3</b>
<b>Figure 3 : Dimension logique d'un document multimédia .....</b>	<b>4</b>
<b>Figure 4 : Dimension hypermédia d'un document multimédia.....</b>	<b>4</b>
<b>Figure5 : Les caractéristiques matérielles et logicielles de différents terminaux. ....</b>	<b>25</b>
<b>Figure 6 : Les composants d'un profil CC/PP .....</b>	<b>28</b>
<b>Figure 7 : Exemple de profil CC/PP en XML .....</b>	<b>29</b>
<b>Figure 8 : L'écrasement d'une valeur par défaut en XML.....</b>	<b>33</b>
<b>Figure 9 : Vue globale sur le système d'adaptation et de négociation du contenu. ....</b>	<b>34</b>
<b>Figure 10: Architecture en couche du protocole SIP (Pile du protocole SIP). ....</b>	<b>37</b>
<b>Figure 11: Syntaxe d'une règle de Style. ....</b>	<b>40</b>
<b>Figure 12: Création d'une structure logique arborescente à partir du document XML.....</b>	<b>42</b>
<b>Figure 13: Unicité de l'ombre d'une personne .....</b>	<b>52</b>
<b>Figure 14 La taille de l'ombre face à la source lumineuse. ....</b>	<b>52</b>
<b>Figure 15: L'ombre d'une personne dans l'obscurcissement.....</b>	<b>53</b>

<b>Figure 16: l'ombre change de forme aux changements apportés à la personne. ....</b>	<b>53</b>
<b>Figure 17: l'ombre suit la personne dans les mouvements. ....</b>	<b>53</b>
<b>Figure 18: Transformation concerne les objets média.....</b>	<b>65</b>
<b>Figure 19: Transformation au niveau de la structure logique.....</b>	<b>66</b>
<b>Figure 20: Adaptation quantitative (l'objet vidéo enlevé). ....</b>	<b>68</b>
<b>Figure 21 : Description des zones de présentation.....</b>	<b>69</b>
<b>Figure 22 : L'adaptation quantitative.....</b>	<b>69</b>
<b>Figure23 : L'adaptation qualitative.....</b>	<b>69</b>
<b>Figure 24 : Application de l'adaptation quantitative et qualitative.....</b>	<b>70</b>
<b>Figure 25: L'architecture Adaptation et Transformation d'une Présentation Multimédia.....</b>	<b>74</b>
<b>Figure 26 : Interface client .....</b>	<b>83</b>
<b>Figure 27 : Authentification du Client et Profil.....</b>	<b>83</b>
<b>Figure 28 : Configuration des paramètres de connexion .....</b>	<b>84</b>
<b>Figure 29 : Sélection de la présentation .....</b>	<b>84</b>
<b>Figure 30 : Gestion des Profils .....</b>	<b>85</b>
<b>Figure 31 : Sélection de la présentation multimédia.....</b>	<b>85</b>
<b>Figure 32 : Extraction des caractéristiques de la présentation sélectionnée .....</b>	<b>86</b>
<b>Figure 33 : Transformation de pixel en pourcentage .....</b>	<b>86</b>
<b>Figure 34 : Visualisation de présentation multimédia transformée .....</b>	<b>87</b>

## ***Résumé***

L'adaptation et la transformation d'une présentation multimédia sont devenus aujourd'hui la solution idoine pour surmonter les obstacles issus des hétérogénéités de différentes entités de connexions et de communications constituant les environnements hétérogènes actuels, afin de satisfaire les préférences des utilisateurs et leurs diversités.

L'adaptation qui a été choisie comme solution, désigne le processus de transformation d'un contenu de son état initial vers un état final, afin de satisfaire l'ensemble de contraintes. Il existe différentes solutions d'adaptation : L'adaptation spécifique et l'adaptation à la demande.

Notre architecture d'adaptation et de transformation de présentations multimédias (**ATPM**), contribue à la transformation de contenu pour pallier les limitations des utilisateurs et les contraintes de leur environnement. La modélisation du profil utilisateur (en utilisant la métaphore de l'ombre) se fait par le standard CC/PP. On utilisera le processus de négociation modélisé par un automate déterministe à états finis [KHO 06]. La transformation consiste à adapter et à transformer le contenu quantitativement et qualitativement.

## ***Mots Clés***

***Environnement hétérogène, multimédia, profil, négociation, adaptation, transformation, métaphore, Web, SMIL, XSLT.***

# Introduction

Dans un passé proche, pour jouer de la vidéo, utiliser du son ou du texte au sein d'une même présentation, le recours à des logiciels spécialisés (coûteux en temps et en ressources) s'imposait, générant des rendus sous forme d'un seul fichier de format vidéo et, ne laissant aucune place à la modification, du fait qu'un détail aussi infime que la taille de police du texte nécessitait la reprise de l'opération à son début [SUS 00].

Aujourd'hui, l'essor croissant du multimédia permet le développement de nombreuses applications alliant les technologies de l'écrit, à de l'image et au son [SUS 00]. Les récentes avancées technologiques permettent aux documents multimédia de s'illustrer sur de nombreuses plates-formes, notamment les téléphones portables, les PDA, les ordinateurs de bureau ...etc. Cette diversification des utilisations et des supports a induit un besoin d'adaptation des documents à leur contexte d'exécution.

Par ailleurs, l'hétérogénéité des moyens et des appareils d'accès s'est accompagnée d'une évolution importante du contenu de l'information disponible sur le réseau d'une part, et des préférences de l'utilisateur d'autre part. La diversité de l'évolution technologique, sa prolifération dans les réseaux et les machines hétérogènes et leurs caractéristiques, la complexité des données et leur présentation, ainsi que les préférences des utilisateurs constituent autant de facteurs à l'origine de la complexité et l'hétérogénéité de l'environnement actuel [LEM 04 a].

Les moyens d'accès ont également subsidiairement changé depuis les débuts du Web, qu'il s'agisse de terminaux ou de moyens de communication. Les réseaux sans fil se sont développés et intégrés à l'Internet, par le biais des réseaux de téléphonie mobile et des réseaux de transmission de données tels que WiFi, GPRS, UMTS, ...etc, constituant l'infrastructure basique de communication du Web [LAY 05 b]. Ses principales caractéristiques sont : la bande passante, le taux d'erreur, gigue...etc.

Dans le même sillage, des appareils de plus en plus variés, en termes d'énergie, de capacité d'affichage et de mémoire et la vitesse d'exécution, tels que les Téléphones cellulaires, les assistants personnels, les téléviseurs, les consoles de jeux et les appareils embarqués se sont connectés sur ces réseaux [LAY 05 b].

D'autres facteurs endogènes à l'origine de la complexité de l'environnement, notamment les contraintes subjectives et physiques, liées soit aux préférences des utilisateurs ou à leurs propres handicaps (malvoyant, aveugle, sourd...), augmentent les difficultés de prise en charge de cet aspect [LAY 05 b].

Un document multimédia est composé d'objets de différents types [SEB 04]: texte, son, image, vidéo...etc. Lors de l'élaboration d'une présentation multimédia, le modèle du document doit permettre la représentation de toutes les relations qui peuvent exister entre les objets multimédia [ROI 98]. Ces relations multimédias peuvent porter sur la description de l'organisation logique du document, la présentation spatiale des différents objets multimédia, leurs synchronisations temporelles ainsi que l'interconnexion entre ses différents éléments : hypermédia.

Toutes ces entités (réseau, matériel, préférences de l'utilisateur, système, et contenu multimédia) et sa diversification constituent l'environnement hétérogène [LEM 04 a]. Cette diversification a entraîné le problème de l'inadaptabilité et de l'incompatibilité entre ces entités. Ce problème peut, se résoudre par la constitution d'un système universel « homogène » capable de passer n'importe quelle présentation sur n'importe quel réseau ou machine. Ce système s'articule principalement sur l'adaptation et la transformation d'une présentation multimédia par rapport aux contraintes imposées par les entités de l'environnement hétérogène.

A cet effet, l'importance est donnée beaucoup plus au contenu multimédia et sa présentation. Avec les nouvelles technologies d'aujourd'hui telle que XML [MIC 00], qui représente le contenu de l'information brute du web indépendamment de sa présentation et de sa structure, on peut accéder et modifier le contenu sans le reconstituer. De ce principe l'idée est née, le travail se base sur les dérivés de XML, tels que SMIL [W3C 00] et XSLT [AMA 02] : SMIL pour la synchronisation et XSLT pour la transformation et la présentation.

Dans l'adaptation on trouve deux solutions, l'une, consiste à développer des contenus spécifiquement ciblés vers un type d'appareil, éventuellement dans un format propre à cet appareil [LAY 04]. L'autre, favorise des formats universels. On produit chaque document une seule fois et on le stocke sous une forme unique [LAY 04]. Ensuite, au moment où le document

doit être envoyé à un client, on engendre une autre forme qui tient compte des spécificités de l'environnement à cet instant.

Notre contribution, s'articule autour de la deuxième solution, car la première son inconvénient majeur, c'est la fragmentation, chose qu'est à l'encontre des objectifs du Web.

Notre mémoire est organisé en trois Chapitres. Premièrement nous exposons les avancements technologiques de XML pour la description de différents types de documents, y compris le document de présentation multimédia SMIL. Dans un second temps, nous définissons la problématique et la description de différents composants constituant les environnements hétérogènes. Dans ce même chapitre, nous présentons quelques solutions d'adaptation qui ont été déjà proposées, et on termine, par notre contribution dans une approche d'Adaptation et de Transformation d'une Présentation Multimédia « **ATPM** ».

## **Plan de la thèse**

### **Chapitre 1   avancement technologique**

Dans ce chapitre, on commence par une introduction posant les problèmes du progrès technologique du Web, puis on présente le document multimédia et ses différentes dimensions. Les langages favorisés dans cette thèse sont : le langage de balisage XML [MIC 00] pour la description de données, et pour la synchronisation des objets constituant une présentation multimédia, on a favorisé le langage SMIL [W3C 00] qui s'appuie sur le XML. En fin, nous présentons la compression et le codage pour le transfert d'un document multimédia,

### **Chapitre 2 : Problématique et solutions**

Dans ce chapitre, nous exposons la problématique des environnements hétérogènes et les obstacles affrontés par des utilisateurs et ses préférences et ses exigences par rapport aux moyens de communication et de transmission utilisés. La solution adoptée, c'est *La transformation et l'adaptation* qui désigne le processus de transformation d'un contenu de son état initial vers un état final afin de satisfaire l'ensemble de contraintes. Différentes solutions d'adaptation sont présentées dans cette thèse, notamment l'adaptation spécifique [LAY 04] et l'adaptation dynamique « universelle ou à la demande » [LAY 04]. Nous poursuivons la solution par des travaux qui ont été déjà réalisés pour l'adaptation d'un document multimédia telles que:

l'Adaptation à priori [SEB 04] et l'adaptation à posteriori [SEB 04] d'un document SMIL : NAC (Negotiation Adaptation Core) [LAY 04] et STAMP (Synchronized Templates for Adaptable Multimedia Presentations) [BIL 05].

### **Chapitre 3 : Contribution**

Dans ce chapitre, nous posons notre propre architecture d'Adaptation et Transformation de Présentations Multimédia (ATPM). L'architecture s'articule premièrement sur la notion de profil qui métaphoriquement est représenté par l'ombre. Deuxièmement on présente le module de conversion de pixel en pourcentage ; et troisièmement l'adaptation quantitative et qualitative. L'architecture **ATPM** consiste à définir des entités qui coopèrent entre elles pour transmettre des contenus qui respectent les caractéristiques et les limites de l'environnement. Afin d'universaliser l'accès au contenu indépendamment de la variété des clients ainsi que leurs moyens, types et endroit d'accès.

### **Conclusion**

L'adaptation et la transformation d'une présentation multimédia sont la solution idoine pour surmonter les obstacles issus des environnements hétérogènes.

Notre architecture d'adaptation et de transformation des présentations multimédias (**ATPM**), contribue à la transformation de contenu pour pallier les limitations des utilisateurs et les contraintes de leur environnement.

ATPM commence le processus de transformation par la formulation du profil de l'utilisateur (description de contexte), qu'est métaphoriquement modulé par l'ombre de l'utilisateur, et leur formalisation s'effectue dans le cadre de CC/PP [CC/PP 04]. A base de ce profil et au contenu demandé, on entame le processus de négociation, qu'est aussi modélisée par un automate déterministe à états finis [KHO 06]. L'objectif de la négociation est de trouver un compromis pour arranger l'utilisateur, et de générer les règles de la transformation. Cette dernière consiste à adapter et à transformer le contenu qu'est métaphoriquement évalué quantitativement et qualitativement, par le filtrage et l'enlèvement des objets médias qui ne seront pas exploités pour ce profil, et d'améliorer la qualité des objets restants par l'application du module de conversion de pixel en pourcentage...etc.

# Chapitre 1

## **Chapitre 1 :      *avancements technologiques***

### **1- Introduction**

Les progrès technologiques récents ont permis l'apparition d'une grande variété de nouveaux moyens permettant à un utilisateur d'accéder et d'utiliser l'information multimédia qui l'intéresse en tout lieu et à tout moment. L'accès au contenu ne s'effectue plus de la même façon ni par les mêmes appareils qu'il y a quelques années.

Les appareils d'accès à l'information ont subi une véritable révolution. En effet, les utilisateurs peuvent accéder au contenu à travers des appareils très divers : ordinateurs portables, assistants personnels, téléviseurs, téléphones cellulaires, etc. Le nombre des utilisateurs de ces nouveaux appareils continue sa croissance exponentielle. Les moyens d'accès au contenu ont également évolué, avec de nouveaux réseaux tels que les réseaux sans fil WiFi, GPRS, UMTS, etc. Ces réseaux se sont développés et se sont intégrés à l'Internet. L'utilisation du World Wide Web ne ressemble donc plus à ce qu'elle était à l'origine, où l'utilisateur accédait à l'information depuis son ordinateur personnel et à travers le réseau filaire [LAY 05 b].

L'hétérogénéité des moyens et des appareils d'accès s'est accompagnée d'une évolution importante du côté du contenu de l'information disponible sur le réseau [LEM 04 a]. Aujourd'hui, on trouve une multitude de formats complexes avec de nouvelles fonctionnalités, telles que la vidéo interactive, les animations 3D et le dessin vectoriel. Ces formats s'appuient sur de nouveaux modèles qui intègrent le monde multimédia (la structure logique, l'espace, le temps et la navigation).

Cette évolution des contenus ainsi que des appareils et des moyens d'accès a fait apparaître des environnements hétérogènes où les utilisateurs ont des facultés différentes pour accéder au même contenu à travers différents réseaux de communication. D'autre part, le contenu peut être trop complexe pour qu'un terminal ayant des capacités limitées puisse le traiter et le présenter correctement. Face à cette réalité il est nécessaire de trouver des mécanismes qui permettent l'accès et l'utilisation de l'information sous une forme qui correspond aux contraintes imposées par l'environnement.

## 2 Les documents multimédias

Un document multimédia est composé d'objets de différents types : texte, son, image, vidéo...Etc. Ces objets sont nommés objets multimédia. Lors de l'élaboration d'une présentation multimédia, le modèle du document doit permettre la représentation de toutes les relations qui peuvent exister entre les objets multimédia [SEB 04].

Ces relations multimédia peuvent porter sur la description de l'organisation logique du document, la présentation spatiale des différents objets multimédia, leurs synchronisations temporelles ainsi que l'interconnexion entre ses différents éléments :hypermédia.

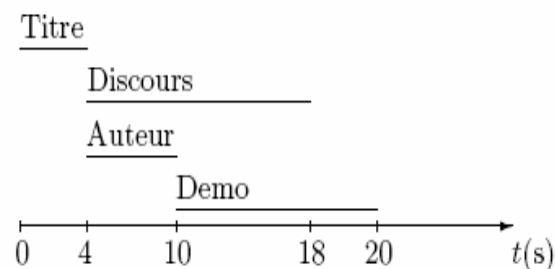
### 2.1- Les dimensions des documents multimédias

Un document multimédia est constitué 4 dimensions [Roi 98] :

#### 2.1.1- La dimension temporelle

La dimension temporelle représente la caractéristique majeure du document multimédia, dont ses objets multimédia sont reliés temporellement entre eux selon un ordre bien défini. Cet ordre peut être modélisé grâce à un scénario temporel [ROI 98].

Exemple



**Figure 1 : Dimension temporelle d'un document multimédia**

Dans cet exemple, on a quatre objets multimédia : le Titre, le Discours, l'Auteur et la Demo. L'organisation dans le temps de ces éléments est la suivante : l'objet Titre se joue pendant 4 secondes de 0 à 4 secondes, puis l'objet Discours et l'objet auteur se jouent en parallèle, dont, le premier commence de 4 à 18 secondes, et le deuxième commence de 4 à 10 secondes, puis l'objet Demo se joue de 10 à 20 secondes en parallèle aussi avec l'objet discours.

Trois types d'objets multimédia peuvent néanmoins se distinguer :

**Objets élastiques.** Ce sont des objets multimédia dont la présentation ne dépend pas du temps, comme par exemple du texte ou des images fixes. On peut leur attribuer une durée de présentation explicite ou implicite selon le contexte d'un scénario donné.

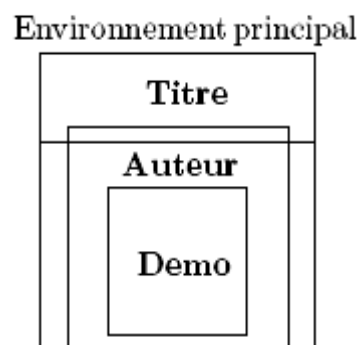
**Objets rigides.** Ils correspondent à des objets multimédia dont la présentation dépend du temps et dont la valeur de durée est connue *à priori*. Ce sont par exemple des vidéos ou bien des flux sonores.

**Objets à durée non-déterminée.** Ce type d'objets se distingue par le fait qu'ils n'ont pas de durée connue *à priori*. Ce sont par exemple des flots audios ou vidéos continus auxquels on accède à travers le réseau.

### 2.1.2- La dimension spatiale

La dimension spatiale d'un document multimédia permet à certains objets multimédia comme du texte, des vidéos ou des images de disposer d'un espace géométrique sur l'écran délimité par une certaine zone.

#### Exemple



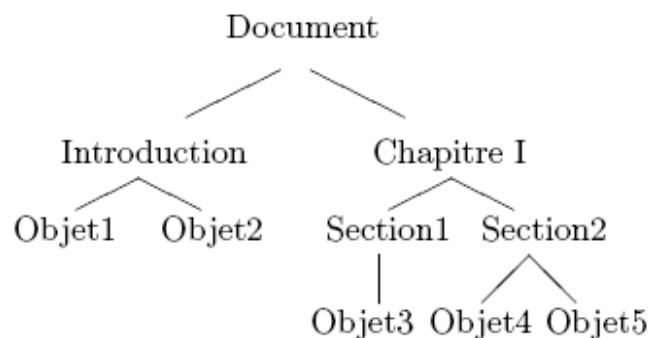
**Figure 2 : Dimension spatiale d'un document multimédia**

Cet exemple comporte un environnement principal contenant des objets multimédia (Titre, Auteur et Demo) qui jouent dans des zones d'affichage prenant des formes rectangulaires.

### 2.1.3- La dimension logique

Dans un document multimédia, il est possible de distinguer des parties du document qui sont sémantiquement liées et qui peuvent être regroupées par le biais de relation logique. Regrouper les éléments sous une même entité revient à définir une structure logique du document c'est-à-dire une structure hiérarchique des éléments multimédia.

#### Exemple :



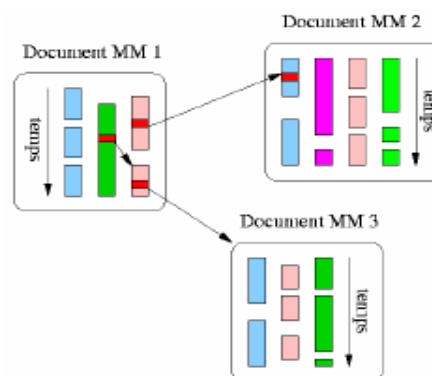
**Figure 3 : Dimension logique d'un document multimédia**

Cet exemple exprime la racine de la structure logique de l'ensemble du document multimédia qui peut se décomposer en plusieurs parties (Introduction, Chapitre . . .) et sous parties (Section . . .). Où chaque partie ou sous partie peut contenir un nombre fini d'objets multimédia.

### 2.1.4- La dimension hypermédia

La dimension hypermédia d'un document permet de décrire des liens qui peuvent exister entre certains éléments, voire même entre différents documents.

#### Exemple :



**Figure : 4 Dimension hypermédia d'un document multimédia**

Dans cet exemple, le Document MM1 comporte trois liens hypermédias, l'un entre les objets qui le constituent, et deux autres pointent sur deux documents différents MM2 et MM3

### **3- Le World Wide Web Consortium: W3C**

W3C [W3C 06], est un organisme officiellement chargé de la normalisation de tout ce qui concerne l'évolution technique du Web de puis 1994, et particulièrement des évolutions du langage HTML. Cette évolution technique se concrétise, par le développement des nouveaux standards pour assurer l'interopérabilité. Le W3C a notamment publié les spécifications des diverses versions du standard HTML [DAN 00], mais aussi de la norme XML [MIC 00], ou encore du format d'image PNG. Sur le long terme, le W3C veut assurer l'universalité de l'accès au web, garantir une utilisation efficace des ressources du web par les utilisateurs, et guider le développement du web en considérant les nouveaux enjeux sociaux, commerciaux et juridiques induits par cette technologie.

### **4- Hyper Texte Markup Language**

HTML [DAN 00], est un langage de balisage avec des liens hypertextes, dérivé de SGML [HER 95] en 1990. HTML est un langage très simple, mais il a des limites tel que le balisage d'éléments sémantiques n'est pas possible, ce qui limite les performances des moteurs de recherche.

### **5- eXtensible Markup Language**

XML [MIC 00], signifie langage à balises extensibles, c'est un compromis la complexité de SGML et la légèreté de HTML. Il a été lancé en 1996 par le groupe du World Wide Web Consortium (W3C) avant sa recommandation en date du 10 février 1998.

XML est un Métalangage, c'est un format de description des données et non de leur représentation, à base d'une DTD [W3C 02] ou XMLSchema [THO 02] qui définissent la structure de données et des feuilles de style pour leur présentation tel que le XSL [QUI 07].

## **6- Synchronized Multimedia Integration Language**

**SMIL** [W3C 00], est un langage de balisage prononcé smile permettant de synchroniser des fichiers de nature différente pour en faire des objets multimédias aptes à être intégrés dans une page Web. Ce langage s'appuie avant tout sur le XML, il est même entièrement basé sur celui-ci. SMIL est une recommandation officielle, en date du 15 juin 1998 du W3C [W3C 06], il est défini comme " une application XML qui décrit le format et le comportement temporel de présentations multimédia, et la façon dont les objets multimédia exploitent les liens hypertextes". Le langage SMIL permet de définir les différentes caractéristiques de gestion d'un fichier multimédia (pause, redémarrage, choix d'une langue, et le choix d'un niveau de qualité en fonction de la bande passante disponible...etc). Comme il précise le moment de visualisation et les durées de ces objets multimédias. Les formats de fichiers supportés dans SMIL sont les suivants : .swf, .rm, .wav, .aif, .mov, .mp3, .gif, .jpg, .rt, .rm, .avi, .mov, .asf, .viv, .mpeg.

