

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université M'hamed Bougara - Boumerdès



Faculté des Sciences  
Département d'Informatique

**MEMOIRE**

*Pour l'obtention du diplôme de*

**MAGISTER**

***Option : Spécification logiciel et traitement de l'information***

**Thème :**

**Adaptation des présentations multimédias pour  
une optimisation de la consommation  
d'énergie**

*Présenté par*

**REZOUG Abdellah**

*Le jury composé de :*

<b>M<sup>r</sup> M.Mezghiche</b>	<b>Professeur</b>	<b>UMBB</b>	<b>Président</b>
<b>M<sup>r</sup> A.Abdelli</b>	<b>MCA</b>	<b>USTHB</b>	<b>Promoteur</b>
<b>M<sup>r</sup> M.Ahmed-Nacer</b>	<b>Professeur</b>	<b>USTHB</b>	<b>Examineur</b>
<b>M<sup>r</sup> M.Benchaiba</b>	<b>MCA</b>	<b>USTHB</b>	<b>Examineur</b>

2007/2008

# Remerciements

Je voudrais remercier mon directeur de mémoire, Monsieur ABDELLI Abdelkrim, pour la confiance qu'il m'a accordé en acceptant de proposer et diriger ce travail, pour les précieux conseils et encouragements, pour ses remarques attentives, la gentillesse et la patience qu'il a manifesté à mon égard, et pour avoir mis à ma disposition la documentation nécessaire à l'accomplissement de ce travail.

Je remercie également, très vivement le Professeur MEZGHICHE Mohamed de l'UMBB, responsable de l'école doctorale et du laboratoire LIFAB, pour les efforts et le temps qu'il a consacré et continue de consacrer aux étudiants.

Je tiens à remercier les membres de jury, Mr. AHMED NACER, et Mr. BENCHAIABA, d'avoir accepté de juger ce modeste travail.

Merci à tous mes amis qui m'ont encouragé et soutenu tout au long de ce travail, en particulier : les amis de ma promotion d'ingénieurs de l'année 2007, ainsi que ceux de ma promotion de magister.

Enfin, je remercie tous ceux qui, de près ou de loin, ont bien voulu m'encourager pour que ce travail puisse être achevé.

## Dédicaces

A ma chère mère pour son affection et son soutien indéfectible, qui ne cesse de prier pour nous,

A mon cher père qui m'a toujours encouragé et soutenu et qui fait tout pour nous rendre heureux,

A mes chers frères Abdenour, Amar et mes chères sœurs Djamila, Hafida, Souad.

A tous les membres de ma famille,

A tous mes amis,

A tous mes enseignants du primaire jusqu'à l'université,

Je dédie ce modeste travail.

# Résumé

Dans un système d'adaptation de documents multimédia, un document subit une série de transformations afin qu'il puisse être exécuté sur un équipement terminal. Ces transformations permettent de rendre le document compatible avec les caractéristiques spécifiques du client (type codec vidéo, taille écran, résolution etc.), comme la conversion d'une image de format JPEG au format GIF, la conversion d'une vidéo à une bande sonore etc.

De nos jours, les applications multimédias investissent le Web et la téléphonie mobile et occupent une grande place dans la vie quotidienne. Cependant, cette tendance a eu comme conséquence une charge supplémentaire importante sur la consommation de la ressource énergétique et par conséquent sur la durée de vie de la batterie. Dans ce mémoire, nous explorons les techniques à même de remédier à ce problème.

Il est question de discuter des fonctionnalités et options offertes par le langage standard de spécification de documents multimédia SMIL (Synchronized Multimedia Integration Language), et également des mécanismes d'économie d'énergie définis dans la littérature afin de dégager une architecture optimisée pour la fourniture de service multimédia.

**Mots clés :** *Adaptation des présentations multimédia, optimisation d'énergie, SMIL, les appareils mobiles, architecture de fourniture de service.*

# Abstract

In a multimedia presentation adapting system, a document undergoes a series of transformations so that it can be executed on a terminal device. These transformations allow rendering a new document which is adapted to the specific characteristics of the terminal (standard video codec, size screen, resolution etc). This implies, for instance, to convert a JPEG image format to GIF, or to reduce the content of a video to its embedded sound etc.

Nowadays, multimedia applications are investing the Web as well as mobile telephony. However, this trend results an additional cost on energy consumption and hence over the battery life expectancy. In this thesis, we explore different techniques defined in the literature to cure this problem.

We discuss the functionalities and options offered by the standard language of multimedia documents specification SMIL (Synchronized Multimedia Integration Language), and also energy saving mechanisms defined for this purpose. This makes it possible to put forward an architecture for provisioning and adapting multimedia presentation with little resource requirements.

**Key words:** *multimedia presentations adaptation, power optimization, SMIL, mobile devices.*

في نظام تكيف المستندات متعددة الوسائط, يتعرض المستند لسلسلة من التغييرات لكي يصبح قابلاً للتنفيذ على جهاز محمول. هذه التغييرات تمكن من جعل المستند متوافق مع خصائص الزبون (نوع الكودك, أبعاد الشاشة, الدقة. الخ), مثل تحويل شريط فيديو إلى شريط سمعي, تحويل صورة من قياس JPEG إلى قياس GIF, الخ.

في أيامنا هذه, البرامج متعددة الوسائط دخلت عالم الواب و الهواتف المحمولة, وأخذت مكانة كبيرة في الحياة اليومية, لكن, هذا التوجه يحمل كنتيجة حمل إضافيا مهما على استهلاك المورد الطاقوي وذلك معناه مدة اقل من حياة البطارية.

في هذه المذكرة, نفحص التقنيات التي تعالج هذا المشكل, لذلك بجدد بنا مناقشة الخيارات التي تمنحها لغة تخصيص المستندات متعددة الوسائط SMIL, وآليات حفظ الطاقة الموجودة, من اجل تحرير هندسة مختصرة لنظام تموين خدمات متعددة الوسائط. **كلمات مفتاحية:** تكيف المستندات متعددة الوسائط, اختصار الطاقة, SMIL, الهواتف المحمولة.

---

# Table des matières

Remerciement .....	i
Dédicace .....	ii
Résumé .....	iii
Abstract .....	iv
Table des matières.....	vi
Liste des figures et tableaux.....	x
Liste des acronymes.....	xii
<b>Introduction générale.....</b>	<b>1</b>
1. Contexte de recherche.....	1
2. Problématique.....	2
3. Objectif.....	2
4. Organisation du document .....	3
<b>I. SMIL 3.0 : Concept et relation avec l'énergie.</b>	
I.1 Introduction.....	5
I.2 Définition.....	5
I.3 Historique du SMIL.....	5
I.4 Pourquoi SMIL.....	6
I.5 La modularisation et Le profilage.....	8
I.6 Les dimensions des documents multimédia.....	9
I.6.1 Dimension spatiale.....	9
I.6.2 Dimension temporelle.....	11
I.6.3 Dimension hypermédia.....	12
I.6.4 Dimension logique.....	14
I.7 Structure d'un document SMIL.....	15
I.8 Profils SMIL 3.0.....	17
I.9 Catégories de modules SMIL.....	18
I.9.1 Animation.....	18
I.9.2 Content Control.....	19
I.9.3 Time manipulation.....	19
I.9.4 Timing & synchronisation.....	20
I.9.5 Transition.....	20
I.9.6 Meta Information.....	21
I.9.7 State.....	21
I.9.8 Text SMIL.....	22
I.9.9 Structure.....	22
I.9.10 Media object.....	22
I.9.11 Layout.....	22
I.9.12 Linking.....	23
I.10 Modules de SMIL et consommation d'énergie.....	23
I.10.1 Le module BasicContentControl.....	231

I.10.2	Le module prefetchControl.....	24
I.10.3	Le module AudioLayout.....	25
I.10.4	Module MediaRenderAttributes.....	25
I.10.5	Attributs accelerate, decelerate et autoReverse.....	26
I.10.6	Module StateTest.....	27
I.10.7	. Synthèse .....	28
I.11	Conclusion.....	28
<b>II.</b>	<b>Généralités sur l’adaptation.</b>	
II.1	Introduction.....	29
II.2	Adaptation des présentations multimédia.....	29
II.2.1	Procédure d’adaptation.....	29
II.2.2	Architecture générale de fourniture de contenus multimédia adaptés.....	30
II.2.2.1	Gestion du contexte.....	31
II.2.2.2	Gestion des contenus multimédia.....	31
II.2.2.3	Gestion de la prise de décision.....	32
II.2.2.4	Gestion de l’adaptation de contenus multimédia.....	32
II.3.	Architectures d’adaptation au contexte.....	34
II.4.	Outils de description du profil utilisateur.....	35
II.4.1	MPEG-21.....	35
II.4.1.1	Adaptation des Items Digitaux.....	36
II.4.1.2	Descriptions d’environnement d’utilisation (UED).....	37
II.4.2	CC/PP.....	40
II.5.	Comparaison entre CC/PP et MPEG-21.....	42.
II.6	Conclusion.....	43
<b>III.</b>	<b>Techniques d’optimisation d’énergie</b>	
III.1	Introduction.....	45
III.2	Techniques d’optimisation de la consommation d’énergie.....	45
III.2.1	Adaptation du contenu.....	45
III.2.1.1	Effet de quelques paramètres sur la consommation d’énergie par les objets vidéo.....	46
III.2.1.2	Effet de quelques paramètres sur la consommation d’énergie des éléments audio.....	50
III.2.2	Optimisation d’énergie de rétro-éclairage.....	52
III.2.3	Optimisation de l’énergie par la technique Dynamic Voltage-Frequency Scaling (DVFS).....	54
III.2.4	Optimisation d’énergie par annotation des données.....	56
III.2.5	Optimisation de la consommation d’énergie lors de la transmission réseau..	59
a.	La technique <i>Traffic Shaping</i> .....	60
b.	Power-Saving Mode.....	60
c.	L’algorithme AB-PSM.....	61
d.	Le système CoolSpot.....	61
e.	Techniques d’ordonnancement optimal.....	62

---



f. Approche historique.....	63
III.2.6 Optimisation inter-couche.....	63
1. Le projet GRACE.....	63
2. Le projet FORGE.....	64
3. Le projet xTune.....	65
III.2.7. Optimisation d'énergie par la technique du cache.....	65
III.3. Adaptation des images.....	66
III.3.1 Types d'images.....	67
III.3.1.1 Images vectorielles.....	67
III.3.1.2 Images matricielles (ou images bitmap).....	67
III.3.2 Adaptation des images numériques.....	68
III.3.2.1 Le transcodage.....	68
III.3.2.2 La compression.....	69
III.3.2.3 Le changement de couleur.....	70
III.3.2.4 Le redimensionnement.....	70
III.4 Conclusion.....	71
<b>IV. Proposition d'une architecture.</b>	
IV.1 Introduction.....	72
IV.2 La notion du profil utilisateur.....	72
IV.3 Types d'adaptation.....	73
IV.3.1 Adaptation de contenus multimédia.....	73
IV.3.2 Adaptation de documents multimédia.....	73
IV.4. Techniques d'adaptation.....	74
IV.5 Architecture générale.....	74
IV.5.1 Fonctions des modules.....	78
IV.5.1.1 Module de gestion de contextes d'utilisation.....	78
IV.5.1.2 Module de gestion de contenus multimédia.....	81
IV.5.1.3 Module de gestion des documents multimédia.....	84
IV.5.2 Le proxy d'adaptation.....	86
IV.5.3 Processus d'adaptation.....	87
IV.5.4 Adaptation énergétique des documents multimédia.....	89
IV. 5.4.1 Adaptation de contenu.....	90
A) <u>Image</u> .....	92
B) <u>Vidéo</u> .....	94
C) <u>Audio</u> .....	95
IV. 5.4.2 Adaptation structurelle.....	96
IV.5.5 Optimisation matériels.....	98
A) <u>Carte réseau</u> .....	98
B) <u>Backlight</u> .....	99
C) <u>CPU</u> .....	99
IV.6 Tests expérimentaux.....	100
IV.6.1 Test de compromis entre facteurs d'optimisation.....	100

IV.6.2 Test des techniques d'adaptations et d'optimisation.....	100
IV.6.3 Résultats attendus.....	101
IV.6.4 Architecture expérimentale de tests.....	101
IV.7 Conclusion.....	103
<b>Conclusion et perspectives.....</b>	<b>104</b>
<b>Références bibliographiques.....</b>	<b>106</b>

---

# Liste des figures

Figure I.1 : dimensions d'un document SMIL.....	9
Figure I.2 : la dimension spatiale dans un document SIML.....	10
Figure I.3 : les informations spatiales dans un en-tête SMIL.....	10
Figure I.4 : la dimension temporelle dans un document SMIL.....	11
Figure I.5 : Utilisation des balises de synchronisation seq et par.....	12
Figure I.6 : Graphe de la synchronisation de la figure I.5.....	12
Figure I.7 : Dimension hypermédia d'un document multimédia.....	13
Figure I.8 : Lien interne d'un élément à une présentation.....	14
Figure I.9 : Lien vers des fragments SMIL.....	14
Figure I.10 : Lien d'un élément vers une sous-partie d'un objet multimédia.....	14
Figure I.11 : Dimension logique d'un document multimédia.....	15
Figure I.12 : exemple d'un document SMIL.....	16
Figure I.13 : structure d'un fichier SMIL.....	17
Figure I.14 : exemple sur l'élément erase.....	26
Figure I.15: exemple sur les attributs de teste.....	27
Figure II.1. interaction entre les éléments d'un système de fourniture de contenu multimédia.....	30
Figure II.2.architecture de base d'un système de fourniture de documents multimédia.....	31
Figure II.3.Illustration d'adaptation des éléments digitaux.....	37
Figure II.4. Informations énergétiques dans l'UED.....	39
Figure II.5. La structure d'un profil utilisateur en CC/PP.....	41
Figure II.6. Exemple d'un profil en CC/PP.....	42
Figure II.7 : Exemple d'un élément digital.....	43
Figure III.1: effet de type de codec sur la consommation d'énergie.....	47
Figure III.2 : Effet de format de fichier sur la consommation d'énergie.....	47
Figure III.3: effet de débit binaire des vidéos sur la consommation d'énergie.....	48
Figure III.4 : effet de la fréquence d'image des vidéos sur la consommation d'énergie	49

Figure III.5 : l'effet du facteur résolution sur la consommation d'énergie.....	50
Figure III.6 : énergie consommé par un fichier audio codé dans différents formats...	51
Figure III.7 : Enrichissement d'un document multimédia en vue d'une adaptation.....	58
Figure III.8 : Spécification de conditions à appliquer au document de la figure III.7	58
Figure III.9 : Résultat de l'application des conditions de la figure III.8 au document de la figure III.7.....	59
Figure III.10 Algorithme pour calculer le comportement optimal.....	62
Figure III.11 : Exemple d'une image vectorielle.....	67
Figure III.12 : Exemple d'une image matricielle.....	68
Figure III.13 : Tailles des images JPEG.....	69
Figure III.14 : Compression d'image JPEG.....	70
Figure IV.1 : adaptation à priori avec SMIL.....	74
Figure IV.2 : environnement matériel de l'architecture .....	75
Figure IV.3 : l'architecture fonctionnelle du système. ....	77
Figure IV.4 : Exemple de description de profil en MPEG-21.....	79
Figure IV. 5 : module de gestion de contextes d'utilisation.....	80
Figure IV.6 : Exemple de description MPEG-7.....	82
Figure IV.7 : module de gestion de contenus multimédia.....	83
Figure IV.8 : module de gestion de documents multimédia.....	85
Figure IV.9 : les relations entre les bases de données.....	86
Figure IV.10 : Le proxy d'adaptation .....	87
Figure IV.11 : le trafic de données dans le système.....	87
Figure IV.12 : Le processus d'adaptation .....	90
Figure IV.13 : Le changement d'échelle .....	91
Figure IV.14 : L'architecture de mesure d'énergie.....	102

---

## Liste des tableaux

Tableau III.1 : énergie consommé avec différentes valeurs de débit binaire .....	51
Tableau III.2: taux de croisement d'énergie consommé lors d'augmentation de débit binaire.....	51
Tableau III.3: énergie consommée pour différentes valeurs de fréquence d'image.....	52
Tableau III.4 : taux de croisement d'énergie consommée lors d'augmentation de fréquence d'image.....	52
Tableau III.5 : Comparaison entre des formats d'images .....	68

---

# Liste des acronymes

## **A**

AAC	Advanced Audio Coding
AB-PSM	Adaptive-Buffer Power Save Mechanism
AC-3	
ADMITS	Adaptation in Distributed Multimedia IT Systems
AP	Access Point
APPAT	Adaptation Proxies PLATform
AVI	Audio Video Interleaved

## **B**

BD	Base de Données
BMP	BitMaP
BNC	Bayonet Neill-Concelma

## **C**

CBCS	Concurrent Brightness and Contrast Scaling
CC/PP	Composite Capability/Preference Profiles
CPU	Central Processing Unit

## **D**

DAISY-DTB	Digital Accessible Information System-Digital Talking Book
DAQ	Data Acquisition Q
DCAF	Distributed Content Adaptation Framework
DIA	Digital Items Adaptation
DivX5	Digital Video Express
DLS	Dynamic luminance scaling
DPM	Dynamic Power Management
DRM	Digital Rights Management
DTD	Document Type Definition
DVS	Dynamic Voltage Scaling

***E***

EDF                    Earliest Deadline First

***F***

ftp                    File Transfer Protocol

***G***

GIF                    Graphics Interchange Format

GPRS                  General Packet Radio Service

GSM                    Global System for Mobile Communication

GUI                    Graphique User Interface

***H***

HTML-TIME            Html Time

http                    HypertText Transfer Protocol

***I***

***J***

JPEG                    Joint Photographic Experts Group

JXTA                    Juxtapose

***K***

Kbps                    Kilo bit per second

***L***

LCD                    Liquid Crystal Display

***M***

MAPS                    Media Accelerating Peer Services

MMS                    Multimedia Messaging Service

MOV

MPEG                    Moving Picture Experts Group

MP2                    MPEG-1 Audio Layer 2

MP3                    MPEG-1 Audio Layer 3

MP4                    MPEG-1 Audio Layer 4

MSSA                    Multimedia Scene Semantic Adaptation

***N***

NAC                    Negotiation and Adaptation Core

NI                        National Instruments

***O***

***P***

PAAM Architecture for the Provision of AdAptable Multimedia composed documents

PCI Peripheral Component Interconnect

PDA Personal Digital Assistant

P2P Peer To Peer

PSM Power-Saving Mode

***Q***

QABS Quality Adapted Backlight Scaling

QoS Quality of Service

***R***

Raw Real Audio Wrapper

RDF Resource Description Framework

RMS Rate-Monotonic Scheduling

RPC Remote Procedure Call

Rtp Routing Transfer Protocol

RTSP Real-Time Streaming Protocol

RVB rouge, vert et bleu

***S***

SATO Service-aware Transport Overlay

SE Système d'exploitation

SMIL Synchronized Multimedia Integration Language

SVG Scalable Vector Graphics

SYMM Synchronized Multimedia Working Group

***T***

TFT Thin-Film Transistor

TIFF Tagged Image File Format

***U***

UED Adaptation QoS

UCD Universal constraints description

UMA Universal Multimedia Access

UMTS Universal Mobile Telecommunications System



## Liste des acronymes

---

URI	Uniform Resource Identifier
URL	Uniform Resource Locator
<b>V</b>	
<b>W</b>	
WAM	Web, Adaptation et Multimédia
WLAN	Wireless Local Area Network
WAVE	WAVEform audio format
WMA	Windows Media Audio
WMV	Windows Media Video
WMV3	Windows Media Video 9
WNIC	Wireless Network Interface Card
W3C	World Wide Web Consortium
<b>X</b>	
XHTML	eXtended HTML
XML	eXtended Markup Language
XML-NS	XML Name Space
XSLT	XML Style Sheet Transformation
<b>Y</b>	
<b>Z</b>	
<b><u>Autre</u></b>	
3GPP	Third Generation Partnership Project

# Introduction générale

## 1. Contexte de recherche

Un document (présentation ou scène) multimédia est composé d'un ensemble de média de différents types (image, vidéo, audio, texte) temporellement synchronisés suivant un scénario orchestré dans une période de temps. Ses éléments visuels et sonores sont rendus suivant une certaine disposition spatiale. La caractérisation d'un document multimédia peut se traduire par une spécification par un langage telle que : SMIL [SMIL], Flash [FLASH] etc. Les documents multimédia, sont présents dans plusieurs domaines : ils sont utilisés dans l'enseignement à distance, en médecine, voire pour des réclames et des publicités, etc.

Les outils logiciels de traitement de documents multimédia (système de fourniture NAC [Le.04], PAAM [Ka,08] ; éditeur LimSee 2.0 [LIMSEE], GriNS [GRINS] et lecteur RealOne player [REALONE], Ambulant [AMBULANT] etc.) ont connu un essor considérable, grâce à leur intégration dans les appareils mobiles. D'ailleurs, l'utilisation de ces équipements a connu une grande émergence, durant ces dernières années, notamment au près du grand public. La prolifération des dispositifs mobiles a eu pour conséquence une grande diversité de conceptions, chacune dotée de ses propres caractéristiques : la taille de l'écran, le type de CPU, la résolution etc. Cela a parallèlement été accompagné par une diversité dans les réseaux de communication, et dans les contenus multimédia. Toute cette diversité exige d'effectuer les transformations nécessaires sur un document multimédia pour qu'il puisse être exécuté par un appareil mobile. Les tendances récentes dans le domaine de recherche, se sont focalisées sur le développement de systèmes d'adaptation de document multimédia. Dans ce contexte, plusieurs travaux [Az.05], [Be,06], [La,08], [Le,04] ont permis d'apporter des solutions aux problèmes d'adaptation, en satisfaisant les contraintes de profil et de qualité de service posées par ce type d'applications.

Par ailleurs, ces applications deviennent de plus en plus exigeantes (par exemple : préférences d'utilisateur, contrainte d'environnement, contraintes d'internet etc.) et convergent vers plus d'intégration des composants participants à l'adaptation (i.e.: gestion de profils, gestion de documents et contenus multimédia, gestion d'adaptation et gestion de

qualité de service etc.). En effet, les applications multimédias sont assez gourmandes en ressources (bande passante, mémoire, CPU), alors qu'elles imposent aux environnements matérielles et logicielles des échéances strictes pour leur délivrance. Cela impose une garantie de qualité de service soutenue par toutes les couches du système de communication.

## 2. Problématique

L'exécution des applications multimédia tend à se populariser, et se développer sur des appareils mobiles. Cela a malheureusement eu comme résultat une plus grande consommation d'énergie. En effet, afin de supporter l'exécution de ces applications, plusieurs modifications ont dues être apportées à ces systèmes. Ceux-ci ont ainsi vu leur capacité de stockage et leur puissance de calcul augmenter de manière significative. Or, ces deux éléments sont à l'origine d'une consommation énergétique supplémentaire plus importante.

Le facteur énergie bien qu'il soit important pour le bon fonctionnement de ce type d'appareil, n'est pas pris en considération lors de processus d'adaptation de document multimédia, et malgré les progrès technologiques réalisés dans le domaine des batteries, l'autonomie des appareils mobiles reste toujours relativement limitée.

Les travaux existants ont proposé de traiter de manière partielle et spécifique l'économie d'énergie dans les applications multimédias. Une optimisation de l'énergie par adaptation du contenu multimédia a été proposé dans quelques travaux, comme [Li,10] pour la vidéo, ou [Li,07] et [Li,08] pour l'audio. Dans d'autres travaux, il a été proposé des techniques d'optimisation de la gestion de l'énergie par adaptation du fonctionnement matérielle, c'est-à-dire par le contrôle optimisé des composants matériels comme la CPU [Bo,01] [Bu,05], le rétro-éclairage [Pa,03] [Ch,06], et la carte réseau [WI,10] [Ch,03]. Cependant très peu de travaux se sont intéressés à la proposition d'architectures pour la prise en charge globale de la gestion de l'énergie lors du déploiement des applications de distribution de présentations multimédias.

## 3. Objectif

Pour ce mémoire de Magister, nous nous sommes fixés les objectifs suivants:

- Faire un état de l'art exhaustif des méthodes et des techniques proposées dans la littérature traitant de l'optimisation de l'énergie dans un environnement mobile. Un intérêt particulier est porté aux méthodes dédiées aux applications multimédias.

- Discuter d'une architecture générale d'un système d'adaptation de document multimédia, dans laquelle les méthodes d'économie d'énergie suscitées peuvent être intégrées. Cette architecture devra permettre de définir et de déployer des environnements matériel et logiciel pour la fourniture de présentations multimédia. La délivrance de telles présentations devra assurer une garantie de qualité de service soutenue vis-à-vis de l'application et de l'utilisateur, tout en imposant une gestion efficace de la ressource énergétique. Pour ce faire, nous sommes amenés à répondre aux questions suivantes :

- Quelles sont les techniques et les méthodes existantes qui permettent l'optimisation d'énergie ?
- Et comment optimiser la consommation d'énergie des applications multimédias dans les appareils mobiles, tout en préservant une bonne qualité de présentation d'un document multimédia ?

#### 4. Organisation du document

Ce mémoire est organisé en quatre chapitres, comme suit :

##### **Chapitre I:** «Le standard SMIL 3.0: concept et relation avec énergie »

Ce chapitre présente la version 3.0 du langage SMIL [SMIL] d'une façon générale, et étudie, module par module, le rapport entre les différents éléments et attributs et la consommation énergétique des documents multimédia SMIL.

##### **Chapitre II:** «Généralité sur l'adaptation»

Ce chapitre a pour but de donner: la description des principales notions du domaine d'adaptation de documents multimédias ; les principaux composants d'une architecture d'adaptation de documents multimédias; et les principaux éléments du profil utilisateur. Il aborde également une comparaison entre les deux normes de description de profil utilisateur CC/PP [CCPP] et MPEG-21[MPEG21].

##### **Chapitre III:** «Techniques d'optimisation d'énergie»

---

Ce chapitre est un état de l'art des techniques d'optimisation d'énergie dans les environnements mobiles. Deux classes de techniques sont discutées : les techniques d'optimisation au niveau contenu distinguées par le type de média (vidéo, audio et image); et les techniques d'optimisation au niveau matériel (carte réseau, rétro-éclairage etc.).

**Chapitre IV:** «Proposition d'une architecture »

Ce chapitre est consacré à présenter les différents modules d'une architecture d'un système d'adaptation et d'optimisation de présentations multimédia SMIL avec les techniques d'optimisation d'énergie que nous recommandons pour de telles applications. L'architecture que nous dégageons et discutons, est suivie par une proposition de plateforme de tests, nécessaire à sa validation.

# I. SMIL 3.0: Concept et relation avec l'énergie

## I.1 Introduction

SMIL [SMIL] est un langage de spécification de documents multimédias basé sur XML. Il permet de caractériser les exigences spatiales, structurelles ainsi que temporelles d'une présentation multimédia, qui doit être jouée sur un large éventail d'équipements, notamment de terminaux mobiles. La toute dernière version du SMIL, la version 3.0, publiée en 2008 a apporté des modifications sur les anciens modules des versions précédentes et a intégré de nouveaux modules permettant aux auteurs une caractérisation de plus en plus flexible basé sur le profil.

Dans ce chapitre nous nous intéressons aux nouveaux modules de SMIL 3.0. Nous explorons comment certains mécanismes définis peuvent aider à optimiser la consommation d'énergie dans les appareils mobiles, tels que les PDAs (Personal digital Assistant) et les téléphones portables.

## I.2 Définition

SMIL [SMIL] (Synchronized Multimedia Integration Language) est un langage déclaratif recommandé par le W3C (World Wide Web Consortium) permettant la spécification des présentations multimédias interactives par l'intégration et la synchronisation des contenus multimédias diversifiés (image, son, vidéo, texte, animation, flux de texte (streaming)).

SMIL est un langage de balisage basé sur le standard XML, la structure du document SMIL décrit l'organisation temporelle d'objets multimédias (aspect temporel), spécifie leur disposition spatiale (aspect spatial), et définit des liens hypermédias (aspect hypermédia), sous forme de documents XML.

## I.3 Historique du SMIL [SMIL]

SMIL, depuis son apparition dans sa première version 1.0 le 15 juin 1998 a connu une évolution constante grâce au groupe de travail SYMM (synchronized multimedia working

group) du W3C. Au début, le groupe a instauré les bases du langage permettant une spécification de documents multimédias.

- permettre la synchronisation des objets multimédia de manière plus ou moins flexible (aspect temporel).
- permettre le placement visuel des objets multimédias de la présentation (aspect spatial).
- permettre le référencement des objets multimédias (aspect hypermédia).
- Permettre l'interaction événementielle avec l'utilisateur.
- Offrir des possibilités d'adaptation de contenu local avec l'élément SWITCH.

Le 7 août 2001, une deuxième version 2.0 a été publiée avec de nouveaux modules qui viennent s'ajouter offrant plus de flexibilité et de fonctionnalités par exemple : le module méta-information et le module animation.

SMIL 2.1, la troisième version publiée en 2005, a amené beaucoup de changement par l'ajout des modules d'effets et de définition des profils du langage en **DTD** (Document Type Definition), à exécuter sur des plateformes mobiles (comme SMIL basic profil).

En fin 2008, la quatrième version de SMIL (SMIL 3.0) est élaborée, celle-ci définit de nouveaux modules et profils très puissants, elle fait l'objet d'une étude dans ce chapitre.

## I.4 Pourquoi SMIL

Le langage SMIL grâce à ces différentes versions a investi une grande partie du marché du multimédia, surtout celui du mobile. De plus en plus d'applications utilisent ce standard :

- Les MMS (Multimedia Messaging Service) ce format largement utilisé sur les téléphones mobiles, utilise un sous-ensemble de modules du langage SMIL.
- HTML-TIME est un langage permettant l'ajout de la dimension temporelle aux pages HTML, ceci est réalisé par l'inclusion des éléments temporels provenant de langage SMIL.
- **SVG (Scalable Vector Graphics)**, est un format de documents basé sur XML pour la description des ensembles de graphiques vectoriels, est dans sa partie animation basé sur le module d'animation du langage SMIL.

— DAISY-DTB (Digital Accessible Information System-Digital Talking Book) est basé sur le standard XHTML et SMIL, permettant la lecture d'un format de livre électronique par synthèse vocale.

En raison de la large utilisation de SMIL, de nombreux outils d'édition et de lecture multimédia exécutant SMIL ont été développés, comme :

#### A. Les éditeurs

- LimSee 2.0 [**LIMSEE**] pour SMIL 2.0 et LimSee 3.0 [**LIMSEE**] pour SMIL 2.1 et SMIL 3.0
- GRINS [**GRINS**] pour SMIL 2.0 ;
- Adobe GoLive [**GOLIVE**] ;
- SMOX Pad 1.2.1.2 [**SMOXPAD**] pour SMIL 2.0 ;

#### B. Les lecteurs

- QuickTime [**QuickTime**] pour SMIL 1.0 ;
- RealOne Player [**REALONE**] pour SMIL 2.0 ;
- Ambulant 2.0.2 [**AMBULANT**] pour SMIL 2.1 et SMIL 3.0.
- Internet Explorer [**IE**] pour HTML-TIME.
- PoketSMIL 2.0 [**POKETSMIL**] exécutant des documents SMIL 2.0 sur PDA.
- Rubic [**Rubic**] exécutant des MMS basées sur SMIL.

Tous ces outils d'édition et de lecture, peuvent être utilisés pour créer plusieurs types de présentations, tout en exploitant les diverses fonctionnalités qu'offre SMIL, dont on cite quelques unes :

- SMIL peut être utilisé pour créer des présentations Internet ou Intranet ;
- Il peut être utilisé pour créer des présentations slide-show ;
- Il est décrit comme l'image Internet de PowerPoint ;
- Les présentations SMIL peuvent afficher de multiples types de fichiers (texte, vidéo, audio.) ;
- Elles peuvent afficher de multiples fichiers en même temps ;



- Elles peuvent afficher de multiples fichiers à partir de multiple serveurs web, grâce aux URL ;
- Elles peuvent contenir des liens à d'autres présentations SMIL ;
- Elles peuvent contenir des boutons de contrôle (stop, start, next, ...) ;
- SMIL a des fonctions pour définir les séquences et la durée des éléments ;
- Il a des fonctions pour définir la position et la visibilité des éléments ;
- Etc.

## I.5 La modularisation et le profilage [SMIL]

La modularisation est une approche dans laquelle la fonctionnalité de balisage est spécifiée comme un ensemble de modules sémantiquement liés aux éléments, attributs et valeurs d'attribut XML. La modularisation permet aux concepteurs du langage de spécifier le balisage dédié destiné à l'intégration avec d'autres profils existants. Des exemples de telles spécifications destinées à l'intégration sont : MathML [**MATHML**], et XForms [**XFORMS**].

Le profilage est la création d'un langage basé sur XML par le biais de la combinaison de ces modules, afin de fournir la fonctionnalité requise par une application particulière. Le profilage introduit la possibilité d'adapter un langage basé sur XML (dans notre cas c'est le langage SMIL) à des besoins spécifiques, par exemple, pour optimiser la présentation et l'interaction aux capacités du client. Le profilage ajoute également la capacité d'intégration d'une fonctionnalité à partir d'autres langages de balisage, Il prévoit en outre la cohérence de balisage par l'utilisation du même modèle à intégrer. Le profilage en plus de la facilité de création de présentations offre la possibilité de réutilisation de code. Par exemple, un code en SMIL qui contient des fonctionnalités de synchronisation peut être utilisé pour des documents codés en XHTML + SMIL, et des documents codés en SVG.