En août 2001, le W3C a recommandé la version SMIL 2.0. Ce dernier apporte de nouvelles fonctionnalités comme l'animation, le contrôle de contenu, la mise en page, les hyperliens, les objets média et les effets de transition. Actuellement c'est la version SMIL 2.1, qui a été publié le 13 décembre 2005, en recommandation du W3C. L'objectif principal de cette version est la réalisation de présentations multimédias sur des terminaux mobiles.

### **6.1- Pourquoi SMIL ?**

Les percées technologiques récentes en matière de multimédia ont permis d'accroître les possibilités d'interaction entre l'homme et la machine. La manipulation numérique de l'image, du son et de la vidéo a changé la nature d'un grand nombre d'applications, et a donné naissance à d'autres applications plus évoluées telles que la visioconférence et le téléenseignement. Ainsi, dans un passé pas si lointain, pour jouer de la vidéo, du son, du texte au sein d'une même présentation, il aurait fallu utiliser des logiciels spécialisés (coûteux en temps et en ressources) qui généraient des rendus sous forme d'un seul fichier de format vidéo. Cela ne laissait aucune place à la modification, car retoucher un détail aussi infime que la taille de police du texte nécessitait de refaire toute l'opération [SUS 00].

De ce problème est né le besoin pour un standard permettant de présenter et de synchroniser différentes sources multimédia indépendamment les unes des autres et sous un format textuel pour l'échange de document, en sollicitant l'intégration d'objet multimédia.

SMIL [W3C 00] verra le jour pour y remédier. Sa force réside dans la facilité d'écriture et d'utilisation, héritée de XML. A ce titre SMIL profite de tous les outils agissant sur les documents XML comme les feuilles de style XSLT.

## **6.2- Structure d'un fichier SMIL**

Un fichier SMIL commence par `<smil>` et finit par `</smil>`. Entre ces balises, on trouve, comme dans un fichier HTML, une partie `<head>` et une partie `<body>`.

### **La partie `<head>`**

La partie `<head>` (en-tête) contient des informations relatives à la description du contenu, telles que les métadonnées (balises `<meta>`), les informations sur le "découpage" de la zone SMIL et la disposition des différents éléments. Chaque élément peut être identifié (attribut `id="nom_identificateur"`).

#### **`<meta>`**

Les balises `<meta>` servent à donner des informations à propos de la présentation SMIL, telles que l'auteur, le titre ou un résumé. Elles ont comme attributs possibles `name` et `content`. Voici un exemple :

Exemples de balises meta

```
<meta name="author" content="D. DJAMEL" />
```

```
<meta name="title" content="Exemple SMIL" />
```

```
<meta name="copyright" content="U.S.T.H.B" />
```

#### **`<layout>`**

Le découpage se fait par la balise `<layout>`. Celle-ci contient la balise `<root-layout>` et la ou les balise(s) `<region>`, correspondant aux diverses zones qui composent le document.

#### **`<root-layout>`**

C'est le `<root-layout>` qui définit les dimensions de la zone SMIL principale, et la couleur de fond.

## <region>

Viennent ensuite les zones, <region>, chacune décrivant les dimensions et la position d'une zone où sera inséré l'objet multimédia. Les zones peuvent se superposer par l'attribut **z-index**. La balise <region> peut avoir les attributs suivants: id, background-color (couleur de fond), title (un titre pour la zone), top, left, width, height, z-index, fit.

On attribue un nom à la zone (attribut id). Ceci permettra d'y faire référence dans la partie <body>, pour situer les éléments. Les attributs top et left, qui indiquent la position de la zone par rapport au coin supérieur gauche de la zone principale du fichier, ne sont pas obligatoires et leur valeur est de 0 par défaut. La valeur peut être exprimée de façon relative ou absolue, c'est-à-dire en pourcentage ou en pixels.

### Attribut fit

L'attribut **fit** détermine comment vont être représentés les éléments dont les dimensions sont différentes de celles spécifiées pour la région où ils se trouvent.

Plusieurs cas de figure se présentent. L'attribut **fit** peut avoir l'une des valeurs suivantes :

*Hidden ,Fill ,Meet ,Slice , Scroll.*

### La partie <body>

La partie <body>, contient les éléments multimédia eux-mêmes, indique la localisation des fichiers composant le document SMIL, et le comportement (la durée et le moment d'apparition) de chacun.

### Les éléments multimédia

Les éléments multimédia que l'on peut trouver sont de 6 types différents:

<audio />, <video /> , <img /> , <text />, <animation />, <textstream />

En cas de doute sur le type de média, on peut utiliser la balise <ref />. Pour chaque élément, on indique le type, au moyen d'une des balises listées ci-dessus, puis, à l'intérieur de cette balise, la source du fichier (attribut **src**, qui donne l'URL du fichier source) et la région où il doit apparaître (attribut **region**). Celle-ci a été définie dans la partie <head>.

En général, les éléments sont inclus dans les balises <par> (pour parallèle) ou <seq> (pour séquence), qui servent à gérer la synchronisation des éléments comme on peut combiner les deux balises <par> et <seq> . Cependant, ils peuvent aussi être insérés directement dans la

balise <body>. Ils apparaissent alors les uns après les autres, dans l'ordre où ils sont dans le code. A noter: la balise <seq> a le même effet.

### Exemple

```
<smil>
  <head>
    <meta name="author" content="DJEDDOU DJAMEL " />
    <meta name="title" content="exemple SMIL" />
    <meta name="copyright" content="U.S.T.H.B" />

    <!-- la section layout contient les informations de positionnements de la présentation -->
    <layout>

      <!-- root-layout definit la fenêtre de présentation (taille, couleur de fond) -->
      <root-layout id="principale"
        height="200" width="320"
        background-color="black"
        skip-content="true" />

      <!-- les régions sont disposées dans le root-layout. Elles sont les zones d'affichage. -->
      <region id="region1"
        height="120" width="80"
        background-color="silver"
        left="5" top="6" z-index="0"
        fit="hidden" />
      <region id="region2"
        height="120" width="80"
        background-color="silver"
        left="152" top="40" z-index="0"
        fit="hidden"
        skip-content="true" />
    </layout>
  </head>
  <!-- Body definit le contenu -->
  <body>
    <par endsync="last" repeat="1">
      
      
    </par>
  </body>
</smil>
```

```

    <audio id="audio0"
      src="zik.mp3"
      dur="24.0s"
      repeat="1"
      fill="remove" />
  </par>
  <audio src="zik.mp3" />
  
  

</body>
</smil>

```

Dans l'exemple ci-dessus, le déroulement de la présentation est comme suit :

- Au départ, les trois objets photo1, photo2 et audio0 se jouent en parallèle, dont la photo1 se joue dans la région1 pendant 15 secondes, la photo2 dans la région2 pendant 15 secondes et l'audio0 se joue pendant 24 secondes.
- Puis l'objet audio "zik.mp3" se joue jusqu'à la fin
- Après la photo4 s'affiche pendant 10 secondes dans la région1
- Puis et dans la même région "région1" la photo5 s'affiche pendant 8 secondes.

### Attributs pour les balises <par> et <seq>

Elles peuvent avoir les attributs suivants :

id ,title , abstract , author , copyright , dur , repeat , region , begin , end , endsync (seulement pour <par>)

### 6.3-Nouveauté dans SMIL 2.0 :

En plus de <par> et <seq>, on peut inclure des éléments médias dans la balise <excl> (pour "exclusive"). Cette balise permet de créer un groupe exclusif dans lequel les éléments sont diffusés un seul à la fois. À la différence de la balise <seq>, qui permet également de faire jouer les éléments un seul à la fois, mais seulement dans un ordre donné, la balise <excl> donne plus de contrôle sur l'ordre dans lequel les éléments seront diffusés.

Autres balises :

### La balise <switch>

<switch> est utilisé pour proposer différents choix à l'utilisateur, par exemple, un texte ou un enregistrement vidéo en différentes langues. La balise <switch> englobe alors les différents éléments proposés :

**system-language** : Choix de la langue.

**system-bitrate** : choix du type de connexion et le contrôle du taux de transmission.

**system-captions** : Ignorer si l'utilisateur ne veut pas de sous-titre.

**system-overdub-or-caption** : Sous-titrage ou non.

**system-required** : Reconnaissance ou non d'une extension.

**system-screen-size** : Spécifie la taille minimale de l'écran.

**system-screen-depth** : Spécifie la résolution minimale.

### Exemple :

```
<smil>
  <head>
    <!-- non developpe -->
  </head>

  <body>
    <par>
      <switch>
        <audio src="french.mp3" system-language="fr" />
        <audio src="spanish.mp3" system-language="es" />
        <audio src="english.mp3" />
      </switch>
      <video src="alien.avi" />
    </par>
  </body>
</smil>
```

### Les liens

#### L'élément <a>

Comme dans HTML, l'élément <a> permet de gérer les liens.

**L'attribut show** contrôle le comportement du document contenant le lien lorsque le lien est

suivi. Il peut avoir l'une des valeurs suivantes : **replace**, **new**, **pause**, et ce pour commencer, suspendre, reprendre, commencer dans une nouvelle fenêtre ou terminer une présentation.

**L'attribut href**, comme dans HTML, indique l'URL de l'élément vers lequel pointe le lien. Cet attribut est obligatoire. L'élément `<a>` doit également avoir un attribut `title`, qui va décrire précisément le fichier de destination. Les liens peuvent se faire sur les éléments suivants : `animation`, `audio`, `img`, `par`, `ref`, `seq`, `switch`, `text`, `textstream`, `video`.

### L'élément `<anchor>`

L'élément `<anchor>` a pour effet de rendre cliquable une partie d'élément seulement. Si l'on souhaite qu'une partie "spatiale" d'un élément visuel soit cliquable, on utilise l'attribut **coords**, qui va définir le rectangle cliquable de la façon suivante:

Les deux premières valeurs vont déterminer le coin supérieur gauche du rectangle, et les deux dernières vont déterminer le coin inférieur droit. Les coordonnées sont relatives au coin supérieur gauche de l'élément. Chaque valeur est séparée de la suivante par une virgule. On peut fournir les coordonnées en pourcentage.

Il est également possible grâce à l'élément `<anchor>` de séparer un objet en sous-parties temporelles. On va pour cela utiliser les attributs **begin** et **end**. Il ne faut pas oublier de nommer chaque sous-partie au moyen de l'attribut `id`.

À noter: dans SMIL 2.0, l'élément `<anchor>` est remplacé par l'élément `<area>`.

### Synchronisation des éléments

Une des propriétés de SMIL est de permettre de gérer la synchronisation des éléments. On peut ainsi préciser quand commencent et se terminent les éléments, combien de temps dure chacun d'eux, etc.

Les éléments de type image ou texte n'ayant pas de durée propre, on spécifie celle-ci par l'attribut `dur`.

#### Exemple :

```

```

On peut définir précisément le moment où l'élément apparaît (commence) et celui où il se termine. Par exemple, dans le cas d'une diffusion séquentielle, il peut y avoir un intervalle entre chaque élément. Les attributs `begin` et `end` sont utilisés pour spécifier un moment de début et/ou fin particulier.

En général, la valeur de l'attribut `begin` est relative à l'élément parent, c'est-à-dire que l'élément débute `x` secondes (0 s si l'attribut n'est pas précisé) après que l'élément parent ait débuté.

**Exemple :**

```
<par>
  
  
</par>
```

La deuxième photo apparaîtra 4 secondes après le début de l'élément `<par>`, c'est-à-dire également 4 secondes après l'apparition de la photo1.

Dans le cas où les éléments multimédia sont inclus dans une balise `<seq>`, l'attribut `begin` sera relatif à l'élément multimédia précédent.

**Exemple :**

```
<smil>
  <head>
    <layout>
      <root-layout height="480" width="640"
        background-color="white" />
      <region id="region1" height="480" width="640"
        background-color="silver" />
    </layout>
  </head>

  <body>
    <seq>
      
      
      
    </seq>
  </body>
</smil>
```

Dans l'exemple précédent, la photo1 sera affichée durant 5 secondes, puis, après 4 secondes, la photo2 apparaîtra durant 5 secondes également. De nouveau, la photo3 apparaîtra pour 5 secondes après 3 secondes de la fin de la photo 2.

Il est possible de faire débiter un élément multimédia relativement à un autre élément multimédia, en exprimant l'attribut begin de la façon suivante : begin="id(id\_element)(2s)".

On peut également remplacer le nombre de secondes par begin ou end, ce qui signifie que l'élément débute lorsque l'autre élément spécifié débute ou se termine.

De la même manière, il est possible de faire terminer un élément relativement à un autre élément, en exprimant l'attribut end comme ci-dessus. Comme on peut faire jouer plusieurs fois de suite le même élément, grâce à l'attribut repeat.

#### **6.4- Intégration de SMIL dans une page web**

Pour intégrer un fichier SMIL dans une page web, de façon à ce que le fichier soit lu directement dans le navigateur en faisant appel au module externe nécessaire et non pas dans une fenêtre d'un player comme RealPlayer, il faut utiliser les balises <object> ou <embed>. Ces deux balises ont le même but, mais les navigateurs Netscape et Internet Explorer reconnaissent chacun l'une d'entre elles seulement.

#### **6.5- La modularisation et le profilage dans SMIL 2.0**

Un *module* [AYA 01] est une collection d'éléments, d'attributs et de valeurs d'attribut sémantiquement liés à XML, qui représente une unité de fonctionnalité. Les modules sont définis dans des ensembles cohérents. Cette cohérence est exprimée dans le fait que les éléments de ces modules sont associés au même espace de nommage.

*La modularisation* [AYA 01] est une approche selon laquelle une fonctionnalité de balisage est spécifiée comme un ensemble de modules qui contiennent des éléments, des attributs et des valeurs d'attributs sémantiquement liés à XML. *Le profilage* est la création d'un langage basé sur XML par la combinaison de ces modules, dans le but de fournir la fonctionnalité requise par une application particulière

Un *profil de langage* est une combinaison de ces modules. Les modules sont *atomiques*, ils ne peuvent être divisés en sous-ensembles quand ils sont inclus dans un profil de langage. En outre, une spécification de module peut inclure un ensemble conditions préalables d'intégration, auxquelles les profils de langage contenant le module doivent satisfaire.

Le profilage introduit la capacité de façonner un langage basé sur XML pour des besoins spécifiques, exemple : optimiser la présentation et l'interaction selon les capacités du client. Le profilage ajoute aussi la capacité d'intégrer des fonctionnalités à partir d'autres langages balisés, libérant ainsi l'auteur du langage de la spécification de ces fonctionnalités.

La modularisation et le profilage utilisent les propriétés d'extensibilité de XML, ainsi que des technologies apparentées comme les espaces de nommage XML et les schémas XML.

Généralement, il y a un profil de langage principal qui inclut pratiquement tous les modules associés à un même espace de nommage. Par exemple, le profil de langage SMIL 2.0 utilise la plupart des modules SMIL 2.0.

### **6.6- Nouveauté dans SMIL 2.1**

SMIL 2.1 [BUL 05] dynamise les présentations multimédias pour les terminaux mobiles. Il apporte, par rapport à SMIL 2.0, la notion de profil pour les supports mobiles, ainsi qu'un profil étendu pour ces mêmes supports. Un profil SMIL 2.1 est par conséquent constitué d'un ensemble de modules. Donc SMIL 2.1 a été conçu sous la forme d'une série de modules composables fondée sur l'infrastructure fournie par la recommandation Modularisation de XHTML [ALT 01]. SMIL 2.1 définit trois nouveaux profils : le profil langage SMIL 2.1 [LAY 05 a], le profil Mobile SMIL 2.1 [DIC 05 a] et le profil étendu SMIL 2.1 [DIC 05 b].

SMIL 2.1 enrichit la boîte à outils du créateur de présentations multimédia, en offrant notamment des transactions plein écran, des capacités de mise en page visuelle améliorées telles que la juxtaposition d'image d'arrière-plan, et des performance audio améliorées.

SMIL 2.1 permet aux auteurs de prédéfinir des ensembles de paramètres pouvant être référencés plusieurs fois dans une présentation et d'une présentation à l'autre. Cette réutilisation facilite la création et la mise à jour des contenus multimédia et permet de réduire la taille des présentations SMIL. Ce qui est un facteur supplémentaire améliorant les performances sur les terminaux mobiles

### 6.7- Conclusion – SMIL –

SMIL est un langage de synchronisation basé sur XML pour décrire le format et le comportement temporel et spatiale d'une présentation multimédia, dont il offre la possibilité d'acheminer un document multimédia vers un document adaptable selon les caractéristiques du matériel et les préférences des utilisateurs. Il utilise la notion de région soit pour la synchronisation (parallèle <par> ou série <seq>), soit pour le mode d'affichage, soit pour ajuster un document selon la région, qui peut avoir une dimension différente de celle spécifiée pour la région où il se trouve.

SMIL contient différents attributs pour la présentation, notamment la balise **switch** pour répondre aux préférences de l'utilisateur tel que le choix de la langue, le type de connexion...etc. L'élément **FIT** pour le mode d'affichage, dont on peut présenter une image d'une façon relative ou absolue, c'est-à-dire en pourcentage ou en pixels, afin de satisfaire certaines contraintes matérielles. SMIL utilise aussi la notion de lien (**show**) pour commencer, suspendre, reprendre la présentation, commencer dans une nouvelle fenêtre ou terminer une présentation.

SMIL résout certains problèmes de l'adaptation tels que la synchronisation et l'ajustement d'un document pour l'affichage, aussi certaines préférences de l'utilisateur telles que la langue, la bande passante..etc.

## **7- La compression de documents**

### **7.1- Introduction**

Les données multimédia (graphiques, audio et vidéo) non compressées réclament une capacité de stockage et une bande passante considérables. En dépit des progrès réalisés en capacité de stockage, en puissance de traitement des processeurs et en performance des systèmes de communications numériques, la demande en capacité de stockage et en bande passante de transmission reste toujours supérieure au potentiel des technologies disponibles. La demande récente en applications web utilisant de manière intensive des données multimédia a généré un besoin d'algorithmes de compression des signaux et des images plus performants et plus souples d'utilisation. Une application émergente est l'utilisation de données multimédia avec la téléphonie portable, qu'elle soit de 2<sup>ème</sup> génération (GSM) avec le protocole WAP ou de 3<sup>ème</sup> génération (UMTS).

### **7.2- Principes de base :**

#### **7.2.1- La vidéo**

La vidéo est une succession d'images animées. Le principe fondamental de la vidéo est que l'oeil humain a la faculté de retenir pendant un certain temps (de l'ordre du dixième de seconde) toute image imprimée sur la rétine. Il suffit donc de faire défiler un nombre suffisant d'images par seconde, pour que l'oeil ne se rende pas compte qu'il s'agit d'images distinctes.

Il existe deux grandes familles de systèmes vidéo : les systèmes vidéo analogiques [LEC 00] et les systèmes vidéo numériques [LEC 00].

#### **7.2.2- Problème de la vidéo**

Une image d'une vidéo non compressée occupe une taille d'environ 1 Mo. Afin d'obtenir une vidéo paraissant fluide, il est nécessaire d'avoir une fréquence d'au moins 25 ou 30 images par seconde, ce qui produit un flux de données d'environ 30 Mo/s, soit plus de 1.5 Go par minute. Il est évident que ce type de débit est peu compatible avec les espaces de

stockage des ordinateurs personnels, ni même avec les connexions réseau de particuliers ou de petites ou moyennes entreprises.

Pour pallier cette difficulté, il est possible de recourir à des algorithmes permettant de réduire significativement les flux de données en compressant / décompressant les données vidéos. On appelle ces algorithmes *CoDec* (pour *COmpression / DECompression*).

### **7.3- Approches de compression**

#### **7.3- 1 Le but de la compression**

Les méthodes de compression [DJA 06] et de codage réduisent le nombre de bits par pixel à stocker ou à transmettre, en exploitant la redondance informationnelle dans l'image.

Les principaux critères d'évaluation de toute méthode de compression sont :

- La qualité de reconstitution de l'image
- Le taux de compression;
- La rapidité du codeur et décodeur (codec).

#### **7.3.2- Comment compresser les images ?** [DJA 06]

Pour compresser des images, il faut éliminer les redondances présentes dans le signal, celles-ci sont de 3 types cités respectivement : la redondance *spatiale* (corrélation entre pixels voisins), La redondance *spectrale* (réduire de manière importante le volume d'informations à traiter) et La redondance *temporelle* (le peu de changements entre les séquences d'images).

#### **7.3.3- La compression MPEG**

Le MPEG [LEC 00] (Motion Picture Experts Group) est d'abord un organisme de standardisation américain du secteur de la vidéo et de l'audio dont le sigle a été repris pour désigner dans le langage courant les formats de compression et de codage qu'il a créés en 1988.

Ces formats visent à transmettre des contenus vidéo (avec leur partie audio) par tout canal de transmission numérique (Internet, diffusion par satellite, etc.), ou à rendre plus compact les contenus vidéo numérisés de façon qu'ils tiennent sur des supports de type DVD.

Ils sont structurés en trois parties ou " couches ", l'une traitant les signaux vidéo, la deuxième les signaux audio et le troisième, nommé système, ayant pour fonction principale de resynchroniser les deux flux.

### **7.3.3.1- Normes et produits de MPEG**

MPEG a plusieurs normes : La première norme c'est MPEG 1 [LEC 00] introduit en 1991, pour traiter les images vidéo sur des supports acceptant un débit de 1,5 Mbit/s. La deuxième norme c'est MPEG 2 [LEC 00] introduite en 1994, pour la compression de la vidéo de qualité télévision à un débit de 4 à 6 Mbit/s. La troisième norme c'est MPEG 3 mais cette norme a été rapidement abandonnée. La quatrième norme c'est MPEG 4 [LEC 00] en 1998 pour la diffusion de la vidéo au plus bas débit possible de 4,8 à 64 Kbit/s. La cinquième norme c'est MPEG 7 [LEC 00] introduite en 2000, pour faciliter l'indexation et la recherche d'un document multimédia. La sixième norme c'est MPEG 21 [LEC 00] introduite en 2001, elle ajoute une couche de sécurité à la norme MPEG 7.

### **7.4- Conclusion**

Le codage est très important pour le transfert d'un document multimédia, notamment pour optimiser la bande passante d'un coté et de répondre d'un autre coté aux caractéristiques, décrivant le matériel final ou les préférences de l'utilisateur. Dont on trouve Mpeg-4 qui utilise la notion d'objet hiérarchique « par Niveau », et selon le profil on adapte le niveau du document. Exemple à partir des images fixes (arrière-plan), des objets vidéo (objets en mouvement sans arrière-plan) et des objets audio (la voix associée à l'objet en mouvement) constituant l'arbre logique du document multimédia on peut enlever l'objet qui ne correspond pas au profil.

Cette approche d'objet permet alors de faire de très nombreuses opérations sur un document multimédia tel que l'ajout, la suppression ou le déplacement d'un objet et la transformation géométrique.

# Chapitre 2

## **Chapitre 2: Problématique et solutions**

### **1- Environnements Hétérogènes**

#### **1.1- Evolution des moyens d'accès (avancement technologique):**

Les moyens d'accès ont également beaucoup changé depuis les débuts du Web, qu'il s'agisse des terminaux ou des moyens de communication. Les réseaux sans fil se sont développés et se sont intégrés à l'Internet. Ce sont à la fois des réseaux de téléphonie mobile et des réseaux de transmission de données du type WiFi (IEEE 802.11). Tous font maintenant partie de l'infrastructure de communication du Web [LAY 05 b].