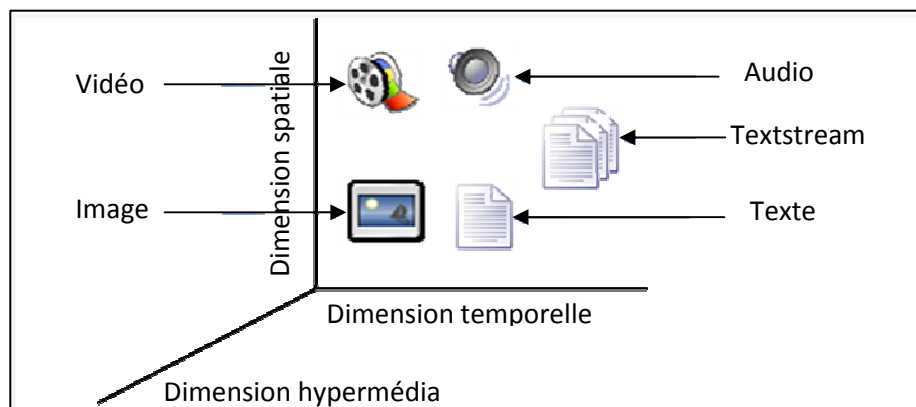
La modularisation et le profilage utilisent des propriétés d'extensibilité de XML, et des technologies connexes comme les espaces de nommages XML (XML-NS) et XML Schéma (XSCHEMA).

## I.6 Les dimensions des documents multimédia

Un document multimédia est composé d'objets de différents types : texte, son, image, et 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 multimédias.

Ces relations multimédias peuvent porter sur la description de l'organisation logique du document, la présentation spatiale des différents objets multimédias, leurs synchronisations temporelles ainsi que l'interconnexion entre ses différents éléments (hypermédia). Il est donc possible d'analyser un document multimédia selon quatre dimensions (**figure I.1**) :

- temporelle (§ I.6.1) : organisation temporelle des objets multimédias.
- spatiale (§ I.6.2) : disposition graphique des objets multimédias.
- hypermédia (§ I.6.3) : relations entre documents ou objets multimédia.
- logique (§ I.6.4) : structure du document en chapitres.



**Figure I.1 : dimensions d'un document SMIL**

La partie suivante va détailler chacune des dimensions de document multimédia.

### I.6.1 Dimension spatiale

La dimension spatiale consiste en la disposition géométrique d'objets visuels d'un document multimédia comme des textes, des vidéos et/ou des images sur l'écran, délimitée par une certaine zone, elle est définie, dans le cas du SMIL dans la partie *Head* de la spécification comme le montre la **figure I.2**.

```

<smil>
  <head>
    <!-- ..... Dimension spatiale ..... -->
  </head>
  <body>
    <!-- .....corps du document..... -->
  </body>
</smil>

```

**Figure I.2 : La dimension spatiale dans un document SMIL**

L'en-tête *Head* d'un document SMIL contient les informations spatiales du document comme la taille de la fenêtre globale de présentation ou le placement des différents objets multimédia visibles du document. Ces informations sont stockées dans une partie nommée *Layout*.

Pour visualiser les objets multimédia, il est nécessaire, dans un premier temps, de définir un environnement principal de présentation où sont exécutés les différents objets multimédia, c'est-à-dire la fenêtre globale de présentation (*root-layout*). Cet environnement principal peut, dans un second temps, contenir d'autres environnements que l'on appelle des régions (*region*). Les régions permettent de disposer aisément les différents objets multimédias. On définit chaque région par un identificateur (*id*), une hauteur (*height*), une largeur (*width*), la distance par rapport au haut (*top*) et au côté gauche (*left*) de l'environnement principal (sauf exceptions), ainsi que la profondeur (*z-index*).

La **figure I.3** illustre le code d'un en-tête SMIL.

```

<smil>
<head>
  <layout>
    <root-layout height="300" width="400"/>
      <region id="video" top="10" left="10" height="200" width="380" z-
index="0"/>
      <region id="text" top="220" left="10" height="70" width="380" z-
index="1"/>
      <region id="bye-msg" top="125" left="100" height="50" width="200" z-
index="3"/>
    </layout>
  </head>
<body>...</body>
</smil>

```

**Figure I.3 : Les informations spatiales dans un en-tête SMIL.**

Dans cette figure, la région identifiée par vidéo dispose d'une hauteur de 200 pixels, d'une largeur de 380 pixels et d'une profondeur fixée à 1. La coordonnée du point supérieur gauche de cette région est (10; 10).

**Note :** Les termes *Head*, *Layout*, *Region* et *Root-layout* ne sont autres que des éléments du module *Layout* (mise en page) du langage SMIL.

## I.6.2 Dimension temporelle

La dimension temporelle définit les liens temporels qui existent entre les objets dont la valeur de durée est connue a priori (audio et vidéo), et ceux qu'ils n'ont pas de durée connue a priori (provenant du réseau) et ceux dont la présentation ne dépend pas du temps (texte et image). Ces liens consistent à la temporisation de chaque objet et la synchronisation entre tous les objets de la présentation. Les informations temporelles constituent la caractéristique majeure d'un document multimédia, elles sont définies, dans le langage SMIL dans la partie *body* de la spécification comme le montre la **figure I.4**.

```
<smil>
  <head>
    <!-- ..... Dimension spatiale ..... -->
  </head>
  <body>
    <!-- .....Dimension temporelle..... -->
  </body>
</smil>
```

**Figure I.4 : La dimension temporelle dans un document SMIL**

Le corps *body* d'un document SMIL (exemple **figure I.5**) contient les informations de synchronisation temporelle des différents objets multimédias d'une présentation. Chaque objet multimédia est identifié par son type : *text* pour les textes, *img* pour les images, *audio* pour les fichiers sonores et *video* pour les vidéos. L'élément *src* permet de localiser la ressource multimédia correspondante et un identifiant (*id*). Les différents objets multimédia d'un document SMIL peuvent être contrôlés dans le temps en spécifiant une durée d'exécution (*dur*), un instant de début (*begin*) et de fin (*end*) de présentation.

Des conteneurs temporels permettent d'exécuter un ensemble d'éléments en séquence (*Seq*), en parallèle (*Par*) ou bien de manière exclusive (*Excl*) où un seul objet multimédia est exécuté à la fois.

```

<body>
  <par>
    <seq>
      
      <text id="T1" src="graph-text" region="left-text"/>
    </seq>
  </par>
  <par>
    <a href="http://www.w3.org/People/Berners-Lee">
      <video id="V1" src="tim-video" region="left-video"/>
    </a>
    <text id="T2" src="tim-text" region="right-text"/>
  </par>
  <seq>
    <audio id="A1" src="joe-audio"/>
    <video id="V2" src="joe-video" region="right-video"/>
  </seq>
</par>
</body>

```

Figure I.5 : Utilisation des balises de synchronisation Seq et Par.

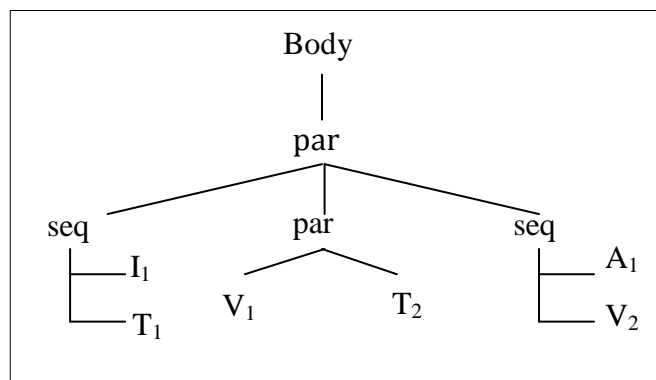


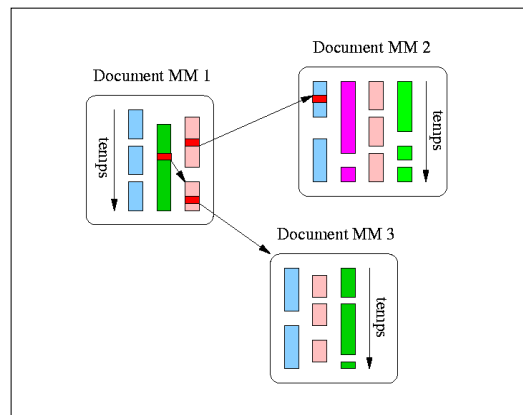
Figure I.6 : Graphe de la synchronisation de la figure I.5

La **figure I.6** illustre une représentation graphique d'une synchronisation possible des objets A<sub>1</sub>, I<sub>1</sub>, T<sub>1</sub>, T<sub>2</sub>, V<sub>1</sub> et V<sub>2</sub> du document illustré dans la **figure I.5**.

### I.6.3 Dimension hypermédia

La dimension hypermédia consiste à décrire les liens hypermédia qui peuvent exister entre certains éléments de même document, ou même entre différents documents. Ces liens permettent d'établir un réseau de documents, ils constituent ainsi le support de navigation entre ces documents dans le Web car SMIL est un langage d'Internet. Les éléments de spécification hypermédia de SMIL sont groupés dans le module *hyperlink* définis dans la

partie <body> du document. La **figure I.7** présente un exemple de liens hypermédia entre objets et documents multimédias.



**Figure I.7 : Dimension hypermédia d'un document multimédia**

Dans cet exemple, le document MM1 comporte un lien entre deux objets multimédias (lien hypermédia entre objets). Un objet multimédia du document MM1 pointe sur le document multimédia MM3 (lien hypermédia entre documents multimédia), le document MM3 est exécuté lorsque l'utilisateur interroge le lien hypermédia qu'il trouve sur l'interface de communication. Les liens hypermédia peuvent aussi avoir une dimension temporelle lorsque l'exécution d'un objet ou d'un document pointé par un lien, dépend du temps d'exécution de l'objet source du lien. Par exemple, le document MM2 est relié à un objet du document MM1, dont l'exécution est enclenchée après un certain temps  $t$ .

Un lien hypermédia peut être interne ou externe, les marqueurs **a** et **area** permettent de fixer un lien simple comme dans le cas de HTML, ils permettent également de fixer le lien hypermédia sur une ancre tandis que la localisation de la cible est contenue dans l'attribut *href*. Un lien hypermédia a une relation de dépendance avec son ancre et est actif pendant une période donnée à un endroit donné. Dans le langage SMIL, les attributs *temporels* (*begin*, *dur*, *end*) et *spatiaux* (*coords*) sont utilisés pour spécifier les informations de présentation d'un lien hypermédia.

– Dans l'exemple suivant (**figure I.8**) l'objet *getout* est affiché; en cliquant dessus, l'objet *choose* sera affiché :

```
<smil>
  <head>
    <layout>
      <region id="c1" top="0" left="0" width="100" height="100"/>
    </layout>
  </head>
  <body>
    <par>
      <a href="choose.jpg">
        
      </a>
    </par>
  </body>
</smil>
```

**Figure I.8 : Lien interne d'un élément à une présentation**

– On utilise également le symbole # pour désigner l'ancre de positionnement (figure I.9) :

```
<a show="new" href="time-time.smil#XVII">
  <text src="archives-dcab.html" region="I-main" dur="indefinite"/>
</a>
```

**Figure I.9 : Lien vers des fragments SMIL**

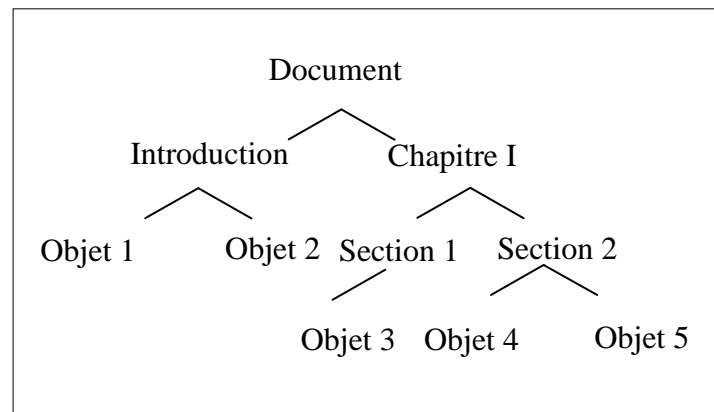
– Il faut cliquer sur getout.jpg dans le cadre indiqué par *coords* pour faire apparaître choose.jpg (figure I.10) :

```
<body>
  
    <anchor href="getout.jpg" coords="0,0,75,100" />
  </img>
</body>
```

**Figure I.10 : Lien d'un élément vers une sous-partie d'un objet multimédia**

### I.6.4 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 multimédias. La **figure I.11** présente une structure logique d'un document multimédia.



**Figure I.11 : Dimension logique d'un document multimédia**

La racine de la structure logique représente l'ensemble du document multimédia. Ce document peut se décomposer en plusieurs parties (Introduction, Chapitre . . .) et sous-parties (Section . . .). Chaque partie ou sous-partie peut contenir un nombre fini d'objets multimédia.

## I.7 Structure d'un document SMIL

L'élément racine d'un document SMIL est l'élément *Smil*. Cet élément contient deux éléments fils *body* et *head*. L'élément *head* ne contient pas des informations temporelles mais renseigne sur l'auteur et la présentation spatiale. L'élément *body* contient des informations sur les médias et leurs relations temporelles. Un document SMIL est structuré sous forme de conteneurs temporels (*time containers* en anglais), appelés également éléments composites ou opérateurs. Un opérateur comprend une sémantique temporelle particulière qui permet de définir le placement temporel des objets média. Ces opérateurs sont les éléments *seq*, *par* et *excl*.

Le conteneur temporel *seq* (séquence) définit une présentation en séquence des ressources médias. Le conteneur temporel *par* (parallèle) permet de jouer les ressources en parallèle. Finalement, l'opérateur *excl* est basé sur *par* mais en ajoutant la contrainte que seul un objet enfant soit joué à un moment donné.



L'exemple de la **figure I.12** présente les principaux éléments d'une présentation multimédia SMIL :

```
<?xml version="1.0" ?>
<smil xmlns="http://www.w3.org/2000/SMIL20/Language">
  <head>
    <layout type="text/smil-basic">
      <region id="left-video" left="20" top="50" z-index="1"/>
      <region id="left-text" left="20" top="120" z-index="1"/>
      <region id="right-text" left="150" top="120" z-index="1"/>
    </layout>
  </head>
  <body>
    <par>
      <seq>
        
        <text src="graph-text" region="left-text"/>
      </seq>
      <par>
        <a href="http://www.w3.org/People/Berners-Lee">
          <video src="tim-video" region="left-video"/>
        </a>
        <text src="tim-text" region="right-text"/>
      </par>
      <seq>
        <audio src="joe-audio"/>
        <video id="jv" src="joe-video" region="right-video"/>
      </seq>
    </par>
  </body>
</smil>
```

**Figure I.12 : Exemple d'un document SMIL**

La structure d'un document SMIL peut être schématisée par un arbre qui contient une racine et des branches, et des sous branches comme le montre la **figure I.13** suivante :

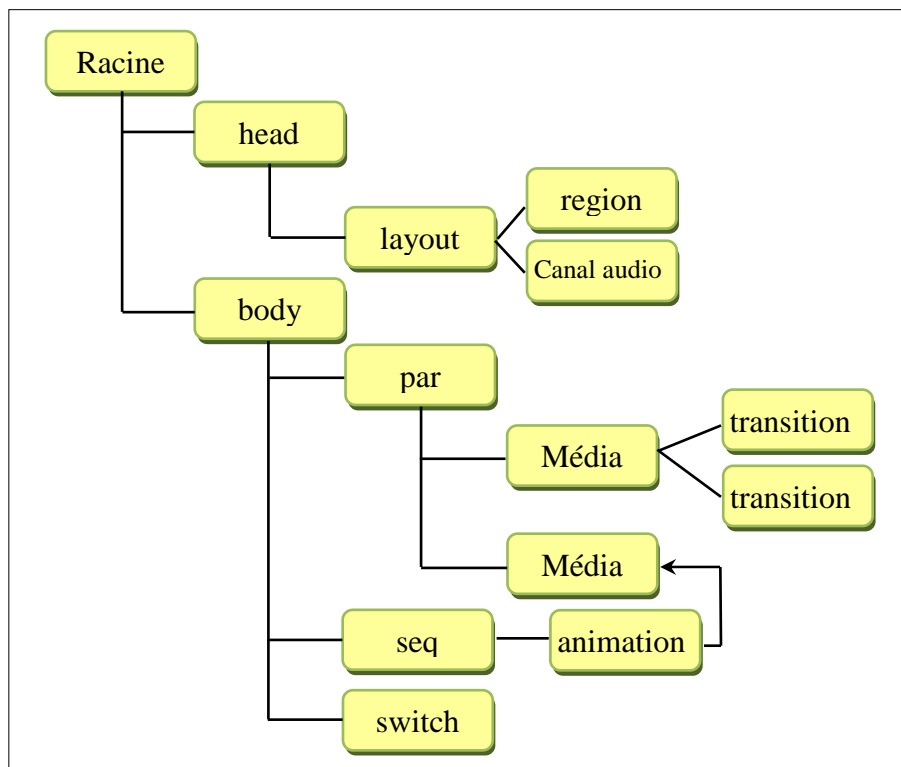


Figure I.13 : Structure d'un fichier SMIL

## I.8 Profils SMIL 3.0 [SMIL]

La spécification SMIL définit le comportement de SMIL en termes d'un ensemble de groupes fonctionnels. Chaque groupe fonctionnel est partitionné en *modules*, qui sont des collections d'éléments et/ou attributs. Les modules représentent un groupe atomique de composants : si un module est utilisé dans une implémentation, toutes les caractéristiques définies à travers ce module sont supportées.

Ce ne sont pas toutes les versions de SMIL qui supportent tous les modules de chaque groupe fonctionnel. La fonctionnalité de SMIL est davantage partitionnée dans une collection de modules appelée *profiles*. La spécification SMIL définit les profils suivants :

□ **LANGUAGE DE PROFILS:** Les auteurs peuvent choisir le modèle approprié de document pour les dispositifs à capacités limitées. Ces modèles s'intègrent dans un LANGUAGE DE PROFILS conçu spécifiquement pour prendre en charge des clients particuliers. Parmi ces LANGUAGE PROFILS, nous citons le SMIL BASIC LANGUAGE PROFILE, qui se compose d'un sous-ensemble réduit de modules SMIL. Le sous-ensemble

défini peut être supporté par une large variété de lecteurs SMIL incluant ceux qui fonctionnent, sur des dispositifs mobiles présentant des ressources limitées : des écrans de petite taille, cartes réseaux supportant un faible trafic, autonomie d'énergie faible, etc.

**UNIFIEDMOBILE:** Plusieurs organisations de standardisation autre que W3C contrôlent les langages implémentés dans les appareils mobiles. Par exemple, le consortium 3GPP (Third Generation Partnership Project) a adopté SMIL comme le standard pour la troisième génération d'infrastructure de lecteur de téléphoné mobile. Pour supporter ces modèles, SMIL3.0 utilise une description d'un profil cadre appelée profil UNIFIEDMOBILE.

**DAISY:** SMIL 3.0 a été préconisé pour les besoins de Daisy Consortium. Le profil DAISY de SMIL est un cadre de travail (framework). L'actuelle standardisation de Daisy est fournie dans la spécification NSI/NISO Z39.86 pour les livres vocaux digitaux.

**TINY:** C'est la collection minimum de modules qui fournissent le support pour le langage SMIL 3.0. Ce profil convient aux systèmes qui exigent des présentations SMIL très simples où les interactions d'utilisateur et la mise en page spécifique de contenu ne sont pas nécessaires. C'est, par exemple, le cas pour des dispositifs avec des capacités de calcul réduites telles que les lecteurs MP3/MP4, les téléphones portables, les systèmes de navigation de voiture, les téléviseurs ou les agents vocaux d'utilisateur. En outre, il est possible d'utiliser le profil TINY Profil dans le développement des playlists de côté du serveur. Ces playlists sont utilisées pour produire des streams d'un fichier audio ou vidéo sans aucune l'intervention de l'utilisateur.

## I.9 Catégories des modules SMIL

Le langage SMIL est un ensemble de modules qui sont classés en plusieurs catégories :

### I.9.1 Animation

Les modules d'animation de SMIL 3.0, sont composés d'un module *BasicAnimation* et d'un module *SplineAnimation*. Ces modules contiennent des éléments et des attributs pour intégrer une animation sur un plan de montage chronologique, et un mécanisme pour composer les effets de plusieurs animations. Puisque ces éléments et ces attributs sont définis dans des modules, les concepteurs d'autres langages balisés peuvent choisir d'inclure ou non

cette fonctionnalité dans leurs langages. Les concepteurs de langage qui intègrent d'autres modules SMIL n'ont pas besoin d'inclure les modules d'animation si la fonctionnalité de l'animation n'est pas utile.

### I.9.2 Content Control

Les modules de contrôle de contenu de SMIL 3.0 contiennent les éléments et les attributs qui offrent des choix de contenu à l'exécution et une livraison de contenu optimisée. La fonctionnalité de contrôle de contenu de SMIL se partage en quatre modules :

- BasicContentControl : qui contient des éléments de sélection de contenu et des attributs de test du système prédéfinis ;
- CustomTestAttributes : qui contient des éléments et attributs de test personnalisés définis par l'auteur ;
- PrefetchControl : qui contient des éléments et attributs d'optimisation de la présentation; et
- SkipContentControl : qui contient des attributs qui gèrent une évaluation sélective des attributs.

### I.9.3 Time manipulation

Ce module introduit de nouveaux attributs pour la manipulation avancée du comportement du temps, comme le contrôle de la *vitesse* ou le *régime* du temps pour un élément. Ces *manipulations du temps* sont particulièrement adaptées aux médias d'animation et aux médias non linéaires, ou médias discrets. Les types de média continus ne gèrent pas tous totalement les manipulations du temps. Par exemple, un flux vidéo MPEG 1 ne gère pas, en général, une lecture en arrière. Un mécanisme de repli est décrit pour ces types de média.

Quatre attributs *accelerate*, *decelerate*, *autoReverse* et *speed* ajoutent une gestion des manipulations du temps aux modules de temporisation de SMIL, incluant le contrôle sur la vitesse d'un élément et la gestion de l'accélération et de la décélération. Une définition des mécanismes de repli raisonnables est fournie pour les lecteurs de média qui ne gèrent pas les manipulations de temps.

### I.9.4 Timing & synchronisation

La gestion de la temporisation et de la synchronisation de SMIL 3.0 est découpée en 15 modules, permettant ainsi une grande flexibilité pour les créateurs d'autres langages intégrant ces fonctionnalités.

La temporisation de SMIL définit les éléments et attributs servant à coordonner et synchroniser la présentation d'un *média* au cours du temps. Le terme de *média* couvre un large champ, incluant les types de médias *discrets* tels que les images fixes, le texte et les graphiques vectoriels, ainsi que les types de médias *continus* étant intrinsèquement basés sur le temps tels que la vidéo, l'audio et les animations.

Trois éléments de synchronisation appelés *conteneurs de temps* gèrent des utilisations de temporisation communes :

- L'élément `<seq>` joue les éléments enfants les uns à la suite des autres en séquence ;
- L'élément `<excl>` joue les éléments enfants un à la fois mais sans imposer l'ordre ;
- L'élément `<par>` joue les éléments enfants comme un groupe permettant une lecture en parallèle.

La temporisation de SMIL fournit également des attributs permettant de spécifier le comportement d'un élément au cours du temps. Les éléments ont un début et une *durée simple*. La *durée simple* définit la durée de présentation de base d'un élément. Les éléments peuvent être définis pour répéter une durée simple, un certain nombre de fois ou encore pendant un certain temps. La durée simple et les effets de répétition sont combinés pour donner la *durée active*. Lorsque la durée active d'un élément est écoulée, celui-ci peut être soit retiré de la présentation, soit *gelé* (conservé dans son dernier état), par exemple pour combler un espace dans la présentation.

### I.9.5 Transition

Les modules de transition de SMIL 3.0 contiennent les éléments et les attributs qui offrent La fonctionnalité de transition à SMIL 3.0 cette catégorie est donnée à travers quatre modules : le module *BasicTransitions*, qui définit la méthode de sténographie style-like de spécification des transitions, le module *FullScreen*, qui définit des mécanismes pour les transitions qui anime l'écran entier, le module *InlineTransitions*, ce module permet un niveau

---

beaucoup plus fin de contrôle des transitions, et en fin le module *TransitionModifiers*, qui fournit le contrôle additionnel de l'aspect visuel d'une transition.

### **I.9.6 Meta Information**

Cette catégorie est composée d'un seul module. Ce module contient les éléments et les attributs qui permettent la description de documents SMIL. Les métadonnées sont des « données sur des données » (par exemple, le catalogue d'une bibliothèque est une métadonnée, étant donné qu'il décrit des publications) ou, spécifiquement dans le contexte de cette caractérisation, des « données décrivant des ressources Web ».

Le principe dans le cas de SMIL est d'utiliser les informations des métadonnées pour décrire les documents SMIL publiés sur le Web. Le module *Metainformation* de SMIL 3.0, défini dans cette spécification, gère pleinement l'utilisation de l'élément *meta*, mais il introduit aussi de nouvelles fonctionnalités pour décrire des métadonnées en utilisant la syntaxe et le modèle du cadre de description de ressource (Resource Description Framework). Ce dernier est un langage de méta-information puissant pour fournir des informations sur des ressources.

### **I.9.7 State**

La catégorie state de SMIL 3.0 fournit un mécanisme pour que l'auteur crée un flux de commande plus complexe que ce que SMIL fournit par les modules de synchronisation et de contrôle de contenu. Il permet à un document d'avoir un certain nombre d'états explicites (au sens variables) adaptant la présentation aux desideratas d'un utilisateur.

Les nouveaux modules dans cette catégorie fournissent un mécanisme par lequel l'auteur de document puisse créer un flux de contrôle plus complexe que celui que SMIL fournit par les modules de synchronisation et de contrôle de contenu. Quatre modules sont présentés dans les domaines fonctionnels d'état state : le module *StateTest*, contenant une sélection étendue du contenu; le module *UserState*, contenant l'état défini ; le module *StateSubmission*, permet l'enregistrement de l'état défini par l'auteur; le module *StateInterpolation*, permettant la modification en temps réel des valeurs des attributs durant la présentation.

### I.9.8 Text SMIL

Les modules *smilText* se composent de module *BasicText* et de deux modules *TextStyling* et *TextMotion* apportant des fonctionnalités additionnelles. Ces modules contiennent des éléments et des attributs employés pour définir le contenu intégré des textes. Le profil *SmilText* permet au *smilText* d'être employé comme format externe.

### I.9.9 Structure

Le module de structure de SMIL fournit les éléments de base pour structurer un contenu SMIL. Ces éléments agissent comme la racine dans le modèle de contenu de tous les profils de langage conformes au langage hôte SMIL. Le module *Structure* est un module obligatoire pour les profils de langage conformes au langage hôte SMIL.

Le module *SMIL Structure* est composé des éléments *smil*, *head* et *body*, et il est compatible avec SMIL 1.0. Les éléments SMIL 1.0 correspondants forment un sous-ensemble du module *Structure*, à la fois dans la syntaxe et la sémantique. Comme leurs attributs et le contenu étant également présentés par le module *Structure*, par conséquent, ce dernier a une compatibilité ascendante avec SMIL 1.0.

### I.9.10 Media object

Les modules d'objet média de SMIL sont composés d'un module de base *BasicMedia* et de cinq modules lui fournissant des fonctionnalités additionnelles : les modules *MediaClipping*, *MediaClipMarkers*, *MediaParam*, *MediaAccessibility* et *MediaDescription*. Ces modules contiennent les éléments et attributs nécessaires à la description des objets médias. De plus, un élément *BrushMedia* est fourni et peut être utilisé comme un objet média grâce à la définition de ces éléments et attributs.

### I.9.11 Layout

Les modules de disposition de SMIL 3.0, sont composés d'un module *BasicLayout* et de trois modules avec des fonctionnalités supplémentaires qui sont basés sur le module *BasicLayout* : les modules *AudioLayout*, *MultiWindowLayout* et *HierarchicalLayout*. Les modules contiennent des éléments et des attributs permettant le positionnement d'éléments médias sur la surface de rendu visuel et le contrôle du volume audio.

### I.9.12 Linking

Les modules de lien de SMIL 3.0 définissent les attributs et les éléments du document SMIL 3.0 pour la navigation hyperlien. Ce sont des navigations au travers une présentation SMIL déclenchées par interaction de l'utilisateur ou par d'autres événements déclencheurs, tels que des événements temporels. SMIL 3.0 n'offre que des éléments de lien en-ligne. Les liens sont limités aux liens uni-directionnels (i.e., tous les liens ont exactement une source et une destination).

Les modules de lien de SMIL 3.0 sont nommés *LinkingAttributes*, *BasicLinking* et *ObjectLinking*. Le module *LinkingAttributes* inclut un ensemble d'attributs employés pour fournir la sémantique des liens SMIL aux éléments de lien. Le module *BasicLinking* inclut les éléments de lien de SMIL 3.0 eux-mêmes. Le module *ObjectLinking* inclut des mécanismes de lien optionnels supplémentaires qu'un profil de langage souhaite inclure. Notez que le module *BasicLinking* inclut explicitement les attributs du module *LinkingAttributes* sur ses éléments.

## I.10 Modules de SMIL et rapport à la consommation d'énergie

On a constaté que parmi les modules de la recommandation SMIL 3.0, certains offrent des fonctionnalités qui peuvent être manipulées de façon à réduire l'effet global de la présentation sur la consommation d'énergie, tout en préservant une bonne qualité de la présentation. Ces constatations sont discutées subséquemment.

### I.10.1 Le module BasicContentControl

SMIL définit des attributs de test relatives aux contraintes de terminaux cibles permettant la création d'alternatives dans l'élément SWITCH pouvant être lus par ces terminaux, parmi lesquels :

- ***systemBitrate***: spécifie la bande passante approximative, en Bps, disponible pour le système.
- ***systemComponent***: spécifie les caractéristiques/composants de l'agent utilisateur, le nombre de canaux audio, les codecs, le décodeur HW MPEG, etc.
- ***systemOverdubOrSubtitle*** : overdub | subtitle; spécifie si un doublage ou des sous-titres sont interprétés.



- ***system-overdub-or-caption*** : caption | overdub; détermine si l'utilisateur préfère le doublage ou des sous-titres quand l'option est disponible.
- ***systemScreenDepth*** : 1 | 4 | 8 | 24 | 32; spécifie la profondeur de la palette de couleur de l'écran en bits requis pour afficher l'élément.

Les attributs de tests de l'élément *Switch* présentés ci-dessus, sont très employés dans les systèmes d'adaptation à priori des documents SMIL, néanmoins ils viennent d'être concurrencés dans la nouvelle version du standard SMIL, par les fonctionnalités offertes dans le module *StateTest*, à savoir les tests par des expressions logiques comprenant des opérateurs logiques. Ce point est traité en (§ I.10.6) pour illustrer l'impact de l'utilisation de ce moyen sur l'optimisation d'énergie comparativement aux attributs de tests.

## I.10.2 Le module Prefetch Control

Ce module définit un élément et des attributs à cet élément, permettant de pré-télécharger un média à partir d'un serveur, avant son utilisation.

Les attributs de l'élément *Prefetch*:

- ***mediaSize***: permet de préciser la taille en octets ou en pourcentage;
- ***mediaTime***: permet de préciser le temps de la ressource;
- ***bandwidth***: permet de préciser la bande passante nécessaire pour son téléchargement.

Exemple:

```
<prefetch id="upimage" mediaSize="100%"  
src="http://www.example.org/up.gif"/>
```

Ce mécanisme fournit aux outils de développement ou aux auteurs prévoyants, la possibilité de planifier la récupération des données quand ils pensent qu'il y a de la bande passante ou du temps pour le faire. Il permet un gain sur le plan de la bande passante si elle est bien gérée et allouée, et une gestion efficace des ressources globales car l'objet se retrouve en mémoire une seule fois quelque soit le nombre de fois qu'il est joué.

Le pré-chargeement peut influencer positivement sur la consommation d'énergie. Si les pré-téléchargements sont planifiés en parallèle voire en blocs de courtes durées espacées dans le temps, le système peut envisager de mettre en veille la carte réseau durant ces périodes d'inactivité ce qui permettra de réduire la facture énergétique.

### I.10.3 Le module *AudioLayout*

L'élément région du module *BasicLayout* possède l'attribut *soundLevel* dénotant le volume associé à un média audio, exprimé comme valeur de pourcentage par rapport au volume d'origine. Cet attribut offre la possibilité de contrôler le volume d'un élément sonore placé dans une région *R*. Le principe est qu'un objet multimédia assigné à une région avec un attribut *soundLevel* explicite aura son audio interprété à l'intensité sonore relative donnée.

- Un réglage de '0%' joue le média silencieusement.
- Une valeur de '100%' jouera le média à son volume enregistré (en dB).
- De façon similaire, une valeur de '200%' jouera le média à peu près deux fois plus fort (6 dB) que son volume enregistré.

Le rapport avec l'énergie est qu'un objet audio avec une intensité élevée provoque un coût plus élevé en termes de consommation d'énergie. De ce fait si on arrive à contrôler le volume d'objet audio de façon qu'il soit audible sans plus, cela permet d'économiser une quantité d'énergie non négligeable.

### I.10.4 Module *Media Render Attributes*

Dans ce module, l'attribut *erase* permet de répéter la lecture d'un objet, autant de fois que l'on veut. Dans l'exemple de la **figure I.14**, chaque image est affichée successivement et continue d'être affichée jusqu'à la fin de la présentation grâce à l'attribut *erase="never"*, cela provoque une perte d'énergie, ce qui peut être évité si on indique une valeur explicite de temps.

```

<par>
  <seq>
    <par>
      
      <audio src="audio1.au" />
    </par>
    <par>
      
      <audio src="audio2.au" />
    </par>
    <!--...éléments parallèles...-->
    <par>
      
      <audio src="audioN.au" />
    </par>
  </seq>
</par>

```

Figure I.14 : Exemple sur l'élément Erase

### I.10.5 Attributs *accelerate*, *decelerate* et *autoReverse*

Deux types de manipulation du temps sont définies et applicables aux médias non linéaires

- 1) *accelerate*, *decelerate* = "un nombre decimal" permet d'accélérer (décélérer) le taux de progression simple de media, en gardant sa durée implicite.
- 2) *autoReverse* = "true/false" : permet de lire un média une fois avant et une fois après.
- 3) *Speed* = "nombre de [0,1]" contrôle la vitesse de l'élément actif,

**Exemple:** <animateMotion dur="10s" repeatCount="2" speed="2.0" path=... />

La cible se déplacera le long du tracé en 5 secondes (dur/speed simples = 10s / 2.0 = 5s) deux fois grâce a *repeatCount*.