Dans le même temps, des appareils de plus en plus variés se sont connectés sur ces réseaux : Téléphones cellulaires, assistants personnels, téléviseurs, consoles de jeux, appareils embarqués et qui sont divers en terme d'énergie de capacité d'affichage, et de mémoires, etc.

Tous sont des terminaux multimédias, dans le sens où ils gèrent plusieurs média, et notamment des média continus. De plus, la plupart accèdent à des réseaux mobiles. On est donc en plein dans le monde du multimédia mobile [LAY 04].

### **2- Problématique**

Le problème majeur des environnements hétérogènes est la présence de nombreuses contraintes relatives aux différentes entités du système. Ces entités désignent l'évolution des moyens d'accès et des contenus avec leurs complexités, ainsi que les préférences des utilisateurs. Face à cette réalité il est nécessaire de trouver des mécanismes qui permettent l'accès et l'utilisation de l'information sous une forme qui correspond aux contraintes imposées par cet environnement hétérogène et qui respecte les principes de base de l'existant.

### **3- La solution : L'adaptation et la transformation**

C'est la constitution d'un système qui prendra en charge, d'une manière automatique l'adaptation et la transformation d'un document multimédia tout en respectant l'ensemble des différentes entités constituant l'environnement hétérogène.

On débutera par l'adaptation de contenu qui est définie généralement comme le processus qui transforme un contenu de son état initial vers un état final afin de satisfaire l'ensemble de contraintes. Ce contenu final peut provenir du même contenu source ou faire usage d'autres ressources de contenu. Satisfaire une contrainte donnée, revient à changer l'état du contenu d'origine afin que l'usage du contenu, dans son état final, ne pose pas de problème en la présence de la contrainte. Si un contenu donné ne peut pas satisfaire une contrainte de l'environnement, cela peut causer un problème d'utilisation du contenu dans cet environnement. Ce problème peut être d'ordre sémantique, c'est-à-dire le contenu reçu par le client ne correspond pas à la sémantique du contenu original tel qu'il a été créé par l'auteur ; ou d'ordre présentation, c'est-à-dire le contenu visualisé par le client ne correspond pas à la présentation du contenu original.

#### **4- Différentes Approches d'adaptation :**

##### **4.1- Approche spécifique.**

L'approche pragmatique [LAY 04] de cette adaptation consiste à développer des contenus spécifiquement ciblés vers un type d'appareil, éventuellement dans un format propre à cet appareil. C'est typiquement l'approche retenue par le WAP, qui utilise un format particulier, et se base sur le WML, qu'est un langage de marquage restreint permettant d'organiser l'information d'une façon qui se prête bien à l'accès sur le petit écran d'un téléphone et à travers un réseau à faible bande passante.

##### **Les avantages**

- Elle est facile à maître en œuvre
- Bien adaptée à un type de matériel

##### **Les inconvénients**

- Ce modèle va à l'encontre des objectifs du Web, qui se veut un espace d'information universel, accessible à tous.
- Le risque de cette approche est la fragmentation, avec un Web pour les téléphones mobiles de première génération, qui ne contient que de l'information produite pour ces appareils, mais aussi un Web pour les téléviseurs, un Web pour les assistants personnels, etc.

- fortes restrictions pour les utilisateurs de chaque type d'appareil : ils n'ont plus accès qu'à une partie de l'information.
- coûts élevés pour les producteurs de contenu qui veulent toucher un large public : ils doivent créer plusieurs fois la même information pour les différents types d'appareils.

#### **4.2- Approche Universelle**

Cette approche fait le contraire, elle favorise des formats universels. On produit chaque document une seule fois et on le stocke sous une forme unique [LAY 04]. Ensuite, au moment où le document doit être envoyé à un client, on engendre une autre forme qui tient compte des spécificités de l'environnement à cet instant. L'environnement comprend l'appareil client et les moyens de communication utilisés, le client lui même (ses capacités, ses handicaps et ses préférences), et voire aussi les tâches qu'il peut effectuer, d'où la nécessité d'une adaptation dynamique qui prend en charge toutes les contraintes relatives aux différentes entités de l'environnement hétérogène.

Cette approche d'adaptation nécessite :

- A- Un mécanisme de description de format du contenu indépendamment de leur présentation
- B- Un mécanisme de description du contexte (Réseau, Matériel, Préférences de l'utilisateur, Navigateur, Logiciel)
- C- Un mécanisme de négociation
- D- Un mécanisme de transformation et de présentation selon les résultats donnés par les différents mécanismes précédents.

#### **5- Description de Format du document**

L'objectif majeur de la description de format du contenu est de séparer ce dernier le plus possible de sa présentation à l'appareil de restitution. XML [MIC 00] va dans ce sens, il sépare le contenu et la structure logique d'un document de leur présentation. XML n'apporte aucune information sur la manière dont son contenu sera présenté (il est nécessaire de

l'associer à une feuille de style). Ce qui garantit à ses utilisateurs l'indépendance de leurs documents de toute technologie ; il ouvre la porte du Web aux documents structurés (BD, protocoles, applications distribuées,...). Bien que la structure du document est définie dans une DTD ou XMLSchema et les présentations sont décrites par différentes feuilles de Styles(CSS, XSL) qui peuvent être appliquées à la même ressource XML.

La manipulation d'un document multimédia nécessite l'utilisation d'un certain nombre de langages dédiés à cet effet, tel que le langage SMIL [W3C 00], SVG [EIS 02], XHTML [DAN 00]. SMIL a été reconnu comme un langage de synchronisation basé sur XML pour décrire le format et le comportement temporel et spatiale d'une présentation multimédia. SMIL est utilisé dans notre approche, pour acheminer un document multimédia vers un document adaptable selon les caractéristiques du matériel et les préférences de l'utilisateur. L'avantage de SMIL, se résume dans l'utilisation de la notion de région, soit pour la synchronisation (parallèle <par> ou série <seq>), soit pour le mode d'affichage, aussi il offre la possibilité d'ajuster un document selon la région, et comment ce document va être présenté.

## **6- Contraintes matérielles et Préférences utilisateurs**

### **6.1- Description du Contexte**

Les documents multimédias qui combinent des médias numériques discrets comme le texte ou l'image ou des médias continus comme le son et la vidéo doivent être adaptés à un environnement mobile qui contient des appareils très hétérogènes, en termes de capacités mémoire, de calcul ou d'affichage tels que des téléphones cellulaires, des assistants personnels, des téléviseurs, des consoles de jeu ou des appareils embarqués.

Pour faire une telle adaptation on a besoin de décrire trois facteurs pour réaliser l'adaptabilité des documents:

- (1) au contexte réseau (débit, qualité de la transmission et taux d'erreur).
- (2) aux différents types de terminaux (du PC, téléphone mobile ou PDA),
- (3) à différentes classes d'utilisateurs (du néophyte au spécialiste)

### **6.1.1- Description du réseau :**

Les moyens d'accès ont beaucoup changés depuis les débuts du Web, qu'il s'agisse des terminaux ou des moyens de communication. Les réseaux sans fil sont développés et se sont intégrés dans l'Internet, ce sont à la fois des réseaux de téléphonie mobile et des réseaux de transmission de données. On doit avoir une description des caractéristiques du réseau pour pouvoir adapter le document selon cette description, par exemple le débit de transfert, type de réseau ...etc.

**Exemple :** Réseau fixe (Ethernet, câble, ADSL, etc.) ou GSM, GPRS, UMTS, ... etc.

### **6.1.2- Description de l'appareil :**

#### **6.1.2.1- Description**

Des appareils de plus en plus variés se sont connectés sur ces réseaux : téléphones cellulaires, assistants personnels, téléviseurs, consoles de jeux, appareils embarqués. Tous sont des terminaux multimédias, dans le sens où ils gèrent plusieurs médias, et notamment des médias continus. De plus, la plupart accèdent à des réseaux mobiles. Ces équipements sont très hétérogènes, en terme de capacité de mémoire, de calcul ou de capacité d'affichage. Cette diversité rend la connaissance des caractéristiques de l'appareil plus qu'indispensable.

**Exemple :** Téléphone cellulaire, Marque Sagem X2, Ecran couleur, Affichage 100\*50 pixel.

#### **6.1.2.2- Caractéristiques des machines hétérogènes**

Une des opérations importantes qui interviennent dans le processus d'adaptation est la description des contraintes et les capacités des périphériques hétérogènes [LAY 99]. Les capacités visées sont d'ordre matériel mais aussi logiciel.

Il faut noter que du point de vue de l'utilisateur, même dans le cas d'un accès à l'information depuis un PC, il pourrait avoir un certain nombre de préférences pour la visualisation de documents multimédia. Par exemple, il peut être amené à vouloir une fenêtre d'affichage plus réduite pour accéder aux documents multimédia. Dans ce cas, on parle de

préférences étant donné que la contrainte ne vient pas des possibilités de la machine mais des souhaits de l'utilisateur. La description des caractéristiques des machines est ainsi étendue pour décrire les préférences.

		ScreenPhone	GSM	PC	Portable PC	PDA
<b>Outils de connexion</b>		Modem : 56 Kbps	9600 bps	Modem > 28,8 Kbps	9600 bps up to 2 Mbps(UMTS)	>9600 bps
<b>Ecran</b>	<b>Affichage</b>	640*480	~90x53	800x600 up to 1600x1200	800x00 up to 600x1200	160x160 up to 240x32
	<b>Couleur</b>	8	Oui	8,16,32 b	8,16,32 b	4 grey scale
	<b>Noir et Blanc</b>	Oui	Oui	Oui	Oui	Oui
<b>Vidéo</b>		Oui	Oui	Oui	Oui	Oui
<b>Image</b>		Gif and Gpeg	Bmp	Oui	Oui	Oui
<b>Audio</b>		.AU and .wav	Oui	Oui	Oui	Oui
<b>Protocole utilisé</b>		HTTP	HDTP	HTTP	HTTP	HTTP
<b>Langage utilise</b>		HTML 3.2 P.java applets	HDML SMIL Tiny	HTML Java, SMIL	HTML Java, SMIL	SMIL Basic

**Figure5 : Les caractéristiques matérielles et logicielles de différents terminaux.**

### **6.1.3- Description du profil utilisateur (client) :**

On doit avoir des connaissances du profil utilisateur, ces préférences reflètent son choix de voir ou de ne pas voir une telle ou telle chose, ses capacités physiques (malvoyant, aveugle, sourd...).

#### **Exemple :**

- Je souhaite avoir la couleur
- Je préfère le format GIF
- Je ne comprends que le français
- Mes centres d'intérêt sont le sport, les recherches scientifiques et les informations d'actualité.

Le profil de l'utilisateur est composé de quatre parties [LOP 05] : le profil statique de l'utilisateur, les caractéristiques dynamiques de l'utilisateur, la description du type de connexion et les caractéristiques de l'environnement. Ces parties sont regroupées en deux, les caractéristiques personnelles de l'utilisateur et les caractéristiques de son contexte.

Les caractéristiques personnelles désignent les aspects statiques et dynamiques. Les données statiques de l'utilisateur sont permanentes (son nom, sa langue maternelle...etc), et les caractéristiques dynamiques représentent les informations pouvant changer (centres d'intérêts de l'utilisateur, ses connaissances, ou encore sa localisation lorsqu'il utilise un dispositif mobile).

Les caractéristiques de son contexte désignent le contexte de l'utilisateur et la connexion et l'environnement. Le contexte de l'utilisateur représente un état à un instant donné, exprimé en termes de caractéristiques liées à la tâche effectuée, la connexion et l'environnement. La définition de la tâche inclut la description de l'objectif de l'utilisateur, son rôle, le temps disponible pour exécuter sa tâche...etc. Les caractéristiques de la connexion sont définies par la description du dispositif (ses capacités) et du réseau (bande passante). Les caractéristiques de l'environnement concernent la localisation et le moment de la connexion.

Pour faire ces descriptions (profil utilisateur et matériel) on doit adopter un langage unique pour permettre la communication et la négociation sur le format du document qui doit être envoyé. Parmi ces langages, on cite le langage CC/PP [CC/PP 04] qui constitue un standard de ce type.

## **6.2- Resource Description Framework (RDF)**

Le RDF [KAL 99] (Resource Description Framework) est un modèle de métadonnées conçu par le World Wide Web Consortium (W3C) et recommandé le 22 février 1999. Il repose sur le support XML pour la description de ressources et la représentation des données, sous forme de graphes étiquetés (également appelés "diagrammes arcs et noeuds").

RDF conçu pour spécifier des propriétés exploitables par des programmes, décrire le contenu du WEB et les capacités des composants qui interviennent dans l'accès ; en plus les préférences associées à un utilisateur.

RDF permet de rendre plus « intelligente » l'information nécessaire à ces moteurs de recherche et, plus généralement, nécessaire à tout outil informatique analysant de façon automatique des pages web. Pour ce faire, RDF propose d'associer à toute ressource du Web un ensemble de descripteurs qui caractérisent au mieux cette ressource : on parle alors de Métadonnées (en anglais, *metadata*) [HER 07].

### **6.3- Capacités Combinées/Profils de Préférences (CC/PP)**

#### **6.3.1- Introduction**

CC/PP [CC/PP 04] est une recommandation du Consortium World Wide Web (W3C), annoncée le 15 Janvier 2004 comme environnement permettant d'exprimer les capacités d'un terminal, logiciel, les préférences utilisateurs et le contexte d'utilisation en utilisant RDF [KAL 99] (Resource Description Framework). C'est un modèle pour l'utilisation des métadonnées et de description de l'ensemble d'informations nécessaires sur le Web.

Un profil CC/PP est une description des capacités des appareils, des caractéristiques logiciel et des préférences de l'utilisateur, qui peut s'utiliser pour guider l'adaptation d'un contenu présenté à cet appareil. Cela signifie qu'un profil CC/PP contient un certain nombre de noms d'attributs CC/PP et de valeurs associées, lesquels sont utilisés par un serveur afin de déterminer la forme la plus appropriée de la ressource à remettre à un client

#### **6.3.2- Objectifs du CC/PP**

L'objectif principal du CC/PP consiste en la mise en œuvre d'un format standard de description de capacités « ressources » et de permettre à tout terminal de communiquer de façon détaillée ses capacités à un serveur donné.

#### **6.3.3- La structure de CC/PP**

De façon générale, un profil CC/PP est construit selon une hiérarchie à deux niveaux :

Un profil a au moins un *composant*.

Chaque composant a au moins un *attribut*.

Les branches initiales de l'arbre du profil CC/PP décrivent les composants principaux du client.

Exemples de composants principaux :

- La plate-forme matérielle sur laquelle le logiciel s'exécute.
- La plate-forme logicielle sur laquelle toutes les applications sont hébergées.
- Une application individuelle, comme un navigateur.
- Une représentation graphique simple d'un arbre CC/PP construit à partir de ces trois

composants (MaterielFinal, LogicielFinal et NavigateurFinal) pourrait être :

[*exemple:LeProfil*]

+--*ccpp:component*-->[*exemple:MaterielFinal*]

+--*ccpp:component*-->[*exemple:LogicielFinal*]

+--*ccpp:component*-->[*exemple:NavigateurFinal*]

**Figure 6: Les composants d'un profil CC/PP**

### **6.3.3.1- Les composants**

Un profil CC/PP contient un ou plusieurs *composants*, qui contiennent à leur tour un ou plusieurs attributs. Chaque composant est représenté par une ressource de type *ccpp:Component* (ou par certaines sous-classes RDF de celle-ci) et relié à la ressource du profil client par une propriété *ccpp:Component*

### **6.3.3.2- Les attributs**

Les profils CC/PP sont établis en utilisant RDF [ORA 99]. Le modèle de données RDF représente les attributs CC/PP par des propriétés nommées, reliant une ressource sujet à une ressource objet associée ou une valeur littérale RDF. Afin de décrire les capacités et les

préférences du client, le client en question est une ressource dont les caractéristiques sont décrites par des arêtes de graphe étiquetées, depuis celui-ci vers des valeurs objets correspondantes. Les étiquettes d'arête de graphe identifient la caractéristique du client (l'attribut CC/PP) en cours de description et les valeurs objets correspondantes sont les valeurs des caractéristiques.

[Ressource composant du client] --nomAttribut--> (valeur-attribut)

Le code XML correspondant pourrait ressembler à ceci :

```
<?xml version="1.0"?>

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ccpp="http://www.w3.org/2002/11/08-ccpp-schema#"
  xmlns:exemple="http://www.example.com/schema#">

  <rdf:Description rdf:about="http://www.example.com/profil#LeProfil">
    <ccpp:component>
      <rdf:Description
        rdf:about="http://www.example.com/profil#MaterielFinal">
        <!-- Les propriétés de MaterielFinal ici -->
      </rdf:Description>
    </ccpp:component>

    <ccpp:component>
      <rdf:Description
        rdf:about="http://www.example.com/profil#LogicielFinal">
        <!-- Les propriétés de LogicielFinal ici -->
      </rdf:Description>
    </ccpp:component>

    <ccpp:component>
      <rdf:Description
        rdf:about="http://www.example.com/profil#NavigateurFinal">
        <!-- Les propriétés de NavigateurFinal ici -->
      </rdf:Description>
    </ccpp:component>
  </rdf:Description>
</rdf:RDF>
```

Figure 7 : Exemple de profil CC/PP en XML

### 6.3.3.3- Les valeurs par défaut

Les attributs d'un composant peuvent être incorporés, ou être déclarés par référence à un profil par défaut, lequel peut être stocké séparément et qu'on accède au travers de l'adresse URI qu'il contient.

Cette utilisation d'un profil par défaut (défini ailleurs) est en quelque sorte similaire au concept d'héritage dynamique. Elle rend possible certaines optimisations importantes. En tant que document distinct, le profil peut résider à un emplacement différent et peut être mis en cache séparément. Ceci est particulièrement appréciable dans les environnements en liaison sans fil, tels que les réseaux cellulaires, où les profils peuvent être volumineux et la liaison du client lente et coûteuse. Avec des valeurs par défaut, seule une petite partie du profil total est transmise par le réseau sans fil.

Les valeurs par défaut d'un composant de profil CC/PP sont signalées par un arc **ccpp:defaults** depuis le composant concerné vers un composant qui définit les valeurs par défaut

Le profil de l'appareil appelant les valeurs par défaut :

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ccpp="http://www.w3.org/2002/11/08-ccpp-schema#"
  xmlns:ex="http://www.example.com/schema#">

  <rdf:Description
    rdf:about="http://www.example.com/profil#LeProfil">
    <ccpp:component>
      <rdf:Description
        rdf:about="http://www.example.com/profil#MaterielFinal">
        <rdf:type
          rdf:resource="http://www.example.com/schema#PlateformeMateriel" />
        <ccpp:defaults
          rdf:resource="http://www.example.com/profilMateriel#MaterielDefaut" />
        </rdf:Description>
      </ccpp:component>

    <ccpp:component>
      <rdf:Description
        rdf:about="http://www.example.com/profil#LogicielFinal">
        <rdf:type
```

```

        rdf:resource="http://www.example.com/schema#PlateformeLogiciel" />
    <ccpp:defaults
        rdf:resource="http://www.example.com/profilLogiciel#LogicielDefault" />
    </rdf:Description>
</ccpp:component>

<ccpp:component>
  <rdf:Description
    rdf:about="http://www.example.com/profil#NavigateurFinal">
    <rdf:type
      rdf:resource="http://www.example.com/schema#AgentUtilNavigateur" />
    <ccpp:defaults
      rdf:resource="http://www.example.com/profilTerminal#AgentUtilDefault" />
    </rdf:Description>
  </ccpp:component>

</rdf:Description>
</rdf:RDF>

```

### Les valeurs par défaut de PlateformeMateriel :

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://www.example.com/schema#">
  <rdf:Description
    rdf:about="http://www.example.com/profilMateriel#MaterielDefault">
    <rdf:type
      rdf:resource="http://www.example.com/schema#PlateformeMateriel" />
    <ex:displayWidth>320</ex:displayWidth>
    <ex:displayHeight>200</ex:displayHeight>
  </rdf:Description>
</rdf:RDF>

```

### Les valeurs par défaut de PlateformeLogiciel :

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://www.example.com/schema#">
  <rdf:Description
    rdf:about="http://www.example.com/profilLogiciel#LogicielDefault">
    <rdf:type
      rdf:resource="http://www.example.com/schema#PlateformeLogiciel" />
    <ex:name>EPOC</ex:name>
    <ex:version>2.0</ex:version>
    <ex:vendor>Symbian</ex:vendor>
  </rdf:Description>
</rdf:RDF>

```

**Les valeurs par défaut de AgentUtilNavigateur :**

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://www.example.com/schema#">
  <rdf:Description
    rdf:about="http://www.example.com/profilTerminal#AgentUtilDefaut">
    <rdf:type
      rdf:resource="http://www.example.com/schema#AgentUtilNavigateur" />
    <ex:name>Mozilla</ex:name>
    <ex:version>5.0</ex:version>
    <ex:vendor>Symbian</ex:vendor>
    <ex:htmlVersionsSupported>
      <rdf:Bag>
        <rdf:li>3.2</rdf:li>
        <rdf:li>4.0</rdf:li>
      </rdf:Bag>
    </ex:htmlVersionsSupported>
  </rdf:Description>
</rdf:RDF>
```

**6.3.3.4- L'écrasement d'une valeur par défaut en XML**

Le profil de l'appareil appelant les valeurs par défaut :

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ccpp="http://www.w3.org/2002/11/08-ccpp-schema#"
  xmlns:ex="http://www.example.com/schema#">
  <rdf:Description
    rdf:about="http://www.example.com/profil#LeProfil">

    <ccpp:component>
      <rdf:Description
        rdf:about="http://www.example.com/profil#MaterielFinal">
        <rdf:type
          rdf:resource="http://www.example.com/schema#PlateformeMateriel" />
        <ccpp:defaults
          rdf:resource="http://www.example.com/profilMateriel#MaterielDefaut" />
        <ex:memoryMb>32</ex:memoryMb>
        </rdf:Description>
      </ccpp:component>

    </rdf:Description>
  </rdf:RDF>
```

**Les valeurs par défaut de PlateformeMateriel:**

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ex="http://www.example.com/schema#">
  <rdf:Description
    rdf:about="http://www.example.com/profilMateriel#MaterielDefaut">
    <rdf:type
      rdf:resource="http://www.example.com/schema#PlateformeMateriel" />
    <ex:displayWidth>320</ex:displayWidth>
    <ex:displayHeight>200</ex:displayHeight>
    <ex:memoryMb>16</ex:memoryMb>
  </rdf:Description>
</rdf:RDF>

```

**Figure 8 : L'écrasement d'une valeur par défaut en XML**

Dans cet exemple, le composant par défaut indique une capacité mémoire de 16Mb, mais cette valeur est écrasée par la propriété memoryMb appliquée directement au composant profil. De ce fait, dans ce profil, l'attribut memoryMb a une valeur de 32.

**6.3.3.5- Conclusion de CC/PP**

Le cadre de travail CC/PP (Composite Capabilities/Preferences Profile), défini par le W3C, représente un modèle de description de contexte (Matériel, logiciel, navigateur et les préférences de l'utilisateur). On débute la description par des valeurs déclarées par défaut et on termine par des valeurs finales (écrasement) dans la nécessité, pour changement de profil, afin de guider le processus de l'adaptation de contenu.