L'utilisation des attributs *accelerate*, *decelerate*, et *speed* peut avoir un rapport sur la quantité d'énergie consommée lorsqu'elle est appliquée sur des médias non linéaires (audio et vidéo). La décélération d'un objet média réduit le taux de diffusion des images vidéo ; cela ressemble à la réduction de taux d'échantillonnage frame-rate d'une vidéo.

### I.10.6 Module StateTest

*StateTest* est un nouveau module; il permet l'utilisation d'une expression booléenne grâce à l'attribut *expr* qui sera évaluée par les appareils cibles, ceci contrairement aux modules *BasicContentControl* et *CustomTestAttributes* qui utilisent un mécanisme d'attribut de test qui définit des pré-conditions qui doivent être vraies. Plusieurs fonctions de tests ont été développées avec le module *StateTest*, afin de donner la possibilité de créer diverses expressions très complexes avec une taille de code réduite. Ci-après quelques exemples de fonctions:

- *boolean* *smil-audioDesc()* ;
- *string* *smil-operatingSystem()* ;
- *number* *smil-bitrate()* ;
- *string* *smil-overdubOrSubtitle()* ;
- *boolean* *smil-captions()* ;
- *boolean* *smil-required(string uri)* ;
- *boolean* *smil-component(string uri)* ;
- *number* *smil-screenDepth()* ;
- *boolean* *smil-customTest(string name)* ;
- *number* *smil-screenHeight()* ;
- *string* *smil-CPU()* ;
- *number* *smil-screenWidth()* ;
- *number* *smil-language(string lang)*.

#### Exemple :

Dans l'exemple de code de la **figure I.15**, le media audio n'est joué que si la connexion est plus rapide que 1Mbps.

```
<switch>
  <audio src="background.mp3" expr="not(smil-audioDesc()) and smil-
bitrate() > 1000000" />
  
  
</switch>
```

**Figure I.15: Exemple sur les attributs de test.**

Comparativement au mécanisme du test de contrôle de contenu, qui consiste à exécuter l'alternative, dont les attributs sont conformes à la configuration du client, l'expression logique *expr*, en plus qu'elle est plus pratique -car elle utilise des fonctions de test prédéfinies et peut comporter des opérateurs logique **and**, **or**, **not** etc-, elle offre l'avantage d'extension

par de nouvelles fonctions, autrement-dit, l'utilisateur peut créer les fonctions dont il a besoin, et les intégrer dans des expressions logiques. L'autre avantage est le fait que les valeurs retournées par des fonctions sont plus significatives car ces dernières s'exécutent en temps réel, et reflètent exactement et réellement l'état du système. Dans le but d'optimisation d'énergie, ces avantages peuvent être profités pour créer des fonctions qui vont se charger de surveiller et de contrôler les composants matériels de l'appareil afin de manipuler d'autres facteurs qui influent positivement sur la diminution de la consommation d'énergie.

### I.10.7 Synthèse

Nous remarquons à partir des constats émis que : premièrement, les fonctionnalités offertes par SMIL 3.0 se classent parmi les techniques d'adaptation structurelle puisque, elles sont applicables sur le code SMIL du document multimédia. Deuxièmement, nous avons dégagé des idées basées sur des paramètres sur lesquels nous pensons pouvoir intervenir dans notre solution pour optimiser la consommation d'énergie :

- Pré-télécharger des médias d'un serveur contenant les médias vers les clients par l'injection de l'élément *Prefetch* de module *PrefetchControl* ainsi que ces attributs dans le code source SMIL des documents multimédia ;
- Diminuer le volume de lecture d'un média audio ou vidéo d'un pourcentage de sorte qu'il puisse être écouté, cela est possible si on peut ajouter l'attribut *soundLevel* de module *BasicLayout* dans le code source de document
- Augmenter ou diminuer le taux de progression des médias non linéaires en injectant les attributs *Accelerate* *Decelerate* *Speed* dans les codes des documents SMIL.
- Définir des nouvelles fonctions de test que l'on pourra utiliser avec des expressions logiques de test permettant le contrôle et la modification de certains facteurs influant sur la consommation énergétique.

## I.11 Conclusion

Dans ce chapitre nous avons présenté, en premier lieu un aperçu sur les principaux modules de la version SMIL3.0. Nous avons ensuite discuté, des fonctionnalités en rapport avec l'optimisation de la consommation d'énergie.

Dans les chapitres suivants, nous présentons d'autres techniques d'adaptation définies dans la littérature permettant l'optimisation de l'énergie.

## II. Généralités sur l'adaptation

### II.1 Introduction

Les dispositifs mobiles connaissent actuellement un essor considérable, notamment au près du grand public, grâce au succès des téléphones cellulaires et des assistants numériques de poche. Les tendances récentes, motivées par des préférences d'utilisateurs se sont concentrées sur l'intégration des applications complexes comme les présentations multimédias dans un dispositif d'usage universel simple, souvent ayant pour résultat une consommation plus importante d'énergie et par conséquent, la réduction de la durée de vie de la batterie.

Malgré les progrès technologiques réalisés dans le domaine des batteries, l'autonomie de ces appareils reste toujours relativement limitée. De nombreuses recherches dans le cadre des systèmes de fourniture de contenu multimédia ont été menées afin de diminuer la consommation d'énergie.

Dans ce chapitre, nous explorons comment le facteur énergie est pris en compte par les standards de description de profil de l'utilisateur, et comment sa consommation est traitée par les processus d'adaptation.

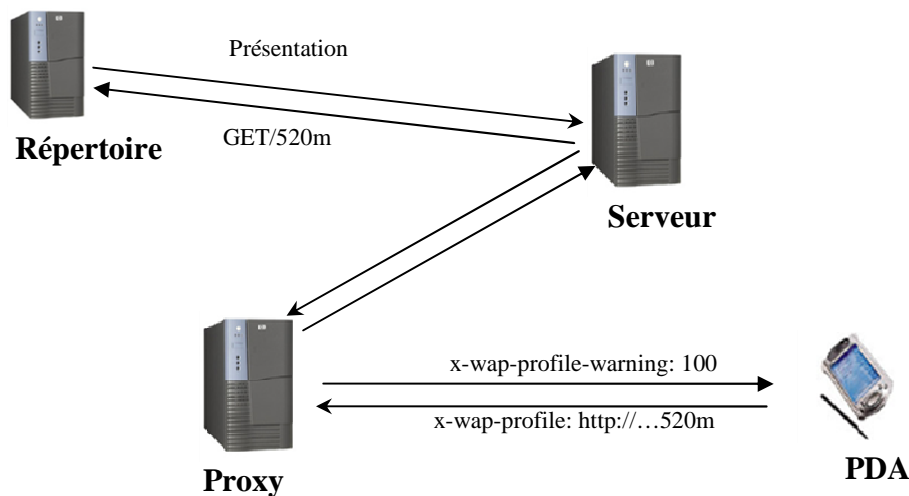
### II.2 Adaptation des présentations multimédia

L'hétérogénéité des réseaux, des terminaux et des contextes d'utilisation contraignent de plus en plus les distributeurs de contenus multimédia à envisager des méthodes de distribution intégrant des modules d'adaptation permettant, suite à une négociation client-serveur, d'adapter le contenu au contexte d'utilisation. Différentes techniques existent pour adapter les média audiovisuels.

#### II.2.1 Procédure d'adaptation

Concernant l'adaptation, les applications clientes transmettent certaines informations relatives à leur contexte. Cela est assuré par les routines de protocole utilisé pour l'accès au contenu. Par exemple, en utilisant le protocole HTTP [HTTP] ou RPC [RPC], l'application cliente transmet son profil à travers l'entête de la requête de la demande (voir **Figure II.1**).

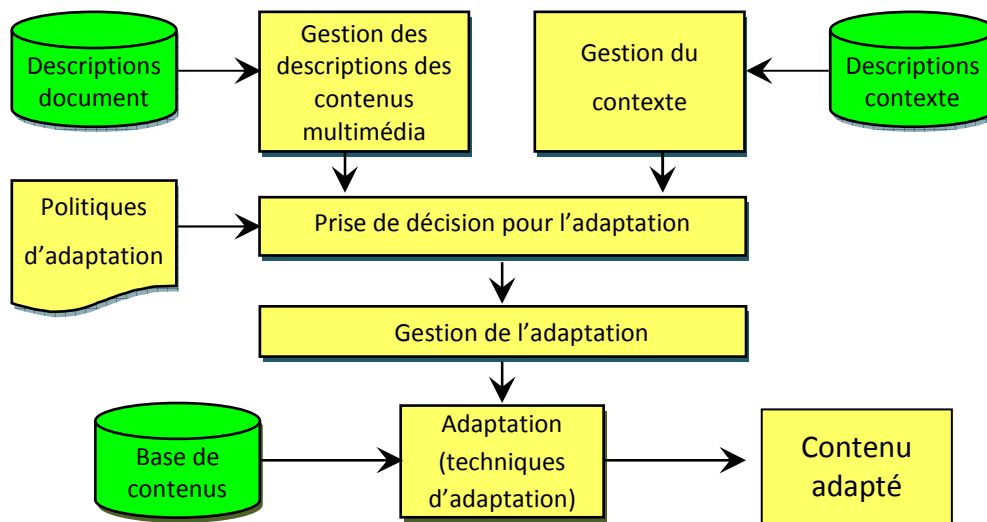
Dans ce cas, le module d'adaptation fait une correspondance automatique entre le profil du client et un ensemble de caractéristiques prédéfinies, que doit vérifier le contenu et la structure du document multimédia. Par exemple, la référence de l'appareil peut donner une idée sur les protocoles d'accès supportés. D'autres informations transmises par l'application cliente informent sur les formats supportés, que ce soit par l'application d'origine ou par les plugins installés. Les informations transmises par les applications clientes sont reçues par le module d'adaptation et font partie des dimensions du contexte utilisées par le processus de négociation de contenu [Ka,08].



**Figure II.1 : Interaction entre les éléments d'un système de fourniture de contenu multimédia**

## II.2.2 Architecture générale de fourniture de contenus multimédia adaptés

Nous distinguons un ensemble de fonctionnalités de base communes à presque toutes les architectures de fourniture et d'adaptation de contenus multimédia : La gestion du contexte utilisateur, la gestion des changements dynamiques dans le contexte utilisateur, la gestion des contenus multimédia, la gestion de la prise de décision pour l'adaptation et la gestion de l'adaptation. La **Figure II.2** illustre l'interaction entre les modules d'une architecture standard ; par la suite nous détaillons chaque fonctionnalité.



**Figure II.2 : Architecture de base d'un système de fourniture de documents multimédia**

### II.2.2.1 Gestion du contexte

C'est la gestion de toute caractéristique ou capacité des entités de l'environnement (utilisateur, terminal, réseau, environnement naturel de l'utilisateur, état et caractéristiques des modules réalisant l'adaptation), qui peut influencer sur la manière de consommer le contenu multimédia, et qui est susceptible de changer dans le temps, suite à des événements. Elle repose sur un langage de description des éléments du contexte de l'utilisateur et des outils pour collecter ces données, les comprendre et les analyser.

Les langages tels que **CC/PP [CCPP]** et **MPEG-21 [MPEG21]** décrivent des informations contextuelles telles que : les informations du profil personnel de l'utilisateur (par exemple le nom et les centres d'intérêt), les préférences de l'utilisateur en termes de présentation des contenus multimédia (par exemple la taille et résolution d'une image et les formats préférés d'un contenu vidéo), les capacités de son terminal (par exemple la taille de l'écran, les types des agents utilisateurs, la capacité de la batterie et la résolution), et les caractéristiques de son réseau d'accès et de son environnement naturel (par exemple la valeur de la bande passante et l'indice de luminosité).

### II.2.2.2 Gestion des contenus multimédia

Un système d'adaptation de contenus multimédia a besoin des descriptions des contenus riches afin de mieux prendre des décisions pour l'adapter. Ces données descriptives sont souvent des métadonnées. Une métadonnée (du grec *meta* "après" et du latin *data*



"informations"), est une donnée servant à définir ou décrire une autre donnée, son utilisation permet de faciliter la recherche d'information dans une page Web sur Internet.

Concernant la gestion des contenus multimédia, plusieurs raisons peuvent pousser un producteur de contenus multimédia à fournir une description, soit accompagnée du document multimédia qu'il souhaite partager, soit carrément incorporée au sein même du document multimédia lorsqu'il s'agit d'un contenu auto-descriptif comme un contenu SMIL [SMIL] ou MPEG-7 [MPEG-7].

### II.2.2.3 Gestion de la prise de décision

Souvent appelé moteur de prise de décision. Il consiste à déterminer si le contenu demandé par l'utilisateur nécessite une adaptation ou non, par rapport aux contraintes exprimées dans le profil de l'utilisateur. Pour ce faire, il est nécessaire d'analyser, d'une part, les informations contextuelles de l'utilisateur, et d'autre part, les informations relatives aux contenus multimédia demandés. La prise de décision peut être guidée par des politiques d'adaptation.

La prise de décision peut concerner des médias simples (élémentaires), comme la réduction de taille d'une image ou le changement d'encodage d'une bande son, elle est appelée adaptation du contenu, comme elle peut devenir assez complexe dans le cas d'adaptation d'un document multimédia composé de plusieurs médias tels qu'une vidéo, une image ou du texte ; on parle alors d'*adaptation sémantique*.

La prise de décision est définie pour être l'entité qui prend les décisions sur le type de l'adaptation à appliquer et également la valeur des paramètres nécessaires pour cette adaptation particulière. La prise de décision doit avoir accès à la description du contexte et du contenu. Elle a besoin également des informations des adaptateurs de ressource, c.-à-d. elle doit pouvoir identifier les disponibilités en termes de ressources des adaptateurs et leur paramétrage.

### II.2.2.4 Gestion de l'adaptation de contenus multimédia

Cette fonctionnalité est souvent intégrée à la fonction de gestion de la prise de décision. En effet, une fois la phase d'identification des adaptations à réaliser est achevée (gestion de la prise de décision), il s'agit de déterminer par la suite le ou les entités sur lesquelles seront réalisées les adaptations.

Le module responsable de la gestion de l'adaptation doit fournir des outils pour identifier le ou les modules ressources capables de réaliser les différents processus d'adaptation, les localiser, les instancier, et gérer le bon déroulement des adaptations. En d'autres termes, il faut instancier la suite des adaptations à réaliser en trouvant ou en choisissant les bons adaptateurs et assurer un bon déroulement de leur exécution.

Les outils d'adaptation de ressource sont les adaptateurs simples de médias, il y a deux catégories principales d'outils d'adaptation de ressource : les adaptateurs de ressource qui fonctionnent sur la base de la modification directe de ressource *bitstream*<sup>1</sup> et des adaptateurs qui fonctionnent sur la base de la conversion de ressource.

- Les anciens outils d'adaptations de ressource effectuent directement les modifications sur le bitstream, et cela, la plupart du temps en enlevant des paquets des données, tandis que les outils récents d'adaptation, convertissent complètement les médias. En d'autres termes, dans la modification directe du bitstream, sa structure demeure sans changement.
- Les adaptateurs de ressource ou convertisseurs de ressource peuvent changer la structure du bitstream après adaptation, par exemple en changeant le format (transcodage ex. la conversion d'une vidéo du format MP4 et format WMV) ou la modalité (transmodage ex. conversion d'une vidéo en une bande sonore) de la ressource. Ces concepts sont définis ci-après :
  - a) **Le transmodage** : Le concept de transmodage se rapporte à la conversion d'une ressource multimédia en changeant sa modalité initiale en une autre modalité. Il s'agit, par exemple, de transformer un texte en image de ce texte pour un client qui ne dispose pas de la police de caractère permettant l'affichage de ce texte. Par conséquent, nous définissons le transmodage comme un type d'adaptation de ressource qui change la modalité de la ressource originale.
  - b) **Le transcodage** : Nous définissons le transcodage en tant qu'une conversion de ressource qui change le format de codage de la ressource sans changer sa modalité, de telle sorte que le contenu d'origine garde le même aspect. Un exemple serait celui de convertir une vidéo du format **MOV** au format vidéo **AVI**.

---

<sup>1</sup>Le format binaire est le format des données contenues dans certains flux de bits utilisés dans une communication numérique ou de l'application de stockage de données.

- c) **La transformation** : La transformation est définie en tant qu'une conversion de ressource changeant des paramètres de ressource autre que le format et la modalité. Ce processus transforme un contenu en réduisant sa taille par exemple, cette catégorie concerne également des adaptateurs spécifiques qui reconstruisent un document multimédia de synchronisation de type **SMIL** ou **MPEG-21** par exemple et qui prennent en entrée un document multimédia et le transforment en un autre document multimédia.

### II.3 Architectures d'adaptation au contexte

On a constaté qu'il existe plusieurs solutions architecturales répondant à la problématique d'adaptation de contenus multimédia au contexte utilisateur. Cette constatation est due au fait que les applications relatives au multimédia, les types de contenus multimédia ainsi que les besoins d'adaptation varient énormément : à chaque problème une solution. Les différentes solutions architecturales classifiées selon la partie qui fait l'adaptation sont définis dans ce qui suit :

**a. Adaptation coté client** : L'approche consiste à envoyer la totalité du document original à la plateforme demandeuse, et c'est au niveau de cette dernière que l'opération d'adaptation va s'effectuer par les composants logiciels intégrés au SE. L'adaptation est faite vis-à-vis du contexte de l'utilisateur et aux contraintes du terminal. Par exemple, un **PDA** accédant à une information qu'il est incapable de présenter peut l'adapter par réduction du nombre de couleurs, et en redimensionnant chaque média.

Néanmoins, notons qu'une telle adaptation peut être parfois source de problèmes. Plus concrètement, la charge supplémentaire imposée à un terminal disposant de capacités et de ressources limitées (mémoire, CPU, stockage, énergie, etc.) peut s'avérer assez importante.

**b. Adaptation coté serveur** : L'adaptation orientée serveur consiste, quant à elle, à effectuer les conversions nécessaires des medias ou de structure de document au niveau de serveur de telle sorte qu'il en résulte un document satisfaisant toutes les contraintes exprimées dans le profil envoyé par le client.

Cette architecture suppose généralement la présence de canaux de retour (client → serveur), par lesquels sont renvoyées des informations. Ces informations permettent à la source de mesurer l'efficacité et la pertinence de sa stratégie d'adaptation et éventuellement

de la peaufiner. Il faut donc que les entités coopèrent pour au moins assurer ce retour d'informations (statistiques par exemple).

**c. Adaptation coté Proxy :** L'architecture orientée Proxy consiste à effectuer l'adaptation au niveau d'un ou de plusieurs intermédiaires ce qui évite d'utiliser les ressources du client (respectivement du serveur), autrement que pour la consommation (respectivement la production et le stockage) du contenu. Cette solution se prête également parfaitement à l'évolutivité du système d'adaptation. UMA [UMA], WAM [WAM] sont des exemples d'architectures qui réalisent l'adaptation au niveau proxy tandis que APPAT [La,05] et DCAF [Be,06] réalisent l'adaptation au niveau de plusieurs Proxy.

**Adaptation selon l'architecture P2P :** Dans ce genre de réseau, chaque nœud peut être à la fois adaptateur, client et producteur de contenus multimédia. Le modèle architectural P2P peut également pallier aux problèmes liés à l'adaptation au niveau du client et au niveau du serveur. Il apporte une souplesse et une évolutivité dans l'adaptation. Ce modèle introduit néanmoins une gestion des adaptateurs plus complexe compte tenu de la volatilité des nœuds. Ce modèle architectural se prête également au passage à l'échelle par l'ajout et la suppression de nouveaux nœuds du réseau, à condition d'avoir une plateforme d'exécution qui le permet. Par exemple, les plates-formes P2P implémentées avec JXTA [JXTA] ne passent pas à l'échelle car elle ne peut pas gérer l'évolution dynamique de la structure du réseau. Un système comme MAPS pour Media Accelerating Peer Services [Li,03] étend les infrastructures Peer to Peer existantes avec quelques modules qui permettent de personnaliser les fonctions de recherche et de livraison de documents. Le système SATO [SATO] intègre également une logique d'adaptation basée sur le modèle architectural P2P où l'adaptation est partiellement réalisée au niveau des nœuds du réseau.

## II.4 Outils de description du profil utilisateur

### II.4.1 MPEG-21

**MPEG-21** (ISO/IEC 21000) est une norme ou cadre de travail (*Multimedia Framework*) multimédia élaboré par le groupe MPEG (*Moving Picture Experts Group*) (ISO/IEC JTC1 SC29 WG11) [MPEG] qui vise à définir un cadre ouvert qui garantit une

utilisation (livraison et consommation), transparente et enrichie de ressources multimédia pour une vaste gamme de lecteurs, de réseaux et de terminaux. Il repose sur deux concepts essentiels :

- **Le concept d'élément numérique (Digital Item)** : Un objet numérique structuré avec une présentation, une identification et des métadonnées standards. C'est la plus petite unité du contenu pour la transaction et la consommation ;
- **Le concept d'utilisateurs (Users)** : qui représente n'importe quelle entité qui interagit dans l'environnement MPEG-21, ou qui utiliserait un Digital Item.

MPEG-21, répond à la nécessité de disposer d'un outil non-propriétaire et normalisé de gestion numérique des droits ou **DRM**. Elle est composée de plusieurs éléments permettant de décrire les droits couverts par les licences :

- un langage de description des droits (*Rights Expression Language*),
- un dictionnaire de données sur les droits (*Rights Data Dictionary*),
- un système de déclaration des objets numériques (*Digital Item Declaration*)

et met en œuvre une méthodologie de notification de l'utilisation des œuvres musicales, audiovisuelles ou graphiques. Elle permet de gérer l'ensemble des droits numériques, des permissions et des restrictions liées à un contenu tout au long de la chaîne qui va du créateur ou du producteur de ce contenu jusqu'au consommateur. Cette norme constitue un système de gestion des droits numérique ouvert et interopérable à part entière. La norme est divisée en 18 modules, la partie qui est destinée aux procédures d'adaptation est définie dans le module sept (7) intitulé, *Adaptation des Items Digitaux (DIA)*, cette partie contient des éléments permettant de décrire le contexte de l'utilisateur et plus particulièrement, les informations énergétiques de son terminal.

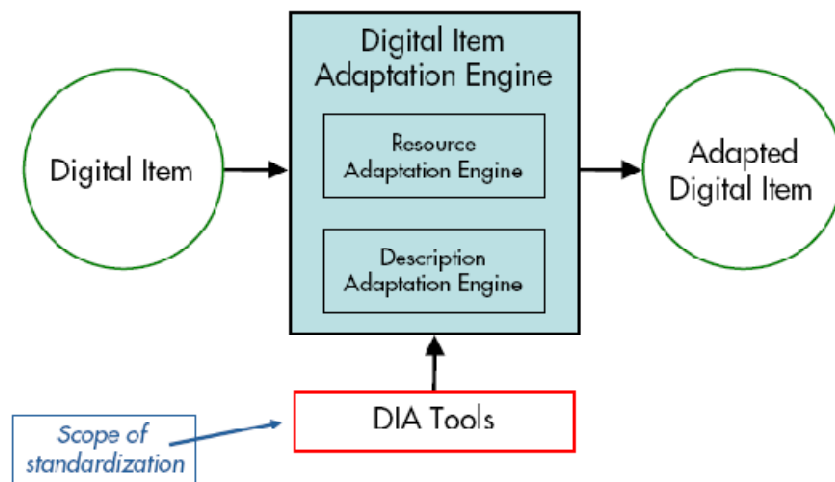
#### II.4.1.1 Adaptation des Items Digitaux

La partie adaptation des éléments digitaux (Partie 7 : *DIA* pour *Digital Item Adaptation*) spécifie la syntaxe et la sémantique des outils qui peuvent être utilisés pour aider l'adaptation des Digital Items, satisfaire les contraintes de transmission, de stockage et de consommation aussi bien que la gestion de qualité de service (QoS), et fournit un ensemble d'outils, laissant décrire les caractéristiques et les capacités des réseaux, des terminaux et des environnements aussi bien que des préférences des utilisateurs.

- **Description d'environnement d'utilisation (UED)** : Un ensemble de descripteurs et de méthodologies pour décrire le contexte de l'utilisation ;

- **Qualité du service d'adaptation (AQoS)** : spécifie le rapport entre les contraintes, les opérations faisables d'adaptation satisfaisant ces contraintes, et les utilités associées (par exemple qualité subjective et/ou débit binaire) ;
- **Description universelle des contraintes (UCD)** : permet la description des contraintes sur des opérations d'adaptation, et un contrôle sur les opérations qui sont exécutées sur le contenu.

Nous mettons l'accent ci-après sur les outils de description de l'environnement d'utilisation (*UED*), car ils constituent la partie qui pourrait répondre le mieux à nos besoins de représentation et de description du contexte utilisateur. En effet, ces outils ( voir **figure II.3**) fournissent une information descriptive concernant les diverses propriétés de l'environnement d'utilisation, notamment en rapport avec la capacité et la jauge de la batterie.



**Figure II.3 : Illustration d'adaptation des éléments numériques**

#### II.4.1.2 Descriptions d'environnement d'utilisation (UED)

Les outils de description d'environnement incluent la description des possibilités et les caractéristiques du terminal et du réseau, ainsi que les caractéristiques de l'utilisateur et les caractéristiques de son environnement. De telles descriptions fournissent des données fondamentales à n'importe quel moteur d'adaptation.

1. **Capacités des terminaux** : Les capacités d'un terminal doivent être décrites en termes de possibilités de réception et de transmission. Une telle description est employée pour

satisfaire la consommation et des contraintes de traitement d'un terminal particulier. Les catégories principales sont brièvement décrites ci-dessous :

- a. **Possibilités de codec** : Spécifie les formats de codage qu'un terminal est capable de coder ou de décoder, les formats de codage et/ou de décodage : image, vidéo, audio, formats de graphiques, et paramètres reliés à la modalité : par exemple, débit binaire maximal.
- b. **Caractéristiques d'entrée-sortie** : Les possibilités d'affichage, par exemple, taille d'affichage, résolution, modèles de couleurs supportés, et profondeur des Pixels. De même que les capacités de rendement audio, par exemple, la puissance de sortie du signal, les gammes de fréquence, types d'échantillonnage. Par ailleurs, diverses propriétés de plusieurs types de dispositifs en entrée, par exemple, la souris, le stylo, peuvent être décrites.
- c. **Propriétés du terminal** : Les propriétés du terminal peuvent être : la classe d'appareils (ex l'assistant personnel (*PDA*), PC), le temps estimé avant épuisement de sa batterie, sa capacité de stockage (ex, la taille du disque et sa vitesse de transfert), les caractéristiques d'affichage i.e. la taille de l'écran ou sa résolution, la fréquence de la CPU, etc.

Dans notre cas, effectuer une opération d'optimisation nécessite au préalable des informations sur les propriétés du terminal et plus précisément des informations sur l'énergie, comme la quantité d'énergie restante. Ces informations sont des attributs de différents types appartiennent à un élément digital de schéma XML de la norme appelé *TerminalCapabilityBaseType* comme montré dans la **figure II.4**. les attributs *averageAmpereConsumption* *batteryCapacityRemaining* et *batteryTimeRemaining* de type entier permettent de savoir la consommation en ampère moyenne, le reste de capacité de la batterie, et le reste de temps de la batterie respectivement, tant que l'attribut de type boolean *runningOnBatteries* permet de savoir si le client est sous batterie.

```

<!-- Definition of PowerCharacteristics -->
<!-- ##### -->
<complexType name="PowerCharacteristicsType">
<complexContent>
<extension base="dia:TerminalCapabilityBaseType">
<attribute name="averageAmpereConsumption" type="integer"
use="optional"/>
<attribute name="batteryCapacityRemaining" type="integer"
use="optional"/>
<attribute name="batteryTimeRemaining" type="integer"
use="optional"/>
<attribute name="runningOnBatteries" type="boolean"
use="optional"/>
</extension>
</complexContent>
</complexType>

```

**Figure II.4 : Informations énergétiques dans l'UED**

**2. Caractéristiques de réseau :** MPEG-21 considère deux principales catégories de que sont :

**a. Capacités du réseau (attributs de charge statique) :** Les capacités du réseau sont décrites par des attributs statiques comme la capacité maximale et minimale de bande passante du réseau, le nombre de paquet que le réseau peut fournir en séquence, le traitement d'erreurs c'est-à-dire, si les paquets contenant des erreurs sont fournies ou supprimées de réseau ou encore la taille maximum d'un paquet en bit.

**b. Les conditions réseau (dynamique et variable dans le temps) :** Les conditions réseau contiennent entre autres : la largeur maximale, minimale et/ou moyenne de la bande passante disponible, le taux de perte dans un paquet, le taux d'erreur en bit, et le délai d'attente, etc.

**3. Caractéristiques d'utilisateur :** Les caractéristiques de l'utilisateur sont classées comme suit :

**a.** Les informations sur l'utilisateur, ses préférences et son historique tels que le type de musique préféré, la langue utilisée, etc.

**b.** Les préférences de présentations, dénote de la configuration préférée pour l'utilisation par exemple d'une ressource audio, pour l'affichage d'une image, etc.



- c. Les caractéristiques d'accessibilité qui sont relatives aux handicaps des personnes qui reçoivent le contenu multimédia (exemple : cécité, surdité),
- d. Les caractéristiques de la position qui couvrent la description de la mobilité et de la destination dans le temps.

**4. Caractéristiques normales d'environnement :** L'**UED** spécifie aussi bien les caractéristiques normales de l'environnement comprenant l'emplacement et la période de l'utilisation d'un Digital Item, que des caractéristiques qui concernent des aspects audiovisuels. Le premier type concerne l'endroit où se trouve l'élément digital et l'instant de son utilisation ; le second type décrit les attributs audiovisuels de l'environnement de l'utilisateur et qui peuvent influencer la réception du contenu. Par exemple, pour les ressources audio, MPEG-21 décrit le niveau du bruit de l'environnement, cette information peut aider le moteur d'adaptation Digital Items pour effectuer une meilleure adaptation.

#### II.4.2 CC/PP

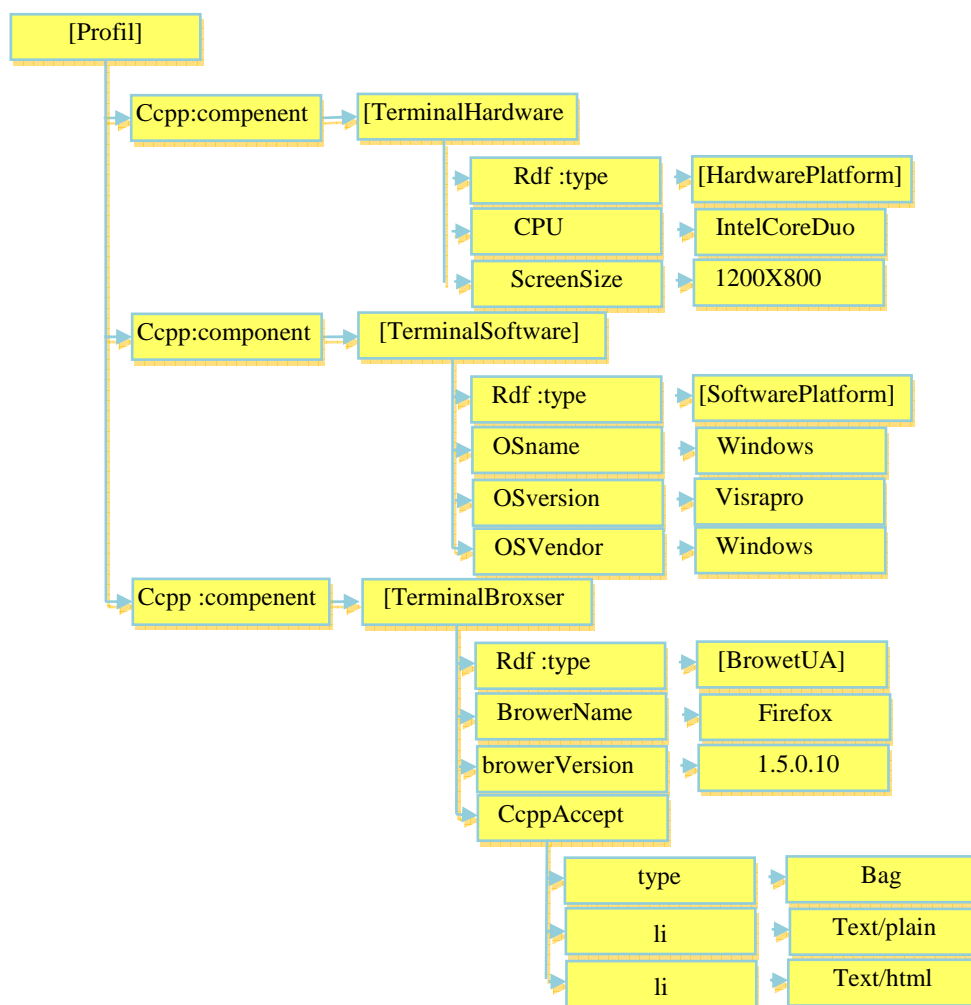
**CC/PP** (*Composite Capability/Preference Profiles*) [**CCPP**] est né des efforts de standardisation de World Wide Web Consortium (*W3C*) [**W3C**], c'est un langage standard de description de profils basé sur Ressource Description Framework (*RDF*) qui est notamment utilisé dans le standard **SMIL**. Le but de CC/PP est d'offrir les moyens pour que le client exprime les capacités logicielles et matérielles de son terminal ainsi que ses préférences.