Cette description reste incomplète, dont elle porte quelques inconvénients pour l'obtention d'un processus cohérent. Ce manque se résume dans deux points :

- 1- CC/PP n'assure pas la confidentialité et la sécurité adéquate pour protéger les informations privées des utilisateurs.
- 2- L'utilisateur doit avoir des connaissances pratiques du CC/PP, et prévoir tous les cas d'adaptation à l'avance pour constituer le corps du profil, tels que la génération des nouveaux vocabulaires et la redéfinition des attributs ainsi que leurs types pour lui affecter des nouvelles valeurs.

Pour pallier cet inconvénient, on doit rajouter à la description de profil CC/PP, les protocoles SIP et SDP (4.2.c.c), afin que chacun d'eux effectue sa tâche.

## 7- La Négociation :

### 7.1- Définition :

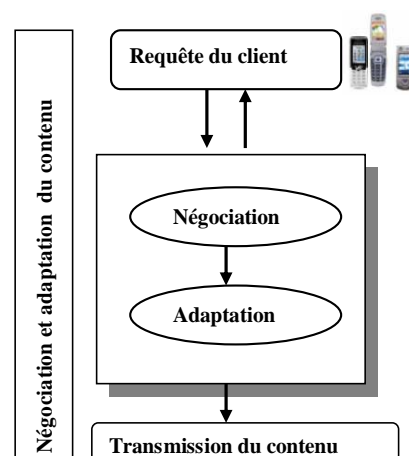
La description du contexte sert de base à une négociation entre un client et un serveur. Grâce à cette description, le serveur peut connaître les contraintes qu'il doit satisfaire. Il se peut qu'il soit en mesure de les satisfaire toutes et, dans ce cas, la négociation est vite conclue. Mais il se peut aussi qu'un dialogue s'engage pour établir un compromis acceptable entre ce que le client demande et ce que le serveur peut offrir ou simplement pour échanger davantage d'information de contexte ; lorsque celle-ci est trop volumineuse pour être transmise en entier systématiquement. C'est le rôle du protocole de négociation (http, Sip...etc) de mener ce dialogue.

L'objectif du processus de négociation est d'assurer une forme de consensus entre les fournisseurs du contenu et les clients. Les fournisseurs du contenu sont représentés par l'ensemble des serveurs et des proxy intermédiaires. Une architecture de négociation et d'adaptation peut être représentée par une boîte noire, qui accepte en entrée les requêtes des clients et retourne en sortie des services adaptés destinés à être utilisés par ces clients [LEM 04 a].

Cette boîte noire, comporte principalement deux processus :

- 1- Le processus de négociation.
- 2- Le processus d'adaptation.

Plusieurs requêtes peuvent être échangées entre le client et le serveur afin d'accomplir l'étape de négociation



**Figure 9 : Vue globale sur le système d'adaptation et de négociation du contenu.**

## 7.2- Classification des stratégies de négociation

Suivant l'invocation du processus d'adaptation, nous distinguons deux types de négociations :

1. **La négociation à la demande:** Dans ce type de négociation, le processus d'adaptation est invoqué lors de la réception de la requête de l'application cliente.

Les résultats de l'adaptation sont directement transmis au client cible. Dans ce cas, l'ensemble des profils document qui peuvent exister au niveau du serveur ne subissent pas de changement. On dit alors que l'ensemble des profils est stable [LEM 04 a].

2. **La négociation fondée sur des caches** [LEM 04 a]: au contraire du premier type de négociation, cette négociation se fonde sur l'utilisation des versions du contenu source préalablement sauvegardées. Cela veut dire qu'à la réception de la requête de l'application cliente, le serveur n'invoque pas directement de tâches d'adaptation. Le serveur (ou l'entité intermédiaire) cherche d'abord des versions existantes. Si la version recherchée n'existe pas le serveur invoque le processus d'adaptation en utilisant la meilleure transformation disponible dans le système. Le contenu source est donc adapté et le contenu généré est stocké pour une utilisation ultérieure. Dans ce cas, l'ensemble des profils s'enrichit par les nouveaux profils qui correspondent au contenu stocké.

Le deuxième type de négociation est préférable si l'utilisation d'un même contenu est fréquente, par un seul client ou par un groupe particulier de clients. En effet, dans ce cas, l'utilisation du cache évite de lancer les mêmes méthodes d'adaptation plusieurs fois, ce qui améliore les performances du système. Cependant, si le contexte du client change fréquemment, l'application du premier type de négociation est plus efficace car il est inutile de stocker des contenus, générés par adaptation, qui seront rarement utilisés.

Une stratégie de négociation qui combine les deux approches peut être définie en se basant sur :

- La trace et la fréquence des requêtes des clients
- Le changement du contexte client

D'autres techniques peuvent être exploitées dans une approche de négociation, telles que les délais de garde pour la sauvegarde des profils, des techniques avancées de cache du contenu.

### **7.3- Les protocoles de négociation**

#### **7.3.1- Le protocole de transfert hypertexte**

**HTTP** [WON 00] (Hypertext Transfer Protocol) est un protocole de communication client serveur développé pour la négociation et le transfert de contenu sur Internet depuis 1990.

#### **Inconvénient de HTTP**

Ce protocole reste toujours limité, malgré les différentes améliorations apportées dans ses versions (0.9, 1.0 et 1.1). Le problème majeur réside dans la gestion de la négociation. HTTP a été créé au départ pour la description de navigateurs, et non pas pour la description des contextes exigés aujourd'hui par l'environnement hétérogène actuel. HTTP devant ses limites, reste toujours un protocole de transfert de données et non pas un protocole de négociation.

#### **7.3.2- Le Protocole d'Initiation de Session (SIP)**

##### **7.3.2.1- Définition**

Le protocole *SIP* (Session Initiation Protocol) normalisé par l'I.E.T.F, R.F.C 2543 [ROS 02], est un protocole de signalisation appartenant à la couche application du modèle OSI. Son rôle est d'ouvrir, modifier et libérer les sessions. L'ouverture de ces sessions permet de réaliser de l'audio ou vidéoconférence, de l'enseignement à distance, de la voix (téléphonie) et de la diffusion multimédia sur IP essentiellement.

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

Enfin, SIP possède l'avantage de ne pas être attaché à un médium particulier et est sensé être indépendant du protocole de transport des couches basses.

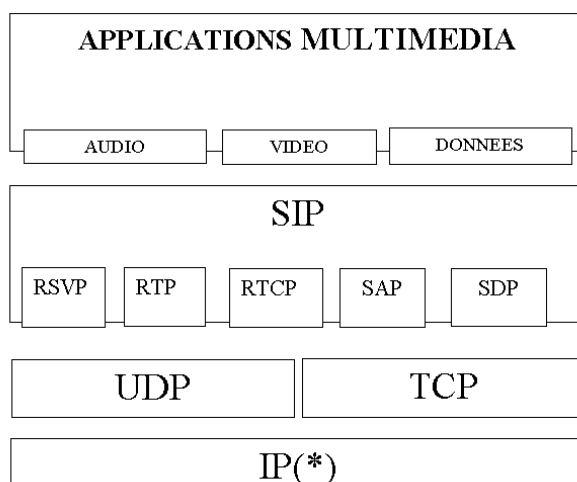
### **7.3.2.2- Fonctionnalités de SIP:**

Le protocole SIP possède plusieurs fonctionnalités, les plus importantes sont les suivantes [ROS 02] :

- Localisation du terminal appelé.
- Analyse du profil et des ressources.
- Négociation du type de média (voix, vidéo, données), et des paramètres de communication.
- Disponibilité de l'appelé : détermine si le poste appelé souhaite communiquer, et autorise l'appelant à le contacter.
- Etablissement et suivi de l'appel : avertit les parties appelant et appelé de la demande d'ouverture de session, gestion du transfert et de la fermeture des appels.

### **7.3.2.3- Architecture de SIP :**

L'architecture en couches de SIP, telle que la présente le modèle OSI, fait apparaître une palette de nombreux protocoles :



**Figure 10 : Architecture en couche du protocole SIP (Pile du protocole SIP).**

A chacune des couches de l'architecture SIP sont associées des protocoles tels que :

**RSVP** (Resource reSerVation Protocol) [SAR 03]: C'est un protocole de signalisation qui permet de réserver des ressources. Il peut aussi être utilisé pour faire connaître les qualités de services « QoS » souhaitées dans un réseau IP.

**RTP** (Real Time Protocol) [PER 02]: Le but de ce protocole, est de fournir un moyen uniforme de transmettre sur IP des données soumises à des contraintes de temps réel (audio, vidéo, etc.). RTP a un court en-tête qui est ajouté aux paquets UDP pour porter des informations en temps réel.

**RTCP** (Real Time Control Protocol) [SCH 03]: le protocole RTCP est basé sur des transmissions périodiques de paquets de contrôle par tous les participants dans la session, il fournit l'information de contrôle telle que la réaction sur la qualité de transmission des données.

**SAP (Session Announcement Protocol)** [HAN 00]: permet aux ressources réseau, d'annoncer leurs adresses réseau et les services qu'elles fournissent. Chaque service est identifié par un numéro appelé identificateur SAP. Les mises à jour SAP sont envoyées toutes les 60 secondes. SAP est également utilisé pour écouter et émettre périodiquement des annonces SAP pour tenir à jour une table des services disponibles sur le réseau.

**SDP (Session Description Protocol)** [HAN 98]: fournit un format universel pour décrire des sessions de multimédia, des annonces ou des invitations (SAP). Il est indépendant de la façon dont cette information est transportée. Il emploie un format de données entièrement textuel pour maximiser la portabilité dans les transports. Il est utilisé par différents protocoles de transport appropriés, y compris le protocole d'annonce de session SAP, le protocole de déclenchement de session SIP, le protocole des flux en temps réel RTP/RTSP, etc. Plus loin, on va voir en détail ce protocole, car c'est le protocole principal utilisé par SIP pour transmettre les types de média utilisés d'une session.

#### **7.4- But de la négociation**

Le principal objectif de la négociation est de faire un compromis entre la demande du client et le service offert par le serveur conteneur de l'information multimédia. Tout en respectant les différentes contraintes du système hétérogène. La négociation s'articule sur les

principales caractéristiques de l'ensemble des entités constituant l'environnement (réseau, matériel, protocole, navigateur, logiciel, préférences de l'utilisateur).

Les services de négociation doivent comprendre :

- La notion du profiling.
- Un mécanisme de recherche des meilleures offres adéquates aux spécifications du client
- L'utilisation des technologies Web telle que XML, RDF et CC/PP.

Mettre en place un bon service de négociation, revient au choix des protocoles de négociation et de transfert adéquats, et aux standards de descriptions de contextes. De ce qui a été étudié auparavant, la description de contexte se fait par le CC/PP et la négociation et le transfert du profil se fait par les protocoles SIP et SDP. Tous ces standards (CC/PP, SIP et SDP) doivent être installés respectivement au niveau du client et au niveau du serveur. Le service de négociation peut être modélisé par un automate déterministe à états finis [KHO 06]

## **8- La transformation**

### **8.1- Définition**

La transformation est la dernière phase du processus d'adaptation. Elle consiste à produire une version bien adaptée au contexte à partir d'une ressource unique. Elle peut intervenir à plusieurs niveaux différents : sur les objets média, sur la structure du document et sur certains aspects de sa sémantique. Le résultat de la transformation se visualise dans des feuilles de style à base des contraintes du contexte.

### **8.2- Présentation des feuilles de style**

Le concept de feuilles de style est apparu en 1996 par une recommandation du W3C intitulée « *Cascading StyleSheets* » (*feuilles de style en cascade*), notée CSS [BAS 06]. Le principe des feuilles de style consiste à regrouper dans un même document des caractéristiques de mise en forme associées à des groupes d'éléments. Il suffit de définir par un nom un ensemble de définitions et de caractéristiques de mise en forme, et de l'appeler pour l'appliquer à un texte.

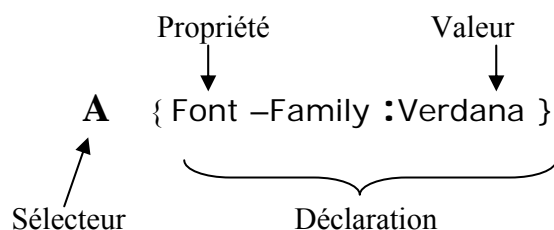
### 8.3- Définition d'un style

La définition d'un style se fait à l'aide de règles en texte simple permettant de décrire l'aspect des éléments de la page. Une règle CSS est caractérisée par deux principaux éléments: Un **sélecteur** de balises, permettant de préciser à quelles balises du document le style s'applique.

Une **déclaration** de style, définie entre accolades, permet de préciser le style à appliquer aux balises sélectionnées. La déclaration est elle-même constituée :

- d'une ou plusieurs propriété(s), suivie du caractère « : » (double point),
- d'une ou plusieurs valeur(s) associée(s) à chaque propriété, entourée de guillemets s'il s'agit de plusieurs mots ou séparés par des virgules s'il y en a plusieurs, suivie d'un point virgule.

La syntaxe est ainsi la suivante :



**Figure 11 : Syntaxe d'une règle de Style.**

La syntaxe suivante définit par exemple le style à appliquer aux liens hypertextes (balise <A>), c'est-à-dire une police Verdana de taille 18 pixels, en gras et de couleur jaune :

```
A { font-family: Verdana; font-size: 18px; font-style: bold; color: yellow }
```

### 8.4- CSS 2.0

Depuis le 12 mai 1998, la norme CSS 2.0 est le standard en vigueur. La norme CSS2 apporte un grand nombre de nouvelles fonctionnalités par rapport à la précédente norme (1.0), parmi lesquelles :

- La prise en compte de la diversité de médias (écran, imprimante, braille, synthétiseur, vocal).
- L'ajout de propriétés auditives pour les aveugles et malvoyants
- La manipulation de curseurs
- La gestion des débordements et de la visibilité des éléments.

## **8.5- Le langage XSL**

### **8.5.1- Définition**

Le XSL [QUI 07] pour *eXtensible Stylesheet Language* ou "langage extensible de feuilles de style" est une recommandation du W3C datant de novembre 1999. C'est un standard dans le domaine de la publication sur le Web. Le XSL est en quelque sorte le langage de feuille de style du XML. Un fichier de feuilles de style reprend des données XML et produit la présentation ou l'affichage de ce contenu XML selon les souhaits du créateur de la page.

Le XSL est lui-même défini avec le formalisme XML, cela signifie qu'une feuille de style XSL est un document XML bien formé. Toutefois, contrairement aux CSS, XSL permet aussi de retraiter un document XML afin d'en modifier totalement sa structure, ce qui permet à partir d'un document XML d'être capable de générer d'autres types de documents (PostScript, HTML, Tex, RTF, ...etc) ou bien un fichier XML de structure différente.

Ainsi la structuration des données (définie par XML) et leur représentation (définie par un langage tel que XSL) sont séparées. Cela signifie qu'il est possible à partir d'un document XML de créer des documents utilisant différentes représentations (HTML pour créer des pages web, WML pour les mobiles WAP [LEC 00], ...).

### **8.5.2- Les différents langages de XSL**

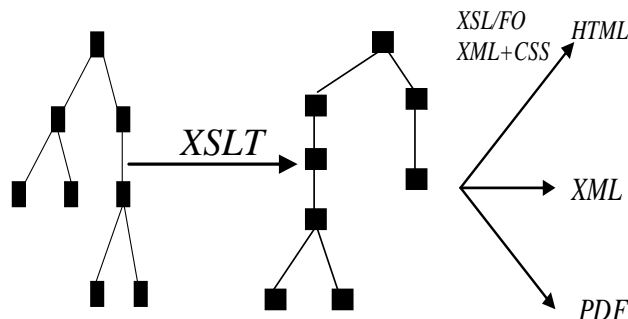
XSL comporte en fait 3 langages :

- Le XSLT est un langage qui transforme un document XML en un format, généralement en Html, reconnu par un navigateur.
- Le Xpath permet de définir et d'adresser des parties de document XML.
- Le XML-Formatter permet de formater un document XML (transformé) de façon qu'il puisse être rendu sur des PCpockets ou des unités de reconnaissance vocale.

#### **8.5.2.1- Le XSLT**

XSLT [AMA 02] (Extensible Style Language Transformations) est un langage destiné à transformer un fichier XML en un autre fichier XML , PDF HTML, Texte Pur selon les

règles (*template rules*) contenues dans la feuille XSL. Il crée une structure logique arborescente (arbre source) à partir du document XML et lui fait subir des transformations selon ces règles, Les composants de l'arbre résultat sont appelés *objets de flux*.



**Figure12 : Création d'une structure logique arborescente à partir du document XML**

### 8.5.2.2- Structure générale d'une feuille de style XSLT

- 1- Le premier élément dans une feuille de style XSLT est « `xsl : stylesheet` », il contient le préfixe `xsl`, et l'espace nominal « `http://www.w3.org/1999/XSL/Transform` ». Cet élément est obligatoire, il peut être dans un document englobant, ou dans la racine d'un document contenant uniquement des règles XSLT.

```
<xsl :stylesheet xmlns :
xsl=" http://www.w3.org/1999/XSL/Transform
Version='1.30' ">
```

- 2- Instruction d'importation ou d'inclusion d'une autre feuille de style XSLT.

```
<xsl : import href ="URI" />
<xsl : include="URI"/>
```

- 3- Définition d'un namespace d'attributs qui pourront qualifier un élément de l'arbre généré

```
<xsl :attribute-set name= "..."> /xsl : attribute-set>
```

- 4- déclaration d'une règle modèle, le sélecteur étant donné par l'attribut *match*

```
sl :template match =" ..."> ..... </xsl :template>
```

5- déclaration de règle nommée .

```
xsl :template name= " ..." > ..... < :xsl template >
```

### **8.5.2.3- Le XSL/FO**

XSL/FO [HOL 03] est un langage de formatage des données, c'est-à-dire un langage permettant de définir la mise en page (affichage de texte ou de graphiques) de ce qui a été créé par XSLT.

Donc après la création de l'arbre source, le *XSL/FO* permet de formater le résultat, c'est-à-dire d'interpréter l'arbre résultat, ou plus exactement les objets de flux le composant en leur appliquant des objets de mise en forme afin d'en faire une représentation visuelle (papier, écran,...).

### **8.6- Transformation des objets média**

On peut transformer les images et la vidéo en appliquant une réduction de couleurs ou de niveaux de gris, en réduisant les dimensions, ou en changeant l'encodage. Une transformation plus drastique peut remplacer une vidéo par une image fixe qui en est extraite, ou une image animée par sa version discrète, ou encore une image fixe ou animée par un texte la décrivant.

### **8.7- Transformation au niveau de la structure logique**

La structure relativement riche d'un document HTML peut être ramenée à une structure XHTML, qui est une version XMLisée de HTML conçue pour les terminaux mobiles. Un autre exemple est la transformation d'un format XML abstrait vers un format plus proche de la présentation, comme XHTML [DAN 00] ou SVG [EIS 02]. Cette transformation de structure peut s'accompagner d'une transformation des objets média qu'elle contient. L'outil de choix des transformations de structures est le langage XSLT, spécialement adapté aux transformations d'un langage XML vers un autre langage XML ou vers une syntaxe textuelle d'un autre type.

## **8.8- Adaptation au niveau sémantique**

Consiste à considérer le sens associé aux éléments structuraux d'un document et aux relations qui les lient. L'adaptation est alors plus ciblée que par de simples transformations structurales, puisqu'elle tire partie du sens associé à la structure. À titre d'exemple, dans SMIL, les éléments *par* et *seq* ont des structures très proches du strict point de vue XML, mais ils portent des sémantiques fort différentes, celles d'objets présentés en parallèle ou en séquence. La prise en compte de cette sémantique permet de les traiter différemment lors de l'adaptation.

## **8.9- Logiciels de transformation**

### **8.9.1- Ixslt :**

iXSLT [QUI 01] est une version incrémentale du processeur Xalan (qui est un processeur XSLT qui transforme les documents XML en texte brut). Le principe de transformation incrémentale consiste à générer des règles de réévaluation qui associent à chaque expression de transformation contenue dans la feuille de transformation XSLT, les instructions à exécuter en fonction des catégories de modifications susceptibles de remettre en cause le résultat de l'expression. Lors d'une opération d'édition du document source ou de la feuille de transformation, les règles de réévaluation sont exploitées pour ne ré exécuter que les instructions nécessaires. Ce processeur incrémental est expérimenté dans un outil d'édition et de présentation de documents multimédia génériques et adaptables développé.

### **8.9.2- VXT :**

VXT [QUI 01] (visual approach to XML transformations) est un environnement interactif mettant en oeuvre un langage visuel dédié à la programmation de transformations de documents XML. Il fournit une représentation graphique originale des structures de documents et des méthodes de manipulation visuelles de ces structures qui permettent à l'utilisateur de spécifier très efficacement des transformations selon le modèle de XSLT. Les transformations sont guidées par la source et peuvent être mises en cascade.

## **9- Solutions proposées**

Des travaux ont déjà été réalisés pour l'adaptation d'un document multimédia tels que:

### **9.1.- Adaptation a priori d'un document SMIL**

Il est possible d'adapter a priori [SEB 04] un document multimédia SMIL. Cette adaptation est donnée par le langage lui-même. SMIL propose un élément *switch* dans lequel il est possible de définir un ensemble d'éléments alternatifs, chacun de ces éléments pouvant être choisi.

Dans un *switch*, l'auteur peut insérer des éléments multimédia ou des éléments composites. Une liste d'attributs est proposée par le langage pour tester les propriétés du système. On peut tester par exemple : la bande passante du système de présentation, les préférences concernant le langage de l'utilisateur, la taille de l'écran... etc. Le système de présentation évalue les éléments contenus dans le *switch* dans l'ordre d'occurrence.

Il est évident de constater que cette méthode d'adaptation comporte plusieurs défauts. Tout d'abord, l'auteur doit prévoir tous les cas d'adaptation à l'avance et réorganiser son document pour chaque condition exprimée. Ceci est d'une part fastidieux et d'autre part mènera dans une grande majorité des cas à un échec car l'auteur ne peut connaître toutes les contraintes qui seront appliquées à son document et notamment les contraintes qui vont apparaître dans le futur. Ensuite, l'auteur doit utiliser la balise *switch* pour chaque combinaison de conditions, ce qui rend le document multimédia complexe.

Enfin, il est impossible de faire comparer les différentes conditions à la volée.

### **9.2- Adaptation a posteriori d'un document SMIL**

Le langage XSLT (eXtensible Stylesheet Language Transformation) [AMA 02] proposé par le W3C permet la manipulation de documents XML à l'aide de règles de transformation. Ces règles sont exprimées dans une feuille de style (*stylesheet*).

Le but de ce langage est d'offrir aux utilisateurs la possibilité de transformer un document XML en un autre document XML au moyen d'un langage déclaratif. Il est ainsi possible d'adapter à posteriori [SEB 04] un document SMIL.

L'avantage des feuilles de style ne nécessite pas de prévoir à priori tous les cas d'adaptation. Néanmoins, cette technique comporte plusieurs défauts. Tout d'abord, les feuilles de style sont figées c'est-à-dire peu adaptées à la situation courante. De plus, les outils d'adaptation sont dans ce cas fortement liés au langage de description de la structure du document. Enfin, la conservation de propriétés du document initial n'est assurée que par l'expertise de l'auteur du programme de transformation.

### **9.3- NAC (Negotiation Adaptation Core)**

L'approche NAC [LAY 04] vise à assurer l'accès adapté aux contenus multimédia, où qu'ils se trouvent dans le réseau et quel que soit le protocole utilisé pour les transférer. Elle interpose un proxy entre le client qui souffre des mêmes limitations et le serveur détenant l'information qui peut être ne dispose aucun moyen d'adaptation, et il utilise un protocole insuffisant en matière de négociation de contenu. Ce proxy prend en charge l'essentiel de la négociation et de l'adaptation, et il a une vision globale de l'environnement.