Un profil CC/PP est une description des capacités d'un appareil ou des préférences d'un utilisateur. Cette description, est employée pour guider l'adaptation du contenu présentée à ce dispositif et utilisateur. Pour ce faire, les profils CC/PP utilisent le cadre de description des ressources (*RDF*), qui permet de représenter tous types d'informations sur Internet. Le vocabulaire d'un profil CC/PP se base également sur des identificateurs (des adresses URI) qui désignent des capacités et des préférences particulières.

Basé sur XML [**XML**], RDF a été conçu pour fournir un langage de description de métadonnées [**RDF**]. CC/PP utilise un RDF, qui lui permet de développer et d'implémenter de manière indépendante un vocabulaire, et lui offre les moyens de la réutilisation et de l'extensibilité de ses métadonnées.

Le cadre de travail de CC/PP définit une structure et un vocabulaire. Une structure définit l'organisation du profil sous forme d'arbre à 2 niveaux de hiérarchie. Cette structure inclut les composants (*components*) et les attributs (*attributes*). Un profil contient au moins un composant et chaque composant contient au moins un attribut. Les composants possibles sont la plateforme matérielle (*TerminalHardware*), la plateforme logicielle (*TerminalSoftware*), l'application qui récupère et présente le contenu à l'utilisateur tel que le navigateur (*TerminalBrowser*). Chaque composant est un sous-arbre dont les branches représentent les capacités et les préférences de ce composant.

RDF permet la modélisation d'un grand nombre de structures de données. L'architecture d'un profil CC/PP est illustrée par la **figure II.5**.



**Figure II.5 : La structure d'un profil utilisateur en CC/PP**

Le code illustré par la **figure II.6**, est un exemple d'un profil CC/PP. Trois composants sont décrits : le premier composant contient la description des informations sur les propriétés hardware du terminal (ex : fréquence CPU), le deuxième composant contient des informations concernant les propriétés software du terminal (ex : format supportés), et le dernier composant qui contient les informations sur les propriétés de navigateur Web du terminal (ex : nom).

```
<?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:example="http://www.example.com/schema#">

  <rdf:Description
rdf:about="http://www.example.com/profile#MyProfile">

    <ccpp:component>
      <rdf:Description
rdf:about="http://www.example.com/profile#TerminalHardware">
        <!-- TerminalHardware properties here -->
      </rdf:Description>
    </ccpp:component>

    <ccpp:component>
      <rdf:Description
rdf:about="http://www.example.com/profile#TerminalSoftware">
        <!-- TerminalSoftware properties here -->
      </rdf:Description>
    </ccpp:component>

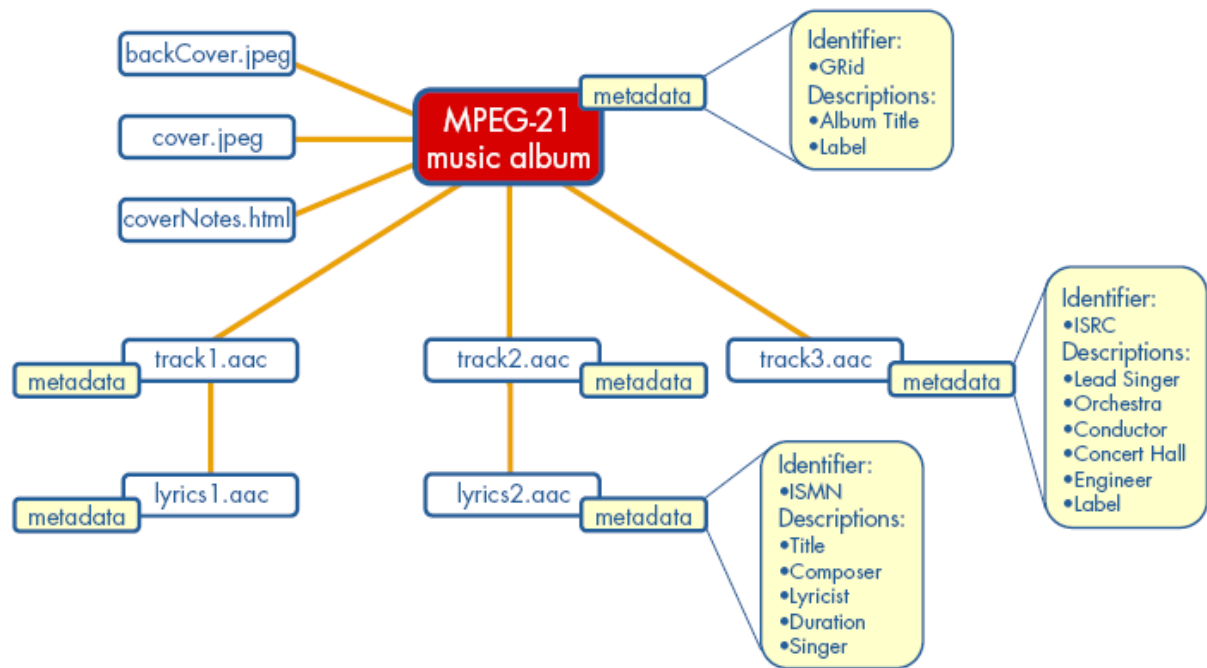
    <ccpp:component>
      <rdf:Description
rdf:about="http://www.example.com/profile#TerminalBrowser">
        <!-- TerminalBrowser properties here -->
      </rdf:Description>
    </ccpp:component>

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

**Figure II.6 : Exemple d'un profil en CC/PP**

## II.5 Comparaison entre CC/PP et MPEG-21

**MPEG-21** est un environnement de spécification, adaptation et fourniture de contenu multimédia, le mot contenu ici signifié des éléments média distincts (image, vidéo, audio etc.), il permet de spécifier un élément média sous une forme d'élément digital (voir **Figure II.7**) :



**Figure II.7 : Exemple d'un élément digital**

Dans MPEG-21, on constate qu'il n'y a pas de notion de présentation. Au fait, on ne peut pas manipuler des présentations multimédia en tant qu'entités avec leurs dimensions spatiales et temporelles. Par conséquent, la synchronisation temporelle entre plusieurs éléments média et leur répartition spatiale sur l'écran n'est pas gérée. Cependant, cet environnement permet d'exprimer et de caractériser la notion d'énergie lors de la description du contexte de manière quantitative.

Au contraire de MPEG-21, le standard CC/PP, est le standard adopté par le W3C pour la caractérisation des profils des présentations multimédia notamment ceux spécifiés avec SMIL. Il permet d'exprimer les informations de contexte d'utilisation, mais il ne permet pas d'exprimer les caractéristiques énergétiques du terminal ni de les manipuler.

## II.6 Conclusion

Dans ce chapitre, après avoir effectué un tour d'horizon sur les principales fonctionnalités des systèmes d'adaptations de contenus multimédia, nous avons exploré la prise en compte du facteur gestion de l'énergie par les langages de description de contexte utilisateur.

Nous avons constaté que le framework **MPEG-21** à travers son Module 7, contrairement à **CC/PP** permet d'exprimer le profil énergétique du client. Cela pourrait être exploité pour l'optimisation de la gestion de la ressource énergie lors du processus d'adaptation d'une présentation multimédia à condition de trouver des mécanismes pour étendre son applicabilité à ce type de documents.

Il reste à explorer maintenant les techniques d'adaptation à appliquer sur le contenu du média et sur l'environnement matériel et logiciel du client afin d'optimiser la consommation d'énergie. Cela fait l'objet du prochain chapitre.

## III. Techniques d'optimisation d'énergie

### III.1 Introduction

L'essor de la téléphonie mobile et, plus globalement, des appareils nomades comme les PDA, les téléphones cellulaires ou les micro-portables, s'accompagne d'un besoin accru en ressources énergétiques (batteries, piles, etc.). En effet, la durée de vie de la batterie est une métrique importante à considérer pour des appareils mobiles, et sa maximisation revient à réduire la consommation de l'ensemble de ses composants : CPU, mémoire, rétro-éclairage, etc. Dans ce contexte, plusieurs techniques d'optimisation de l'énergie ont fait l'objet de travaux dans la littérature. Nous présentons dans ce chapitre un panorama de ces techniques.

### III.2 Techniques d'optimisation de la consommation d'énergie

#### III.2.1 Adaptation du contenu

Dans [Ta,04] les auteurs proposent deux techniques pour économiser la consommation de l'énergie lors de l'exécution d'un flux vidéo sur un terminal mobile. Ils proposent en premier lieu de réduire la qualité visuelle avec un transcodeur ; cela est possible par l'ajustement d'un ensemble de paramètres (Débit binaire, Nombre de trames et taille d'images). Cette opération est réalisée par un proxy sur un nœud intermédiaire (ou sur le serveur du contenu), nommé *transcoding proxy*.

En second lieu, les auteurs proposent une technique qui ressemble au streaming, elle consiste à transférer périodiquement une grande partie de données vidéo à travers le réseau sans fil à un débit élevé. Le terminal d'utilisateur reçoit chaque fragment de données de paquet au taux de transmission le plus haut possible, puis stocke les données dans le buffer local, à ce moment là, il met sa carte réseau sans fil sous tension jusqu'à ce que le buffer devient vide. L'opération se répète jusqu'à ce que toutes les données vidéo soient transmises ; cette technique est nommée *buffered playback*.

Dans le même travail, il est proposé une technique pour estimer les valeurs de chaque paramètre (telles que la taille de l'image, la fréquence d'image et le débit binaire). Cela permet d'envisager la lecture d'une vidéo d'une durée spécifique en considérant la durée de vie restante de la batterie. Un modèle de consommation d'énergie pour un terminal

d'utilisateur a été développé, dans lequel les valeurs des paramètres suscités peuvent être estimées automatiquement.

Il est proposé aussi un mécanisme de contrôle de qualité de service QoS qui permet la lecture de multiples segments vidéo avec des qualités différentes, basées sur les préférences de l'utilisateur. Au fait c'est un algorithme pour distribuer la quantité d'énergie entre les multiples segments vidéo résidents. Quand un utilisateur spécifie l'importance relative parmi les segments vidéos et la propriété visuelle préférée (proportion parmi la taille d'image, le taux d'échantillonnage et le débit binaire), pour chaque segment, l'algorithme détermine la qualité de lecture de chaque segment vidéo de sorte que la lecture de la vidéo peut durer le temps relatif à la quantité d'énergie allouée.

### III.2.1.1 Effet de quelques paramètres sur la consommation d'énergie par les objets vidéo

Dans cette section, nous exposons les résultats de trois expériences, reportés dans [Li,10]. Dans l'expérience 1 et 2, les auteurs ont mesuré la consommation d'énergie de trois films de genres différents : animation, action, et romancier.

- Dans l'expérience 1, les films examinés sont « *the Simpsons* », « *Spider Man 3* », et « *August Rush* » respectivement, avec une valeur fixe de résolution, de fréquence d'image et du débit binaire, et en assumant différents codecs de compression pour un même fichier conteneur WMV.
- Dans l'expérience 2, cinq films différents ont été testés pour chaque genre, avec une valeur fixe de la résolution, de la fréquence d'image, et du débit binaire, mais codé par divers codec et en considérant différents formats de fichier.

Dans l'expérience 3, ils ont modifié la résolution, la fréquence d'image, et le débit binaire un à la fois pour étudier leurs effets sur la consommation d'énergie. Les résultats des trois expériences ont permis de tracer les histogrammes suivants :

#### *Expérience 1 : Type de codec*

Comme le montre la **figure III.1**, on remarque qu'avec les mêmes paramètres de résolution, de débit binaire, et de fréquence d'échantillonnage, les films codés en DivX5, H.263+, et XviD consomment moins d'énergie par rapport à ceux codés en WMV3 et H.264. Cependant, il est à constaté que es films codés en H.264 on une meilleure qualité visuelle.

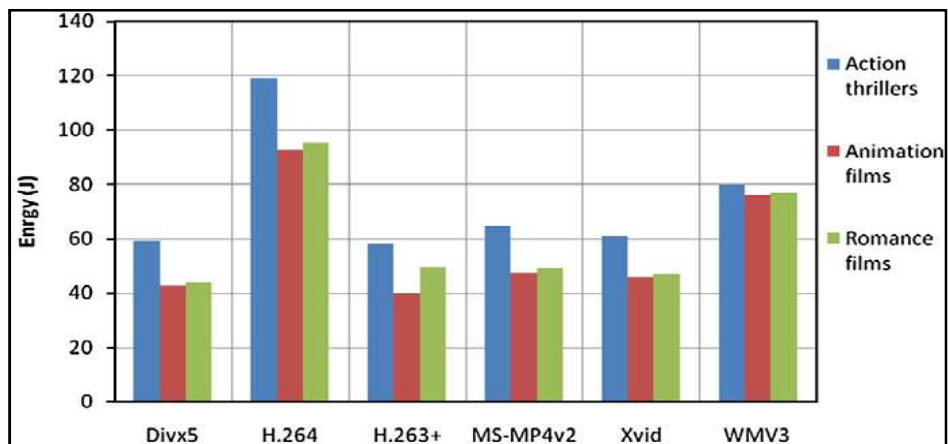


Figure III.1: Effet de type de codec sur la consommation d'énergie

*Expérience 2 : Format du fichier*

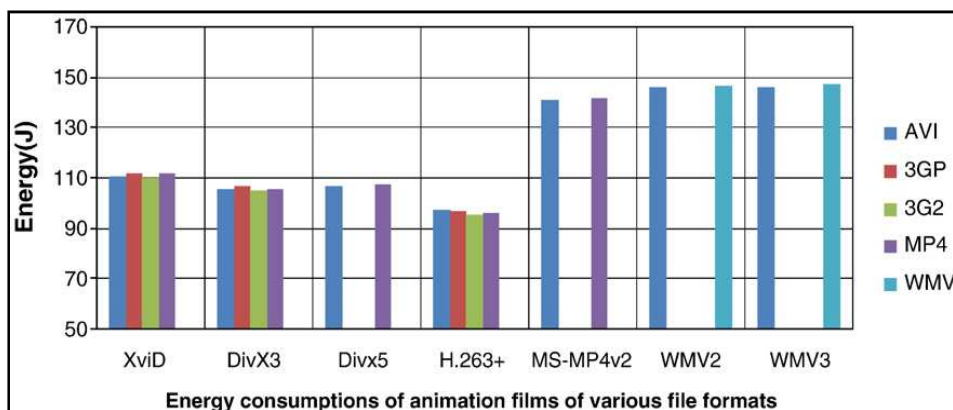


Figure III.2 : Effet de format de fichier sur la consommation d'énergie

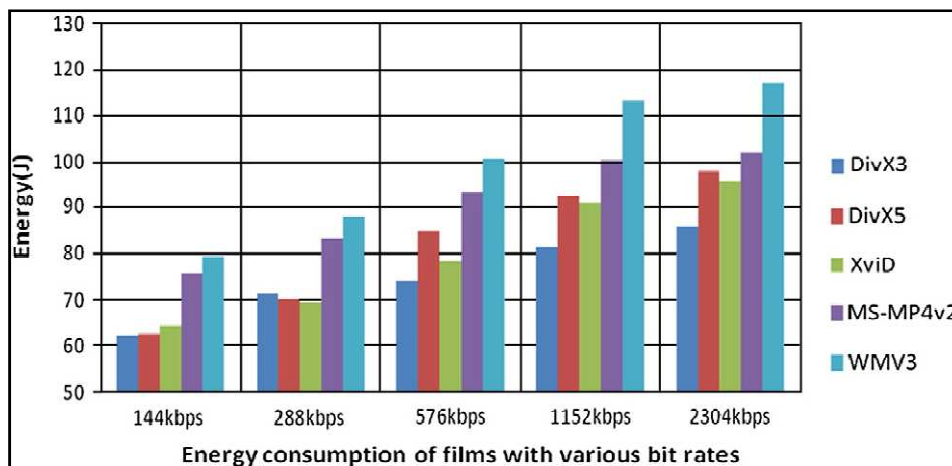
Dans l'expérience 2, la consommation d'énergie est mesurée pour des films codés dans divers formats visuels : 3GP, 3G2, AVI, MP4, WMV. Les films sont codés en XviD, DivX3, DivX5, MS MPEG-4v2, H.263+, WMV2, et WMV3 avec une résolution de 352×288, une fréquence d'image de 25 images/s, un débit binaire de 512 Kbps, et un temps de lecture avoisinant les 5 minutes pour chaque film.

Selon les histogrammes de la **figure III.2**, nous observons que les films codés par le même codec ont la même consommation d'énergie indépendamment des formats de fichier. On peut déduire que le format n'a pas un grand effet sur la consommation d'énergie pour les équipements mobiles.

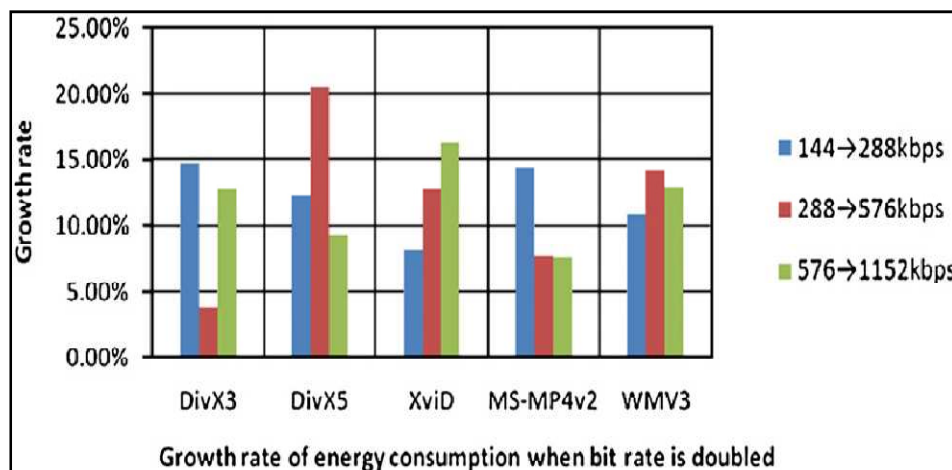


*Expérience 3 : débit binaire, fréquence d'image et résolution*

Pour le premier test de l'expérience 3, les auteurs ont mesuré la consommation d'énergie pour différentes valeurs de débit binaire (voir **figure III.3 a**), en le variant de 144 Kbps à 288 Kbps. De la même manière, ils ont calculé les variations de la consommation d'énergie (voir **figure III.3 b**). La moyenne de taux de croissance de la consommation d'énergie est estimée à 11.85%.

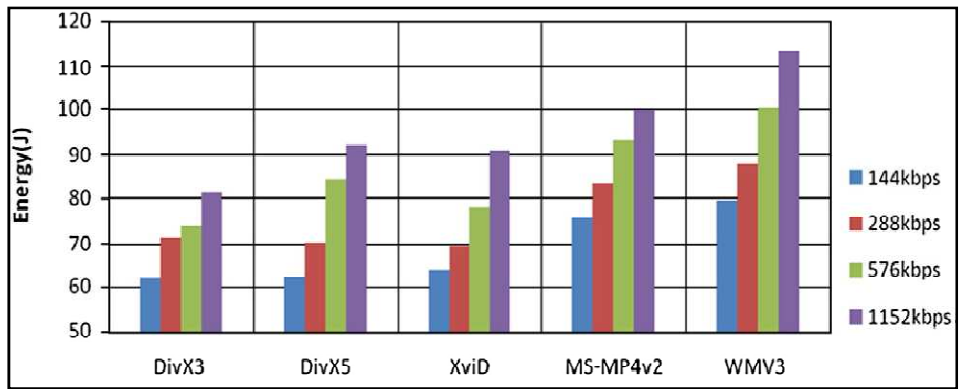


a)

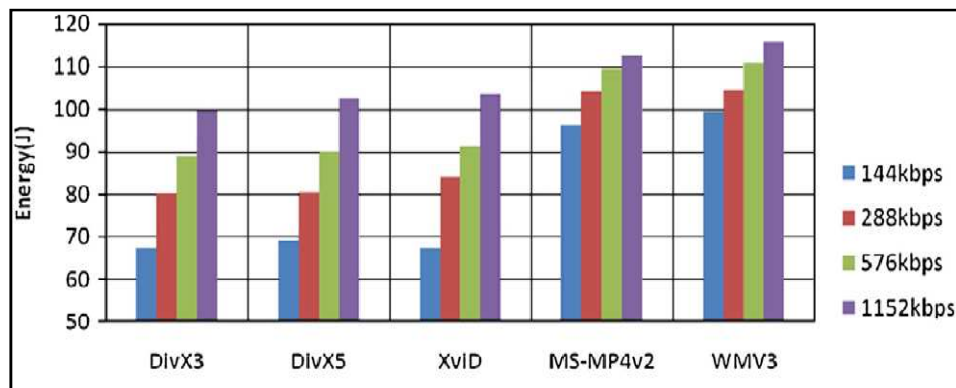


b)

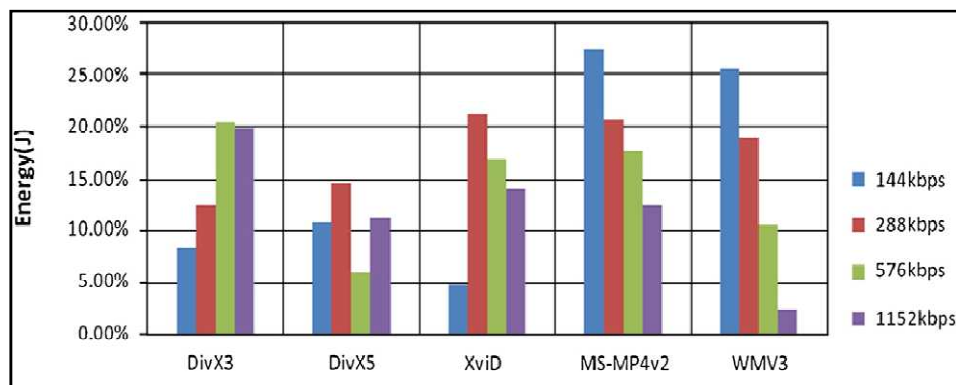
**Figure III.3: Effet de débit binaire des vidéos sur la consommation d'énergie**



a)



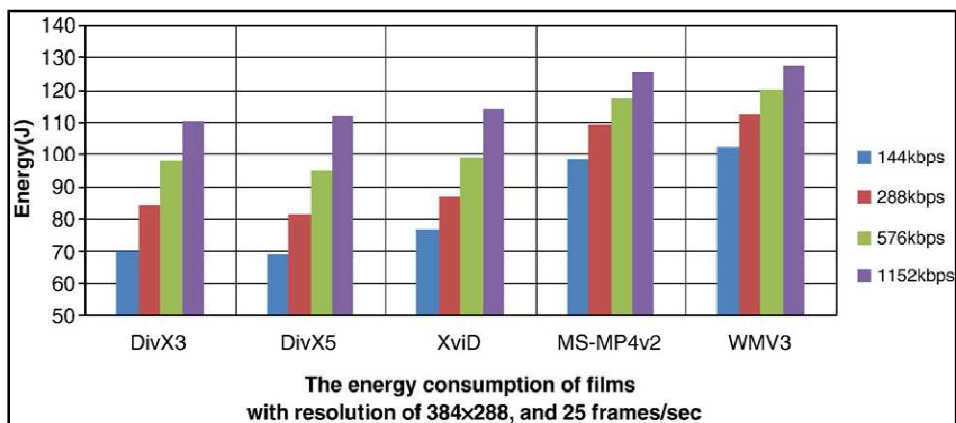
b)



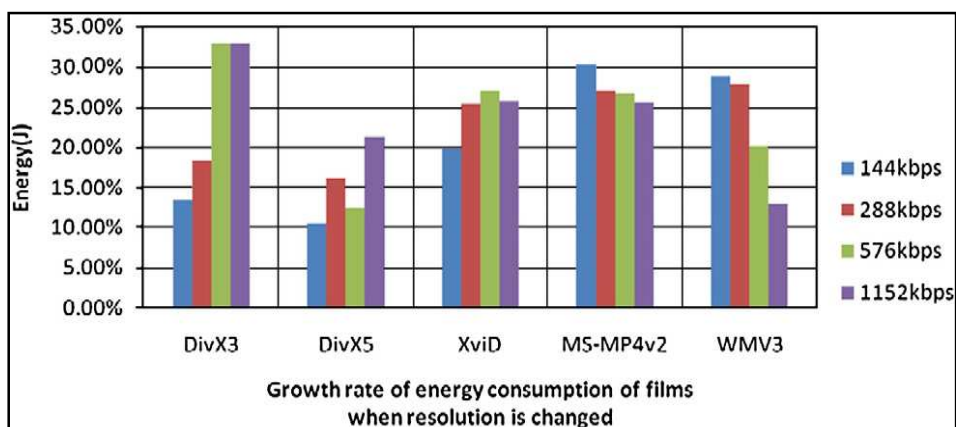
c)

**Figure III.4 : Effet de la fréquence d'image des vidéos sur la consommation d'énergie**

Le second test de l'expérience 3, compare la consommation d'énergie des films codés avec une résolution de 320×240 (voir **figure III.4 a)**), et 25 Images/s, avec celle des films codés avec la même résolution mais avec une fréquence d'échantillonnage plus rapide (30 Images/s) (voir **figure III.4 b)**). Il a été démontré que le taux de croissance moyen de la consommation d'énergie est de 15.76% quand la fréquence d'échantillonnage est changée de 25 à 30 Images/s (voir **figure III.4 c)**).



a)



b)

**Figure III.5 : Effet du facteur résolution sur la consommation d'énergie**

Dans le troisième test de l'expérience 3, en considérant des résolutions différentes (voir **figure III.5 a)**), il a été constaté que le taux de consommation d'énergie augmente de l'ordre de 22,44% lorsqu'on passe d'une résolution  $320 \times 240$  à  $384 \times 288$  (voir **figure III.5 b)**).

### III.2.1.2 Effet de quelques paramètres sur la consommation d'énergie des éléments audio.

Comme dans le cas des objets vidéo, nous allons citer l'effet de quelques paramètres sur la consommation d'énergie lors de la délivrance des média audio. Pour ce faire, nous présentons les résultats expérimentaux reportés dans [Li,07] et [Li,08].

Dans ces expériences, les auteurs ont utilisés plusieurs codecs AAC, AC-3, MP2, MP3, Ogg et WMA. Les résultats de la consommation d'énergie sont affichés en **figure III.6**.

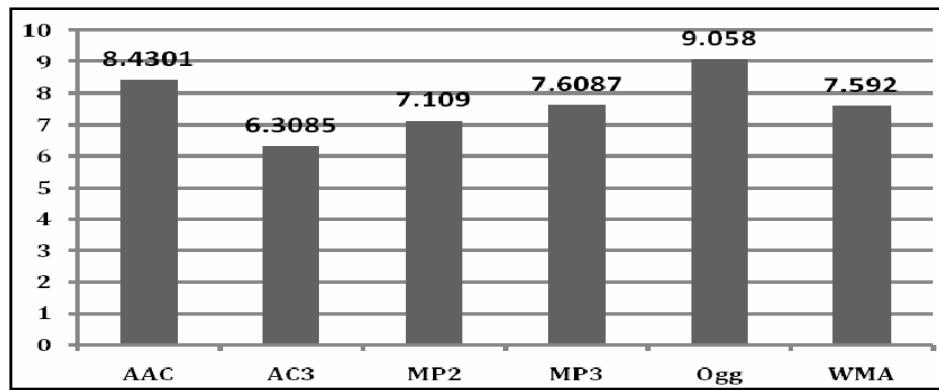


Figure III.6 : Energie consommée par un fichier audio codé dans différents formats

A partir de cet histogramme, nous observons que un objet audio codé en Ogg consomme le plus d'énergie, suivi du codec AAC. Bien que celui codé en AC3 consomme le moins d'énergie, néanmoins la qualité du son est la plus mauvaise.

Dans l'expérience qui suit, deux types de paramètres ont été manipulés : le débit binaire et le taux d'échantillonnage. Ces paramètres ont été ajustés pour observer leur influence sur la consommation d'énergie et détecter quel paramètre a le plus d'influence. Les résultats expérimentaux sont résumés dans les **Tableaux III.(1,2,3,4)**.

	64kbps	96kbps	128kbps	160kbps	192kbps
AAC	8.3624	8.3875	8.4301	8.4855	8.5032
AC3	6.2322	6.2855	6.3085	6.3588	6.3601
MP2	7.022	7.088	7.109	7.126	7.138
MP3	7.56	7.595	7.6087	7.6233	7.6598
Ogg	8.788	8.955	9.058	9.266	9.388
WMA	7.366	7.408	7.592	7.6533	7.8044

Tableau III.1 : Energie consommée pour différentes valeurs de débit binaire

Le **Tableau III.1** montre les valeurs de consommation d'énergie de l'objet audio en variant les débits binaires et en fixant le taux d'échantillonnage à 44100Hz.

	64->96kbps	96->128kbps	128->160kbps	160->192kbps
AAC	0.3%	0.51%	0.65%	0.21%
AC3	0.85%	0.36%	0.79%	0.02%
MP2	0.94%	0.29%	0.24%	0.16%
MP3	0.46%	0.18%	0.19%	0.47%
Ogg	1.9%	1.15%	2.29%	1.31%
WMA	0.57%	2.48%	0.81%	1.97%

Tableau III.2: Taux de croissance de l'énergie consommé lors de l'augmentation du débit binaire.

Le **Tableau III.3** montre les valeurs de consommation d'énergie d'un objet audio en variant les taux d'échantillonnage et en fixant le débit binaire à 128k.

	22050 Hz	24000 Hz	44100 Hz	48000 Hz
AAC	12.022	12.854	8.4301	8.695
AC3	5.188	5.487	6.3085	7.016
MP2	6.265	6.589	7.109	7.955
MP3	6.047	6.358	7.6087	7.899
Ogg	7.655	7.937	9.058	10.874
WMA	6.58	6.84	7.592	7.986

**Tableau III.3: Energie consommée pour différentes valeurs de fréquence d'échantillonnage**

	22050->24000 Hz	44100->48000 Hz
AAC	6.92%	3.14%
AC3	5.76%	11.21%
MP2	5.17%	11.9%
MP3	5.14%	3.81%
Ogg	3.68%	20.04%
WMA	3.95%	5.19%

**Tableau III.4 : Taux de croissance d'énergie consommée lors de l'augmentation de la fréquence d'échantillonnage**

Le **Tableau III.4** montre les valeurs des taux de croissance de consommation d'énergie quand le taux d'échantillonnage est varié.

En conclusion, nous observons que, quand le débit binaire est augmenté, le taux de croissance moyen de la consommation d'énergie est d'environ 1.2%. Par ailleurs, quand le taux d'échantillonnage est augmenté, alors le taux de croissance moyen de la consommation d'énergie est d'environ 7.4%. Par conséquent, le taux d'échantillonnage a une influence plus significative que le débit sur la consommation d'énergie.

### III.2.2 Optimisation d'énergie de rétro-éclairage

La plupart des équipements mobiles sont équipés d'un écran **TFT** (*Thin-Film Transistor*) ou **LCD** (*Liquid Crystal Display*). Pour ces périphériques, l'unité d'affichage repose sur l'illumination du rétro-éclairage. Le rétro-éclairage consomme un pourcentage considérable, entre 20% et 40% de la consommation totale pour un Compaq iPAQ.

La réduction dynamique de l'illumination du rétro-éclairage, et l'augmentation de la luminance des pixels des images des séquences vidéo sont souvent considérées comme une méthode efficace d'optimisation de l'énergie. Cependant, l'augmentation de la luminance des pixels des images affecte la fidélité de la qualité vidéo.

1. Dans [Pa,03] trois politiques d'adaptation sont définies, et sont utilisées en conjonction avec un algorithme de compensation de luminosité, afin de réduire la consommation d'énergie du rétro-éclairage dans les terminaux mobiles. Les auteurs constatent que la compensation agressive de la luminosité est possible pour le streaming vidéo, par rapport aux images fixes, sans pour autant qu'il y ait un impact considérable sur la qualité de la vidéo. Au fait, les petits défauts (introduits grâce à la compensation agressive), se faisant sentir en image fixe sont moins discernables en streaming vidéo où plus de 25 images sont affichées sur l'écran chaque seconde.
2. Dans [Cho,04], les auteurs proposent une technique appelée **DLS : Dynamic luminance scaling** (*graduation dynamique de la luminance du rétro-éclairage*). Le principe de DLS est de sauver la puissance par l'obscurcissement de rétro-éclairage tout en reconstituant la luminosité de l'image par la compensation appropriée d'image. DLS ne sacrifie pas la luminosité globale de l'image mais adopte des déformations mineures de couleur. Pour réaliser l'économie de puissance maximum pour une limite donnée de déformation de couleur, DLS mesure dynamiquement la luminance du rétro-éclairage pendant que l'image sur le panneau LCD change. Pour une économie de puissance plus agressive, DLS peut également effectuer la modification active des couleurs, ce qui permet l'extrême obscurcissement du rétro-éclairage. Cette technique réduit la consommation électrique d'un rétro-éclairage LCD TFT couleur tout en conservant la luminosité ou le contraste.
3. Dans [Sh,04] est proposé une technique appelée **EDLS : Extended DLS**. un cadre de travail (*framework*) de gestion dynamique de l'alimentation du rétro-éclairage des écrans LCD. Cette technique commute entre DLS et l'autre méthode dite Dynamic Contrast Enhancement (*DCE*). Le DCE augmente également la qualité de l'image sous un rétro-éclairage obscurci, mais par augmentation du contraste de l'image. Tandis que DLS préserve les couleurs originales, le DCE peut avoir comme conséquence un changement des couleurs originales à la suite d'un contraste plus élevé et d'une lisibilité améliorée. DCE doit donc être considéré comme une méthode très agressive

de gestion d'énergie pour les écrans LCD. Le gain est de l'ordre de 25% d'économie d'énergie dans le rétro-éclairage pour les images fixes et animées.

4. Dans [Ch,05], les auteurs intègrent de façon explicite la qualité vidéo dans la stratégie de commutation du rétro-éclairage et proposent une technique adaptative dite **QABS** (*Quality Adapted Backlight Scaling*). pour réaliser une économie d'énergie de rétro-éclairage pour les applications vidéo sur les dispositifs mobile. Les auteurs présentent un algorithme rapide pour optimiser l'obscurcissement de rétro-éclairage tout en maintenant la dégradation dans la qualité d'image à un minimum, de sorte que la qualité globale de service QoS soit proche d'un seuil spécifique. En plus, ils proposent deux techniques efficaces pour empêcher la commutation fréquente du rétro-éclairage, qui affecte négativement la perception d'utilisateur de la vidéo.
5. Dans [Ch,06], une méthode analogue, nommée **CBCS** (*Concurrent Brightness and Contrast Scaling*) est proposée. CBCS vise à la conservation d'énergie en réduisant la luminance du rétro-éclairage tout en conservant la qualité de l'image par la préservation du contraste. Leur définition de déformation et la technique de compensation proposées donne de bons résultats pour les applications basées sur les images statiques, telles que l'interface utilisateur graphique (*GUI*) et des cartes géographiques (*maps*), mais pourrait ne pas être appropriée pour le streaming des scénarios vidéo, car leur compensation du contraste compromet davantage la fidélité des images.

La compensation de luminance, cependant, conduit inévitablement à une distorsion de la qualité. Pour une application de streaming vidéo, la qualité est normalement définie comme la ressemblance entre la vidéo originale et celle traité.

### III.2.3 Optimisation de l'énergie par la technique Dynamic Voltage-Frequency Scaling (*DVFS*)

Au niveau système, la technique d'ajustement conjoint en tension et en fréquence (*Dynamic Voltage Scaling : DVS*) s'est particulièrement distinguée par sa grande efficacité à réduire la consommation du processeur. Elle permet d'exécuter différentes tâches d'une application à des couples tension/fréquence différents en fonction de la charge de travail du processeur. Notons que diminuer la tension d'alimentation  $V_{dd}$  impose une fréquence  $f$  de fonctionnement plus faible. Comme la puissance consommée est proportionnelle à  $V_{dd}^2$  et à  $f$ ,

par conséquent, en réduisant la vitesse du processeur de moitié, la consommation d'énergie peut être réduite à environ un tiers. Le défi majeur est de prévoir avec précision les temps d'exécution de la charge de travail pour les tâches futures. Pour des applications temps réel, les systèmes qui prennent en charge *DVFS* doivent équilibrer les économies d'énergie réalisées avec les contraintes de temps des applications exécutées.

**Dynamic Voltage and Frequency Scaling (DVFS) [Bo,01]**, est une technique efficace pour atteindre une faible consommation d'énergie tout en répondant aux exigences de performance. La dissipation d'énergie est réduite par la variation dynamique de la tension d'alimentation du CPU, de sorte qu'il fonctionne à une vitesse minimale requise par la tâche à exécuter. La technique consiste principalement à l'ordonnancement des tâches en vue de déterminer le moment exact où chacune sera exécutée par le processeur, dans le cas où aucune tâche n'est prévue DVFS permet de réduire la fréquence du processeur, ce qui réduit la tension qui lui est affectée, et donc consomme moins d'énergie.

Des priorités peuvent être assignées aux tâches de manière statique (lorsque les priorités sont fixes), ou dynamique (lorsque les priorités sont modifiées d'une requête à une autre). Un exemple bien connu de la gestion dynamique de la priorité est EDF (*Earliest Deadline First*) (temps de fin le plus tôt passe en premier) [Li,73], et un exemple pour l'allocation statique est RMS (*Rate-Monotonic Scheduling*) [Bu,05] pour des tâches périodiques.

Les techniques DVFS peuvent être classés selon les contraintes de temps, de granularité de scaling, et la politique de base d'ordonnancement. La contrainte de timing basée sur l'ordonnancement peut être ou ne pas être en temps réel. Les auteurs dans [Pi,01] ont présenté une nouvelle classe d'algorithmes pour le **DVS** en temps réel qui modifie l'ordonnancement temps réel du système d'exploitation et le service de gestion des tâches pour assurer des économies d'énergie importantes tout en respectant des délais des contraintes temps réel. Leurs résultats montrent une réduction de consommation d'énergie de 20% à 40%.

DVFS peut être qualifiée de préemptive ou non-préemptive. Dans l'ordonnancement préemptif, le temps est généralement divisé en intervalles fixes. DVFS est ensuite appliquée par un planificateur de niveau de tâches basé sur les statistiques d'utilisation du processeur sur l'intervalle précédent. Alors que dans l'ordonnancement non-préemptif, les tâches s'exécutent jusqu'à la fin et ne peuvent être interrompues [Da,00].

Une technique d'ordonnancement non-préemptive qui exploite la tension comme une variable est proposée dans [Ho,99]. Jeffay et al [Je,91] ont analysé le problème de



l'ordonnancement d'un ensemble de tâches périodiques ou sporadiques (ie: non périodiques) en temps réel, sur un monoprocesseur sans préemption et sans temps mort. Les auteurs ont exhibé un ensemble de conditions nécessaires et suffisantes C sur un ensemble de tâches périodiques ou sporadiques pour qu'elles puissent être planifiables pendant des temps arbitraires de déclenchement des tâches. Ils ont prouvé que quelque soit l'ensemble des tâches périodiques ou sporadiques qui satisfait les conditions C, cet ensemble pourra être ordonnancée avec l'algorithme EDF.

L'ordonnancement DVFS peut également être classé en des techniques en ligne et hors ligne. Les prises de décisions d'ajustement de la tension et de la fréquence de fonctionnement du processeur sont faites soit hors ligne, lors de la phase de conception i.e. avant exécution, soit en ligne c'est-à-dire en cours d'exécution. [He,07]. L'ordonnancement hors ligne [We,94] a une surcharge de traitement d'exécution inférieur, en supposant le pire des cas avec une approche d'ordonnancement pessimiste, entraînant ainsi des économies d'énergie plus faible. La techniques d'ordonnancement en ligne telle que proposé dans [Ch,04] pourrait se traduire par des économies de puissance plus élevés que leurs homologues statiques, mais elles sont associées à une surcharge de traitement plus élevée.

[Ju,08] présentent un cadre de travail de gestion dynamique de puissance ou DPM (*Dynamic Power Management*) à base d'apprentissage dirigé pour un processeur multi-noyaux, où un gestionnaire de puissance PM (*Power Management*) apprend à prévoir l'état d'exécution du système pour chaque tâche entrante. Cela est effectué par une analyse simple et efficace de quelques caractéristiques d'entrée disponibles telles que : l'état d'occupation de la file d'attente de service et la cadence d'arrivé des tâches. Puis emploie cet état pour rechercher l'action optimale de gestion de puissance dans une table de consultation de politique pré-calculée. Cette technique - en dépit d'être en ligne - pourrait se traduire par la surcharge d'énergie d'exécution réduite.

Dans [Sa,07] les auteurs proposent une technique hybride qui prend avantage de la faible surcharge du traitement avec une plus grande efficacité énergétique. La technique utilise à la fois l'information statique et dynamique et est basée sur l'ensemble des points de réglage fréquence/tension appropriées insérés comme code d'exécution par le compilateur.

### III.2.4 Optimisation d'énergie par annotation des données

L'annotation de données consiste à analyser un paquet de données et de le compléter avec un résumé d'informations collectées ; ces informations sont employées plus tard pour des

optimisations temps réel du traitement de données. Plusieurs approches basées sur l'annotation des données ont été proposées dans le but de réduire la consommation d'énergie dans les environnements mobiles.

1. Dans [Co,06] les auteurs présentent une approche basée sur une annotation logicielle, pour l'optimisation d'énergie. Son application sur une technique d'ajustement du rétro-éclairage des écrans LCD pendant la lecture d'un contenu multimédia a été démontrée. La technique annote le paquet vidéo avec l'information qui est utilisé plus tard, pendant le playback, pour ajuster le niveau du rétro-éclairage. Afin de réduire la consommation d'énergie pendant le playback, la technique consiste à obscurcir le rétro-éclairage et compenser la dégradation en même temps en augmentant la luminance de l'image montrée.
2. Dans un autre travail [Cor,06] les mêmes auteurs proposent une technique permettant l'optimisation de la consommation d'énergie par l'annotation des paquets du trafic réseau. La technique est basée sur les annotations qui se sont produites pendant une étape de prétraitement, au niveau serveur. Ces annotations sont injectées dans le paquet MPEG et envoyés à travers le réseau. Le paquet traverse les éléments de réseau jusqu'à arriver au point d'accès sans fil. Ici des paquets du réseau seront combinés dans des rafales de transmissions qui seront envoyés au lien sans fil puis aux dispositifs mobiles qui sont lues par un décodeur. Pour adapter la nature de rafale de transmission avec le reste, et la consommation périodique de données par le décodeur, un buffer intermédiaire est placé entre le réseau et le décodeur. Le décodeur (étape finale) analyse le paquet MPEG, décode les canaux audio et vidéo et envoie les données décodées à l'interface de sortie correspondante.
3. Dans [Cnd,06] le travail se focalise sur la réduction de la consommation d'énergie de la CPU lors de l'exécution d'une vidéo. Dans MPEG, le traitement des macroblocks **I** a des conditions de traitement plus élevées, que des macroblocks **P** et **B**. La distribution des différents macroblocks dans une trame permet donc d'estimer le moment requis pour décoder une trame (contenant toujours un nombre constant de macroblocks). Comme chaque trame a une distribution différente des types de macroblock, cette information est stockée comme annotation au niveau de la trame, ainsi que la taille de la trame. Les auteurs utilisent des annotations pour aider à estimer

le calcul requis pour décoder une image (temps de décodage). Cette estimation est alors appliquée dans un programme de gestion d'énergie DVS.

4. Le langage HTML 4.01 est un langage de description de documents multimédia basé sur une extension temporelle du langage HTML (*Hypertext Markup Language*) et offrant des nouveaux marqueurs et attributs [Ro,99]. Celui-ci permet de définir des alternatives enrichies par des annotations portant sur le contenu d'une présentation multimédia. La **figure III.7** présente un exemple basé sur ce modèle. L'auteur a la possibilité d'enrichir son document avec des annotations précisant des valeurs d'importance de certains objets multimédia, le type de versions proposées, etc. Des conditions relatives à un profil cible sont par la suite appliquées au document pour satisfaire les contraintes et offrir un document adapté. Un exemple de conditions à appliquer au document de la **figure III.7** est proposé dans la **figure III.8**. Dans cet exemple, il n'est retenu par la requête que les informations du document considérées par l'auteur comme importantes (*importance, high*), des versions courtes (*version, short*) et une exécution simultanée (*version, par*).