L'approche NAC consiste à séparer la négociation et le transfert de contenu dans deux protocoles différents. Ainsi, même si un client ne supporte pas la négociation, le proxy proposé peut s'engager pour lui, à partir de son profil, dans une négociation avec un serveur et lui retourner ensuite le contenu négocié.

L'architecture NAC se base sur la description des contextes qui les stocke dans la base de profils. Celle-ci décrit les terminaux clients, les préférences des utilisateurs, l'état du réseau, les capacités des serveurs...etc. Le client n'a pas besoin d'envoyer une description complète de ses caractéristiques, puisqu'à partir de la seule indication du modèle d'appareil, le proxy peut retrouver le détail de son profil dans la base. Cette base doit évidemment être mise à jour régulièrement pour intégrer les nouveaux terminaux, changer les préférences des utilisateurs ou tenir compte des fluctuations du réseau.

L'architecture NAC comporte cinq composants :

**Le proxy de communication** assure l'accès au contenu et le transfert habituel entre le serveur et le client. Il transmet au client les réponses du serveur, après les avoir transformées dans le module d'adaptation ANM.

**Le module ANM** effectue la négociation et l'adaptation. Il coopère avec les autres composants pour prendre la meilleure décision dans la négociation : choix de version du contenu, choix de méthode d'adaptation, choix de méthode de transmission du contenu final.

**Le module de contexte utilisateur** UCM réside sur le client. Il prend en charge la négociation de façon indépendante de l'application cliente. C'est lui qui gère le profil client et il permet à l'utilisateur d'en changer à tout moment.

**Le protocole de négociation** met en oeuvre la stratégie de négociation. Il définit les interactions entre le module UCM du client et le processus de négociation (ANM) du proxy.

**Le système de gestion de profils** assure la gestion et l'interrogation des descriptions de contexte stockées dans la base de profils. Il travaille notamment pour le compte du module ANM.

### **Le UPS : représentation de profils**

Dans les bases de profils, les descriptions de contexte utilisent un modèle appelé UPS [LAY 04] qui se présente comme une extension de CC/PP [CC/PP 04]. Comme CC/PP, UPS utilise RDF pour représenter les profils sous la forme de graphes étiquetés. Les descriptions exprimées selon ce modèle concernent les profils et les ressources clients, les instances et les ressources de documents, les méthodes d'adaptation et l'état du réseau.

### **9.4- STAMP (Synchronized Templates for Adaptable Multimedia)**

Le modèle STAMP [BIL 05] vise la génération dynamique de présentations multimédia à partir de contenus semi-structurés XML.

STAMP supporte la présentation extraite au moyen de requêtes effectuées sur des sources de données XML. Partant de la prémisse que le nombre, le type et la taille des

contenus informationnels fournis par une source de données ne sont pas connus à priori, STAMP propose des modèles ou templates qui décrivent l'organisation spatiale, temporelle, navigationnelle, données, composition et décorative d'une présentation multimédia dont le contenu peut varier. L'instanciation d'un Template s'effectue en tenant compte d'un ensemble de contraintes spatiales et temporelles associé au contexte de diffusion.

Le modèle template a pour objectif de permettre la spécification, au niveau conceptuel, de présentations capables de s'adapter à des éventuelles variations concernant le contenu (quantité de données à présenter, support media différent...ect ) ou le dispositif d'accès utilisé (PC fixe ou portable versus PDA, par exemple ).

Cette approche est basée sur la description des templates qui sont instanciés lors de la génération d'une présentation.

Un template constitue un format prédéfini utilisé comme point de départ pour la génération d'une présentation adaptée

Le modèle STAMP est basé sur six dimensions dont les trois premières sont communes pour toutes les propositions de formalisation d'une présentation multimédia tels que la dimension spatiale qui nécessite la disposition spatiale des éléments constituant la présentation, la dimension temporelle qui décrit l'ordonnement des événements de la présentation, la troisième dimension, navigationnelle qui fait les liens de navigations (externe ou interne) activables dans la présentation. Les trois autres dimensions complémentaires caractérisant STAMP sont : la dimension données, elle modélise le contenu à insérer dans la présentation. Cette modélisation, permet lors de la description des autres dimensions, de faire abstraction du fait que le contenu n'est pas connu au moyen de la conception. La dimension composition décrit la présentation au moyen de structures emboîtées construites à partir des contenus informationnels à délivrer. La dimension décorative correspond à la personnalisation de l'apparence de la présentation.

Le modèle STAMP qui repose sur la génération dynamique de la présentation multimédia, permet au SIW (système Informatique de Web) de délivrer un contenu adapté en termes de présentation sans intervention explicite de la part du concepteur.

## **10- Conclusion**

L'adaptation des documents multimédia est abordée à travers trois aspects complémentaires:

Le premier aspect concerne les **formats**. L'objectif est de concevoir des formats de document multimédia qui soient autant que possible indépendants des appareils qui devront les présenter.

Le deuxième aspect de l'adaptation est celui des **profils** et de la **négociation**. L'idée consiste à décrire le contexte d'utilisation d'un document (caractéristiques de l'appareil de présentation, l'état du réseau, préférences de l'utilisateur) et les caractéristiques du document lui-même. A partir de ces données, les parties prenantes (schématiquement, le serveur et le client) négocient pour trouver la forme du document la plus appropriée qui puisse être produite à partir de l'existant. C'est cette forme qui est ensuite engendrée et délivrée à l'appareil.

Le dernier aspect de l'adaptation, est la **transformation** des structures et des contenus en fonction des résultats de la négociation. Ces transformations sont examinées sous plusieurs angles. On s'appuie sur certains aspects de la sémantique des documents multimédia, comme leur sémantique temporelle, spatiale ou navigationnelle. Où on s'appuie sur la modularité des formats XML du Web pour transformer les structures. Enfin, on tire partie des différents modes de codage des contenus.

# Chapitre 3

## **Chapitre 3 Contribution**

### **1- problématique**

La diversité de l'évolution technologique, sa prolifération dans les réseaux et les machines hétérogènes et leurs caractéristiques, la complexité des données et leur présentation, ainsi que les préférences des utilisateurs constituent autant de facteurs à l'origine de la complexité et l'hétérogénéité de l'environnement actuel.

Celui-ci s'articule sur cinq (5) paramètres, à savoir l'aspect matériel, l'aspect réseau, l'aspect système ou logiciel, l'aspect contenu et l'aspect préférentiel des utilisateurs, lesquels constituent la condition sine qua non de toute solution standard.

A ce titre, le quatrième paramètre, en l'occurrence celui lié au contenu, par référence dans ce travail de recherche, au document multimédia, constitue le centre même de cette étude.

En effet, le document multimédia, tel qu'il apparaît sur le web, se présente sous une forme unique, indépendante de sa présentation [LEM 04 a], qui nécessite des moyens software (logiciel, navigateur et feuille de style) et hardware (matériel et réseau), pour permettre la visualisation de son contenu.

A ce titre, compte tenu des effets conduits de la corrélation de ces facteurs exogènes, ce travail de recherche s'est fixé comme objectif l'identification des opportunités d'adaptation et de transformation de document multimédia en fonction de cet environnement hétérogène et ses paramètres.

### **2- La solution : L'adaptation et la transformation :**

La solution est de mettre en place un système qui prend en charge les différentes étapes d'adaptation et de transformation étudiées dans le Chapitre 2. Donc la réponse à cette problématique passe à priori, par la création d'un profil (la description de contexte), puis la génération des règles qui y sont issues après une négociation sur l'offre, dans l'optique d'appliquer la transformation du contenu du document multimédia, en fonction des règles générées, sur des feuilles de styles et de présentation.

### **3- La description de contexte**

#### **3.1- Notion de profil**

Le profil est un ensemble de valeurs affectées, à un instant donné, aux différents attributs caractérisant l'ensemble des entités de l'environnement hétérogène (réseau, matériel, préférences de l'utilisateur...etc). D'où, le profil reflète une image unique de cet ensemble, dans un contexte bien déterminé à un instant donné.

#### **Exemple :**

##### **- Préférences de l'utilisateur**

Je souhaite avoir la couleur. Je préfère le format GIF. Je ne comprends que le français. Mes centres d'intérêt sont le sport, les recherches scientifiques et les informations d'actualité.

##### **- Matériel utilisé :**

Téléphone cellulaire, Marque Sagem X2, Ecran couleur, Affichage 100\*50 pixel.

##### **- Réseau d'accès : GSM.**

##### **- Langage utilisé : WML.**

#### **3.2- Les métaphores**

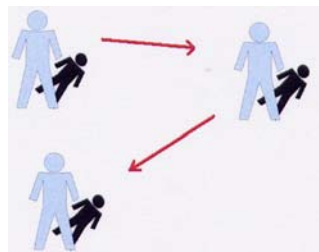
La métaphore [PAU 97] est une figure qui consiste à désigner un objet ou une idée par un mot qui convient pour un autre objet (ou une autre idée) lié au précédent par une analogie. La métaphore fusionne donc en un seul les deux termes de la comparaison.

La métaphore c'est le remplacement d'un mot par un autre selon un rapport de ressemblance

#### **3.3- Représentation du profil par la métaphore**

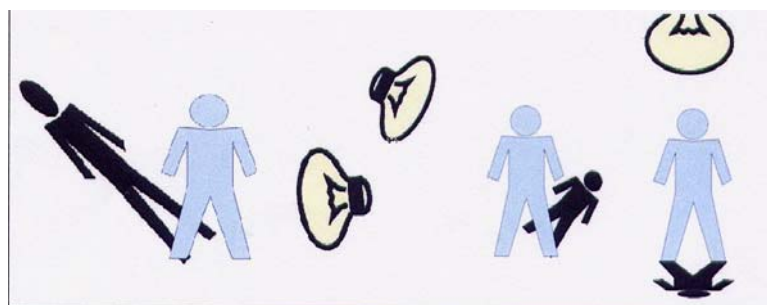
Le profil est à un utilisateur ce que l'ombre est à une personne, d'où le profil est métaphoriquement représenté par l'ombre d'une personne.

Tout comme l'ombre, le profil doit être unique et doit suivre partout son propriétaire.



**Figure 13. Unicité de l'ombre d'une personne**

La taille de l'ombre d'une personne peut être petite ou grande selon la position de la source lumineuse (soleil, lampe...etc). Pour le profil, ceci peut être interprété par l'état du réseaux à un instant donné : congestion, début de la connexion...etc.



**Figure 14 La taille de l'ombre face à la source lumineuse.**

Dans le cas où il n'y a pas de source lumineuse « l'obscurcissement », l'ombre n'apparaît pas. Dans ce cas nous ferons appel à la dernière ombre si elle existe avant la disparition de la lumière, sinon l'ombre prend à défaut la forme de la personne tel qu'elle est. C'est ce que nous appelons le profil par défaut, dans le cas où l'utilisateur ne donne pas la totalité des informations le concernant.



**Figure 15 : L'ombre d'une personne dans l'obscurité**

La personne peut grandir en taille, prendre ou perdre du poids, l'ombre s'adaptera aux nouvelles formes. Ainsi le profil doit suivre les changements matériels du client et s'adapter quelque soit leurs importance



**Figure 16 : l'ombre change de forme aux changements apportés à la personne.**

En passant d'une zone éclairée à une autre qui l'est moins, l'ombre changera instantanément de forme, aussi le profil doit pouvoir être mis à jour automatiquement.

Tout en gardant la même taille, une ombre pourra changer de forme selon les mouvements de la personne. Nous pouvons traduire ce phénomène par les préférences de l'utilisateur.



**Figure 17 : l'ombre suit la personne dans les mouvements.**

De cette métaphore nous arrivons, à mieux cerner la notion de profil utilisateur et ses différentes composantes.

Tout changement dans la forme de l'ombre métaphoriquement désigne le changement d'un ou plusieurs paramètres composants le profil tels que :

#### **Les paramètres du réseau :**

- **Localisation de l'utilisateur :** correspond à l'adresse URL de l'utilisateur dans le réseaux.

- **Etat du réseau** : correspond au débit de la connexion du client ainsi que la congestion du réseau.

- **Type du réseau** : WIFI, GSM, GPRS, UMTS.... etc.

#### **Les paramètres du Matériel :**

- La taille de l'écran.
- Le nombre de couleurs.
- La taille de la mémoire vive.
- La taille de la mémoire secondaire.
- La résolution.
- La vitesse d'exécution « CPU ».
- L'audio.
- La marque du matériel

#### **Les protocoles utilisés :**

HTTP, RTSP, RTP, SIP.

#### **Les logiciels utilisés :**

SMIL, SVG, MMS, XHTML.

#### **Le système d'exploitation :**

Windows, UNIX, Symbian Os

#### **Les préférences de l'utilisateur :**

- **Vision de l'utilisateur** : le contenu multimédia doit pouvoir s'adapter à l'acuité visuelle de l'utilisateur.

- **Audition de l'utilisateur** : le contenu multimédia doit aussi être adapté à l'ouïe de l'utilisateur.

- **La langue.**

- **Domaine d'intérêt.**

- **La mise à jour automatique.**

Le profil est le socle sur lequel repose toute adaptation de contenu multimédia. Il représente les caractéristiques utilisateurs, qui sont dénominateurs communs à tous les clients potentiels.

Pour une meilleure représentation des informations, le profil suit la structure CC/PP, qui est basé sur le langage de description des métadonnées RDF et de XML pour toute extensibilité de description de données. Lequel il permet la description de n'importe quel attribut nouvellement apparu dans le profil.

Le cadre CC/PP permet de générer systématiquement l'ensemble des règles de gabarit « template rules », qui vont être par la suite les conditions de base de la transformation du contenu demandé.

### **3.4- Formulation du Profil**

La formalisation du profil [LOP 05] se base sur les standards W3C (RDF et CC/PP). Le CC/PP propose trois composants inclus dans RDF : BrowserUA pour la description du navigateur, SoftwarePlatform pour la description des capacités logiciels et HardwarePlatform pour la description des capacités matérielles. La description des caractéristiques utilisateur nécessite l'ajout des composants CC/PP nouveaux, pour représenter les préférences de l'utilisateur. Ces composants sont intégrés dans deux ressources RDF appelées UserCharacteristic [LOP 05] et UserContext [LOP 05]. La ressource UserCharacteristic est constituée de deux composants CC/PP (UserStatic et UserDynamic), définissant respectivement les caractéristiques statiques et dynamiques. La ressource UserContext est constituée aussi de deux composants CC/PP (Environmental et TaskCharacteristic). Le composant Environmental permet de décrire la localisation de l'utilisateur et le moment de la connexion, et le composant TaskCharacteristic décrit les préférences de l'utilisateur.

## **4- La négociation**

La négociation dans notre approche s'effectue entre le Serveur contenant la présentation multimédia (le contenu), le Client et le Serveur qu'est dédié pour la gestion des Profils des utilisateurs, de la négociation et de la génération des règles de gabarit.

Avant la négociation, le Serveur qui a été spécifié pour gérer le Profil, la négociation et génération des gabarits procède à l'opération d'extraction de différentes caractéristiques du profil écrit en CC/PP et du document multimédia demandé écrit en SMIL. Ceci dans le but d'appliquer une série de comparaison et de compatibilité de ce qui a été extrait (la faisabilité de satisfaire la demande)

La comparaison peut mener à l'inadaptabilité de la demande (le profil) par rapport à l'offre (la présentation multimédia), de cet effet la négociation est née. Cette dernière s'effectue par échange d'information entre le client et le Serveur contenant le Profil à travers le serveur SIP, afin de satisfaire le client en concertation avec le serveur de contenu [KHO 06]

La négociation finalise l'échange d'information, soit par un accord, ou un refus. Le cas positif nécessite la mise à jour du profil écrit en CC/PP dans la base de profil, puis la génération de l'ensemble de règles de gabarit «template rules» servant la transformation par la suite. Le cas d'échec, lié à la non satisfaction du client de l'offre met fin au dialogue.

### **5- La transformation**

La transformation est l'étape finale de l'adaptation, elle se base sur les règles d'adaptation générées dans la phase précédente « la négociation ». Une règle d'adaptation est une procédure appliquée pour transformer une structure ou une ressource média d'un état vers un autre. Par exemple, une règle peut juste remplacer le nom d'un nœud dans un arbre ou changer les dimensions d'une image. La transformation s'effectue sur les caractéristiques du document multimédia qui peuvent être quantitative, qualitative ou les deux à la fois (une série d'adaptations) afin de satisfaire le profil de l'utilisateur.

La transformation consiste en :

- l'adaptation au Matériel
- l'adaptation au réseau
- l'adaptation au système et du logiciel utilisé
- l'adaptation aux préférences utilisateur (couleur, format, langue, malvoyant, aveugle, sourd...)
- l'adaptation au contenu
  - adaptation spatiale
  - adaptation temporelle

- adaptation logique
  - adaptation sémantique
- l'adaptation globale (série d'adaptations).

### **5.1- L'adaptation au Matériel**

L'adaptation au matériel consiste en l'adaptation aux différents paramètres (taille d'écran, mémoire, capacité...etc) qui le caractérisent.

#### **5.1.1- Module de Conversion de pixel en pourcentage**

Ce module est très important. Il s'occupe de transformer les valeurs, taille et de placement spatial de tous les médias, en pourcentage par rapport aux dimensions globales de la présentation. Cette transformation a pour but, de faciliter une adaptation éventuelle de la taille des médias à de nouvelles dimensions d'écrans.

SMIL permet la synchronisation des fichiers de natures différentes dans des régions bien déterminées (positionnement et taille de chacune). Mais cette présentation reste spécifique pour des types bien déterminés de profils. En plus on trouve l'encodage et la langue...etc. Ce qui évoque l'un de nos objectifs dans cette approche, présenté dans ce module : conversion de Pixel en Pourcentage. Lequel permet de parcourir la partie « HEAD » du fichier SMIL contenant les informations sur les placements spatiaux des régions, afin d'extraire et de rendre l'entête du document SMIL accessible, et variable selon le profil, puis, il convertit les mensurations des régions calculées par pixel en pourcentage.

Le positionnement et la taille des régions de présentations deviennent relatifs par rapport aux dimensions globales de présentation.

#### **Exemple**

**Soit le document SMIL suivant :**

```
<smil>
<head>
  <!--presentation attributes-->
  <meta name ="title" content="House of Dreams"/>
  <mata name ="author" content "RealNetwork, Inc"/>
  <mata name ="copyright" content ="© 1998"/>
```

```

<layout>
  <root-layout width ="580" height "213" background-color=" black"/>
  <region id = "text_region" left="0" top="0" width="255.2" height="213"/>
  <region id = "images_region" left="255.2" top="0" width="319" height="213"/>
</layout>

</head>

<body>
  <seq>
    <video scr="fumer.mpeg" region="images_region"begin="1"end="15"/>
    <par>
      <text scr=" house.rt" region="text_regin" begin="1s" end="20"/>
      <img scr="house.rp" regin= "images_region" begin="1s" end="20"/>
      <audio scr="musique.mp3" begin="1" end="20"/>
    </par>
  </seq>
</body>
</smil>

```

**L'applique du module conversion pixel en pourcentage sur cet exemple donne :**

```

<smil>
<head>
  ....
<layout>
  <root-layout width ="580 height "213" background-color=" black"/>
  <region id = "text_region" left="0%" top="0%" width="44%" height="100%"/>
  <region id = "images_region" left="44%" top="0%" width="55%" height="100%"/>
</layout>

</head>

<body>
  .....
</body>
</smil>

```

### **5.1.2- L'adaptation à l'affichage**

Différents langages offrent la possibilité d'adapter l'affichage d'un document selon le matériel ou le choix de l'utilisateur, notamment le langage SMIL (la version 2.0 et 2.1). Cette adaptation reste restreinte aux services offerts par ces langages.

L'idée est d'élargir cette adaptation selon les préférences et les moyens utilisés (la taille de l'écran). Le module de conversion de Pixel en Pourcentage répond à ce problème, il applique les valeurs extraites du profil écrit en CC/PP (profil matériel) sur ce qui a été converti en pourcentage «relative».

Les paramètres de l'affichage du profil matériel deviennent relatifs par le même module de conversion, afin qu'ils soient par la suite les règles d'affichage appliquées dans les langages de transformation XSLT et XSL-FO.

### **5.1.3- L'adaptation à la couleur et à la résolution**

Dans cette adaptation on a deux types de couleur, le premier c'est la couleur de fond de chaque région (zone d'affichage) et le deuxième c'est la couleur du contenu lui-même.

Pour le premier, on trouve ses informations dans la description de chaque région donc l'idée est de convertir ces valeurs de type texte en valeur numérique (énumérer) puis on applique la même idée sur la qualité de couleur du matériel (2, 16, 32, 256 couleur ou 16 ou 32 bits). Par la suite on fait le rapprochement de ces valeurs relatives à la meilleure présentation qui correspond au profil du matériel par rapport à la valeur originale du document SMIL.

On procède de la même façon pour la couleur du contenu de document sans oublier les préférences de l'utilisateur tel que son choix de voir ou ne pas voir une telle chose, ses capacités physique (malvoyant, aveugle, sourd...). L'adaptation de la résolution se réalise par l'élément **system-screen-depth** contenant dans la balise SWITCH (SMIL 2.0) pour indiquer la résolution minimale de l'écran.

### **5.1.4- L'adaptation à la mémoire et à la vitesse d'exécution**

La diversité du contenu et sa complexité notamment sa synchronisation et sa taille, rend difficile sa présentation sur des machines de capacités limités et de puissances faibles. Malgré ces difficultés, on doit réfléchir à une adaptation qui les prend en charge, comme

l'adaptation à la mémoire et à la vitesse d'exécution. Celle-ci est combinée avec les préférences de l'utilisateur, dont le but d'assurer le bon déroulement de la présentation, et qu'il doit être supporté par les différentes machines.

Cette adaptation est beaucoup plus technique, tout en respectant la sémantique du document, dont elle permet de réduire la taille du document multimédia tel que :

- la compression pour diminuer la taille du document présenté.
- la synchronisation des objets multimédias constituant le document, et ce, soit dans la visualisation des objets parallèles qui devient en série, soit dans l'ordre et le moment de leurs défilements.

**Exemple:**

```
<par>
  
  <audio src="Discours.au" begin="4s" end="18s" />
</par>
```

Dans l'exemple ci-dessus, le déroulement de la présentation est comme suit: Les deux objets se jouent en parallèle, dont l'objet Titre est joué de 0 à 4 secondes, l'objet Discours de 4 à 18 secondes. Les deux objets doivent être chargés en mémoire, malgré que l'objet discours commence en même temps que la fin de l'objet titre.

La solution : pour alléger la mémoire on doit modifier la présentation tel que l'attribut **par** est remplacé par l'attribut **seq** et l'attribut **begin** et **end** remplacés par l'attribut **dur**. Afin que les deux objets se trouvent en mémoire central en alternative.

Le déroulement de la présentation est comme suit :

```
<seq>
  
  <audio src="Discours.au" dur="15s" />
</seq>
```

- Dans SMIL 2.0 l'élément *switch* offre la possibilité d'ignorer certaines informations (sous titre) afin d'économiser la capacité du document et de la bande passante.
- SVG pour la présentation vectorielle à base des coordonnées sous forme de texte au lieu l'utilisation de la présentation en pixels.

## **5.2- L'adaptation au réseau**

Dans les réseaux, les ressources peuvent varier pendant l'exécution, telle que la bande passante, elle peut diminuer selon le nombre d'utilisateurs (GPRS). Comme il est nécessaire d'adapter le flux de données au profil du réseau. De plus le flux de données peut traverser différents réseaux ayant des caractéristiques différentes et spécifiques où les protocoles classiques s'avèrent inefficaces, à l'instar des réseaux sans fils (Wifi, GPRS, UMTS...) où la bande passante est partagée entre tous les utilisateurs du point d'accès.

Dans SMIL 2.0 l'élément *switch* (**system-bitrate**) pour le choix du type de connexion et pour contrôler le taux de transmission, et **system-required** pour la reconnaissance ou non d'une extension.

MPEG 21 pour l'encodage et le décodage (la méthode utilisée, c'est avec perte) puis la compression du document encodé et la décompression avant le décodage.

Optimiser l'utilisation des ressources du réseau si la bande passante est limitée revient à réduire le temps de réponse et les délais de transmission afin de réaliser l'adaptation au débit de connexion.

### **5.2.1- Adaptation au débit de connexion :**

Le débit représente le facteur le plus influent dans le processus d'adaptation. Il représente la quantité d'information pouvant être envoyée sur le réseau durant une unité de temps. Ainsi pour les débits limités, il est impératif de réduire la quantité d'information à transmettre.

Pour ce faire, plusieurs techniques existent relativement à chaque type de médias. Nous en citerons les suivantes :

**A - Pour l'audio :**

Réduire la quantité d'information transmises sur le réseau, revient à supprimer un certain nombre de données. La suppression ne se fait pas de manière arbitraire, mais selon les besoins.

En premier lieu, sont supprimées, les données des hautes et basses fréquences inaudibles à l'oreille humaine.

La seconde phase consiste à ne garder que les fréquences souhaitées et à supprimer les autres. Ainsi par exemple, les signaux vocaux favorisent les fréquences se trouvant dans la gamme de la parole. Alors que la musique fait appel à une plus large gamme de fréquences.

Il est à souligner que la qualité d'une bande sonore diminue avec la suppression des données.

**B - Pour l'image**

Pour réduire la quantité d'information à transmettre dans une image, on doit se baser sur la compression MPEG et la réduction de la taille de l'image dans un fichier texte (SVG).

**C - Pour la vidéo**

Une vidéo est une succession d'images affichées à intervalles réguliers associés à l'audio. La fluidité de la vidéo est fonction du nombre de ces images par seconde. L'adaptation à la vidéo revient à l'utilisation de l'une des adaptations précédentes (l'adaptation pour l'audio ou l'image).

**5.3- L'adaptation au logiciel**

L'adaptation du logiciel concerne le passage d'un protocole d'utilisation vers un autre selon la machine connecté et le type de réseau utilisé tel que de HTML vers WML ou bien l'utilisation de XHTML Mobile profile (XHML MP) ou XHTML Basic.

**5.4- L'adaptation aux préférences de l'utilisateur**

L'utilisateur peut influencer sur le processus d'adaptation selon ses préférences et ses handicaps. Les préférences se résument dans la manière de voir les choses notamment : le format, la langue, le centre d'intérêt, la résolution ...etc. Par contre les handicaps concernent les déficiences physiques tel qu'un malvoyant, un aveugle, un malentendant et un sourd.

La formulation de ces caractéristiques constituant le profil utilisateur se fait par le standard CC/PP.

Le profil de l'utilisateur est métaphoriquement représenté par l'ombre à un instant donné dans un contexte bien déterminé. Une ombre peut changer de forme aux changements de mouvements de la personne (aux changements de préférences), (voir la partie 2.1 chap 3).

L'utilisateur peut effectuer manuellement certaines adaptations d'après leurs capacités matérielles telles que : la résolution, la couleur, la substitution de la police de caractère, le re-taillage d'image en basant sur le ré-encodage des ressources ou bien la sélection de la version de présentation selon leurs capacités de traitement et d'affichage.

L'utilisateur peut faire une transformation et transcodage lors de l'analyse de feuilles de styles XSLT ou l'exécution de certains scripts. Ces techniques peuvent inclure la conversion ou la traduction du contenu, le filtrage, la sélection ...etc. Dans ce cas, la contrainte ne vient pas des possibilités de la machine mais des souhaits et des connaissances de l'utilisateur.

L'adaptation manuelle exige parfois certaines connaissances pratiques de manipulation de la machine, chose qui n'est pas le cas pour tous les utilisateurs. De cette idée notre approche s'appuie sur l'adaptation automatique selon les handicaps et les préférences enregistrés dans le profil de l'utilisateur. Cette adaptation est concrétisée par un module qui prend en charge les différentes contraintes de l'utilisateur d'un côté, et de parcourir le fichier SMIL, afin de comparer la liste des objets multimédia constituant le document demandé par rapport au choix de l'utilisateur.

#### **5.4.1- L'adaptation pour un handicapé visuel : malvoyant**

Dans ce cas, le module enlève les objets visuels (scénario, texte et image), et il ne laisse que l'audio d'après le profil de l'utilisateur. Ainsi il peut faire la réduction de couleur ou de niveau de gris s'il s'agit d'un utilisateur malvoyant, comme il peut substituer la taille de police du caractère par une autre taille ou style.

#### **5.4.2- L'adaptation pour un handicapé auditif**

Dans ce cas, le module effectue l'opération inverse, dont il enlève les objets audio, et il ne laisse que l'objet visuel.

#### **5.4.3- L'adaptation à la langue**

L'adaptation de la langue nécessite l'utilisation d'un traducteur universel pour qu'il puisse traduire un texte écrit d'une langue originale vers une autre langue souhaitable.

L'élément *switch* (system-language) de SMIL répond sur le choix de la langue souhaité par l'utilisateur.

#### **5.5- L'adaptation au contenu**

La transformation de contenu signifie l'adaptation de l'encodage, du format, du type et de la taille des objets multimédias constituant le document. Par exemple, pour appliquer l'adaptation des images et de la vidéo, on doit faire, une réduction de couleurs ou de niveau de gris, un re-taillage ou une conversion de format d'encodage des objets constituant le document multimédia. Pour le flux audio, on le transforme en une succession de messages textes et vice versa. Cette transformation nécessite l'application des algorithmes spécifiques pour la réaliser.

L'adaptation au contenu concerne leurs principales caractéristiques :

- L'adaptation au niveau spatial
- L'adaptation au niveau temporel
- L'adaptation au niveau sémantique et structurelle
- L'adaptation au niveau logique

##### **5.5.1- L'adaptation au niveau Spatial**

Cette adaptation concerne beaucoup plus l'affichage, ce point a été étudié dans la partie de l'adaptation à l'affichage utilisant le module de conversion de pixel en pourcentage.

On peut changer la qualité d'affichage tel que le passage d'une présentation en pixel vers une présentation vectoriel (SVG [EIS 02], SVG Tiny [LEM 04 b] ou SVG Basic [LEM 04 b]) selon les caractéristiques du matériel utilisé [MOB 03]. L'attribut *fit* (**Hidden, Fill,**

**Meet, Slice, Scroll**) de SMIL2.0 offre aussi autres possibilités d'adaptation, dont il détermine comment les objets multimédia vont être représentés dans les régions.

### **5.5.2- L'adaptation au niveau temporel**

Un document multimédia est beaucoup plus caractérisé par sa dimension temporelle. Cette dernière est exprimée dans le document de présentation SMIL par les opérateurs parallèles ou séquentiels sur les intervalles (Begin, End, Dur..etc) [SEB 04]. La synchronisation se résume dans la visualisation des objets parallèles qui devient en série, soit dans l'ordre et le moment de leurs défilements

#### **Exemple :**

Soit un film qui a comme générique un texte décrivant les acteurs plus la musique et en même temps le film se joue. Donc on doit séparer les objets comme suit : on commence par le générique puis l'action du film.

Cette adaptation est complémentaire à l'adaptation à la mémoire et à la vitesse d'exécution. Afin d'optimiser la bande passante, l'affichage ou pour mieux exécuter le déroulement du document multimédia, lequel ne perde pas sa sémantique.

### **5.5.3- L'adaptation au niveau sémantique et structurel**

Une transformation plus drastique peut remplacer une vidéo par une image fixe ou une image animée par sa version discrète, ou encore une image fixe ou animée par un texte la décrivant.



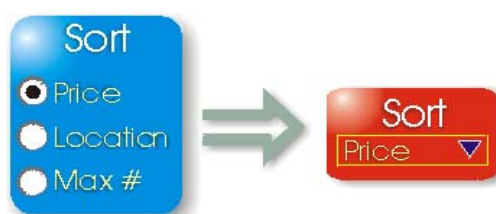
**Figure 18 : Transformation concerne les objets média**

L'adaptation sémantique consiste à considérer le sens associé aux éléments structuraux d'un document et aux relations qui les lient.

#### **5.5.4- L'adaptation au niveau de la structure logique**

Dans un document multimédia, il est possible de distinguer des parties du document qui sont sémantiquement liées et qui peuvent être regroupées par le biais de relation logique. Regrouper les éléments sous une même entité revient à définir une structure logique du document c'est-à-dire une structure hiérarchique des éléments multimédia. La racine de la structure logique représente l'ensemble du document multimédia.

L'adaptation au niveau logique revient d'une part à enlever ou substituer certains objets de l'arbre logique, et d'autre part de transformer l'arbre logique d'un document HTML vers un autre arbre XHTML, ou de XML vers XHTML ou SVG.



**Figure 19 : Transformation au niveau de la structure logique**

#### **5.5.5- Adaptation au niveau de l'hypermédia**

L'adaptation au niveau de l'hypermédia concerne les liens existents entre les objets du même document, voire même entre différents documents. L'adaptation revient à localiser et parcourir les objets référencés par ces liens, afin d'éliminer certains d'entre eux qui ne correspondent pas aux intérêts de l'utilisateur ou matériels. Afin d'optimiser la bande passante, l'affichage ou bien pour mieux passer le document, à condition que le document ne perde pas sa sémantique.

#### **5.5.6- Inconvénients**

Les adaptations au contenu présentées ci-dessus exigent à l'utilisateur d'intervenir dans tous les niveaux, et prévoir tous les cas d'adaptation à l'avance pour exprimer ses besoins.

La solution, est de voir une présentation multimédia d'une autre vision. Cette dernière est concrétisée métaphoriquement par la valorisation du document multimédia qualitativement et quantitativement.

### **5.5.7- La représentation quantitative et qualitative d'un document multimédia**

SMIL exprime l'organisation des objets multimédia par des opérateurs parallèles ou séquentiels sur les intervalles (Begin, End, Dur...etc) [SEB 04]. D'où la précision de ces présentations peut s'exprimer quantitativement.

Exemple :

```
<par>
  
  <audio src="Discours.au" begin="4s" end="18s" />
  <video src="Auteur.avi" begin="4s" end="10s" />
  <video src="Demo.avi" begin="10s" end="20s" />
</par>
```

Dans l'exemple ci-dessus, le déroulement de la présentation est comme suit: Les quatre objets se jouent en parallèle, dont l'objet Titre est joué de 0 à 4 secondes, l'objet Discours de 4 à 18 secondes, l'objet Auteur de 4 à 10 secondes, et l'objet Demo de 10 à 20 secondes.

Cette description *quantitative* du document multimédia décrit précisément le début et la fin de chaque objet multimédia.

En général, un document multimédia est spécifié *qualitativement* car il est plus commode à l'auteur de laisser l'interprétation de la spécification à la machine.

La description qualitative du document écrit dans l'exemple précédent est comme suit :

```
<seq>
  
<par>
  <audio src="Discours.au" />
<seq>
  <video src="Auteur.avi" />
```

```

    <video src="Demo.avi" />
  </seq>
</par>
</seq>

```

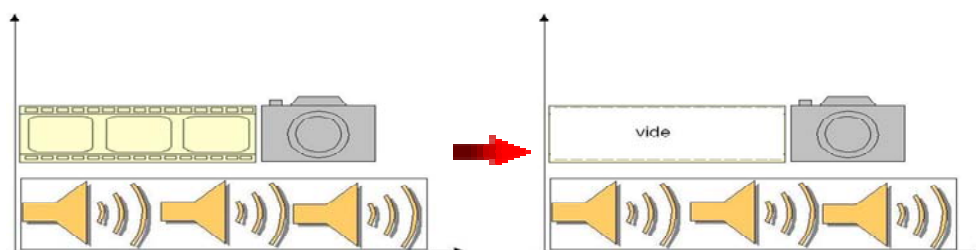
Cette description se déroule de façon suivante :

- Lorsque l’objet Titre se termine, l’objet Discours commence.
- L’objet Discours commence en même temps que l’objet Auteur.
- Lorsque l’objet Auteur se termine, l’objet Demo commence.

Dans ce cas la relation entre les objets s’exprime qualitativement.

### **5.5.7.1- L’adaptation Quantitative**

L’adaptation quantitative se base sur la sélection de l’objet multimédia souhaitable (son, vidéo ou image) selon les contraintes imposées par le profil, et on exclut les objets non conformes à ce dernier.



**Figure 20 : Adaptation quantitative (l’objet vidéo enlevé).**

### **5.5.7.2- L’adaptation Qualitative**

L’adaptation qualitative est applicable sur l’ensemble des objets quantitativement sélectionnés, afin d’augmenter la qualité de la présentation par rapport aux contraintes imposées par le profil.

### **5.5.7.3- Application de l’adaptation quantitative et qualitative**

#### **- Adaptation à l’affichage**

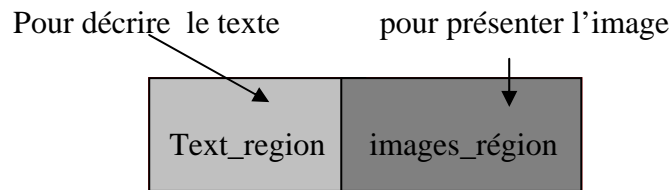
L’adaptation à l’affichage consiste à la récupération des zones désignées pour présenter des objets multimédias supprimés par l’adaptation quantitative, et de les fusionner avec les autres zones restantes selon les cas de chaque profil.

**Exemple**

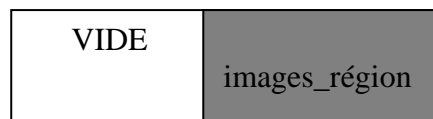
```

<layout>
  <root-layout width ="580 height "213" background-color=" black"/>
  <region id = "text_region" left="0%" top="0%" width="44%" height="100%"/>
  <region id = "images_region" left="44%" top="0%" width="55%" height="100%"/>
</layout>

```

**Figure 21 : Description des zones de présentation**

L'adaptation quantitative pour cet exemple enlève le texte, ce qu'implique l'inutilisation de zone text\_region,

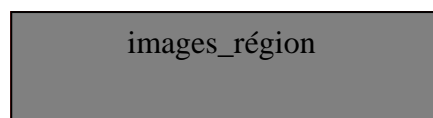
**Figure 22 : L'adaptation quantitative**

L'adaptation qualitative récupère l'information de ce qui a été enlevé, afin d'améliorer la qualité de la présentation : le résultat est le suivant

```

<layout>
  <root-layout width ="580 height "213" background-color=" black"/>
  <region id = "images_region" left="0%" top="0%" width="100%" height="100%"/>
</layout>

```

**Figure23 : L'adaptation qualitative**

Cette adaptation englobe aussi l'amélioration de la qualité de l'image par rapport à la couleur et à la résolution de l'image afin d'offrir une meilleure qualité de service.

### **- L'adaptation au réseau**

L'adaptation quantitative « la suppression de certaines données à transmettre » augmente le débit de connexion, et l'adaptation qualitative améliore la qualité de service

Améliorer la qualité de service revient, soit, à minimiser la compression au maximum, car l'inconvénient majeur de la compression est la perte de l'information. Mais ceci, augmente le volume du fichier transféré, tandis que le débit est bon. Soit à augmenter le nombre des images transférées par seconde afin d'augmenter la fluidité de la vidéo

### **- L'adaptation aux préférences de l'utilisateur**

L'adaptation qualitative pour un malvoyant revient à substituer la taille de police de caractère par une autre taille ou style. Pour un handicapé auditif, on augmente la qualité de l'image à la place de l'audio enlevée, afin qu'il puisse déchiffrer les séquences vidéo muettes.

### **- Exemple d'adaptation quantitative et qualitative**

Amel est anglophone

- Elle veut un document multimédia concernant une chanteuse de variétés
- La batterie de son PDA est faible ainsi elle décide qu'elle ne veut pas de contenu vidéo
- Elle souhaite cependant garder les contenus audio (extraits de la vidéo ou pas)



Avant l'adaptation:  
L'audio, la vidéo, l'image et le texte en français

Après l'adaptation:  
L'audio, l'image et le texte en anglais

**Figure 24 : Application de l'adaptation quantitative et qualitative**

### **5.5.8- L'adaptation globale**

L'adaptation globale « générale » c'est la succession des adaptations précédentes, afin d'avoir un document présentable selon le profil (matériel, réseau et les préférences de l'utilisateur), tout en respectant la sémantique du document original, et ce par la validation du contenu transformé par rapport à la DTD ou XMLSchema du document original.

## **6- Conclusion**

Dans cette partie, nous avons présenté une contribution d'adaptation et de transformation, qui met en œuvre plusieurs solutions d'adaptation selon ce qui a été étudié dans le chapitre précédent. La solution contient notamment:

- La description de contexte qui métaphoriquement représenté par l'ombre, afin de décrire le profil à l'aide du standard CC/PP.
- La négociation qui a été modélisée par un automate déterministe à états finis [KHO06]
- L'adaptation et la transformation qui a été décortiquée en plusieurs adaptations élémentaires, selon la description du contexte et du contenu. Cette adaptation est beaucoup plus caractérisée notamment par : le module de transformation de pixel en pourcentage, l'adaptation quantitative et qualitative.

Le regroupement de ces solutions particulières et l'assemblage de différentes techniques et outils d'adaptation sont regroupés dans une solution globale qui s'articule autour d'une architecture Client/serveur.

Dans ce qui suit, nous décrivons l'implémentation de notre architecture d'Adaptation et de Transformation de Présentations Multimédias (ATPM).

## **7 - L'architecture proposée**

L'architecture proposée s'articule autour d'une architecture Client/Serveur. Le Client est représenté par une application cliente, dont le rôle, est de gérer l'interface qui assure toutes les fonctionnalités de connexion et de communication avec les différents serveurs de l'architecture. Cette application contient également les différents moyens de visualisation de la présentation multimédia.

Le Serveur regroupe respectivement : Le Serveur de Profils et de Négociation et de Génération (SPNG), Le Serveur de contenu (SC), Le Serveur de Transformation et de Validation (STV), où chaque serveur s'acquitte d'une tâche bien précise.

### **7.1 - Principe de l'architecture ATPM**

L'architecture **Adaptation et Transformation de Présentations Multimédia (ATPM)** consiste à définir des entités qui coopèrent pour transmettre des contenus qui respectent les caractéristiques et les limites de l'environnement. Afin d'universaliser l'accès au contenu indépendamment de la variété des clients ainsi que leurs moyens, types et l'endroit d'accès.

ATPM assure l'adaptation et la transformation du document multimédia en fonction du contexte au moment de son utilisation.

ATPM inclut trois grandes parties pour réaliser l'adaptation qui sont : la description du contexte, la négociation et l'adaptation et la transformation de contenu demandé.

La description du contexte assure la formulation des caractéristiques de l'environnement et ses différents composants. Cette description, se fait entre le client, le serveur SIP et le SPNG, à travers les protocoles CC/PP, SIP et SDP.

La négociation [KHO 06] se fait à base de ce qui a été demandé (extrait) par rapport au contexte, afin d'avoir un contenu adaptable au profil final.

L'adaptation et la transformation fait l'objet de notre mémoire. Elle génère l'ensemble des règles extraites du profil, pour les appliquer au contenu demandé après la négociation, tout en respectant la sémantique et le sens de ce document.

### **7.2- Objectifs de l'architecture ATPM**

L'architecture devra remplir les objectifs suivants :

#### **- Assure la communication avec les utilisateurs :**

La communication est un préalable à toute solution d'adaptation. Que ce soit les requêtes des clients ou les informations sur les présentations, il faut mettre en place un

mécanisme utilisant un protocole de communication et d'initialisation des sessions qui puisse assurer l'émission et la réception de différents messages.

**- Gérer l'hétérogénéité des utilisateurs :**

Le concept d'adaptation est motivé par l'hétérogénéité des utilisateurs, ainsi l'architecture devra reconnaître et s'adapter à tout type d'utilisateur de la même manière.

**- Gérer l'hétérogénéité des contenus :**

L'architecture doit pouvoir gérer une diversité de présentations multimédias, du plus simple au plus complexe, et son adaptation doit se faire de manière transparente à l'utilisateur.

**- Gérer les profils utilisateurs :**

Notre architecture doit être capable de collecter les différentes informations concernant les clients leurs environnement et le contexte dans lequel ils se trouvent. Elle doit mettre ces informations sous une forme standard qui puisse être utilisée dans différents processus de l'adaptation.

Aussi, elle se doit de garder trace de ces profils et doit pouvoir les mettre à jour automatiquement.

**- Offrir à l'utilisateur le meilleur service possible :**

Le système doit être capable de satisfaire au mieux les requêtes des clients, en se rapprochant au plus près de leurs besoins et caractéristiques, et ce dans les meilleurs délais

**- Atteindre des performances optimales du système :**

La qualité du service ne doit pas être au détriment des performances du système. L'architecture doit délivrer le meilleur contenu possible à un utilisateur sans pour autant altérer le rendement du système. Ce qui entraînerait une dégradation du service pour les demandes ultérieures.