```

<object id="China" description="Offres de voyages pour la Chine">
  <object id="video" importance="high" default src="china-clip-v1.avi"/>
  <object id="video" importance="low" src="china-clip-v2.avi"/>
  <htlink orig="beg" target="video.beg"/>
</object>
<object id="Japan" description="Offres de voyages pour le Japon">
  <object id="video" version="long" default src="long-japan-clip.avi"/>
  <object id="video" version="short" src="short-japan-clip.avi"/>
  <htlink orig="beg" target="video.beg"/>
</object>
<htlink target="China.beg"/>
<htlink id="JapanBeg" version="seq" default orig="China.end" target="Japan.beg"/>
<htlink id="JapanBeg" version="par" orig="China.beg" target="Japan.beg"/>

```

**Figure III.7 : Enrichissement d'un document multimédia en vue d'une adaptation**

```

<condition>
  <disjunction>
    <choice type="importance" value="high"/>
    <choice type="version" value="short"/>
    <choice type="version" value="par"/>
  </disjunction>
</condition>

```

**Figure III.8 : Spécification de conditions à appliquer au document de la figure III.7.**

Le résultat de l'application des conditions spécifiées dans la **figure III.8** au document de la **figure III.7** est présenté dans la **figure III.9**.

```
<object id="China">
  <object id="video" src="china-clip-v1.avi"/>
  <htlink orig="beg" target="video.beg"/>
</object>
<object id="Japan">
  <object id="video" src="short-japan-clip.avi"/>
  <htlink orig="beg" target="video.beg"/>
</object>
<htlink target="China.beg"/>
<htlink orig="China.beg" target="Japan.beg"/>
```

**Figure III.9 : Résultat de l'application des conditions de la figure III.8 au document de la figure III.7.**

Ce type d'adaptation par sélection du contenu peut être apparenté à l'application d'une feuille de style au document. Par exemple, il est possible de réaliser cette opération au moyen du langage XSLT [XSLT] pour des documents de type XML. Cependant, l'adaptation de tels documents impose que les conditions de sélection ou de transformation du contenu reposent sur un langage dépendant du langage de description du document initial.

### III.2.5 Optimisation de la consommation d'énergie lors de la transmission réseau

Les appareils mobiles tels que les PDA et smartphones doivent être conçus de telle sorte qu'ils utilisent l'énergie efficacement. Comme de plus en plus d'appareils mobiles peuvent s'intégrer dans des réseaux au lieu d'être utilisés dans le mode autonome, plusieurs approches ont été proposées visant à minimiser la consommation l'énergie de l'interface réseau sans fil. Cette dernière étant devenue un consommateur d'énergie en puissance dans les systèmes mobiles. En effet, pour les périphériques portables, la communication sans fil consomme une quantité d'énergie qui peut s'élever jusqu'à 40% de l'énergie totale du système.

Un réseau sans fil (*WLAN*) a généralement trois modes de consommation d'énergie : transmission, réception et repos. La transmission de données consomme le plus haut niveau d'énergie. En mode repos, l'interface consomme presque la même quantité d'électricité que lorsqu'elle reçoit des données.

Les paragraphes qui suivent donnent de brèves descriptions de quelques techniques d'optimisation d'énergie, traitant comme élément consommateur la carte réseau des appareils mobiles.

#### **a. La technique *Traffic Shaping***

Le Traffic shaping (aussi connu sous "packet shaping") [WI,10] est le contrôle du trafic d'un réseau informatique dans le but d'optimiser ou de garantir les performances en réduisant au minimum la probabilité de rejet du trafic. C'est un mécanisme qui peut offrir des économies d'énergie en permettant au client de se mettre en veille pendant de plus longues périodes. Plus particulièrement, le trafic shaping désigne toute action sur un groupe de paquets (souvent appelé *stream* ou *flow*), qui impose un délai supplémentaire à ces paquets pour qu'ils se conforment à une contrainte prédéterminée (contractuelle ou à un certain type de trafic).

Cette technique a été étudiée dans [Ch,02] et [Ch,03], pour le streaming des formats médias de Microsoft, Real media et Quicktime. Ces travaux montrent que la gestion de l'alimentation par défaut dans l'interface réseau IEEE 802.11 est limitée et ils proposent donc une politique pour contrôler le trafic réseau afin de permettre de plus longues périodes d'inactivité de la carte réseau entre les transferts. En mettant la carte sans fil en mode économie d'énergie pendant les périodes d'inactivité, ils sont capables de réaliser des économies d'énergie. Une extension de ces travaux est présentée dans [Cha,04] où l'auteur étudie différents mécanismes basés sur la technique de trafic shaping et discute de leurs efficacités.

#### **b. Power-Saving Mode**

Le standard courant IEEE 802.11b supporte un mécanisme appelé le mode économie d'énergie (*PSM*) pour réduire la dissipation d'énergie. Le PSM permet à un client de passer en mode sommeil à faible consommation quand il n'a pas de trafic sortant ou entrant. Le client doit notifier à l'AP (point d'accès) son passage en mode veille. L'AP met en mémoire alors toutes les données destinées à ce client quand il est endormi, le transfert de données se fera chaque fois que le client s'éveille dans des intervalles périodiques. La gestion se fait en suivant un protocole d'échange de requêtes tel que TIM (*Traffic Indication Map*) qui informe les clients mobiles de données en attente, ou encore PS-Poll (*Power-Saving sondage*) pour indiquer que le client est éveillé.

### c. L'algorithme AB-PSM

Dans [Ad,07] les auteurs proposent un algorithme d'optimisation de la consommation réseau de l'énergie dans les terminaux mobiles. L'algorithme considère une architecture client/serveur, quand le client demande des données au serveur, le serveur envoie en streaming le contenu demandé à travers le réseau sans fil. Au début, le contenu est envoyé sans que les mécanismes d'économie d'énergie, autre que le standard IEEE 802.11 PSM, ne soient appliqués. Le client reçoit le contenu et compare la durée de consommation d'énergie restante à la durée de temps exigée pour accomplir la requête de streaming. Si le niveau de puissance de la batterie ne permet pas au dispositif mobile d'accomplir la tâche, alors le client informera le serveur que des mécanismes d'économie d'énergie doivent être mis en application.

L'économie d'énergie propose différents mécanismes qui sont divisés par étape avec chaque étape ayant trois états différents. Chaque fois qu'un mécanisme d'économie d'énergie est mis en application, le client attendra quelque temps et effectue encore la comparaison entre la durée de vie de la batterie restante et la durée de vie de la batterie requises pour accomplir la tâche. Si la tâche n'est pas accomplie, le prochain mécanisme d'économie d'énergie est mis en application.

Les mécanismes d'économie d'énergie seront exécutés d'une façon incrémentale. Les premiers mécanismes d'économie d'énergie utilisés seront ceux liés à l'étape de réception, pour économiser l'énergie dans l'étape de réception, il est proposé l'Adaptive-Buffer Power Save Mechanism AB-PSM. C'est un mécanisme qui permet au trafic d'une station mobile d'être caché temporairement, lui permettant de se mettre en veille pour longtemps, réalisant l'économie d'énergie. L'avantage d'AB-PSM est qu'il n'exige d'apporter aucune modification à la norme existante d'IEEE 802.11. L'économie d'énergie dans l'étape de décodage est réalisée en envoyant le contenu multimédia qui a été codé aux débits binaires inférieurs. Le débit binaire est maintenu à un niveau qui répond à la qualité perçue par utilisateur tout en préservant la batterie autant que possible.

### d. Le système CoolSpot

CoolSpots [Pe,06] est un système qui permet à un dispositif mobile sans fil de commuter automatiquement entre les interfaces radio multiples, telles que WiFi et Bluetooth,

afin d'augmenter la vie de la batterie. CoolSpots emploie un système hybride WiFi/Bluetooth pour fournir des possibilités d'améliorer la communication quand un dispositif mobile est dans une région CoolSpot-permise. La contribution principale de CoolSpot est une exploration des politiques qui permettent à un système de choisir parmi ses interfaces, chacune avec des caractéristiques différentes afin d'économiser l'énergie. Le système et les politiques n'exigent aucun changement aux applications mobiles, et les changements appliqués à l'infrastructure existante sont minimes.

### e. Techniques d'ordonnement optimal

Dans [Zh,05] sont proposés deux algorithmes de gestion de trafic réseau pour améliorer la technique de trafic shaping pour d'autres économies d'énergie. Les algorithmes construisent un programme de transmission au serveur proxy local qui informe l'AP (*Access Point*) du plan avec lequel les paquets devraient être livrés. Le premier régime propose un plan d'ordonnement optimal qui minimise de façon significative la durée de temps du mode puissance-élevée des clients mobiles. Dans Le second, les profils énergétiques et les capacités résiduelles des batteries des clients mobiles sont prises en considération, pour l'établissement du programme de transmission. C'est une approche de programmation dynamique qui calcule le programme optimal de transmission basé sur des nouvelles métriques. Pour cela l'auteur utilise la somme pondérée des temps d'arrivée comme métrique de performance (voir **figure III.10**). Cela a pour effet de favoriser les clients à faible énergie résiduelle de la batterie et qui ont le taux de consommation d'énergie de mode veille élevé.

**Input:**  $n, m, t_i$  ( $1 \leq i \leq n$ ).

**Output:**  $S_j$  ( $1 \leq j \leq m$ ).

```

1      Sort streams in non-increasing order of  $t_i$  ( $1 \leq i \leq n$ );
2      for  $i = 1, 2, \dots, n$ 
3           $j = i \bmod m$ ;
4          Add  $i$  to  $S_j$ ;
5      for  $j = 1, 2, \dots, m$ 
6          Sort  $S_j$  in non-decreasing order of  $t_{j,k}$ ;
```

**Figure III.10** Algorithme pour calculer le comportement optimal [Zh,05].

## f. Approche historique

L'idée de base de cette stratégie consiste à choisir de façon sélective des périodes appropriées du temps de suspension de la communication par commutation de la carte réseau sans fil à l'état sommeil. Une prédiction linéaire basée sur une technique de prévision des séries chronologiques est utilisée pour prédire l'avenir des intervalles d'absence de données.

Dans [We,04], les auteurs proposent un schéma de prévision statistique pour des intervalles d'inactivité et montrent que les résultats obtenus sont meilleurs que la moyenne simple. Dans [We,06] il est exploré des stratégies coté client à base de prédiction linéaire qui réduisent la consommation d'énergie lors de la réception de flux multimédia.

### III.2.6 Optimisation inter-couche

La manière dont les techniques d'optimisation (ex : optimisation CPU, carte réseau, etc.) sont implémentées dans les systèmes d'optimisation est jugée par certains chercheurs insuffisante, du fait qu'elle réalise des économies d'énergie partielles. Pour plus d'économie d'énergie, l'apparition des middlewares tel que CORBA [CORBA] a créé une convergence vers un nouveau type de système d'adaptation qui vise l'intégration de multiples techniques d'optimisation à plusieurs niveaux dans un même système (ex : optimisation aux niveaux hardware, réseau et contenu etc.).

Cependant, un système avec plusieurs couches d'optimisation exige une coordination minutieuse de cette optimisation pour cumuler les économies de toutes ses couches. Le système d'adaptation et d'optimisation inter-couche doit concilier les exigences d'adaptation et le temps, pour ne pas créer de latence.

#### 1. Le projet GRACE

Le cadre de travail GRACE [Yu,03] est un projet développé à l'université d'Illinois, il s'appuie sur des adaptations à plusieurs niveaux pour maximiser l'utilité du système au plus bas coût d'énergie. GRACE est le premier système mis en application qui effectue des adaptations multicouches

- (1) aux échelles de temps multiples (globales et par-application),
- (2) dans des couches multiples (application, matériel, et programmeur),
- (3) exécutant des applications multiples et,



(4) en réponse aux contraintes multiples (largeur de bande passante, d'unité centrale de traitement).

Il propose trois niveaux d'adaptation : l'*adaptation globale* qui considère ensemble toutes les applications et couches systèmes, mais est déclenchée seulement dans le cas des modifications systèmes de grande taille (par exemple, en entrée du client ou à la sortie vers le réseau). L'*adaptation par-application* considère une seule application à la fois, adaptant toutes les couches de système aux demandes courantes de cette application. Enfin, l'*adaptation interne*, c'est l'adaptation d'une seule couche système (éventuellement en examinant plusieurs applications) pouvant être invoquée plusieurs fois par une application.

Pour l'adaptation de la CPU, GRACE utilise DVFS ; pour les contenus il utilise l'encodeur vidéo H.263 etc. Tous les niveaux d'adaptation sont étroitement couplés, en s'assurant que les décisions d'allocation des ressources prises par le coordinateur sont respectées par les adaptations locales.

## 2. Le projet FORGE

Le projet FORGE [Co,03] est un cadre de travail multimédia /client/proxy/serveur. Les communications entre appareils mobiles et serveur multimédia sont acheminées à travers un serveur proxy, qui traite les flux en temps réel. FORGE vise à étudier les compromis entre la performance énergétique, et les exigences de qualité de service à travers les différentes couches de calcul. L'objectif est de développer et coordonner les approches d'optimisation (application, middleware, système d'exploitation) et l'énergie existante de la batterie, pour des améliorations tant en économies d'énergie que en satisfaction de l'utilisateur, dans le cadre d'applications multimédia distribuées. Différentes techniques sont employées dans [Mo,05], pour parvenir à une gestion coordonnée d'énergie entre les différents niveaux de fonctionnalités du système. L'utilisation d'un cadre de travail middleware réparti adaptatif (appelé "Dynamo"), qui coordonne les adaptations globale et locales. L'annotation des données de l'application avec des informations spécifiques qui peuvent être exploitées lors de l'exécution à différents niveaux d'abstraction (matériel via application). Enfin, examiner des compromis des paramètres définis en multiples couches, telles que la couche application, couche OS, couche réseau et la couche matérielle.

### 3. Le projet xTune

La gestion adaptative des ressources est indispensable pour assurer la qualité des communications multimédia en temps réel, en particulier pour la limite d'énergie des appareils mobiles. *Minyoung Kim* propose le cadre de travail **xTune** [Ki,07], qui emploie une politique itérative utilisant une légère vérification formelle lors de l'exécution et la communication des résultats pour l'adaptation dynamique en temps réel. Une spécification formelle exécutable est élaborée pour chaque couche du mobile en temps réel, pour spécifier formellement les problèmes de synchronisation du tous les niveaux et étudier leurs corrélations. La spécification formelle est ensuite analysée à l'aide de modèle des vérifications statistiques et d'analyse statistiques quantitatives, afin de déterminer l'impact des différentes politiques de gestion des ressources sur les propriétés de synchronisation end-to-end.

#### III.2.7 Optimisation d'énergie par la technique du cache

Les mémoires internes, généralement de technologie SRAM, offrent un temps d'accès de l'ordre de 1 à 2 cycles processeur. Ces mémoires sont soit des mémoires *Cache* gérées matériellement soit des mémoires dédiées ou *scratchpad* gérées par logiciel par le concepteur, où a priori les données les plus fréquemment utilisées sont stockées. Ces mémoires SRAM internes sont exploitées dans quelques approches, pour réduire la consommation d'énergie ou minimiser la dissipation énergétique par le processeur.

Une approche appelée *cache decay* pour réduire la consommation statique dans les caches est introduite dans [Ka,01]. Cette technique consiste à mettre hors tension (*gated- $V_{dd}$* ) une ligne de cache non accédée pendant une période de temps. Les cellules mises hors tension perdent alors leurs contenus (*state-destroying*). Un accès à ces cellules nécessite une recharge des données du niveau hiérarchique supérieur donc une consommation dynamique supplémentaire. Alors que l'approche i-cache possède une granularité de plusieurs blocs contigus, la technique cache decay considère individuellement les blocs du cache.

La technique *Drowsy cache* [Fl,02] utilise l'ajustement en tension et fréquence DVS et consiste à mettre une ligne de cache en mode repos sous une tension plus faible. Les données des cellules mises sous une faible tension sont préservées (*statepreserving*), mais elles ne sont pas accessibles. Ces cellules ont besoin de plusieurs cycles pour être réveillées et retourner à la tension normale de fonctionnement.

Plusieurs approches d'optimisation d'énergie de cache se sont focalisées sur le principe de l'ajout d'un petit tampon supplémentaire, nommé aussi cache filtre, que le système utilise pour récupérer les données directement, au lieu d'accéder au cache d'origine. Il s'agit de réduire les dissipations de puissance dans les hiérarchies de mémoire en utilisant des tampons spéciaux. Le travail de [Pi,02] utilise ESB (*Energy-Saver Buffers*) qui se trouve entre le cache L2 et la mémoire principale. ESB permet de réduire les frais généraux supplémentaires encourus en raison de resynchronisation fréquente des modules de mémoire dans un état de faible puissance.

Dans [Ki,97], les auteurs proposent un cache de faible taille de niveau L0 nommé *filter cache*. L'accès au niveau L1 du cache se produit uniquement lors d'un défaut de cache dans le filter cache. Les auteurs montrent que le filtre cache réduit de 51% le produit énergie  $\times$  temps d'accès sur un ensemble d'applications multimédia, comparé à une architecture conventionnelle de cache.

Diviser des mémoires en composants séparés permet d'optimiser ces composants pour réaliser une meilleure exécution ou pour conserver l'énergie. Cela consiste à diviser les cellules de mémorisation en plusieurs partitions pour réduire la consommation dynamique dissipée lors de la charge ou la décharge des lignes de cache. Ce partitionnement est soit horizontal, soit vertical soit à la fois horizontal et vertical. Dans le cas de partitionnement horizontal, il s'agit de partitionner horizontalement les cellules mémoire en plusieurs bancs de façon que seul le banc en accès soit actif. La taille de la ligne de cache (*word line*) est réduite de même que le nombre de cellules mémoire actives. Cette architecture permet d'améliorer la performance et de réduire la consommation, alors que le partitionnement horizontal répartit une ligne de cache sur plusieurs bancs en gardant une taille fixe de colonnes. Le partitionnement vertical ou Bit-line partitioning répartit la colonne du cache en plusieurs modules construisant une hiérarchie mémoire. Cette technique est largement exploitée pour améliorer les performances et réduire la consommation. Plusieurs travaux d'optimisation sont basés sur les techniques de partitionnement, comme la ligne tampon [In,99], le cache filtre [Ka,01], et le cache subbanking [Ge,05].

### III.3 Adaptation des images

Dans le cadre de notre recherche nous avons constaté un manque de documentations traitant de l'adaptation des images et la préservation de l'énergie dans les environnements à faible ressources. Cependant, cela ne nous empêche pas de développer quelques idées en se

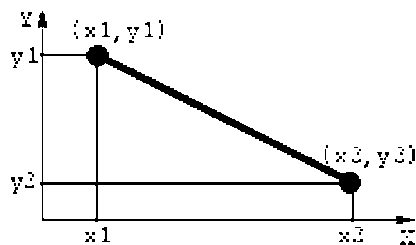
basant sur les différents facteurs caractérisant ce type de média tels que : les formats d'images, les techniques de compression, la résolution et la définition, dans le but de réduire la consommation d'énergie dans les appareils mobiles.

### III.3.1 Types d'images

Il existe divers formats d'images qui se classent en deux grandes catégories, à savoir les images vectorielles et les images matricielles qu'on traite en bref dans cette section. Sur le web, seules les images matricielles existent sous les formats GIF, JPEG et PNG. Ils ont l'avantage d'être indépendants des plates-formes matérielles et des logiciels (ils peuvent être échangés d'un système à un autre). De plus, ce sont des formats compressés permettant ainsi d'optimiser les temps de chargement, point faible du web.

#### III.3.1.1 Images vectorielles

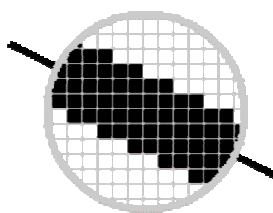
Elle consiste en la représentation des données de l'image par des formules géométriques, autrement-dit, c'est la mémorisation de la représentation des coordonnées des points caractéristiques des formes qui constituent l'image (carrés, rectangles, ellipses, cercles, courbes, etc.). Par exemple (**Figure III.11**), un dessin peut être mémorisé par l'ordinateur comme « une droite tracée entre les points  $(x_1, y_1)$  et  $(x_2, y_2)$  ».



**Figure III.11 : Exemple d'une image vectorielle**

#### III.3.1.2 Images matricielles (ou images bitmap)

La numérisation matricielle des images consiste à découper l'image en des milliers de points colorés, chaque point porte des informations de position et de couleur. Ce sont les pixels (contraction de l'anglais *picture element*) **Figure III.12**. Ces points forment une matrice avec plusieurs dimensions ou chacune des dimensions représente une classe d'informations (spatiale, temporelle etc.). Exemple de format d'images bitmap : BMP, PCX, GIF, JPEG, TIFF, Raw, etc.



**Figure III.12 : Exemple d'une image matricielle.**

Le **Tableau III.5** contient une comparaison entre les principaux formats d'images, et leurs caractéristiques.

	Compression des données	Nombre de couleurs supportées	Affichage progressif	Animation	Transparence
<b>JPEG</b>	Oui, réglable (avec perte)	16 millions	Oui	Non	Non
<b>JPEG2000</b>	Oui, avec ou sans perte	32 millions	Oui	Oui	Oui
<b>GIF</b>	Oui, Sans perte	256 maxi (palette)	Oui	Oui	Oui
<b>PNG</b>	Oui, sans perte	Palettisé (256 couleurs ou moins) ou 16 millions	Oui	Non	Oui (couche Alpha)
<b>TIFF</b>	Compression ou pas avec ou sans pertes	de monochrome à 16 millions	Non	Non	Oui (couche Alpha)

**Tableau III.5 : Comparaison entre des formats d'images**





Dans ce qui suit, nous présentons les principaux formats (algorithmes de compression) d'images matricielles.

### III.3.2 Adaptation des images numériques

Dans cette section nous citons quelques opérations usuelles d'adaptation des images, utilisées dans les différents systèmes de fourniture de contenus ou présentations multimédia.

### III.3.2.1 Le transcodage

Le transcodage des images dans les systèmes d'adaptation multimédia consiste à convertir l'image source d'un format de codage à un autre, cette opération est nécessaire dans le cas où l'appareil cible n'est pas compatible avec le format d'image originale. Contrairement aux formats JPEG et GIF, les formats comme PNG ou TIFF sont peu utilisés, car seuls les navigateurs et lecteurs récents reconnaissent ces formats. Le transcodage vers les formats usuels comme JPEG et GIF est ainsi obligatoire, ce passage souvent proposé par les modules d'adaptation du contenu, modifie la structure binaire des images d'origines et peut par conséquent changer la taille du stockage de l'élément. Les images JPEG ont l'avantage d'avoir une taille plus petite que les autres formats (voir **Figure III.13**), ce qui lui donne le privilège sur les autres formats et justifie sa présence dans pratiquement toutes les nouveaux appareils mobiles à ressources limités.

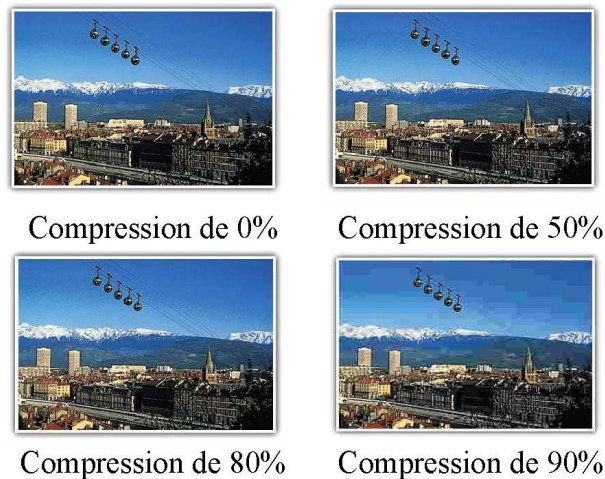
Taille	9.979	6.012	4.539	2.753
				
Qualité	Haut	Moyen sup.	Moyen inf.	Bas

**Figure III.13 : Tailles des images JPEG**

### III.3.2.2 La compression

L'image bien plus que le son, consomme une quantité impressionnante d'octets quand elle est numérisée. Cette augmentation de la taille implique nécessairement plus d'intégration de la carte réseau quand les images sont transférées dans un réseau sans fil, cela cause automatiquement une consommation plus grande des ressources énergétiques.

Pour remédier à ces problèmes, et arriver à des fichiers de petites tailles, il est nécessaire de compresser les images ; les méthodes de compression se classent en deux catégories : celles qui se contentent de compresser les données sans les altérer (compression sans pertes) et celles qui les compactent en les modifiant (avec pertes).



**Figure III.14 : Compression d'image JPEG [TL,05].**

Le degré de compression des images de type JPEG par exemple, peut atteindre jusqu'à 90%, sans pour autant une grande perte d'information comme on le voit dans la **Figure III.14**. De telles opérations de compressions sont bénéfiques, à partir de moment où elles permettent un gain considérable en temps de transmission, elles réduisent la taille de stockage de données et elles gardent en plus une bonne qualité visuelle.

### **III.3.2.3 Le changement de couleur**

Le nombre de bit alloués pour le stockage d'une couleur d'un pixel définit le mode de couleur d'une image. Les images RVB (*rouge, vert et bleu*) sont de couleurs vraies. Le pixel est codé sur trois octets, où chaque octet représente une couleur dans l'espace RVB. Cependant, ce mode pose le problème de la taille de stockage. Les images en 256 couleurs sont caractérisées par le fait qu'un pixel soit codé sur un octet, ce mode est utilisé pour réduire l'espace occupé par l'information couleur des images. Dans le contexte des environnements mobiles, l'adaptation par le changement des couleurs consiste en effet à diminuer nombre de bit sur lequel on stocke l'information couleur des images. Le résultat de cette réduction est la dégradation de la qualité de l'image.

### **III.3.2.4 Le redimensionnement**

En plus des informations sur les couleurs, l'image est caractérisée par des informations concernant ses dimensions spatiales. L'adaptation par le redimensionnement consiste à réduire les dimensions spatiales de l'image pour qu'elles puissent être affichées sur l'écran de terminal mobile.

### III.4 Conclusion

Parmi les techniques d'optimisation d'énergie citées dans ce chapitre certaines peuvent être intégrées dans le cadre d'une architecture d'un système d'adaptation de présentation multimédia sous la forme d'un module supplémentaire, qui communique avec la partie chargée d'effectuer l'adaptation du contenu et de présentation afin de produire des présentations multimédia adaptées et optimisées. D'autres plus générales doivent être appliquées sur les composants matériels de l'architecture globale du système comme le type des appareils la taille de leurs mémoire cache, le type de carte réseau, etc.

Dans le prochain chapitre, nous allons concevoir une architecture d'adaptation de documents multimédia permettant l'optimisation d'énergie des appareils mobiles, et cela tout en exploitant les techniques d'optimisations citées dans ce chapitre.



## IV. Proposition d'une architecture

### IV.1 Introduction

L'intégration du multimédia dans les appareils mobiles est confronté à des contraintes dues d'un coté, au développement rapides des caractéristiques de ces appareils (PDAs, ordinateurs portables, téléphones) ainsi qu'à leur diversité, et d'un autre coté, au développement rapide des réseaux de télécommunication (GSM<sup>1</sup>, GPRS<sup>2</sup>, UMTS<sup>3</sup>, WLAN, Bluetooth) et leur intégration dans Internet. Par conséquent, il devient nécessaire d'adapter ces applications afin de satisfaire les contraintes imposées (temporelles, spatiales, énergétiques etc.) pour la présentation de ces documents sur un appareil portable.

L'objectif de ce chapitre est de proposer une architecture d'un système d'adaptation et de fourniture de présentations multimédias. Les principales fonctionnalités des différents modules de notre architecture sont détaillées. Nous discutons aussi des différentes expérimentations à entreprendre dans un travail futur, visant à développer et valider une telle approche.

### IV.2 La notion du profil utilisateur

L'ensemble des contraintes est exprimé d'une façon formelle dans ce qu'on appelle « *profil utilisateur* » ou « *contexte d'utilisation* ». Un profil est la compilation des contraintes du réseau (le type de réseau, la taille de la bande passante etc.) ; des contraintes sur le terminal : des contraintes hardware (le type de CPU, la taille de la mémoire interne etc.) et des contraintes software (nature du système d'exploitation, les codecs autorisés, le navigateur web etc.) ; et des contraintes sur les préférences et les spécificités de l'utilisateur (la langue préférée, malvoyance etc.).

Les contraintes exprimées dans le profil de l'utilisateur peuvent être classées par rapport à des contraintes sur *le contenu* et des contraintes sur *la structure*. Un document multimédia est composé d'un ou plusieurs éléments média (image, texte, vidéo, audio) organisés

---

<sup>1</sup> GSM : Global System for Mobile Communication

<sup>2</sup> GPRS : General Packet Radio Service

<sup>3</sup> UMTS : Universal Mobile Telecommunications System

spatialement et synchronisés temporellement par une structure descriptive (métadonnées dans le cas du langage SMIL).

### IV.3 Types d'adaptation

Pratiquement deux types d'adaptation existent, en occurrence l'adaptation de contenu et l'adaptation de document. Le paragraphe suivant donne la définition des deux types d'adaptation.

#### IV.3.1 Adaptation de contenus multimédia

Selon [LA,05], l'adaptation de contenus média est l'opération de transformer un objet multimédia en un autre objet satisfaisant un profil donné. Cette adaptation peut être classée en trois catégories : la première catégorie est le *transcodage* qui consiste au changement de codage de l'objet média (par exemple : conversion d'un élément audio de format *mp3* au format *wma*) ; la seconde catégorie est le *transmodage* qui signifie le changement de la modalité de l'élément média (par exemple : conversion d'une vidéo en bande sonore) ; la dernière catégorie est la transformation qui consiste au changement dans les caractéristiques de l'élément média mais en préservant le même encodage et la même modalité (par exemple : réduire les dimension d'une image, traduction d'un texte). Il existe des systèmes destinés à ce type d'adaptation (les systèmes de fourniture de contenus multimédia) comme DCAF [Ha.06] (*Distributed Content Adaptation Framework*), ADMITS [Bo.03] (*Adaptation in Distributed Multimedia IT Systems*).

#### IV.3.2 Adaptation de documents multimédia

L'adaptation de documents multimédia concerne la satisfaction des contraintes concernant principalement la synchronisation entre les objets ainsi que leur disposition graphique sur l'écran du dispositif, ainsi l'existence de l'adaptation temporelle et spatiale du document multimédia. Généralement, l'adaptation de document dans les systèmes d'adaptation de documents multimédia SMIL se fait selon des règles définies sous forme de *templates* codés avec le langage de transformation XSLT [XSLT], c'est le cas de système NAC [Le.04] (*Negotiation and Adaptation Core*).

Cependant, la satisfaction de toutes les contraintes du profil utilisateur nécessite la combinaison de ces deux types d'adaptation.

## IV.4 Techniques d'adaptation

L'adaptation des documents multimédia peut se faire de deux manières différentes ; la première s'appelle l'adaptation à priori, elle se traduit par l'édition au préalable de toutes les alternatives susceptibles de document où chaque alternative est adaptée et correspond à un profil client spécifique. Le langage SMIL grâce à l'élément `<switch>` (**Figure IV.1**) permet la spécification de plusieurs alternatives, dans cette technique d'adaptation il n'est pas obligé de définir les profils utilisateurs. Cependant, l'implémentation de ce type de solution est très difficile vu le nombre très important d'alternatives qu'il faut créer. La deuxième manière s'appelle l'adaptation à posteriori, elle repose sur la définition des règles d'adaptation et des stratégies d'adaptation de documents et de contenus. Selon le profil d'utilisateur, les composants logiciels qui se chargent de l'adaptation établissent un plan d'adaptation qui sera appliqué par la suite afin de satisfaire toutes les contraintes.

```
<par>
  <video id="Demo" src="Demo.avi" region="r1" />
  <switch>
    <audio systemOperatingSystem="palmos" systemLanguage="en"
src="Commentaire-palm.au" ...="" />
    <audio systemLanguage="en" src="Commentaire-en.au" ...="" />
    <audio systemLanguage="de" src="Commentaire-de.au" ...="" />
    <audio src="Commentaire.au" ...="" />
  </switch>
</par>
```

**Figure IV.1 : Adaptation à priori avec SMIL**

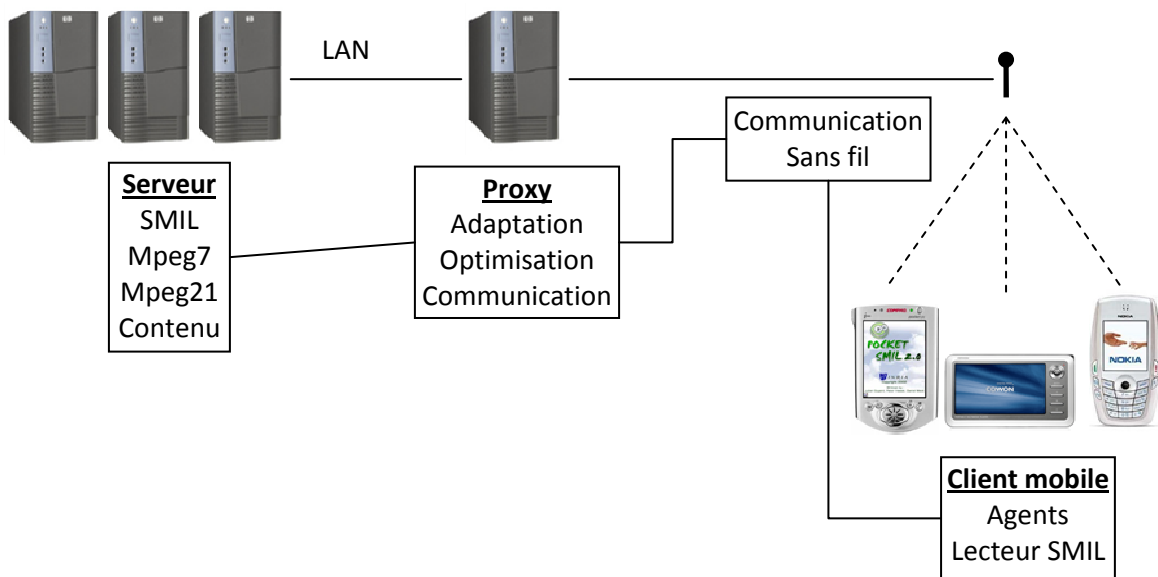
## IV.5 Architecture générale

Jusqu'à lors, nous nous sommes attelés à présenter dans ce document un état de l'art plus ou moins exhaustif des techniques d'optimisation d'énergie (voir chapitre III). Dans le contexte d'une adaptation d'une présentation multimédia, les expérimentations discutées dans le chapitre précédent permettent d'explorer un algorithme global d'optimisation d'énergie, pouvant être par la suite intégré dans une architecture d'adaptation de présentations multimédia que nous présentons subséquemment.

Vue la nouveauté et la complexité du domaine, peu de systèmes d'adaptation de présentation multimédia ont été développés, et même ceux existants comme **PAAM** [Ka,08] (*Architecture for the Provision of AdAptable Multimedia composed documents*) et **MSSA** [Az,05] (*Multimedia Scene Semantic Adaptation*), ne prennent pas en considération la contrainte énergétique. Par exemple dans le système **MSSA**, le choix de la technique

d'adaptation revient à l'utilisateur (le client), à travers des préférences exprimées dans les informations du contexte d'utilisation. Ces dernières peuvent parfois être non décisives pour la satisfaction de la qualité de service offerte aux serveurs, si les contraintes énergétiques ne sont pas garanties.

L'architecture qu'on propose est une architecture de type trois tiers (client/intermédiaire/serveur), elle peut être composée, comme le montre la **figure IV.2** de :



**Figure IV.2 : Environnement matériel de l'architecture**

1. **Les clients** : sont les utilisateurs qui accèdent au système en utilisant des appareils mobiles pouvant interpréter des documents multimédia SMIL, et possédant les droits d'accès aux contenus et aux documents qu'offrent les serveurs, grâce à un profil déclaré dans la base de données profil **Mpeg-21**. Pour le fonctionnement du système nous avons besoin d'installer dans chaque client les deux applications suivantes :

*i. Lecteur SMIL:* un lecteur SMIL est une application logicielle permettant d'interpréter, selon un scénario, les documents multimédias SMIL exécutées dans l'appareil mobile : RealOnePlayer, Ambulant Player, PocketSMIL sont des exemples de lecteurs SMIL.

*ii. Agent énergie:* Ce composant est très important dans notre architecture. Son rôle est double : d'une part, il établit —après mesure— un rapport sur le taux d'évolution de la consommation d'énergie dans le temps de l'appareil mobile, et le communique au proxy, en réaction à une demande lui parvenant de ce dernier.

D'autre part, il optimise la consommation d'énergie de la carte réseau en la mettant en veille après chaque téléchargement d'une quantité précise importante d'objets média de serveurs, et de rétro-éclairage en diminuant sa luminosité.

2. **Le proxy** : C'est une station intermédiaire, reliée avec les clients par une connexion sans fil et avec les serveurs par des connexions filaires. Il transmet les requêtes provenant des clients aux serveurs, ainsi que les données provenant des serveurs aux clients destinataires, par le biais des protocoles de communication (par exemple : HTTP ou RTSP), et assure les fonctionnalités suivantes :

*i. La communication:* A travers le proxy, transitent toutes les requêtes de documents multimédia des clients mobiles avant d'être envoyées aux serveurs. En parallèle, transitent toutes les données (présentation et média), des serveurs avant d'être adaptées et transmises vers les clients.

*ii. L'adaptation et l'optimisation:* C'est l'ensemble des transformations et traitements effectuées par le proxy sur les documents que les clients invoquent, ainsi que sur leurs contenus média, afin de satisfaire les contraintes énergétiques et celles du contexte d'utilisation. L'adaptation et l'optimisation s'effectuent au niveau du proxy en collaboration des différents serveurs et clients. Le proxy a besoin que les serveurs lui envoient le document SMIL, le contenu, le profil etc. ; et que l'agent énergie lui envoie le rapport sur l'évolution de la consommation d'énergie au niveau du client destinataire du document multimédia. Le résultat de cette étape est un document multimédia adapté avec une consommation énergétique optimisée.

*iii. La transmission:* Après son adaptation, le document et ses médias sont transmis de proxy vers le client.

3. **Les serveurs**: désigne les différentes stations hébergeant les informations nécessaires au déploiement des applications multimédia. Nous distinguons quatre types de serveur :

*i. Serveur de gestion de documents multimédia* : il gère une base de données qui contient les codes SMIL des documents multimédia ; la gestion comprend l'ajout la suppression, la mise à jour et la recherche.

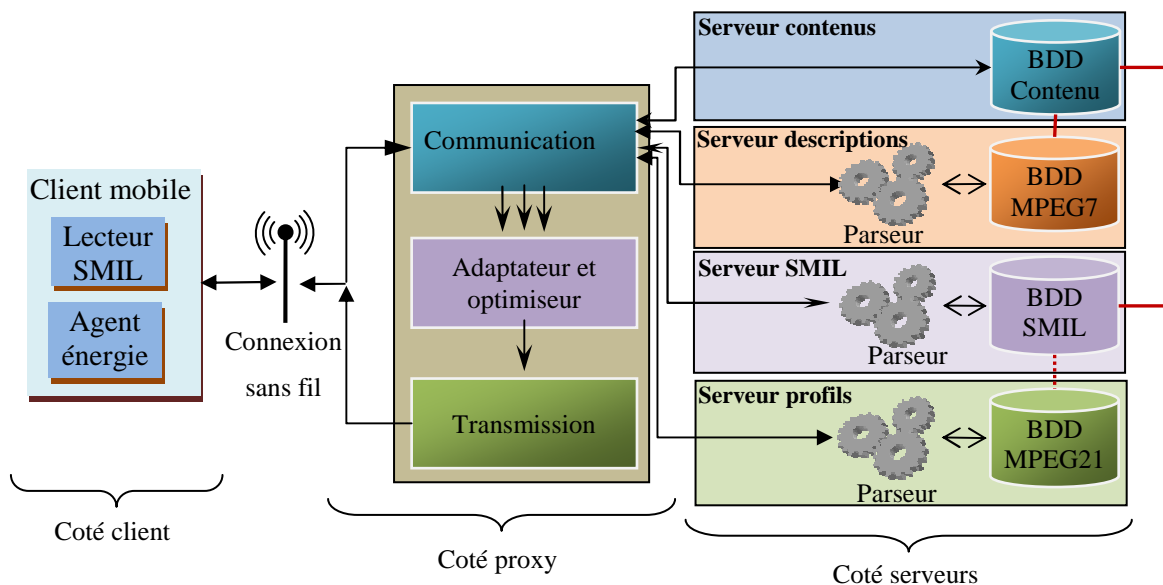
*ii. Serveur BD du contenu multimédia:* serveur gérant l'ajout la suppression la mise à jour et délivrance des médias (image vidéo etc.), où chaque élément possède un nom unique pour l'identifier.

*iii. Serveur de Gestion contexte* : (Serveur des profils), serveur gérant une base de données XML qui contient des descriptions sur des informations concernant les clients sous forme de code MPEG-21. Les informations énergétiques qui sont récupérées grâce à l'agent énergie forment la partie de profil énergétique Mpeg-21 actuel du client, ces informations sont définies par les attributs Mpeg-21 suivants :

- La consommation moyenne : `averageAmpereConsumption`,
- La capacité restante de la batterie : `batteryCapacityRemaining`,
- Le temps restant de la batterie : `batteryTimeRemaining`,
- Fonctionne sur batteries : `runningOnBatteries`.

*iv. Serveur de Gestion de description de contenu* : Serveur gérant une base de données XML qui contient les informations descriptives supplémentaires sur chaque élément média de la base de données de contenu exprimées en Mpeg-7 : il permet de faciliter la recherche et l'extraction des données multimédia à partir de la base de données du contenu, où chaque élément est référencé par une URL permettant de le récupérer.

La **figure IV.3** illustre \_ conceptuellement \_ comment les serveurs seront réparties dans le système.



**Figure IV.3 : L'architecture fonctionnelle du système**

- **un module de gestion de contextes d'utilisation** basé sur le standard MPEG-21, le choix de ce standard peut être justifié par le fait qu'il est d'une part, basé sur XML et d'autre part, il permet d'exprimer la contrainte énergie ;
- **un module de gestion de contenus multimédia** qui fait appel aux serveurs de contenu multimédia et des descriptions. Ce module va gérer à la fois la base de données des contenus qui contient les objets média et une base de données de descriptions MPEG-7 de ces objets et les alternatives de chaque élément. Cette possibilité est offerte par MPEG-7 grâce aux descripteurs, qui permettent d'ajouter des informations descriptives à chaque objet multimédia ;
- **un module de gestion de documents multimédia** basé sur le standard SMIL, qui s'occupe de la manipulation, le stockage, l'ajout et la suppression des présentations SMIL ;
- **un adaptateur optimiseur** ; ce composant est le cœur de l'architecture, il gère l'adaptation des présentations et de contenu selon les contraintes exprimées dans le profil récupéré du module de gestion de contextes, il assure également la mise à jour de la base de données des profils et la communication (négociation) avec le client.

### IV.5.1 Fonctions des modules

L'architecture du système contient plusieurs modules, chaque module s'occupe d'une fonction particulière. Les modules qui composent le système sont discutés par la suite. Ces modules sont les suivants : le module de gestion de contextes d'utilisation (§ IV.5.1.1), le module de gestion de contenus multimédia (§ IV.5.1.2), le module de gestion de documents multimédia (§ IV.5.1.3) et le proxy d'adaptation (IV.5.1.3).

#### IV.5.1.1 Module de gestion de contextes d'utilisation

Le module de gestion de contextes d'utilisation est la partie du système qui est chargée de gérer toutes les opérations de manipulation nécessaires des informations contextuelles de l'environnement d'utilisation du contenu multimédia. Ce module est basé sur un langage de description, dans le cas de l'architecture que nous proposons, il se base sur les descriptions de la norme MPEG-21 pour décrire les informations relatives à l'environnement d'utilisateur telles que ses préférences, les caractéristiques de son terminal ainsi que de son réseau d'accès et les paramètres de son environnement naturel et plus précisément le profil énergétique. La **figure IV.4** présente un exemple de code de quelques paramètres d'un profil, où on trouve des

informations sur les capacités du terminal comme la résolution de son écran *<Display>*, les capacité audio *<AudioOutput>*, etc.