**7.3 - Schéma général de l'architecture TAPM**

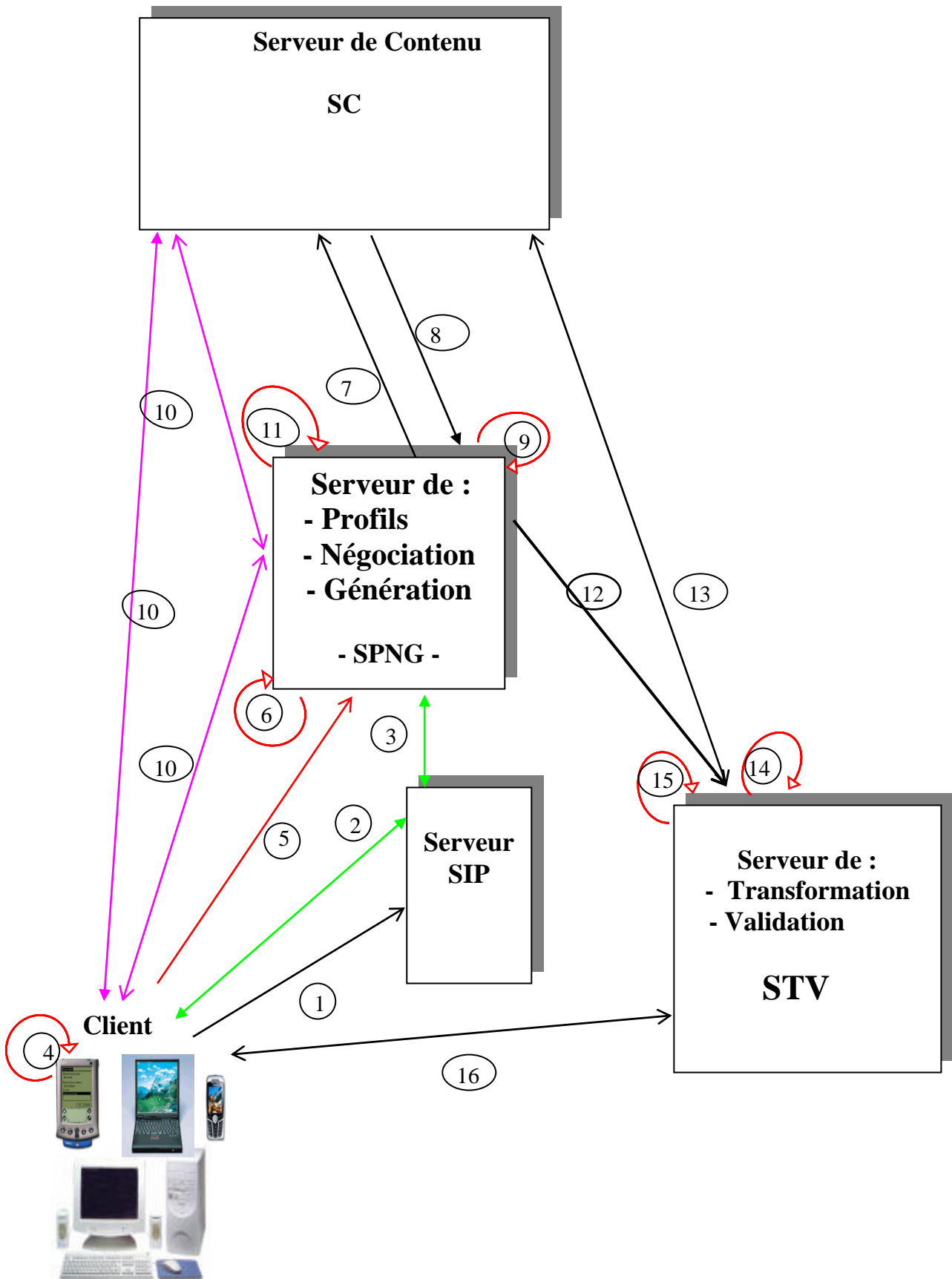


Figure25 : L'architecture Adaptation et Transformation d'une Présentation Multimédia

### **Schéma de parcours de l'adaptation**

1- Le client émet la requête " INVITE " au serveur SIP pour inviter le SPNG à ouvrir une session de communication.

2-3 L'échange des différents messages pour l'établissement de la session entre le client et SPNG à travers le serveur SIP en utilisant les protocoles SIP et SDP.

4- Formalisation du profil de l'utilisateur

5- Une fois la connexion est établie, le client envoie son profil au SPNG dans une requête CC/PP en utilisant les protocoles CC/PP, SIP SDP et un des protocoles de transmission UDP ou TCP.

6- Enregistrement ou mises à jour du profil de l'utilisateur dans la base des profils au niveau du SPNG.

7- Le SPNG envoie l'adresse URL de la présentation demandée au SC.

8- Le SC reprend sur la requête du SPNG par l'envoi de la présentation demandée.

9- Extraction des caractéristiques du document demandé, puis la comparaison avec le profil.

La comparaison engendre deux cas: la conformité étape 12 ou la non-conformité étape 10.

10- Le SPNG met le lien entre le client et SC pour entamer la négociation et la faisabilité d'arranger les deux côtés à travers le serveur SIP.

La négociation mène soit à la possibilité de l'adaptation (étape 11) ou à l'échec

11- le SPNG met à jour le profil dans la base de profils selon la négociation, d'un côté et d'un autre, il génère les règles de gabarit « template rules » qui servent la transformation par la suite.

12- Le SPNG envoie le résultat de la génération (les règles de gabarit) et la présentation du document multimédia au STV.

13- Elaboration d'une session de communication entre le STV et le SC sur les différents documents nécessaires pour le transfert et la visualisation du contenu demandé.

14- Le STV applique le module de conversion de Pixel en pourcentage, l'adaptation Quantitative et qualitative sur des feuilles de styles « XSLT et XSL FO », selon les règles de gabarit qui ont été générées par le SPNG.

15- Le STV valide la transformation de la présentation par rapport à la DTD ou au XMLSchema définissant la structure de données.

La validation donne l'un des deux résultats : soit la non-conformité du document (échec : fin de session), soit la conformité du document (valide : la réussite de la transformation « étape 16 »).

16 – L'envoi de la présentation multimédia demandée au client.

### **3.4 - Rôles des différentes entités de l'architecture ATPM**

#### **7.4.1 - L'application Cliente**

L'application cliente, est une interface intermédiaire entre le client et les différents serveurs de l'architecture ATPM. Elle gère le dialogue client/Serveur ; pour décrire, émettre le contenu de sa requête, ainsi que le traitement des réponses reçues et la description du profil. Elle est installée dans l'appareil du client contenant un ensemble de protocoles de communication notamment SIP, SDP, TCP, HTTP, CC/PP...etc.

#### **7.4.2 - Le Serveur de Contenu : SC**

Il contient l'ensemble des documents originaux à savoir: les documents multimédias, les documents de présentations « SMIL », les documents de validation DTD ou le XMLSchema et les feuilles de style, ainsi que les différentes possibilités d'arrangement et de modification qui peuvent être apportées sur les documents de présentation.

#### **7.4.3 - Le Serveur SIP :**

Les tâches principales de ce Serveur sont :

- Ouvrir, modifier et libérer les sessions entre les utilisateurs et le Serveur de profils et de Négociation et de Génération (SPNG).
- Localisation du terminal appelé.
- Analyse du profil et des ressources du destinataire.
- Négociation du type de média (voix, vidéo, données,...), et des paramètres de communication.
- Disponibilité de l'appelé : détermine si le poste appelé souhaite communiquer, et autorise l'appelant à le contacter.
- Etablissement et suivi de l'appel : avertit les parties appelant et appelé de la demande d'ouverture de session, gestion du transfert et de la fermeture des appels.

#### **7.4.4 - Le serveur de Profils et de Négociation et de Génération : SPNG**

Représente le cœur de l'architecture, il se charge des fonctionnalités suivantes :

##### **A - La gestion des utilisateurs :**

Qui inclut :

- L'identification et l'authentification des utilisateurs.
- L'ajout et la suppression des utilisateurs.
- La récupération des informations clients et leur mise sous forme de profils.
- La mise à jour automatique du profil.

L'utilisateur n'est pas censé transmettre tout le détail de son profil, sauf à la première connexion. Dans le cas de modifications, le client envoie uniquement les valeurs à modifier (dynamiques).

##### **B - L'extraction et la comparaison de différentes caractéristiques :**

SPNG commence l'extraction de différentes caractéristiques du profil et du document multimédia demandé, puis, il exécute le processus de comparaison à partir de ce qui a été extrait précédemment. De la comparaison résulte le degré de satisfaction et de conformité de l'offre par rapport à la demande.

##### **C - La négociation :**

SPNG assure la communication et la négociation sur la possibilité d'arrangement entre le client et le serveur de contenu détenant les différentes présentations multimédias.

##### **D - La génération de différentes règles de gabarit « template rules »**

Les règles de gabarit représentent les critères de base de la transformation extraites du profil utilisateur selon le résultat de la négociation

##### **E - La coordination entre les différentes entités du système.**

SPNG est en relation avec les différentes entités de l'architecture ATPM, il relie le client avec Serveur de contenu (SC), ainsi que ce dernier avec le Serveur de transformation et de Validation (STV).

**7.4.4.1- Algorithme de fonctionnement de SPNG**

Début

- INVITE {invitation envoyée par le client au SPNG à travers le Serveur SIP}  
   {Etablissement de connexion, utilisant le protocole SIP et SDP}
- Si utilisateur = nouveau alors
  - {pour la première fois }
  - Enregistrement dans la base de profil ;
  - Affecter un nom d'utilisateur et un mot de passe à ce profil ;
- Sinon
  - Authentification;{le nom et le mot de passe envoyés par l'utilisateur au SPNG}
  - Lire (Nom\_utilisateur, Mot\_passe) ;
  - Si authentification = faux alors
    - SPNG envoie la réponse SIP « 401 ERROR » ;
    - Répéter;{boucle : le client doit être entré ses informations à nouveau}
  - Sinon
    - Message SIP « 200 OK » ; {authentification valide}
    - Message ACK {confirmer l'établissement de la connexion par le client}
    - Le client envoie son profil au serveur SPNG utilisant le protocole SIP, SDP et l'un des protocoles de transmission UDP ou TCP ;
    - Si profil = modifier alors {le profil est modifié}
      - Mise à jour du profil dans la base de profils ;
    - Fsi ;
- Fsi ;
- Fsi ;
- Formulation de profil sous forme de profil CC/PP au niveau SPNG;
- SPNG extrait l'URL du document demandé par l'utilisateur;
- SPNG envoie l'URL au SC ;
- SC envoie le document spécifié dans l'URL selon :
  - 1 : URL envoyée par le SPNG avant la négociation ;
  - 2 : URL envoyée par le SPNG après négociation ; {document similaire}
- SPNG reçoit la présentation multimédia spécifiée dans l'URL envoyée par le SC ;
- SPNG extrait l'ensemble des caractéristiques constituant le document multimédia ;  
   {l'encodage, le type des objets medias, la langue, les dimensions d'affichage...etc}
- SPNG compare ce qui a été extrait du document demandé par rapport au profil ;
- Si la comparaison = vrai alors {demande faisable et conforme au profil}
  - Envoyer la présentation à l'utilisateur pour la visualisation
- Sinon
  - La négociation ;
  - Le SPNG entame la négociation avec le client sur la possibilité d'arranger et de

```

satisfaire le besoin, tout en consultant le serveur de contenu;
{l'encodage, le type des objets medias, la langue, les dimensions d'affichage..etc}

Si la négociation = vrai alors {il y a un compromis}
  - SPNG génère les règles de gabarit «template rules» qui vont être appliqué
    sur les feuilles de présentation;
  - SPNG met à jour le profil de l'utilisateur selon le résultat de la négociation ;
  - SPNG envoie l'ensemble de documents concernant la présentation
    (SMIL, DTD) et les règle de gabarit ainsi le profil utilisateur au serveur de
    transformation et de validation : STV ;

Sinon
  Ecrire échec de négociation ;
  Message SIP : Bye {Fin de Session};
Fsi
Fsi
Fin {programme}

```

### **7.5 - Le Serveur de Transformation et de Validation : STV**

STV s'occupe de transformer les valeurs, de taille et de placement spatial de tous les médias, en pourcentage par rapport aux dimensions globale de l'adaptation.

A partir des informations envoyées par le SPNG, le STV procède à la transformation par l'application des règles de gabarit sur les feuilles de styles « XSLT » pour visualiser la présentation SMIL spécifiée dans l'URL. L'adaptation et la transformation consistent en l'ensemble de modifications quantitative et qualitative apportées sur le document de présentation

STV utilise différents outils nécessaires pour l'adaptation tels que les navigateurs, le filtrage d'objets, les logiciels d'encodage, les traducteurs universels, les feuilles de style, module de conversion de pixel en pourcentage... etc.

Après la transformation, le STV passe à la phase de validation du résultat selon la DTD du document original contenu dans le serveur de contenu. Le résultat peut mener à deux cas : le premier, à un document valide et sémantiquement correcte, et à ce moment le résultat sera transféré au client bénéficiaire de l'accès. Le deuxième cas le document n'est pas valide et sémantiquement n'est pas correct, donc le résultat a échoué, et le client sera avisé.

### **7.5.1- Optimisation dans l'envoi du Contenu multimédia**

Dans un souci de minimisation du temps de réponse, le client doit pouvoir visualiser la présentation avant même la fin de l'adaptation.

Pour cela, dès le début de l'adaptation des médias, le serveur de transformation et de validation crée un flux multimédia qu'il envoie au serveur de contenu. Au même moment le STV envoie l'adresse RTSP de ce flux au client qui commence à visualiser le contenu multimédia transformé.

De ce fait, l'adaptation et l'envoi du résultat se font en parallèle. Pour réaliser cela, il est impératif de synchroniser les processus de transformation des médias, pour arriver à faire concorder les moments d'encodage et d'envoi du média, avec celui où il doit être joué chez le client.

### **8- Algorithme de parcours de l'architecture ATPM**

**Déclaration :** Les serveurs: SIP, SPNG, STV ;  
 L'application Cliente ;  
 Les Protocoles : SIP, SDP, IP, TCP, UDP, HTTP, RTSP,  
 Les Standards : XML, SMIL, SVG, RDF, CC/PP, SXMLT ;  
 Les Modules : Module de conversion de pixel en pourcentage ;  
 Module de Filtrage, d'adaptation quantitative, d'adaptation  
 Qualitative, de traduction ;

#### **Début**

```

- INVITE {invitation envoyée par le client au serveur SIP pour inviter le Serveur SPNG}
          {Etablissement de connexion, utilisant le protocole SIP, SDP}
- Echange des différents messages pour l'établissement de la session entre le client et
  SPNG à travers le serveur SIP en utilisant les protocoles SIP et SDP.
Si Connexion = faux alors
  - Echec de connexion selon :
    - Serveur indisponible ;
    - Serveur introuvable {fausse adresse du serveur} ;
Sinon
  - Connexion établie ;
    Si utilisateur = nouveau alors
      {pour la première fois }
      - Enregistrement dans la base de profil
      - Affecter un nom d'utilisateur et un mot de passe à ce profil;
    Sinon
  
```

```

-Authentification; { le nom et le mot de passe envoyés par l'utilisateur au
  SPNG }
- Lire (Nom_utilisateur, Mot_passe) ;
Si authentication = faux alors
  -SPNG envoie la réponse SIP « 401 ERROR » ;
  -Répéter;{boucle : le client doit être entrer ses informations à nouveau}
Sinon
  - Message SIP « 200 OK » ; {authentification valide}
  - Message ACK {confirmer l'établissement de la connexion par le client}
  - Le client envoie son profil au serveur SPNG utilisant le protocole SIP,
    SDP et l'un des protocoles de transmission UDP ou TCP ;
    Si profil = modifier alors {le profil est modifié}
      -Mise à jour du profil dans la base de profils ;
    Fsi ;
  Fsi ;
Fsi ;

- Le client envoie son profil au serveur SPNG utilisant les protocoles SIP, SDP et l'un des
  protocoles de transmission UDP ou TCP ;
- Formulation de profil sous forme de profil CC/PP au niveau SPNG;
  {Le profil est métaphoriquement modélisé par l'ombre qui change de forme aux
  changements journaliers apportés au utilisateur}

- SPNG extrait l'URL du document demandé par l'utilisateur;
- SPNG envoie l'URL au SC ;
- SC envoie le document spécifié dans l'URL selon :
  1 : URL envoyée par le SPNG avant la négociation ;
  2 : URL envoyée par le SPNG après négociation ; {document similaire}

- SPNG reçoit la présentation multimédia spécifiée dans l'URL envoyée par le SC ;
- SPNG extrait l'ensemble des caractéristiques constituant le document multimédia ;
  {l'encodage, le type des objets medias , la langue, les dimensions d'affichage...etc}

- SPNG compare ce qui a été extrait du document demandé par rapport au profil ;

Si la comparaison = vrai alors {demande faisable et conforme au profil}
  - Envoyer la présentation à l'utilisateur pour la visualisation
Sinon
  - La négociation ;
  - Le SPNG entame la négociation avec le client sur la possibilité d'arranger et de
    satisfaire le besoin, tout en consultant le serveur de contenu;
    {l'encodage, le type des objets medias, la langue, les dimensions d'affichage ..etc}

Si la négociation = vrai alors {il y a un compromis}
  - SPNG génère les règles de gabarit «template rules» qui vont être appliquées
    sur les feuilles de présentation;

```

```

    - SPNG met à jour le profil de l'utilisateur selon le résultat de la négociation ;
    - SPNG envoie l'ensemble de documents concernant la présentation
      (SMIL,DTD) et les règle de gabarit ainsi le profil utilisateur au serveur de
      transformation et de validation : STV ;

    Sinon
      Ecrire échec de négociation ;
      Message SIP : Bye {Fin de Session};
    Fsi
  Fsi

- Le STV reçoit les règles de gabarit, la présentation SMIL et la DTD ;
- Le STV sélectionne les médias qui seront utilisés lors de la présentation finale ;
- Le fichier est parcouru (comme pour la création de version), et transforme les objets
  sélectionnés selon les critères et règle de gabarit générés ;

- STV procède la transformation ;

- Applique l'algorithme de conversion de pixel en pourcentage,
- Appliquer l'adaptation quantitative; {enlèvement de différents objets médias qui ne
  correspond pas}
- Appliquer l'adaptation qualitative sur ce qui a été donné par de l'adaptation
  quantitative ;
- Le STV valide le résultat obtenu de la transformation par rapport à la DTD ou
  XMLSchema
Si la validation = vrai alors
  {Document valide et sémantiquement correcte}
  - envoyer la présentation au client ;
  - Optimisation dans l'envoi du Contenu multimédia ;
Sinon
  {Document non valide}
  Aviser le client de l'échec de l'adaptation ;
  Fin de session ;
Fsi
Fsi
Fin. {Fin du programme}

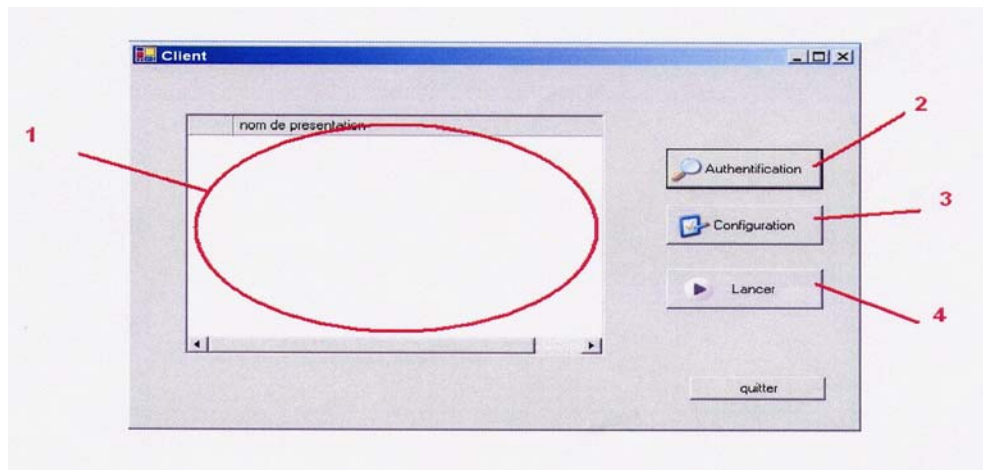
```

## 9- Exemple : d'application

L'exemple suivant décrit les différentes étapes du processus d'adaptation et de transformation d'une présentation multimédia selon l'algorithme précédent :

### 1 - Au niveau du Client :

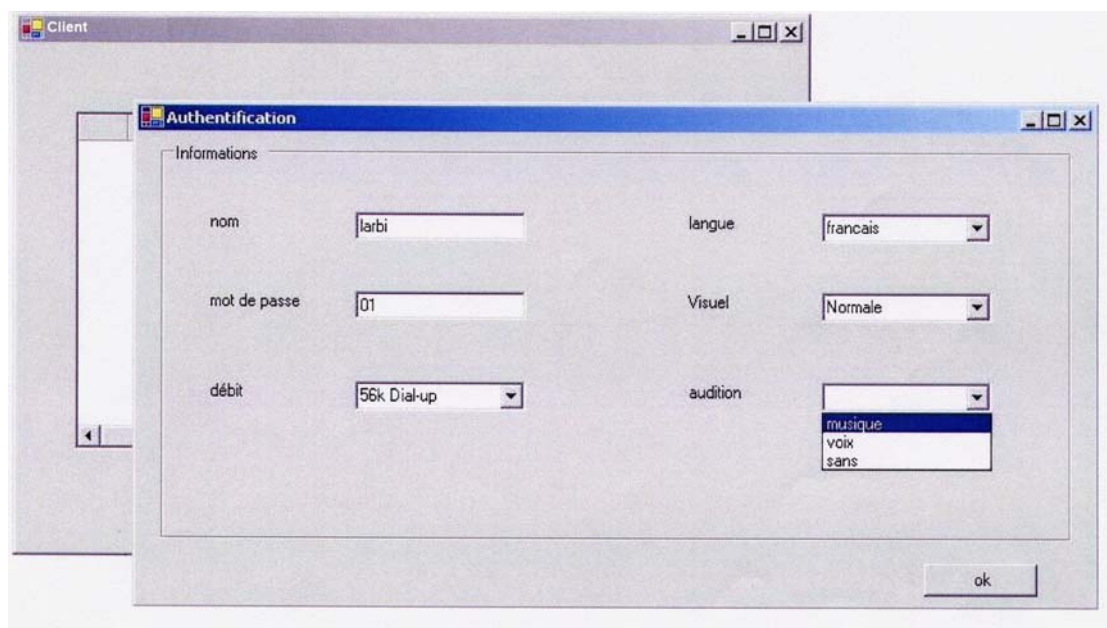
L'interface Client s'articule sur quatre aspects:



**Figure 26 : Interface client**

#### **a- L'authentification**

Avant de faire une requête, le client doit tout d'abord se connecter au SPNG, en cliquant sur le bouton « Authentification », afin d'entrer ses informations d'authentification, ses caractéristiques, et ses préférences.



**Figure 27 : Authentification du Client et profil**

## b- La configuration

La configuration des différents paramètres de connexion, en cliquant sur le bouton configuration.

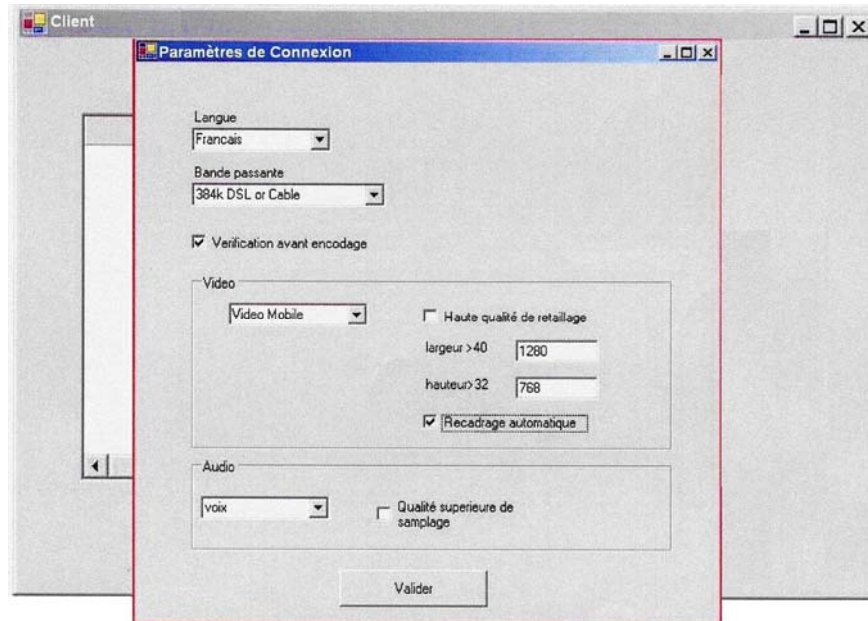


Figure 28 : Configuration des paramètres de connexion

## c- La sélection de la présentation demandée

Une fois l'Authentification et la configuration effectuées, le client peut sélectionner la présentation souhaitable, puis il clique sur le bouton « Lancer » pour commencer la visualisation.