```

<DIA>
<Description xsi:type="UsageEnvironmentType">
  <UsageEnvironmentProperty xsi:type="TerminalsType">
    <Terminal>
      <TerminalCapability xsi:type="DisplaysType">
        <Display id="primary_display">
          <DisplayCapability xsi:type="DisplayCapabilityType">
            <Mode>
              <Resolution horizontal="720" vertical="480"/>
            </Mode>
          </DisplayCapability>
        </Display>
        <Display id="secondary_display">
          <DisplayCapability xsi:type="DisplayCapabilityType">
            <Mode>
              <Resolution horizontal="176" vertical="144"/>
            </Mode>
          </DisplayCapability>
        </Display>
      </TerminalCapability>
      <TerminalCapability xsi:type="AudioOutputsType">
        <AudioOutput xsi:type="AudioOutputType">
          <AudioOutputCapability xsi:type="AudioOutputCapabilitiesType">
            lowFrequency="30" highFrequency="8000" numChannels="2"/>
          </AudioOutputCapability>
        </AudioOutput>
      </TerminalCapability>
      <TerminalCapability xsi:type="UserInteractionInputsType">
        <UserInteractionInput>
          <UserInteractionInputSupport xsi:type="MicrophoneType"/>
        </UserInteractionInput>
        <UserInteractionInput>
          <UserInteractionInputSupport xsi:type="KeyInputType">
            <KeyInput href="urn:mpeg:mpeg21:2003:01-DIA-KeyInputCSNS:1">
              <mpeg7:Name xml:lang="en">PCKeyboard</mpeg7:Name>
            </KeyInput>
          </UserInteractionInputSupport>
        </UserInteractionInput>
        <UserInteractionInput>
          <UserInteractionInputSupport xsi:type="MouseEvent">
            <Mouse buttons="2" scrollwheel="true"/>
          </UserInteractionInputSupport>
        </UserInteractionInput>
      </TerminalCapability>
    </Terminal>
  </UsageEnvironmentProperty>
</Description>
</DIA>

```

**Figure IV.4 : Exemple de description de profil en MPEG-21**

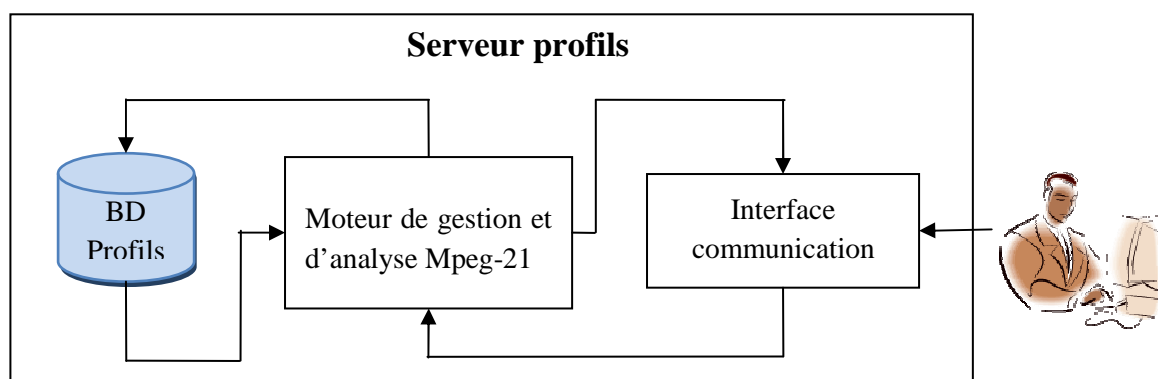
Comme décrit en chapitre II (§ II.4), le **DIA** de **MPEG-21** définit un ensemble complet d'outils de description des contraintes de contexte d'utilisation, connus sous le nom d'outils de description d'environnement d'utilisation(UED). Un contexte utilisateur peut être défini par



les caractéristiques personnelles de l'utilisateur (ex : la langue usitée, son handicap et ses centres d'intérêt), ses préférences de présentation (ex : son lecteur multimédia préféré ou la résolution de l'affichage), les capacités de son terminal (ex : la taille de l'écran du terminal ou les lecteurs multimédia présents) et les caractéristiques de son réseau d'accès (ex : la bande passante).

Le module de gestion de contextes d'utilisation côté client se charge de la collection et la communication de toutes les informations relatives au profil énergétique et statique et sa sauvegarde au niveau de serveur profils.

Côté serveur, un serveur MPEG-21 s'occupe de stocker les descriptions de l'environnement de l'utilisateur. Ce serveur stocke toutes les informations décrivant le contexte. Ces informations sont écrites sous forme d'éléments digitaux formant ainsi une base de données XML. Il gère également l'extraction des informations de la base contenant les éléments digitaux grâce à un module de *parsing* (analyse), et peut effectuer la mise-à-jour des informations dynamiques. Ces informations contextuelles dynamiques sont déterminées lors de l'exécution des requêtes de données ; ce serveur doit gérer et sauvegarder les alternatives de profil avec les informations sur le profil énergétique et cela après chaque opération d'adaptation, ainsi que le lien vers les alternatives de document SMIL correspondant au profil sauvegardé; il doit aussi assurer la communication avec les autres modules du système. La **figure IV.5** illustre les composants de ce module :



**Figure IV.5 : Module de gestion de contextes d'utilisation**

**Le profil statique** : est un fichier XML (mpeg-21) qui décrit par des valeurs exactes, les caractéristiques qui ne change pas avec le temps d'un client mobile. Ces caractéristiques peuvent être : la taille de son écran, sa résolution, la longueur de l'utilisateur, le type de CPU

etc. Tous les profils statiques des clients ayant déjà passé une requête de données au proxy sont en principe sauvegardés dans la base de données des profils, et sont distingués l'un de l'autre par leurs *IdProfilStatique*. Ces profils contiennent les contraintes à base desquelles les documents et leurs contenus seront adaptés.

**Le profil énergétique :** c'est également un fichier XML (mpeg-21), ce profil consiste à un rapport temporel sur les valeurs des caractéristiques énergétiques au moment  $t$  de son établissement par l'agent du client mobile. La sauvegarde se fait de la même manière que le profil statique, sous un identifiant *IdProfilEnergie*. Il est à relever que pour chaque profil statique il correspondra plusieurs profils énergétiques, qui dénotent le contexte dynamique du client. La base de données des profils MPEG 21 permet de sauvegarder pour chaque profil statique des profils dynamiques. La séparation entre les deux a pour but de faciliter la recherche et l'adaptation des présentations multimédia.

Le serveur dès qu'il reçoit un nouveau profil statique, il lui associe un identifiant *IdProfilStatique*, et le sauvegarde dans la base de données et envoie une copie de cet identifiant aux clients associées à ce profil statique. Dorénavant, le client utilise cet identifiant au lieu de son fichier profil statique dans les futures requêtes de données, dans ce cas l'authentification sera faite entre le client et le serveur profils grâce à l'*IdProfilStatique*.

#### IV.5.1.2 Module de gestion de contenus multimédia

Le module de gestion de contenus multimédia se charge de gérer tous les objets média qui forment les documents SMIL, avec toutes leurs descriptions MPEG-7. Il est formé par le groupement des deux serveurs : le serveur contenus et le serveur des descriptions MPEG-7. Ce module assure le fonctionnement de la base de données des objets média et ses interactions avec les utilisateurs (auteurs) en cas d'ajout, mise à jour ou suppression des objets, et gère ses communications avec les autres serveurs ; il gère également la base de données XML qui contient les descriptions MPEG-7 de ces objets, et assure sa manipulation par les utilisateurs (c'est -à-dire : l'ajout, la suppression et la modification).

Un objet média peut faire partie d'un ou plusieurs documents SMIL, et peut avoir plusieurs alternatives où une alternative est une copie de l'objet mais avec des caractéristiques différentes (comme le biterate, la résolution etc.), pour chaque objet il existe une description MPEG-7 correspondante détaillant quelques caractéristiques nécessaires à l'opération d'adaptation de contenu et facilitant la recherche d'information.

La gestion des descriptions explicites du contenu multimédia est de très grande importance dans le cadre d'un système d'adaptation de présentation multimédia. Pour réaliser cette tâche correctement, un système d'adaptation doit avoir une description exacte et complète du contenu à adapter et un moteur qui le gère. Il est vrai que quelques caractéristiques de contenu peuvent parfois être directement extraites du contenu lui-même, comme sa modalité ou son format. Cependant, cela peut parfois s'avérer difficile.

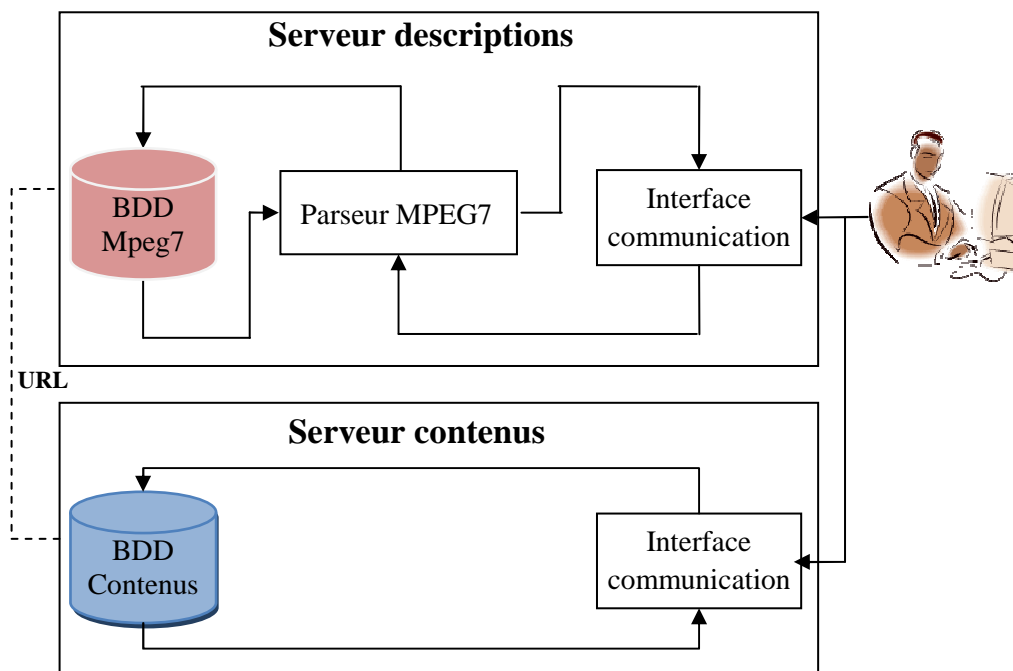
```
<?xml version="1.0" encoding="UTF-8"?>
<mpeg7:Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2004"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-
7_schema_files/mpeg7-v2.xsd">
  <mpeg7:DescriptionMetadata>
    <mpeg7:PrivateIdentifier>video.wmv</mpeg7:PrivateIdentifier>
  </mpeg7:DescriptionMetadata>
  <mpeg7:Description xsi:type="ContentEntityType">
    <mpeg7:MultimediaContent xsi:type="AudioVisualType">
      <mpeg7:AudioVisual>
        <mpeg7:MediaInformation>
          <mpeg7:MediaProfile>
            <mpeg7:Name xml:lang="en">asf</mpeg7:Name>
          </mpeg7:FileFormat>
          <mpeg7:FileSize>64446874</mpeg7:FileSize>
          <mpeg7:BitRate variable="false">2099203</mpeg7:BitRate>
          <mpeg7:VisualCoding>
            <mpeg7:Format>
              <mpeg7:Name xml:lang="en">WMV2</mpeg7:Name>
            </mpeg7:Format>
            <mpeg7:Pixel aspectRatio="1.000" bitsPer="8"/>
            <mpeg7:Frame aspectRatio="1.770" height="904"
width="1600" rate="5.000" structure="" />
          </mpeg7:VisualCoding>
          <mpeg7:AudioCoding>
            <mpeg7:Format>
              <mpeg7:Name xml:lang="en">WMA</mpeg7:Name>
            </mpeg7:Format>
            <mpeg7:AudioChannels>2</mpeg7:AudioChannels>
            <mpeg7:Sample rate="44100" bitsPer="16"/>
          </mpeg7:AudioCoding>
        </mpeg7:MediaFormat>
      </mpeg7:MediaProfile>
    </mpeg7:MediaInformation>
    <mpeg7:MediaTime>
      <mpeg7:MediaTimePoint>T00:00:00:0F1000</mpeg7:MediaTimePoint>
      <mpeg7:MediaDuration>PT0H4M5S3N5F</mpeg7:MediaDuration>
    </mpeg7:MediaTime>
    </mpeg7:AudioVisual>
  </mpeg7:MultimediaContent>
</mpeg7:Description>
</mpeg7:Mpeg7>
```

Figure IV.6 : Exemple de description MPEG-7

Il y a ainsi certaines caractéristiques du contenu qui doivent être données expressément par l'auteur.

Les informations sémantiques telles que le débit binaire d'un fichier audio, les paramètres de codage ou d'autres paramètres tels que la résolution minimale d'une vidéo. Ces informations sont très utiles pour le processus d'adaptation et facilitent la navigation dans les bases de données multimédia.

Dans notre cas, le module de gestion de contenus multimédia (**Figure IV.7**) est l'entité qui manipule différents éléments multimédia pris en charge par le système. Il est composé d'une base de données multimédia, où sont stockés et classés des objets multimédia et leurs alternatives ; et une base de données XML où sont stockées les métadonnées MPEG-7 de description des objets médias ainsi que leurs alternatives, chaque description MPEG-7 contient une URL vers la ressource correspondante.



**Figure IV.7 : Module de gestion de contenus multimédia**

Ce module fournit aux auteurs des interfaces facilitant l'ajout de nouveaux items dans la base de données multimédia de contenus, avec leurs descriptions MPEG-7 dans la base de données XML (par exemple le code de la **figure IV.6**), ou la suppression des items existants. Il doit être capable d'effectuer les modifications sur les bases de données, sur le contenu lui-même ou sa description, de telles modifications peuvent causer une incohérence de données

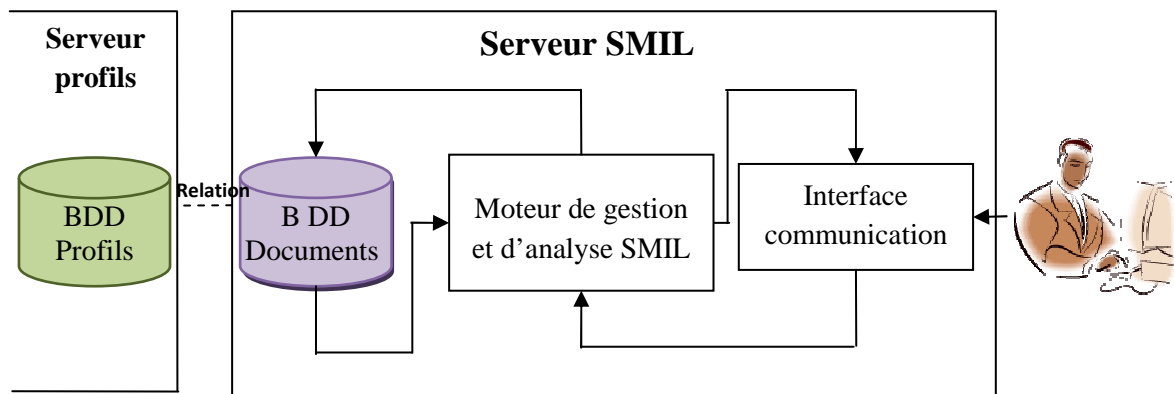
que le système doit prendre en considération par la propagation des mise-à-jours en liens avec l'élément modifié.

Il est à noter que les documents multimédia SMIL sont stockées dans la base de données des documents, et font référence aux URL des éléments média, cela est aussi le cas pour la description MPEG-7 de chaque élément. Le fait que MPEG-7 soit basé sur XML tout comme MPEG-21 et SMIL, facilite la manipulation des trois langages dans un même système. Il est à noter que d'autres standards comme XPath [XPATH], CMML [CMML] peuvent être aussi utilisés, étant donné qu'ils sont basés sur XML. L'analyse (la recherche et la navigation dans un code XML (une base de données XML)) est généralement réalisée par un outil d'analyse appelé *parseur*. Vu le rôle important qu'il joue sa présence est primordiale dans notre architecture.

#### IV.5.1.3 Module de gestion de documents multimédia

Les documents SMIL représentent le cœur du système d'adaptation de présentation multimédia. En plus de leur importance, les documents multimédias doivent être aisément accessibles par l'utilisateur, l'auteur et le proxy d'adaptation. Ce module de gestion de documents multimédia (voir **figure IV.8**) est le composant qui se charge de stocker, rechercher, ajouter, supprimer et modifier les documents multimédias SMIL et leurs alternatives.

Plus concrètement, dans notre architecture ce module est composé d'une base de données XML, dans laquelle seront sauvegardés les documents SMIL et leurs alternatives (leurs versions) ; et d'un parseur SMIL qui assurera la recherche et la navigation dans la base de données. Des interfaces assureront les liens entre ce module et les modules de gestion des contextes et de gestion des contenus (les documents contiennent les liens vers les objets média) et le proxy d'adaptation qui effectuera l'adaptation et créera les alternatives de documents qui devront être sauvegardés dans la base de données de documents. La manipulation des documents SMIL (ajout, suppression et modification), seront mise à la disposition de l'auteur par des interfaces graphiques qui devront tenir compte de la cohérence d'information et de la gestion des versions.



**Figure IV.8 : Module de gestion de documents multimédia**

Un document SMIL original peut avoir plusieurs alternatives sauvegardées dans la base de données, ces alternatives sont créées suite à des opérations d'adaptations, il est possible aussi qu'un profil statique possède plusieurs profils énergétiques sauvegardés dans la base de données des profils. Une alternative d'un document SMIL est adaptée à un seul profil (profil statique + profil énergétique).

Toute alternative de présentation adaptée selon un profil statique et un profil dynamique devra être sauvegardée dans la base des documents. Par conséquent, les identifiants des profils statique et énergétique pour lesquels l'alternative a été adaptée, devront figurer dans la base de données des documents afin de permettre la recherche du document et de son indexation.

La **figure IV.9** résume les relations entre ces éléments constituant les bases de données (des profils MPEG21 et des documents SMIL). Ces sont primordiaux, d'une part parce qu'ils permettent de respecter la structure des bases de données et les règles d'intégrité, et d'autre part parce qu'ils faciliteront les différentes manipulations comme la recherche, la mise à jour etc.

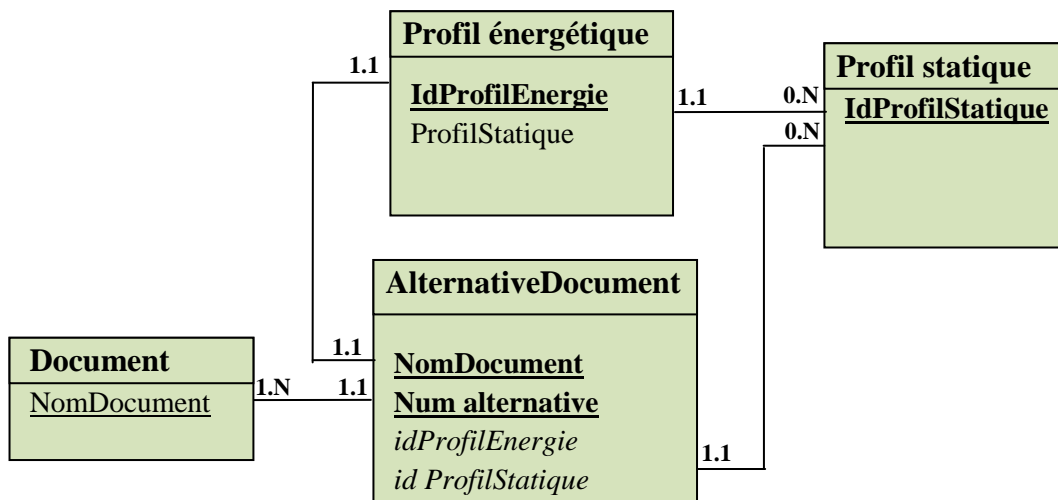


Figure IV.9 : Les relations entre les bases de données

### IV.5.2 Le proxy d'adaptation

Le proxy d'adaptation est le module le plus important dans notre architecture. L'utilisation d'un proxy au lieu d'une architecture client/serveur simple, est motivée par plus d'une raison. D'un côté, l'adaptation est une opération très gourmande qui requière beaucoup de ressources matérielles et énergétiques. Son implémentation au niveau d'un serveur réduit les performances de son fonctionnement et peut conduire à une dégradation de la qualité de service ; d'ailleurs le regroupement des données et des traitements sur une même station de travail est fortement déconseillé dans la plupart des architectures. Par ailleurs, son installation sur les dispositifs mobiles des clients est pratiquement impossible, vue leurs limitation en ressources de calcul, de stockage et d'énergie.

Le proxy est une station intermédiaire entre les clients et les serveurs comme le montre la **figure IV.10**, servant à assurer la communication entre les deux, en se basant sur des protocoles de communication ou de transfert de données ftp, http, rtsp etc.

Le proxy de notre architecture est un proxy d'adaptation, son rôle principal est l'adaptation des documents multimédia. Ces derniers sont délivrés par le serveur de documents multimédia en rapport avec le profil des clients. Pour ce faire, un module que nous appelons module d'adaptation, applique des règles et des stratégies d'adaptation et d'optimisation, cela avec la collaboration : du client qui lui fournit ses disponibilités énergétiques actuelles, grâce à un agent capteur ; et les différents serveurs qui lui fournissent

les informations concernant le profil du client et les documents multimédia requis et les objets média qu'ils contiennent.

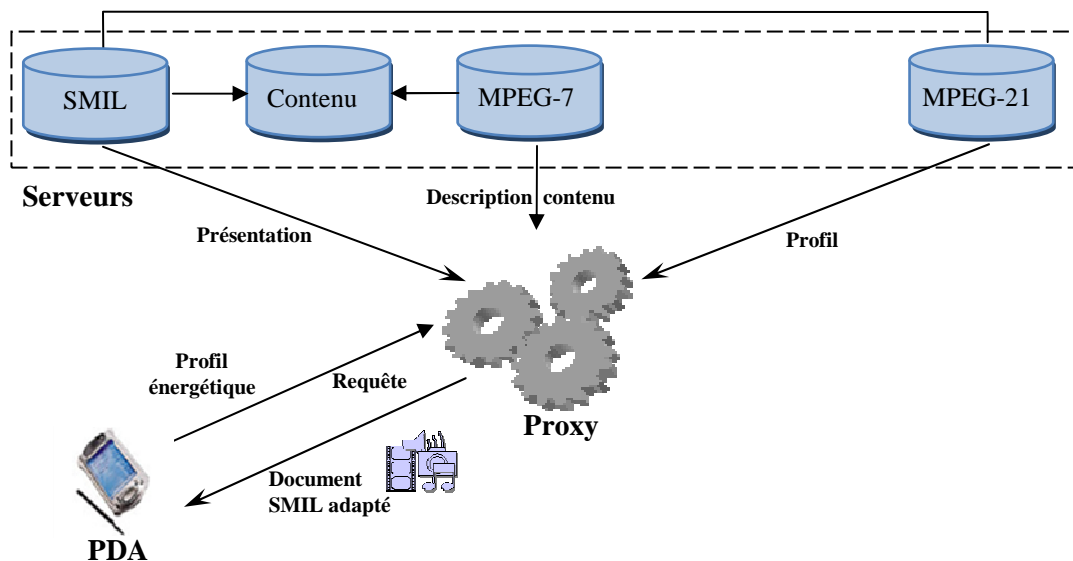


Figure IV.10 : Le proxy d'adaptation

### IV.5.3 Processus d'adaptation

L'adaptation de documents SMIL est une séquence d'opérations de communication et de transfert de données qui fait intervenir plusieurs acteurs (clients, proxy et serveurs). Elle se déroule grâce à des protocoles spécifiques à développer, formés de commandes de communication entre les clients, le proxy et les serveurs ; en plus des protocoles de la couche application habituels (Http, Rtp, etc.) (Figure IV.11).

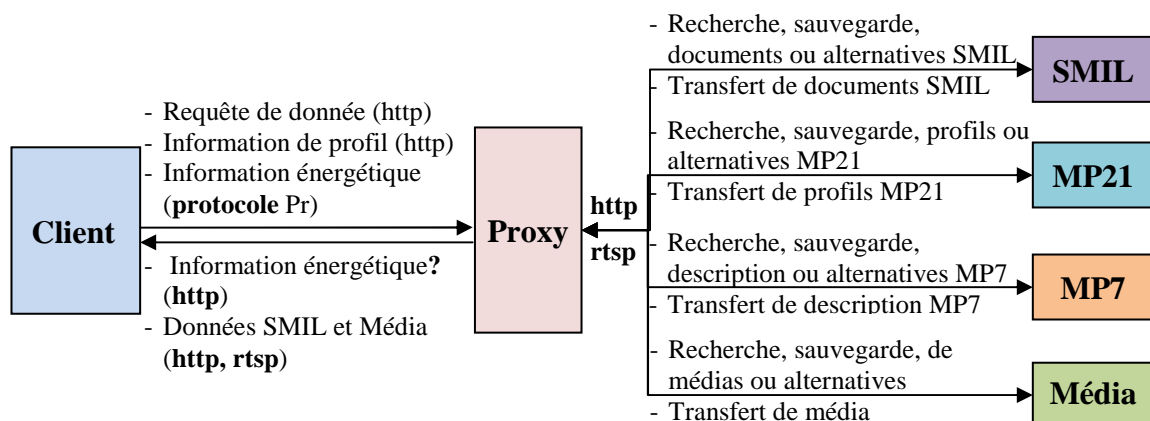


Figure IV.11 : Trafic de données dans le système



En premier lieu, la communication entre un client et le proxy consiste en des requêtes, des informations sur le profil statique et énergétique envoyées par le client vers le proxy. Nous relevons aussi la description et le contenu d'une présentation SMIL envoyés par le proxy vers le client, etc. En second lieu, la communication entre les différents serveurs et le proxy est constituée : de requêtes de recherche, de récupération de données envoyées par le proxy ; et de flux de données de média, de descriptions MPEG7, de profils MPEG21, des OK et NOK envoyés par les différents serveurs.

Certaines de ces communications nécessitent de développer des protocoles de communication au niveau applicatif, assurant la communication entre les différents modules, d'autres nécessitent d'étendre les fonctions des protocoles usuels (comme la fonction POST de protocole http).

Le processus d'interrogation et de fonctionnement des différents modules dans le but d'adapter une présentation multimédia se déroule selon le scénario suivant :

## **LE SCENARIO D'ADAPTATION :**

### **1- Le client ne s'est jamais connecté à la plateforme**

- a) Il récupère son profil statique et interroge le serveur des profils si sa description Mpeg-211 existe dans la base de données. Si le profil statique existe<sup>4</sup>, alors son *IdProfilStatique* est récupéré et envoyé vers le client. Sinon ce nouveau profil est inséré dans la base de données et son *IdProfilStatique* est envoyé vers le client afin qu'il puisse s'identifier lors des transactions futures.

### **2- Le client désire consulter une présentation SMIL**

- a) L'agent installé synthétise le profil énergétique du client.
- b) Ce profil est envoyé via le proxy vers le serveur des profils Mpeg-21 pour vérifier s'il y est associé à son profil statique ce profil énergétique. Si ce dernier existe, son *IdProfilEnergie* est récupéré et envoyé vers le client via le proxy. Sinon ce nouveau profil est rajouté dans la base de données et son *IdProfilEnergie* est envoyé vers le client.
- c) lorsque le client reçoit *IdProfilEnergie* alors il envoie une requête contenant *IdProfilStatique* plus *IdProfilEnergie* plus l'URL du document SMIL au serveur des

---

<sup>4</sup> D'autres terminaux clients se sont déjà enregistrés dans la base de données avec un profil similaire.

documents SMIL.

- d) Le serveur des documents SMIL recherche une alternative du document SMIL correspondante au profil statique et énergétique. Si cette dernière est trouvée, elle est alors récupérée et envoyée en premier lieu au proxy qui se chargera de récupérer les objets requis dans la présentation avant d'envoyer le document SMIL au client.
- e) Sinon le serveur des documents SMIL envoie une alternative du document SMIL la plus proche en terme de profils statique et énergétique vers le proxy pour son adaptation.

### **Adaptation**

Le proxy récupère le profil statique et énergétique du client, estime si l'adaptation est possible en fonction des contraintes exprimées dans le profil client. Si cette adaptation est jugée réalisable, il procède à une adaptation analytique, puis recherche les alternatives des médias dans la base de données correspondant à cette adaptation, en invoquant leurs descriptions MPeg7.

Si une description Mpeg-7 correspondant au média existe alors ce dernier est récupéré et envoyé vers le proxy. Sinon le proxy récupère l'objet source et procède à son adaptation et sauvegarde le résultat dans la base de données avec sa description Mpeg-7.

Il procède ensuite à l'adaptation structurelle du document en terminant par remplacer les anciennes URL par les nouvelles URL des objets adaptés.

Une fois le document adapté, il est rajouté dans la base des documents SMIL comme nouvelle alternative avec comme référents *Nomdocument* *Numalternative* *IdProfilStatique* et *IdProfilEnergie*.

Enfin, le proxy envoie le document adapté au client.

- f) Lorsque le client reçoit le document SMIL, il exécute son code et envoie des requêtes de délivrance de médias au proxy selon le scénario du document. Ce dernier procédera à l'envoi des objets requis vers le client. Ces objets ont été au préalable soit récupérés à partir du serveur du contenu soit adaptés à partir des objets sources.

### **IV.5.4 Adaptation énergétique des documents multimédia**

L'adaptation d'un document multimédia le permet d'être exécuté sur un terminal mobile, doit garantir un coût énergétique inférieur à son profil énergétique. Le processus d'adaptation est divisé en deux phases, comme illustrée dans la **figure IV.12**. La première est basée sur le contenu concerne la transformation des objets et est discuté en § IV.5.4.1. La

deuxième est basée structure concerne la transformation de code source SMIL et est présentée en section § IV.5.4.2.

Le profil statique, et le profil énergétique qui conduisent l'adaptation, regroupent toutes les contraintes que le document doit surmonter avant d'être transmis au client. Le profil codé en MPEG-21 ainsi que le document SMIL et les objets média qu'il référence, sont envoyés des serveurs appropriés vers le proxy, une fois que l'opération d'adaptation soit enclenchée.

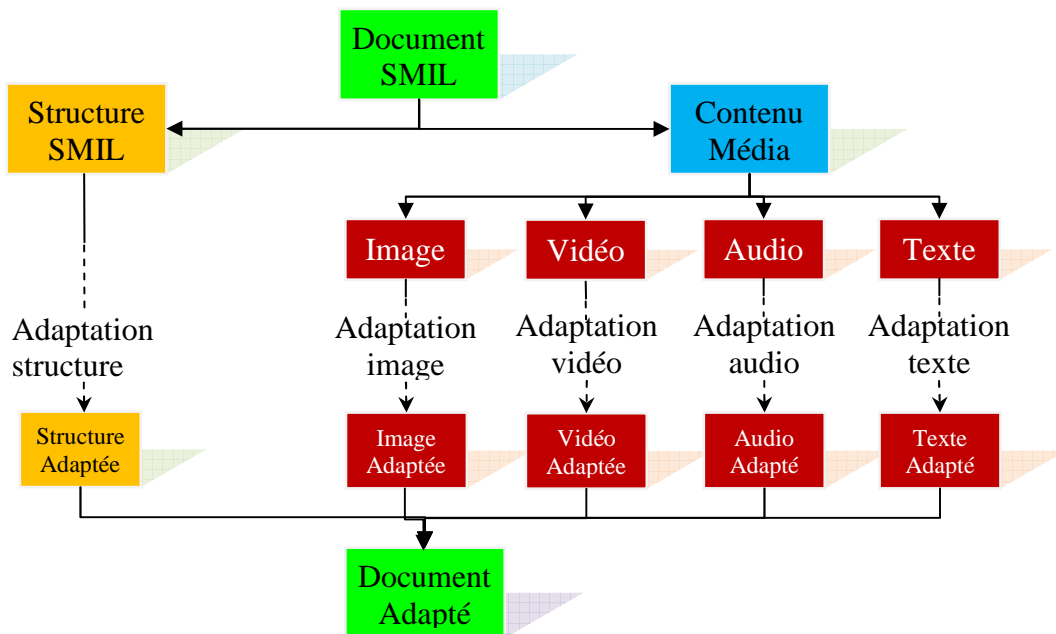


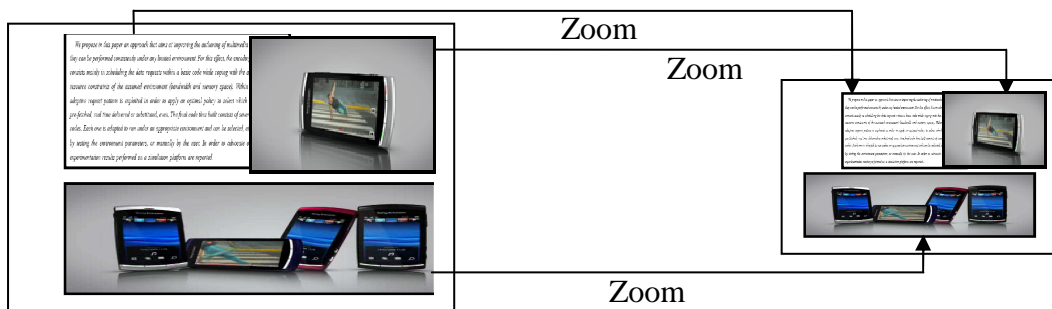
Figure IV.12 : Le processus d'adaptation

#### IV.5.4.1 Adaptation de contenu

L'adaptation du contenu se résume en trois opérations : le transmodage, le transcodage et la transformation. Les opérations d'adaptation du contenu sont aussi dépendantes de type (image, vidéo, audio ou texte) de contenu à adapter, et de l'évolution de la consommation d'énergie dans le temps au niveau client.

Généralement, lors de l'édition d'une présentation SMIL, nous ne savons pas, au préalable, les dimensions de l'écran sur lequel elle sera affichée. Souvent ces dimensions ne correspondent pas aux écrans des appareils mobiles. La solution consiste souvent à la réduction de leurs résolutions. Cette opération effectue une sorte de changement d'échelle en réduisant la taille des éléments visuels à des valeurs plus petites correspondant aux paramètres de l'appareil cible. Ce type d'opération s'appelle « transformation ». Contrairement au

transmodage, les transformations ne changent pas la modalité d'éléments, mais elle modifie leurs caractéristiques. L'opération est schématisée par la **figure IV.13** :



**Figure IV.13 : Le changement d'échelle**

Les conditions booléennes *VidéoCapable*, *TexteCapable*, *ImageCapable* et *AudioCapable*, servent à vérifier à travers le profil, la capacité du client à traiter les différentes modalités d'objet média (image, audio, texte et vidéo). Dans le cas où le client n'est pas en mesure d'en traiter une (ou plusieurs), les objets de cette modalité qui peuvent se trouver inclus dans les documents SMIL doivent être convertis à une modalité que le client peut traiter. Dans ce sens, plusieurs conversions sont possibles, par exemple : une vidéo vers une suite d'images, une vidéo vers une bande son, un texte vers une image etc. cette conversion s'appelle transmodage. Le transmodage est aussi nécessaire quand le client donne privilège à une modalité par rapport aux autres.

Dans notre cas, au lieu des adaptations standards, les opérations d'adaptation s'intéressent principalement à l'application des règles d'optimisation d'énergie que ce soit sur le contenu ou sur la structure des documents multimédia. L'adaptation des images, des vidéos, des audio ou des textes aura comme but de réduire la quantité d'énergie consommée.

Avant l'adaptation, le système doit effectuer quelques vérifications nécessaires. Tout d'abord, il a besoin de vérifier si la quantité d'énergie restante permet la lecture de document après son adaptation. Cela permet d'envisager si une adaptation du document est possible ou non en fonction des disponibilités énergétiques du client. Deuxièmement, pour chaque type d'objet média inclus dans le document (image, vidéo, audio et texte), il y a un ensemble de caractéristiques (par exemple : format, résolution, codec etc.) où chacune possède un ensemble de valeurs (par exemple : pour la caractéristique format d'image jpg, gif etc), il doit vérifier quelle est la valeur permettant d'économiser le plus d'énergie, et en même temps qu'elle soit compatible avec le profil mpeg-21 du client.

Ensuite, chaque élément sera défini par un ensemble de caractéristiques avec des valeurs optimisées pour une minimisation de la consommation énergétique. Enfin, des recherches sur les éléments possédant ces caractéristiques sont effectuées dans la base Mpeg-7 ; les éléments trouvés seront récupérés par le proxy, et ceux qui ne sont pas trouvés vont être créés par une série d'adaptations déclenchée au niveau du proxy.

La consommation d'énergie des objets média composant un document SMIL peut être considérable. Pour la réduire, le proxy aura pour tâche d'appliquer quelques techniques d'optimisation d'énergie. Les techniques que nous proposons ici se basent principalement sur le choix des paramètres des objets média tout en respectant le profil et les préférences d'utilisateurs. Ces techniques sont classées dans la suite de ce manuscrit selon le type d'objet concerné, elles consistent en des transformations et conversions des ressources ; par exemple un changement de format d'une image JPEG<sup>5</sup> en GIF<sup>6</sup> ou même une vidéo en GIF.

### A) Image

L'image est un media fondamental pour de nombreux domaines professionnels tels que la météorologie, la production de films, les publications, la télédétection, etc. En particulier, sa présence dans les documents multimédia se concrétise d'un jour à l'autre, il se peut parfois qu'on trouve une présentation entière composée que d'images. Par image, on entend l'image numérique, qui se caractérise par son format de compression, sa résolution, son type de couleurs etc. L'image est un élément visuel qui requière l'utilisation des ressources d'affichage. Les caractéristiques d'une image peuvent ajustées afin de réduire de façon considérable la consommation énergétique d'un document multimédia quelconque.

#### — *Format*

Dans le chapitre précédent, nous avons montré à travers quelques expériences que le facteur format a une influence sur la consommation d'énergie. Si une image est sous un format qui est censé consommer plus d'énergie, il faudra la convertir vers un format moins gourmand, tout en maintenant la qualité de service et le profil utilisateur requis<sup>7</sup>. Les formats sont classés selon leur consommation d'énergie. Pendant l'adaptation du contenu, le proxy

---

<sup>5</sup> JPEG : Joint Photographic Experts Group

<sup>6</sup> GIF : Graphics Interchange Format

<sup>7</sup> Afin d'éviter de tomber sur un format non compatible avec le profil du terminal cible.

vérifie la compatibilité du format qui consomme le moins d'énergie. S'il est compatible, le proxy convertit alors l'objet vers ce format, sinon il vérifie la compatibilité du prochain format. Une vérification se fait aussi avec le profil d'utilisateur transmis par le serveur ; cela garantit que l'objet puisse être lu par le terminal. L'opération de conversion de format permet de réduire la taille de l'objet, cela implique une réduction aussi du trafic réseau. Par ailleurs, le changement de couleur réduit le temps CPU et traitement au niveau des périphériques d'affichage.

#### — *Compression*

Compresser une image c'est l'opération qui donne une nouvelle image d'une taille inférieure, avec comme effet secondaire une dégradation de sa qualité, la compression d'image est donc une arme à double tranchant. Une taille inférieure signifie un trafic réseau inférieur, cela-dit une consommation d'énergie moindre de la carte réseau sans fil. Cependant, la compression implique une décompression à la lecture impliquant un traitement additionnel consommateur d'énergie qui s'ajoute à la dégradation de la qualité suite à la perte d'informations. Une compression à un taux 20% et 50% permettra -à notre avis- de réaliser une économie d'énergie tout en préservant une qualité acceptable.

#### — *Changement de couleurs*

La troisième transformation que l'on peut appliquer sur l'image est la réduction de ses couleurs. Si une image dans son état d'origine est codée sous RGB<sup>8</sup>, l'opération consiste à réduire le nombre de bits sur lequel l'image est codée. Une définition à 256 couleurs peut être un bon compromis, elle permet d'un côté de réduire la consommation d'énergie, et d'un autre de maintenir une bonne qualité visuelle.

#### — *Luminosité*

Nous suggérons que la luminosité d'une image soit réduite d'un pourcentage à définir. Cette opération doit préserver les préférences de l'utilisateur, tout en économisant de l'énergie. Cette proposition repose sur le fait que la luminosité dans le cas des images, contrairement aux vidéos, ne peut être réduite d'un grand pourcentage, car cela a un effet négatif sur la qualité de service perçue par l'utilisateur.

---

<sup>8</sup> RGB : Red, Green, et Blue

## B) Vidéo

Les vidéos sont des éléments extrêmement gourmands en énergie. Une vidéo utilise plusieurs ressources matérielles en même temps. Elle requière l'intervention des périphériques d'affichage pour sa visualisation, la CPU pour son décodage et la carte réseau pour son transfert. Nous avons montré précédemment qu'on peut optimiser la consommation énergétique des vidéos à travers la manipulation de quelques paramètres, nous avons aussi vu que SMIL dans sa dernière version offre quelques éléments et attributs qui peuvent être utiles pour l'optimisation d'énergie. Quelques uns de ces paramètres sont exploités dans notre architecture, nous les traitons dans cette section.

### — *Format et Codec*

L'influence du choix du format vidéo sur la consommation d'énergie n'est pas prépondérante, par contre, ce facteur conditionne le choix du codec. Dans notre architecture, les codecs XviD, DivX3, DivX5, MS MPEG-4v2, H.263+, WMV2, sont privilégiés car ils impliquent une consommation moindre d'énergie à condition qu'ils soient compatibles avec les profils clients.

### — *Bit-rate*

Le bit-rate n'est pas modifiable pour tout type de vidéo. Par conséquent, son adaptation n'est appliquée que pour les vidéos qui le supportent. La réduction du bit-rate d'une vidéo permet un gain en termes d'énergie. Par ailleurs, il a été démontré qu'une baisse de bit-rate n'a pas un grand impact sur la perception visuelle du client.

### — *Frame-rate*

Le frame rate est un facteur ayant un impact très significatif sur la consommation d'énergie dans les appareils mobiles. La diminution de la valeur du frame-rate cause un défilement moins rapide mais non perceptible dans de tels équipements. Les gains obtenus peuvent être importants : d'une part, la qualité de service est garantie, d'autre part, la quantité d'énergie sauvegardée est très conséquente.

### — *Luminosité*

En outre, le facteur luminosité a aussi un effet sur la consommation d'énergie des périphériques d'affichage (le rétro-éclairage). Constatant que le degré de luminosité des

vidéos est souvent élevé, une baisse en luminosité ne sera pas remarquable par le client et aura un impact sur la préservation de la ressource énergétique. Au niveau du proxy, les vidéos sont soumises à une diminution de leurs luminosité, cette diminution sera légère, pour qu'il n'y ait pas un effet destructrice sur la qualité du média.

#### — *Résolution*

Les écrans des appareils mobiles se caractérisent par leurs tailles réduites, en revanche, une vidéo dans sa forme originale est de taille plus grande que les dimensions d'écran. Pour surmonter cette contrainte, la solution sera de réduire la résolution de la vidéo. Ce traitement permet de satisfaire la contrainte spatiale du dispositif mobile, mais aussi diminuer la facture énergétique.

### C) **Audio**

Les médias de type audio se classent en deuxième position juste après les vidéos en termes de consommation d'énergie. Ils se caractérisent par un certain nombre de paramètres comme la fréquence d'échantillonnage, la taille de l'échantillon, le pas de quantification, le nombre de canaux utilisés, etc. Par ailleurs, le son requiert de gros volumes ce qui explique pourquoi nous le trouvons généralement sous une forme compressée. Parmi les formats audio les plus connus, nous citons le format Wave de Microsoft et le format MP3. La modalité sonore est très présente dans les documents SMIL. En se basant sur ses caractéristiques, nous discutons dans cette section comment adapter un son pour réduire sa consommation d'énergie.

#### — *Format*

Il est nécessaire, dans un premier temps de rappeler que certains formats audio, populaires de surcroît, ont une faible consommation d'énergie (voir chapitre III). Toutefois, la conversion d'un format d'encodage à un autre est une opération qui demeure difficile. AAC, Ogg, MP3, MP2, WMA ou WAVE représentent les formats audio les plus connus, tout un chacun consomme une quantité d'énergie différente des autres. Par conséquent, dans notre architecture nous proposons, pour l'adaptation des objets audio, le transcodage de leurs formats vers un autre format, à condition qu'elle soit compatible avec le dispositif cible, et qu'elle consomme moins d'énergie que le format original.



---

— *Bit-rate*

Ce paragraphe a pour objectif d'expliquer comment contribuer à la réduction de la consommation d'énergie des objets média audio, en modifiant leurs débits binaires. Pour cela, nous sommes amenés à adapter ce paramètre pour tout objet audio référencé dans le document SMIL. Généralement, l'impact du changement du débit binaire sur la qualité du son n'est pas très important. Dans le but de bénéficier de toutes les options qui peuvent se présenter, nous intégrons cette technique dans notre architecture, malgré que la quantité d'énergie économisée soit faible.

— *Sample-rate*

Comparativement au débit binaire, le taux d'échantillonnage (sample-rate) donne des résultats plus motivants, c'est-à-dire il permet un gain d'énergie plus important. A cet effet, en plus de la réduction du débit binaire les objets audio subiront une deuxième réduction qui concernera, cette fois, leurs taux d'échantillonnage. Toutefois, la réduction ne peut être trop agressive de peur d'altérer la qualité sonore du média.

#### **IV.5.4.2 Adaptation structurelle**

La particularité de notre architecture est le fait que l'adaptation est effectuée pour satisfaire —en plus des contraintes de profil— la contrainte énergétique. Les techniques d'adaptation énergétique discutées dans le chapitre I, et que le système d'adaptation pourra implémenter, sont détaillées ici.

L'adaptation énergétique de la structure du document, est effectuée au niveau du proxy lors de l'adaptation, par rapport aux contraintes exprimées dans le profil énergétique du client, elle va consister à des transformations que le proxy effectue sur le code SMIL du document, par l'intégration des éléments et attributs spécifiques (<préfetch>, <accelerate> etc.), ou lors de l'édition de document par l'ajout des éléments d'annotation ou de test.

**Approche 1 :** dans le chapitre III, nous avons parlé de l'adaptation par annotation de document. L'annotation de documents SMIL est considérée comme une adaptation structurelle, elle va consister à ajouter -lors de l'édition des documents- un attribut dit de valorisation, à chaque objet média du document SMIL, permettant de savoir, selon la valeur de cet attribut, le degré d'importance de la présence de l'objet média dans le document, par la

suite lors de l'adaptation, le proxy supprimera les éléments ayant un degré d'importance bas pour l'optimisation d'énergie.

**Approche 2 :** L'élément `<prefetch>` permet à un média d'être pré-chargé en entier ou partiellement par le client avant d'être présenté. En générale, cet élément offre aux auteurs la possibilité de pré-charger, du serveur au client, des médias dont le retard de délivrance peut avoir un effet sur la cohérence du document. Cependant un pré-chargement agressif précédant la présentation des objets ne peut être envisagé que si la bande passante et l'espace mémoire disponibles sont suffisants. Par ailleurs, ce mécanisme peut être également exploité dans le but de réduire le temps de fonctionnement de la carte réseau, dans un but d'économie d'énergie. Pour cela le proxy devra planifier, selon les contraintes de bande passante et de mémoire, le pré-chargement des objets en parallèle de telle sorte à maximiser l'utilisation de la bande passante et dégager ainsi des slots de temps durant lesquels la carte réseau sera inactive et donc mise en veille. Cette opération pourra être gérée, par exemple, par le module pré-installé au niveau client pour l'estimation du profil énergétique.

Le transfert des média entre le proxy et les clients peut se faire de deux manières : en parallèle, c'est-à-dire télécharger plusieurs média à la fois, ou en séquence l'un après l'autre. Le choix entre les deux manières dépend certainement de l'impact de chacun sur l'énergie, mais aussi il dépend de l'espace de stockage au niveau des clients. Le résultat d'un transfert en séquence est une séquence de périodes de mise en veille et de reprise de la carte réseau sans fil, alors que celui d'un transfert en parallèle est une durée de mise en veille plus grande. Cela nous permet de dire que le transfert en parallèle est souhaitable à condition que l'espace mémoire du client le permette.

**Approche 3 :** La configuration appropriée des attributs `<accelerate>` `<decelerate>` caractérisant une vidéo dans un document SMIL permet de réduire sa consommation d'énergie. Ces attributs ne sont intégrés dans les éléments, mes plutôt dans l'entête `<head>` de document, plus exactement dans l'élément `<region>` qui délimite les dimensions des objets média. Ils sont utilisés particulièrement avec les média de type audio ou vidéo pour changer leurs vitesses de lecture, et faire gagner de l'énergie. Durant de la phase d'adaptation le proxy les intègre dans le code source des documents multimédia.

**Approche 4 :** Des possibilités de test sont offertes par le langage SMIL, les expressions logiques de test de l'élément `<switch>`. Permettent l'adaptation des documents multimédia

aux contraintes qui concernent : le type de système d'exploitation, la bande passante disponible, le type de CPU etc. Ces expressions sont généralement utilisées dans les systèmes avec adaptation à priori, dans un contexte de spécification des alternatives de document avec chaque composition distincte de ses paramètres spécifique et compatible à un dispositif particulier. En chapitre I, nous avons montré qu'il est possible d'ajouter des fonctions à ce module (*StateTest*) et de les utiliser dans des expressions logiques de test, afin d'optimiser la consommation d'énergie. Les fonctions qu'on propose ici complètent les autres approches d'optimisation discutées dans ce document. Le pré-chargement (pre-fetching) des éléments média, comme montré précédemment, dépend de la bande passante et l'espace mémoire disponibles au niveau de l'appareil client. Une fonction F1 à développer va se charger de vérifier si la bande passante disponible permet le pré-chargement, alors qu'une autre fonction F2 s'occupera de vérifier si l'espace mémoire disponible l'autorise.

Ainsi, le contrôle de la fréquence de la CPU, la luminosité du rétro-éclairage, etc. peuvent également être manipulés grâce à des fonctions à développer, et qui en communiquant avec l'Agent énergie permettront d'effectuer les changements nécessaires afin d'optimiser l'énergie. L'intégration de ces fonctions dans le code de document, va certainement s'effectuer au niveau du Proxy lors de l'adaptation structurelle.

#### **IV.5.5 Optimisation matériels**

Dans la partie optimisation du chapitre III, nous avons présenté un état de l'art non exhaustif des techniques d'optimisation d'énergie dans les systèmes ayant des ressources énergétiques limitées. Nous avons donc distingué des techniques centrées sur la consommation des différents composants matériels des appareils mobiles (exemple : la carte réseau sans fil). Dans l'architecture proposée; nous essayons d'exploiter ces techniques de telle sorte que la consommation d'énergie soit optimisée.

Nous discutons ici comment on peut intervenir sur certains équipement matériels du client afin de réduire la facture énergétique. L'optimisation de la consommation d'énergie du hardware repose sur quelques travaux discutés précédemment (chapitre III). La carte réseau sans fil, le rétro-éclairage, et la CPU représentent les trois composants que nous traitons dans ce qui suit.

### A) Carte réseau

La carte réseau est sans doute le plus grand consommateur d'énergie. Il est évident également que si on peut arriver à mettre en veille la carte ou réduire le trafic réseau, la consommation d'énergie globale va être réduite d'un pourcentage considérable. Rappelons qu'une carte réseau sans fil en mode veille, consomme autant qu'en mode réception et largement moins qu'en mode émission. Cette procédure peut être gérée par l'agent installé au niveau client. Cet agent sera chargé de la surveillance de la carte réseau sans fil et de la mettre en veille une fois les transferts de données terminés.

### B) Backlight

Dans le chapitre précédent nous avons vu que les éléments visuels représentent les éléments les plus consommateurs d'énergie. Nous pouvons envisager de minimiser la puissance d'énergie que le rétro-éclairage consomme. L'agent au niveau du client se charge de diminuer la luminosité de l'écran lors de la présentation exclusive d'images ou de clips audio ; ces derniers sont peu sensibles à la modification de la luminosité que la vidéo.

### C) CPU

Nous avons montré dans les chapitres précédents que les techniques DVFS (Dynamic Voltage and Frequency Scaling), permettent de diminuer la facture énergétique de la CPU. Elles reposent sur des réductions et le contrôle planifiés de la fréquence de fonctionnement de la CPU, grâce à un programme d'ordonnancement des tâches et des activités de la CPU. L'ordonnancement vise à : donner à la CPU un nombre limité de tâche durant des périodes de temps précises ; éviter d'exécuter beaucoup de tâches en même temps (en parallèle) ; prévoir les temps de non-activité (absence d'activité) ; et mettre la CPU à une basse fréquence et impliquant par conséquent un voltage réduit.

Dans notre architecture, il faut d'abord envisager, si c'est possible, que des médias d'un document SMIL soient mis en séquence au niveau du Proxy, par le remplacement structurellement de l'élément SMIL de parallélisation **<par>** par l'élément **<seq>**, ou une combinaison des deux éléments (permettant de rendre en séquence un sous bloc de bloc parallèle délimité par l'élément **<par></par>**). L'agent implanté au niveau client pourra grâce à un échéancier des tâches fourni par le proxy, procéder à l'ordonnancement des activités et la surveillance de la CPU. Le but est d'arriver à exécuter les médias, qui sont déjà

pré-chargés au niveau de la mémoire, à des fréquences basses et de réduire la fréquence de la CPU dès qu'une tâche est achevée.

## IV.6 Tests expérimentaux

L'état de l'art présenté dans les chapitres précédents, a permis d'inventorier les techniques utilisées pour optimiser l'énergie et augmenter ainsi la durée de vie de la batterie. Ces techniques bien qu'elles soient efficaces, n'ont été appliquées et testées que séparément. Aucune approche n'a tenté de combiner ces techniques ensemble afin d'arriver à une économie globale plus importante d'énergie. Il est à noter qu'une telle combinaison peut provoquer des résultats contradictoires ou inattendus, pouvant causer la production d'un contenu de mauvaise qualité visuelle. Par conséquent, l'algorithme d'adaptation d'une présentation multimédia devra être un compromis entre la qualité du produit et la facture de consommation énergétique.

Dans cette section, nous discutons d'une plateforme de tests à réaliser afin de valider toute architecture, voire un algorithme d'adaptation de présentation multimédia. Dans notre cas, ces tests permettront de valider un algorithme ou une technique globale d'économie d'énergie, et cette technique pourra être intégrée dans un système d'adaptation de présentations multimédia, et lui permettre de fournir des présentations adaptées et optimisées suivant le profil énergétique de client.

### IV.6.1 Test de compromis entre facteurs d'optimisation

Dans le chapitre précédent, plusieurs approches de conservation d'énergie ont été exposées, par la donnée de résultats expérimentations [Li,08][Li,10]. Ces expérimentations nous permettent de déduire d'une manière générale que, l'ajustement d'un paramètre ou d'un sous ensemble de paramètres (biterate, framerate, résolution, choix du codec, format de codage, etc.), peut diminuer considérablement la quantité d'énergie consommée. Cependant le bon ajustement de la valeur de ces paramètres ne peut pas se faire d'une manière aléatoire, sinon cela pourra causer involontairement, une dégradation de la qualité de la présentation. Au fait, la solution à ce problème est expérimentale, elle consiste à utiliser un encodeur de média, pour coder des contenus média de type audio et vidéo. Les encodeurs des medias ont la capacité de modifier les valeurs des paramètres biterate, framerate et résolution, de fixer le codec et le format de contenu. En utilisant un encodeur on va encoder un échantillon

d'alternatives avec des valeurs bien précises de paramètres (biterate, framerate et résolution) qu'on doit définir au préalable.

Une fois l'étape d'encodage terminée, on mesure la consommation d'énergie en faisant jouer la présentation SMIL adaptée sur un équipement mobile. Ces mesures devront être réalisées sur une architecture d'expérimentation et traduites par la suite en des histogrammes comparatifs. Ces histogrammes vont déterminer le compromis idéal entre les valeurs des différents paramètres garantissant une qualité de service satisfaisante d'un côté, et une gestion de consommation d'énergie optimale, de l'autre.

#### **IV.6.2 Test des techniques d'adaptations et d'optimisation**

L'opération d'adaptation d'un contenu multimédia aux contraintes exprimées dans le profil utilisateur, est soit une conversion des ressources ou une modification de bitstream c'est-à-dire une modification au niveau binaire de l'élément. Dans la littérature on trouve que : pour une même opération d'adaptation, il peut exister plusieurs approches. Les techniques d'adaptation de contenu multimédia sont classées en transcodage (conversion de codage ex. de format flv au format wmv), transmodage (conversion de modalité ex. conversion d'une vidéo en son) et transformation (changement des caractéristiques du contenu ex. réduction de la taille d'une image). L'idée de ce test est d'évaluer la consommation énergétique des présentations avant et après l'adaptation pour décider si le système réalise les objectifs voulus.

#### **IV.6.3 Résultats attendus**

Le sujet que nous traitons dans ce mémoire est : l'optimisation de la consommation d'énergie dans les terminaux mobiles, par adaptation des présentations multimédia. L'objectif de notre travail est d'arriver à définir les grandes lignes d'une architecture globale d'un système d'adaptation des présentations multimédia optimisées selon les informations énergétiques dynamiques du terminal demandeur. De ce fait, les expérimentations que nous voulons faire, nous permettront de mesurer la quantité d'énergie que le système va permettre de conserver, et de valider ainsi l'architecture de système d'adaptation et d'optimisation de présentations SMIL proposée. Dans ce qui suit, nous détaillons l'architecture expérimentale.

#### IV.6.4 Architecture expérimentale de tests

L'architecture expérimentale est illustrée en **figure IV.14**, cette architecture est composée d'un terminal mobile dont on a retiré la batterie, relié à un serveur, par une connexion sans fil. Le serveur est la station où les présentations SMIL et leurs éléments sont hébergés. Cette station doit être d'une grande capacité en termes de mémoire, d'espace de stockage et de puissance de calcul.

Le terminal, quant à lui, est placé en série entre un générateur électrique de 5 volts, et une résistance électrique de 1 ou 2 Ohms. Pour mesurer l'énergie, un BNC 2110 connecté via une carte d'acquisition est placé en parallèle à la résistance. La carte d'acquisition est de type National Instruments DAQ Board est installé sur une station, sur laquelle le logiciel de mesure de la consommation énergétique LabView [**LABVIEW**] est installé.

Nous utilisons une carte d'acquisition de données DAQ PCI NI, pour mesurer la différence de tension à travers la résistance à 1000 échantillons/sec. La mesure d'énergie est faite en utilisant LABVIEW, un logiciel de présentation d'analyse et d'acquisition de données. Nous calculons la consommation d'énergie instantanée correspondante à chaque échantillon et l'énergie totale utilisant les deux équations (1) et (2):

$$P_{Inst} = \frac{V_R}{R} \times V_{PDA} \quad (1)$$

$$E = \sum P_{Inst} \times T (T = 1/1000) \quad (2)$$

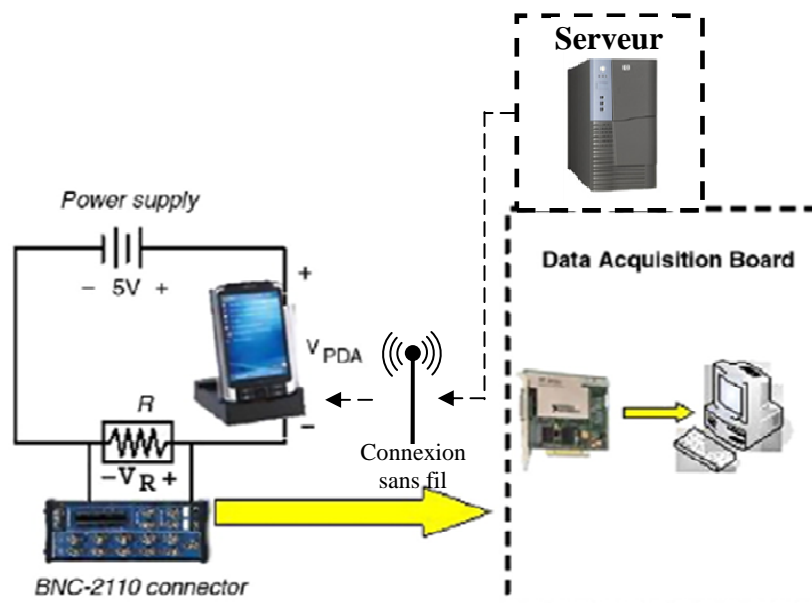


Figure IV.14 : L'architecture de mesure d'énergie

## IV.7 Conclusion

L'architecture que nous avons présentée dans ce chapitre, se compose de quatre (04) modules :

- Module de gestion de contenus multimédia;
- Module de gestion de contextes d'utilisations ;
- Module de gestion de documents multimédia ;
- Proxy d'adaptation et d'optimisation.

Diverses techniques et méthodes d'adaptation pouvant être utilisées pour l'adaptation de contenu et de la structure de document existent : transcodage, transmodage et transformation.

L'objectif principal de ce chapitre était de montrer comment intégrer ces techniques à travers une architecture de fourniture de service multimédia. Pour terminer nous avons présenté une plateforme de tests expérimentaux afin de valider cette architecture.



# Conclusion et perspectives

## 1. BILAN

De nos jours, les applications multimédias tendent à se vulgariser en invertissant les environnements mobiles. Beaucoup de recherches visent d'ailleurs à promouvoir des technologies pour le déploiement des applications multimédias ubiquistes. Le but est d'offrir des services partout, n'importe quand, et à n'importe quel équipement.

Les difficultés du domaine, sont généralement, liées à la diversité des documents et des contenus multimédias, à l'hétérogénéité des réseaux d'accès et à la diversification des équipements mobiles. En effet, c'est avec cet objectif et sous ces contraintes que de nombreux mécanismes et systèmes d'adaptation multimédia ont été proposés, chacun destiné à répondre à des besoins particuliers, en essayant de résoudre les problèmes rencontrés. Toutefois, le facteur de la préservation du facteur énergétique, malgré son importance, n'a pas suscité un grand intérêt, dans le processus d'adaptation. De ce fait, nous avons essayé à travers ce mémoire d'inventorier les principales techniques et approches proposées dans la littérature abordant la gestion du facteur énergétique. Il était question de dégager une solution permettant l'optimisation de l'énergie consommée par les applications multimédias déployées dans les appareils mobiles.

A cet effet, dans un premier temps, nous avons présenté une analyse complète de la version 3.0 du standard libre de spécification de présentations multimédia SMIL. Cela afin d'en tirer les options et les fonctionnalités qui peuvent aider à la résolution du problème posé.

Par la suite, après présentation de quelques définitions fondamentales sur le principe d'adaptation, nous avons procédé, dans un second temps, à une synthèse classifiée des différentes techniques d'optimisation d'énergie recensées dans la littérature. Cela nous a permis de dégager une solution d'une architecture globale pour l'adaptation des documents multimédias dans le but d'optimiser la gestion de la ressource énergétique.

Pour conclure, ce travail a représenté pour nous une première expérience dans le domaine de la recherche scientifique, qui fût à plus d'un titre bénéfique, enrichissante et positive. Premièrement, elle fût bénéfique car elle nous a permis de nous initier à un domaine de recherche scientifique ; deuxièmement, elle fût enrichissante, vue les connaissances qu'elle nous a permis d'acquérir ; enfin, elle fût positive car, nous pensons avoir contribué, un tant soit peu, à une problématique assez récente et encore peu investie.

## 2. PERSPECTIVES

Comme suite à ce travail de recherche, nous proposons de :

1. Réaliser des expérimentations afin de valider des approches d'adaptation pour l'optimisation de la consommation d'énergie;
2. Concevoir et Réaliser l'architecture proposée en y intégrant des techniques d'adaptation validées ;

# Références bibliographiques

## 1- Articles et thèses

### A

[Ad,07] : Janet Adams, Gabriel-Miro Muntean, “*Power Save Adaptation Algorithm for Multimedia Streaming to Mobile Devices*”. In IEEE ICC (2007), 2007.

[Az,05] : Mariam Kimiaei Asadi. “*Adaptation de Contenu Multimédia avec MPEG-21 : Conversion de Ressources et Adaptation Sémantique de Scènes*”. Thèse de doctorat soutenue à l’Ecole Nationale Supérieure des Télécommunications de Paris, France, le 30 juin 2005.

### B

[Be,06] : GIRMA berche hagos, “*accès et adaptation du contenu multimédia pour les systèmes pervasifs*”, thèse de doctorat soutenue à l’institut national des sciences appliquées du Lyon, France, le 25 septembre 2006.

[Bo,01]: S. Borkar, “*Low power design challenges for the decade,*” in Proceedings of the 2001 conference on Asia South Pacific design automation, 2001.

[Bo,03] : Böszörményi L., Hellwagner H., Kosch H., Libsie M., Podlipnig S. “*Metadata driven adaptation in the ADMITS project*”, EURASIP Signal Processing : Image Communication Journal, vol 18, n° 8, Septembre 2003, pages : 749-766.

[Bu,05] : G. C. Buttazzo, “*Rate monotonic vs. EDF: judgment day, Real-Time Systems*”, Springer Science + Business Media, Inc. Manufactured in The Netherland vol. 29, pp. 5-26, 2005.

### C

[Ch,02] : Surendar Chandra and Amin Vahdat. “*Application-specific network management for energy-aware streaming of popular multimedia formats*”. In ACM Usenix Annual Technical Conference, 2002, pp.329 – 342;

[Ch,03] : Surendar Chandra. “*Wireless network interface energy consumption*”. *Multimedia Systems*, 9(2) : pp.185–201, 2003.

[Cha,04] : Surendar Chandra. “*Energy conservation in ad hoc multimedia networks using traffic-shaping mechanisms*”. *SPIE*, 5305:40, 2004.

[Cho,04] : Kihwan Choi, Wonbok Lee, Ramakrishna Soma, and Massoud Pedram, “*Dynamic voltage and frequency scaling under a precise energy model considering variable and fixed components of the system power dissipation*”. In *IEEE/ACM International conference on Computer-aided design*: IEEE Computer Society, 2004.

[Ch,04]: Naehyuck Chang, Inseok Choi, Hojun Shim: “*DLS: Dynamic Backlight Luminance Scaling of Liquid Crystal Display*”. *IEEE transactions on very large scale integration (VLSI) systems*, vol. 12, no. 8, august 2004, pp.837-846.

[Ch,05] : Liang Cheng, Stefano Bossi, Shivajit Mohapatra, Magda El Zarki, Nalini Venkatasubramanian, and Nikil Dutt, “*Quality Adapted Backlight Scaling (QABS) for Video Streaming to Mobile Handheld Devices*”. pp.662–671, 2005.

[Ch,06]: Wei-Chung Cheng, Chain-Fu Chao: “*Minimization for LED-backlit TFT-LCDs*”, *ACM, DAC 2006*, July 24–28, 2006, San Francisco, California, USA, 2006, pp.608–611.

[Co,03] :Radu Cornea, Nikil Dutt, Rajesh Gupta, Ingolf Krueger, Alex Nicolau,Doug Schmidt, Sandeep Shukla. *FORGE: “A framework for optimization of distributed embedded systems software*”. In *IPDPS*, avril 2003, pp.1-13.

[Co,06] : Radu Cornea Alex Nicolau Nikil Dutt: “*Software Annotations for Power Optimization on Mobile Devices*”, *Design, Automation, and Test in Europe*, proceedings of the conference on Design, automation and test in Europe: Proceedings Munich, Germany, 2006, pp.684 – 689.

[Cor,06] : R. Cornea, A. Nicolau, N. Dutt, and D. Bren: “*Annotation based multimedia streaming over wireless networks*”, inProc. *IEEE/ACM/IFIP Workshop on Embedded Systems for Real Time Multimedia*, 2006, pp. 47–52.

[Cnd,06] : Radu Cornea Alex Nicolau Nikil Dutt: “*Video Stream Annotations for Energy Trade-offs in Multimedia Applications*”, *IEEE/ACM Proceedings of the Proceedings of The*

---

Fifth International Symposium on Parallel and Distributed Computing, Washington, DC, USA 2006, pp.17-23.

## D

[Da,00]: J.-W. Dai, "The Scheduling to Achieve Optimized Performance of Randomly Addressed Polling Protocol," *Wireless Personal Communications*, vol. 15, pp. 161-179, 2000.

## F

[Fl,02] : K. Flautner, N. Kim, S. Martin, D. Blaauw, and T. Mudge. "Drowsy caches: Simple techniques for reducing leakage power". In Proc. 29th IEEE/ACM International Symposium on Computer Architecture, pp.147–157, Mai 2002.

## G

[Ge,05] : M. Geiger, S. McKee, and G. Tyson. "Beyond basic region caching: Specializing cache structures for high performance and energy conservation". In Proc. 1st International Conference on High Performance Embedded Architectures and Compilers, Nov. 2005.

## H

[Ha,06] : Girma Berhe HAGOS. "Accès et adaptation de contenus multimédia pour les systèmes pervasifs". Thèse de doctorat soutenue à l'Institut National des Sciences Appliquées de Lyon, le 25 septembre 2006.

[He,07] : J. Henkel and S. Parameswaran, "Designing Embedded Processors: A Low Power Perspective". Springer Netherlands, 2007.

[Ho,99] : I. Hong, D. Kirovski, G. Qu, M. Potkonjak, and M. B. Srivastava, "Power Optimization of Variable- Voltage Core-Based Systems", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, pp.1702-1714, December 1999.

## I

[In,99] : K. Inoue, T. Ishihara, and K. Murakami. "Way-predicting set-associative cache for high performance and low energy consumption". In Proc. IEEE/ACM International Symposium on Low Power Electronics and Design, pages 273–275, Aug. 1999.

**J**

[Je,91] : K. Jeffay, D. F. Stanat, and C. U. Martel, “*On non-preemptive scheduling of period and sporadic tasks*”, in Proceedings of Twelfth Real-Time Systems Symposium, 1991. , 1991, pp. 129-139.

[Ju,08] : H. Jung and M. Pedram, “*Improving the Efficiency of Power Management Techniques by Using Bayesian Classification*”, in 9th International Symposium on Quality Electronic Design, 2008. ISQED 2008. San Jose, CA, 2008.

**K**

[Ka,01] : S. Kaxiras, Z. Hu, and M. Martonosi, “*Cache decay: Exploiting generational behavior to reduce cache leakage power*”. In Proc. 28th IEEE/ACM International Symposium on Computer Architecture, pp.240–251, June 2001.

[Ka,08] :Zakia Imane Kazi-Aoul , “Une architecture orientée services pour la fourniture de documents multimédia composés adaptables”. L’École Nationale Supérieure des Télécommunications Au département Informatiques et Réseaux, 18 Janvier 2008.

[Ki,07] : Minyoung Kim, Mark-Oliver Stehr, Carolyn Talcott, Nikil Dutt, and Nalini Venkatasubramanian, “*A Probabilistic Formal Analysis Approach to Cross Layer Optimization in Distributed Embedded Systems*”. In IFIP FMOODS (2007), pp.285–300, 2007.

[Ki,97] : Kin J., Gupta M., Mangione-Smith W.H., “ *The filter cache: an energy efficient memory structure*”, 30th Annual International Symposium on Micro-architecture (Micro '97) Los Angeles, CA, USA. 1997, p. 184 – 193.

**L**

[La,08]: sébatien LABORIE: “*Adaptation sémantique de documents multimédia*” ; thèse de doctorat soutenue à l’université Joseph fourier - Grenoble en France, le 28 Mai 2008.

[La,05] : Lapayre J-C., Renard F., “*Appat : a New Platform to Perform Global Adaptation*”, Actes de la 1st IEEE International Conference on Distributed Frameworks for Multimedia Applications (DFMA'2005), Besançon, 6-9 février 2005, pages : 351-358.

[Le,04] : Tayeb LEMLOUMA. “*Architecture de Négociation et d'Adaptation de Services Multimédia dans des Environnements Hétérogènes*”. Thèse de doctorat soutenue à l'institut national polytechnique de GRENOBLE le 09 juin 2004.

[Li,03] : Rainer Lienhart, Igor Kozintsev, Yen-Kuang Chen, Matthew Holliman, Minerva Yeung, Andre Zaccarin, and Rohit Puri: “*Challenges in Distributed Video Management and Delivery*”. Handbook of Video Databases. CRC Press, Boca Raton, Florida, pages : 961-990, 2003.

[Li,07]: Chu-Hsing Lin, Jung-Chun Liu, Chun-Wei Liao, “*Energy Consumption Analysis of Audio Applications on Mobile Handheld Devices,*” TENCON 2007 – IEEE Region 10 Conference, Taipei, October 31, 2007, pp.124-125.

[Li,08]: Chu-Hsing Lin, Chih-Hsiong Shih, Jung-Chun Liu, Mao-Hua Cheng, Yan-Wei Lee: “*Energy Efficiency Measurement for Multimedia Audio Decoding on Embedded Systems*”, Conference On Ubiquitous Information Management And Communication – ACM New York, USA, 2008, pp.404-408.

[Li,10]: Chu-Hsing Lin, Jung-Chun Liu, Chun-Wei Liao: “*Energy analysis of multimedia video decoding on mobile handheld devices*”. In Computer Standards & Interfaces 32 – Elsevier, 2010, pp.10–17.

[Li,73]: Liu, C.L., Layland, J.W. “*Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment*”, Journal of the ACM, Vol. 20, No. 1, (January 1973), pp. 46-61.

## M

[Mo,05] : Shivajit Mohapatra, Radu Cornea, Hyunok Oh, Kyoungwoo Lee, Minyoung Kim, Nikil Dutt, Rajesh Gupta, Alex Nicolau, Sandeep Shukla, Nalini Venkatasubramanian, “*A Cross-Layer Approach for Power-Performance Optimization in Distributed Mobile Systems*”. In the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05), 2005.

## P

[Pa,03] : Sudeep Pasricha, Shivajit Mohapatra, Manev Luthra, Nikil Dutt & Nalini Venkatasubramanian, “*Reducing Backlight Power Consumption for Streaming Video Applications on Mobile Handheld Devices*”. 2003.

[Pe,06] : Trevor Pering, Yuvraj Agarwal, Rajesh Gupta, Roy Want, “*CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interface*”. In MobiSys'06, Uppsala, Sweden ACM, pp.220-232, June 19–22, 2006.

[Pi,01] : P. Pillai and K. G. Shin, "Real-time dynamic voltage scaling for low-power embedded operating systems,"in *Proceedings of the eighteenth ACM symposium on Operating systems principles*” Banff, Alberta, Canada: ACM, 2001.

[Pi,02] : Jayaprakash Pisharath, Alok N. Choudhary: “*An Integrated Approach to Reducing Power Dissipation in Memory Hierarchies*”. Proceedings of the 2002 International Conference on Compilers, Architecture, and Synthesis for Embedded Systems, Grenoble, France (2002) 88-97

## R

[Ro,99] : Franck Rousseau, J. Antonio Garcia-Macias, José Valdeni de Lima, Andrzej Duda: “*User adaptable multimedia presentations for the World Wide Web*”, 1999 Published by Elsevier Science B.V page 195 à 211.1999.

## S

[Sa,07] : H. Sasaki, Y. Ikeda, M. Kondo, and H. Nakamura, “*An intra-task dvfs technique based on statistical analysis of hardware events*”, in Proceedings of the 4th international conference on Computing frontiers Ischia, Italy: ACM, 2007.

[Sh,04] : Hojun Shim, Naehyuck Chang, Massoud Pedram, “*A Backlight Power Management Framework for Battery-Operated Multimedia Systems*”. In IEEE Design and Test of Computers, Special Issue on Embedded Systems for Real-Time Multimedia, 2004.

## T

[Ta,04]: Morihiko Tamai, Tao Sun, Keiichi Yasumoto, Naoki Shibata and Minoru Ito: “*Energy-aware Video Streaming with QoS Control for Portable Computing Devices*”.



ACM International Workshop on Network and Operating System Support for Digital Audio and Video Proceedings of the 14th international workshop on Network and operating systems support for digital audio and video, New York, USA, 2004, pp.68 – 73.

## W

[We,94] : M. Weiser, B. Welch, A. Demers, and S. Shenker, “*Scheduling for reduced CPU energy*”, Proceedings of USENIX Symposium on Operating Systems Design and Implementation”, pp. 13–23, 1994.

[We,04] : Yong Wei, Surendar Chandra Suchendra Bhandarkar. “*A statistical prediction-based scheme for energy-aware multimedia data streaming*”. In Wireless Communications and Networking Conference, (WCNC). 2004 IEEE, pp.2053–2057, 2004.

[We,06] : Yong Wei Surendar Chandra Suchendra Bhandarkar, “*A Statistical Prediction-based Scheme for Energy-aware Multimedia Data Streaming*”. 2006.

## X

## Y

[Yu,03] : Wanghong Yuan, Klara Nahrstedt, Sarita V. Adve, Douglas L. Jones, Robin H. Kravets , “*Design and Evaluation of a Cross-Layer Adaptation Framework for Mobile Multimedia Systems*”. In SPIE/ACM Multimedia Computing and Networking Conference (MMCN), 2003.

## Z

[Zh,05] : Fan Zhang, Samuel T. Chanson, “*Proxy-Assisted Scheduling for Energy-Efficient Multimedia Streaming over Wireless LAN*”. pp.980–991, 2005.

## 2- Pages Web

[AMBULANT]: Ambulant player 2.0.2 ; <http://www.ambulantplayer.org>

[CMML]: Continuous Media Markup Language <http://www.ietf.org/internet-drafts/draftpfeiffer-cmml-02.txt> 2004

[CCPP]: Composite Capabilities/Preference Profiles, <http://www.w3.org/Mobile/CCPP/>

- [CORBA]: Common Object Request Broker Architecture
- [FLASH]: <http://www.adobe.com/product/flashplayer>
- [GOLIVE]: Adobe GoLive; <http://www.adobe.com>
- [GRINS]: "GRiNS Editor", Oratrix. <http://www.oratrix.com/>
- [HTTP]: <http://www.w3.org/Protocols/>
- [IE]: Internet Explorer <http://go.microsoft.com/fwlink/?LinkId=54758>
- [JXTA]: <https://jxta.dev.java.net/>
- [LABVIEW]: National Instruments Corp. <http://www.ni.com>
- [LIMSEE]: Projet Opéra, INRIA. <http://limsee2.gforge.inria.fr>
- [MATHML]: <http://www.w3.org/TR/2003/REC-MathML2-20031021/>
- [MPEG]: Moving Picture Experts Group, <http://mpeg.chiariglione.org/>
- [MPEG21]: ISO/IEC JTC1/SC29/WG11, <http://mpeg.chiariglione.org/standards/mpeg-21/mpeg-21.htm>
- [MPEG-7]: MPEG-7 overview, <http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm>
- [POKETSMAIL]: <http://opera.inrialpes.fr/pocketsmil/>.
- [RDF]: Resource Description Framework, <http://www.w3.org/RDF/>
- [REALONE]: Real Networks. RealOne player, <http://www.realnetworks.com>
- [RPC]: <http://www.xmlrpc.com/>
- [Rubic] : <http://www.rubicdesign.com/>
- [SATO]: <http://www.ambientnetworks.org/Files/deliverables/>
- [SMIL]: Synchronized Multimedia Integration Language, <http://www.w3.org/AudioVideo/>
- [SMOXPAD]: SMOX Pad 1.2.1.2 (Studio Multimédia Orienté XML, Personnel Assistant Developpement) <http://www.manalee.com>
- [WAM]: <http://wam.inrialpes.fr/>
- [WI,10]: <http://fr.wikipedia.org>
- [XFORMS]: <http://www.w3.org/TR/xforms11/>
- [XML]: eXtensible Markup Language, <http://www.w3.org/XML/>
- [XPath]: <http://www.w3.org/TR/1999/REC-xpath-19991116>
- [XSLT]: James Clark : XSL Transformations (XSLT). W3C, <http://www.w3.org/TR/xslt>, 1999.