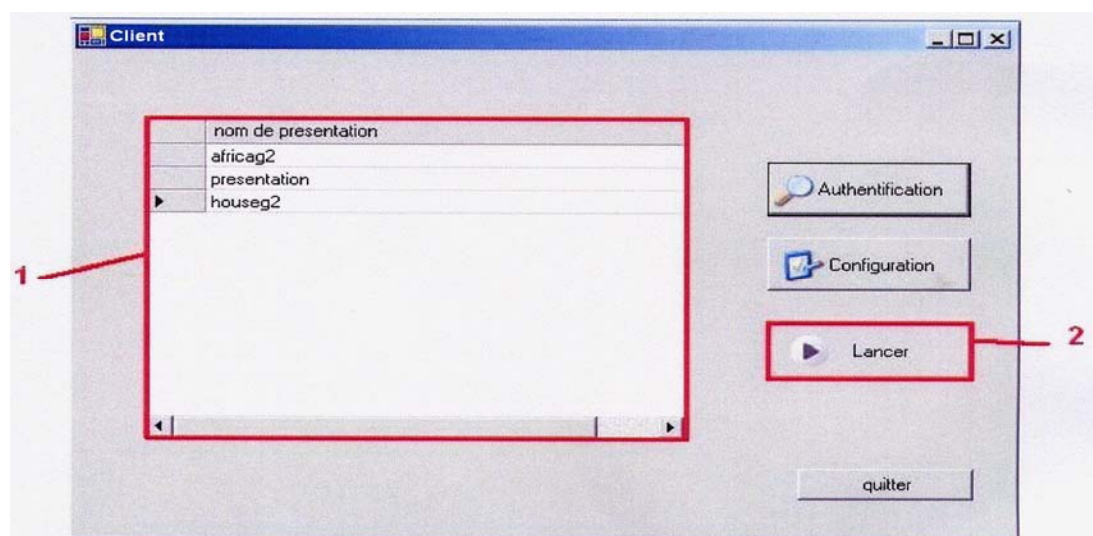
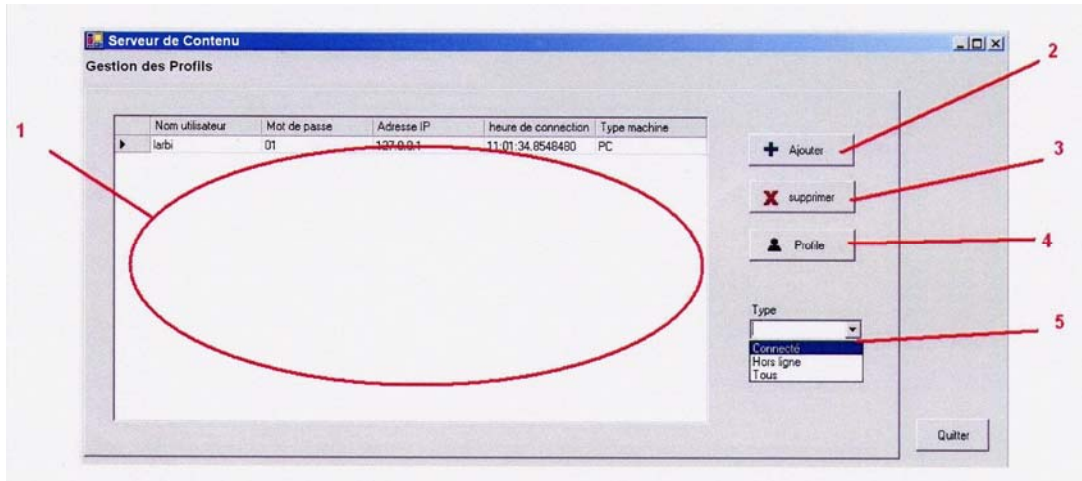


Figure 29 : Sélection de la présentation

## 2- Au niveau du SPNG :

### a- Gestion des profils

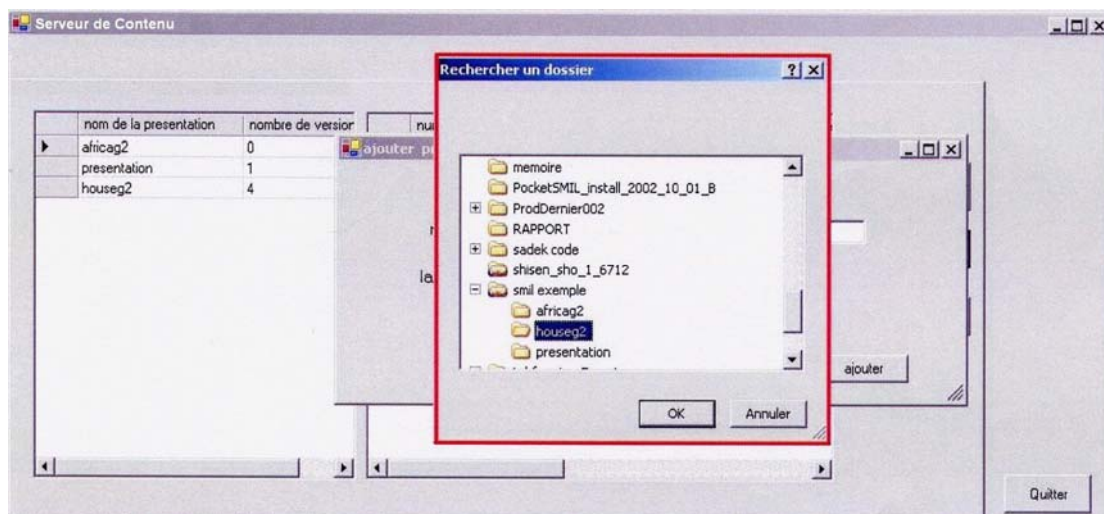
La base des profils est gérée par l'interface suivante :



**Figure 30 : Gestion des Profils**

- 1 : Représente un tableau sur lequel s'affiche l'ensemble des utilisateurs.
- 2 : Bouton permettant de rajouter un utilisateur dans la base de données des utilisateurs
- 3 : Bouton permettant de supprimer un utilisateur de la base de profils.
- 4 : Bouton permettant d'afficher le profil de l'utilisateur.
- 5 : Liste déroulante permettant de choisir le type des utilisateurs à afficher (connecté, hors ligne, tous).

### b- Gestion des présentations multimédia



**Figure 31 : Sélection de la présentation multimédia**

Le SPNG, extrait les différentes caractéristiques des présentations multimédia afin d’entamer la négociation et la génération des règles de gabarit.

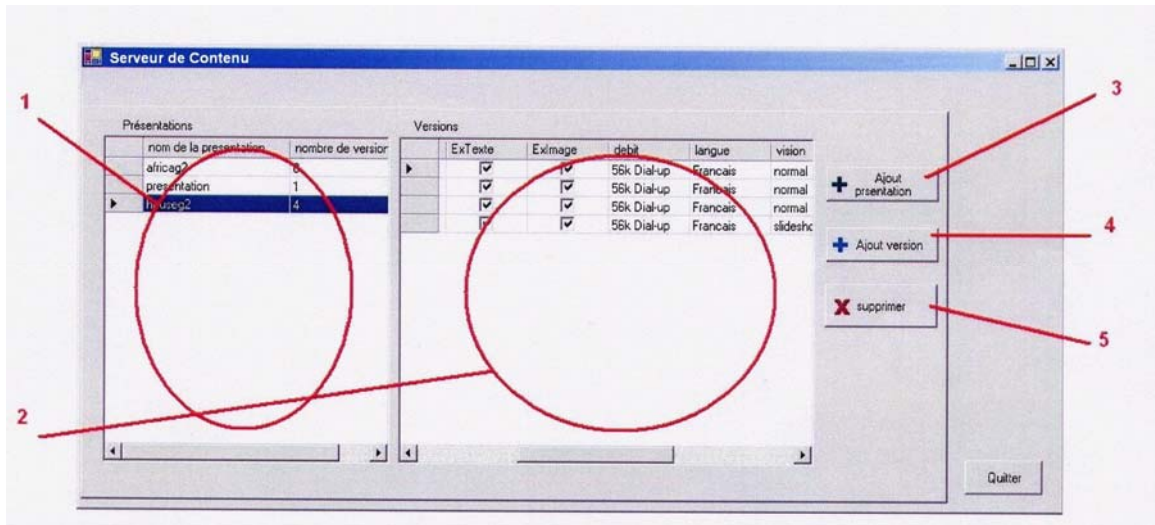


Figure 32 : Extraction des caractéristiques de la présentation sélectionnée

### 3- Au niveau du STV :

#### a- Application de l’algorithme de conversion de pixel en pourcentage.

STV applique l’algorithme de conversion de pixel en pourcentage. L’exemple est donné dans la Figure suivante :

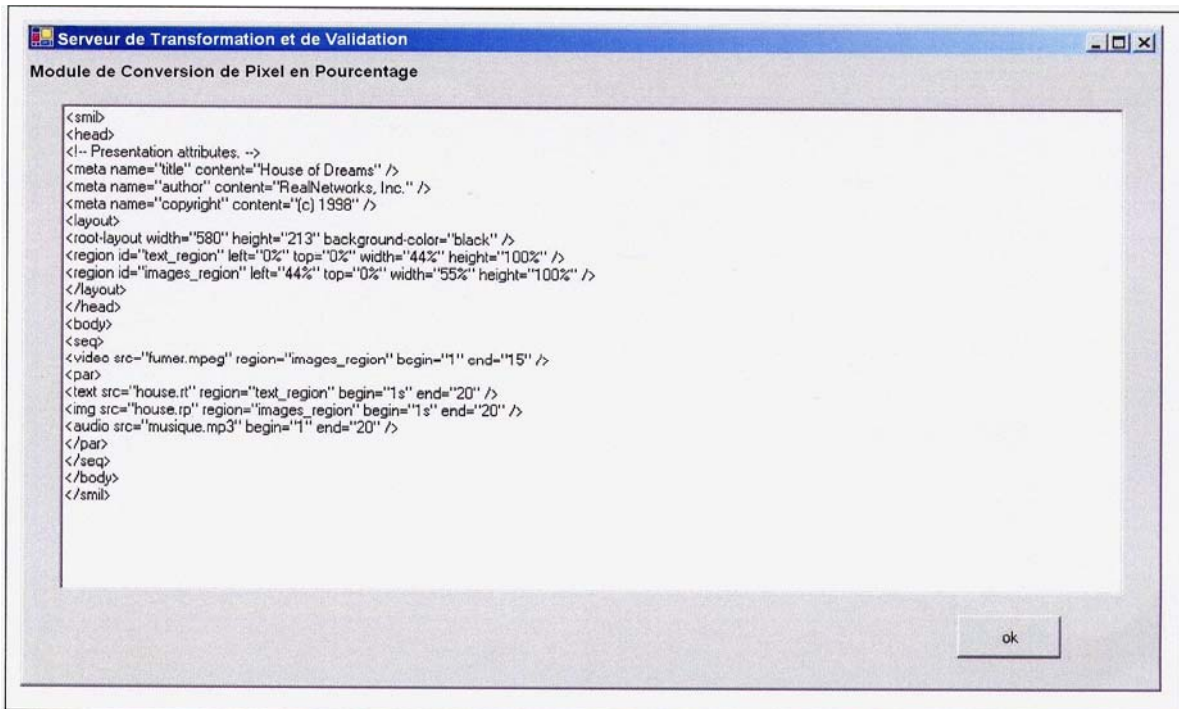


Figure 33 : Transformation de pixel en pourcentage

### b- Application de l'adaptation quantitative et qualitative

L'adaptation quantitative et qualitative s'effectuent selon les règles de gabarit envoyées par le SPNG.

### c- Visualisation de la présentation



Figure 34 : Visualisation de présentation multimédia transformée

## **10 - Conclusion**

L'architecture d'adaptation et de transformation des présentations multimédias (**ATPM**), contribue à la transformation de contenu pour pallier les limitations des utilisateurs et les contraintes de leur environnement.

**ATPM** commence le processus de transformation par la formulation du profil de l'utilisateur (description de contexte), qu'est métaphoriquement modulé par l'ombre de l'utilisateur, et leur formalisation s'effectue dans le cadre de CC/PP. A base de ce profil et au contenu demandé, on entame le processus de négociation sur la faisabilité de ce qui a été demandé. Ceci, caractérise l'étape qui succède la première phase (la description de contexte), dont l'objectif est de trouver un compromis pour arranger l'utilisateur. Dans ATPM la négociation est modélisée par un automate déterministe à états finis [KHO 06], et sa réalisation s'effectue par l'utilisation de différents protocoles de négociation et de communication tels que : SIP, SDP, TCP ou UDP, afin de générer les règles de la transformation. Cette dernière représente l'étape finale dans l'architecture ATPM. Dans cette phase on adapte et transforme le contenu qui métaphoriquement évalué quantitativement et qualitativement, par le filtrage et l'enlèvement des objets médias qui ne seront pas exploités pour ce profil, ou d'appliquer le module de conversion de pixel en pourcentage...etc. Toutes ces adaptations et transformations sont engendrées à partir des règles de gabarit générées par la négociation, elles sont effectuées à l'aide des feuilles de styles XSLT et XSLT-FO. ATPM finalise son processus d'adaptation et de transformation par la validation et la conformité (certitude) du résultat transformé par rapport à la présentation originale à travers sa DTD ou XMLSchema.

# Conclusion

## **Conclusion et Perspectives**

L'avancement technologique (le contenu multimédia et les moyens de communications (appareils et réseaux)), ainsi que les préférences des utilisateurs constituent autant de facteurs à l'origine de la complexité et l'hétérogénéité de l'environnement actuel. Cette complexité rend mal tout genre d'adaptation et de compatibilité entre les entités constituant cet environnement. L'adaptation et la transformation de contenu constituent un moyen de surmonter les obstacles de visualisation d'une présentation multimédia dans ces environnements hétérogènes.

Notre perspective majeure, est d'universaliser la transformation, afin qu'elle prenne en charge automatiquement les difficultés et les souhaits de l'utilisateur tels que :

- L'adaptation pour un utilisateur aveugle, qui consiste à transformer les images ou les textes en audio.
  
- L'adaptation pour un utilisateur sourd, qui consiste à convertir l'audio en texte ou succession d'images fixes ou animées. Cette transformation nécessite une nouvelle région dans le document SMIL afin de présenter la barre de défilement du texte converti.
  
- L'adaptation de la langue, qui nécessite l'utilisation d'un traducteur universel, pour qu'il puisse traduire un texte écrit d'une langue originale vers une autre langue souhaitable, ou l'interprétation d'un audio vers un autre audio d'une autre langue différente de l'originale.

Reference:

- [ALT 01] Murray Altheim. Modularization of XHTML.  
W3C Recommendation 10 April 2001  
<http://www.W3C.org/TR/2001/REC-xhtml-modularization-20010110/>
- [AMA 02] Bernd Amann et Philippe Rigaux, Comprendre XSLT, O'Reilly, 28 février 2002
- [AYA 01] Jeff Ayaras, Synchronized Multimedia Integration Language(SMIL 2.0), 2001  
W3C recommendation 07.08.2001 <http://www.W3c.org/TR/2001/REC-smil20-20010807/>
- [BAS 06] Fabien Basmaison , Antoine Cailliau , Jean Remy Duboc , CSS : Le guide  
Complet, Micro Application, 22 août 2006
- [BIL 05] Marius Bilasco, Jérôme Gensel, Marlène Villanova-Oliver  
STAMP: A Model for Generating Adaptable Multimedia Presentations,  
Edition Springer Netherlands 2005
- [BUL 05] Dick Bulterman Synchronized Multimedia Integration Language (SMIL 2.1)  
W3C Recommendation 13 December 2005  
<http://www.w3.org/TR/2005/REC-SMIL2-20051213/>
- [CC/PP 04] Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0  
W3C Recommendation 15 January 2004, <http://www.w3.org/TR/CCPP-struct-vocab/>
- [DAN 00] Koch, Daniel, Html 4. Xml-Xhtml, Micro Application, 06/02/2000
- [DIC 05 a] Dick C.A SMIL 2.1 Mobile Profile ,2005  
<http://www.w3.org/TR/2005/REC-SMIL2-20051213/smil21-mobile-profile.html>
- [DIC 05 b] Dick C.A, SMIL 2.1 Extended Mobile Profile, 2005  
<http://www.w3.org/TR/2005/REC-SMIL2-20051213/smil21-extended-mobile-profile.html>
- [DJA 06] ZEROUAL Djazia, Implémentation d'un environnement parallèle pour la  
compression d'image à l'aide des fractales, Mémoire de Magister  
Université de Batna/Département d'Informatique, 2006
- [EIS 02] J. Eisenberg , SVG Essentials, O'Reilly, 2002.

- [HER 07] Ivan Herman, Ralph Swick ,Dan Brickley Resource Description Framework (RDF), 2007/01/29, <http://www.w3.org/RDF/>
- [HAN 98] M. Handley, V.Jacobson SDP: Session Description Protocol April 1998, <http://www.ietf.org/rfc/rfc2327.txt>
- [HAN 99] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg, SIP: Session Initiation Protocol, March 1999, <http://www.ietf.org/rfc/rfc2543.txt>
- [HAN 00] M. Handley, C. Perkins, E. Whelan, Session Announcement Protocol October 2000, <http://www.ietf.org/rfc/rfc2974.txt>
- [HER 95] Eric Van Herwijnen, SGML pratique, Intern. Thomsom Publicat, 19 mai 1995
- [HOL 03] G. Ken Holman Definitive Xsl-Fo, Prentice Hall PTR, Mars 2003
- [KAL 99] Karl Dubost Spécification du modèle et la syntaxe du cadre de description des ressources (Resource Description Framework ou RDF), Recommandation du W3C 22/02/1999 <http://www.la-grange.net/w3c/REC-rdf-syntax/>
- [KHO 06] H.KHORSI , La négociation de services dans le Web, Mémoire de Magister, USTHB, 2007
- [LAY 99] Nabil Layaïda Adaptabilité : pistes d'étude pour la définition d'une infrastructure d'accès au contenu multimédia pour des machines hétérogènes1  
Projet Opéra, INRIA <http://www.opera.inrialpes.fr/people/Nabil.Layaida/publi/Adaptabilite.pdf> ,1999
- [LAY 04] Nabil Layaïda, Tayeb Lemlouma, Vincent Quint  
Adaptation et multimédia mobile sur le Web ,  
<http://wam.inrialpes.fr/publications/2004/Mcube2004.html>, 2004
- [LAY 05 a] Nabil Layaïda. SMIL 2.1 Language Profile , INRIA  
<http://www.w3.org/TR/2005/REC-SMIL2-20051213/smil21-profile.html>
- [LAY 05 b] Nabil Layaïda, Tayeb Lemlouma, Vincent Quint  
NAC, une architecture pour l'adaptation multimédia sur le web  
<http://wam.inrialpes.fr/publications/2005/TSI-NAC.pdf>

- [LEC 00] D. Lecomte, Les normes et les standards du multimédia - xml, mpeg-4 et 7, mpeg-21, jpeg 2000, MP3, web3d, wap... et les autres, Dunod, 2<sup>ième</sup> édition-2000
- [LEM 04 a] Tayeb LEMLOUMA Architecture de Négociation et d'Adaptation de Services Multimédia dans des Environnements Hétérogènes 09 juin 2004  
<http://opera.inrialpes.fr/people/Tayeb.Lemlouma/Papers/SAITechnicalDescription.pdf>
- [LEM 04 b] SVG Tiny (SVGT) et XHTML basic : langages pour des composants JSF mobiles Tayeb Lemlouma Novembre 2004, Projet WAM, INRIA Rhône Alpes  
<http://opera.inrialpes.fr/people/Tayeb.Lemlouma/Papers/MobileComponents.pdf>
- [LOP 05] Céline Lopez-Velasco, Marlène Villanova-Oliver, Jérôme Gensel, Hervé Martin, Adaptabilité à l'utilisateur dans le contexte des services Web, Atelier MASIW EGC 2005, Paris, 18 Janvier 2005, [www-lsr.imag.fr/masiw2005/](http://www-lsr.imag.fr/masiw2005/)
- [MIC 00] Alain Michard, XML: langage et applications - 2ème édition, Eyrolles, 12/2000
- [MOB 03] Mobile SVG Profiles: SVG Tiny and SVG Basic,  
<http://www.w3.org/TR/2003/REC-SVGMobile-20030114/>.
- [ORA 99] Spécification du modèle et de la syntaxe du cadre de description de ressource (RDF); Ora Lassila, Ralph Swick; recommandation du 22 février 1999:  
<http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>
- [QUI 01] Vincent Quint , Outils pour les documents électroniques, recherche et applications rapport d'activité INRIA Rhône-Alpes, 2001  
<http://www.inria.fr/valorisation/rencontres/web-semantique/panorama/opera.fr.html>
- [PER 02] Colin Perkins, RTP: Audio and Video for the Internet Addison-Wesley Professional, 30 Juin 2002
- [PAU 97] Ricoeur Paul ,La métaphore vive , Seuil, 2 avril 1997
- [QUI 07] Liam Quin, The Extensible Stylesheet Language Family (XSL), 05/02/2007  
<http://www.w3.org/Style/XSL/>
- [ROI 98] Cécile Roisin. Authoring structured multimedia documents. In Conference on Current Trends in Theory and Practice of Informatics, 1998

- [ROS 02] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M.Handley, E. Schooler. 'SIP : Session Initiation Protocol'. Dynamicsoft, ACIRI, AT & T, Columbia 2002.
- [SAR 03] Manuelli Sara, RSVP, Rotovision -2003
- [SEB 04] Laborie Sébastien Adaptation de documents multimédia : Approche sémantique de la dimension spatiotemporelle des documents SMIL 30 juin 2004  
[www.inria.fr/rapportsactivite/RA2004/exmo2004/bibliography.html](http://www.inria.fr/rapportsactivite/RA2004/exmo2004/bibliography.html)
- [SCH 03] H.Schulzrinne, S.Casner R.Frederick,V. Jacobson ,RTP: A Transport Protocol for Real-Time Applications, July 2003 : RTCP  
<http://www.rfc-editor.org/rfc/rfc3550.txt>
- [SUS 00] Jean-françois Susbielle, Inetrnet, multimédia et temps réel, Eyrolles 2000.
- [THO 02] Jean-Jacques Thomasson, Schémas XML, Eyrolles, 19 novembre 2002
- [W3C 00] Worldwide Web Consortium, Synchronized Multimedia Integration Language Smil 1.0 Specification, Iuniverse Inc - Novembre 2000
- [W3C 02] W3C, Recommended DTDs to use in your Web document, 2002  
<http://www.w3.org/QA/2002/04/valid-dtd-list.html#DTD>
- [W3C 03] Scalable Vector Graphics (SVG) 1.1 Specification, W3C Recommendation 14 January 2003. <http://www.w3.org/TR/SVG11/>.
- [W3C 04] Les profils composites de capacités/préférences (CC/PP) : structure et vocabulaires 1.0 Recommandation du W3C du 15 janvier 2004  
<http://www.w3.org/TR/2004/REC-CCPP-struct-vocab-20040115/>
- [W3C 05] W3C Recommendation 07 January 2005, Synchronized Multimedia Integration Language (SMIL 2.0) - [Second Edition]  
<http://www.w3.org/TR/2005/REC-SMIL2-20050107/> , 2005
- [W3C 06] World Wide Web Consortium, 2006, <http://www.w3.org>.
- [WON 00] Clinton Wong, HTTP précis et concis, O'Reilly - 1 janvier 2000