Escola Politècnica Superior de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER THESIS

TITLE: Multimedia Quality Assessment

MASTER DEGREE: Master in Science in Telecommunication Engineering & Management

AUTHOR: Oriol Solé Molina

DIRECTOR: Antoni Oller Arcas

CO-DIRECTOR: Alberto José González Cela

DATE: November 2nd 2009

Títol: Multimedia Quality Assessment

Autor: Oriol Solé Molina

Director: Antoni Oller Arcas

Co-director: Alberto José González Cela

Data: 2 de novembre de 2009

Resum

Aquest projecte presenta 3 contribucions que permeten decidir quin és el millor códec multimèdia segons l'entorn en que és utilitzat: xarxa i dispositiu. Per dur a terme aquesta decisió es defineix que un códec de media és millor que un altre si aquest presenta la millor relació qualitat-relació de compressió.
Per dur a terme aquesta relació és necessari conèixer com realitzar l'anàlisi de qualitats multimèdia.

En aquest document es descriuen les diferents tècniques per realitzar l'anàlisi de qualitats multimèdia. Tenint com a base aquest estudi s'ha realitzat el disseny d'un mòdul analitzador de qualitats multimèdia que realitza les mesures de qualitat de diferents tipus de media: imatge, vídeo i àudio.

La implementació d'aquest mòdul d'anàlisis multimèdia permet la realització de proves sobre el comportament dels códecs en escenaris de transmissió multimèdia amb pèrdua de dades. Els resultats dels tests i les conclusions són una de les principals contribucions a entendre la importància d'utilitzar un códec específic depenent del medi en el que serà utilitzat.

Finalment s'explica com l'execució d'aquest projecte permet a un projecte d'investigació en desenvolupament realitzar la seva principal funció: decidir la millor adaptació de continguts multimèdia segons la xarxa, les capacitats del terminal i les preferències d'usuari.

Title: Multimedia Quality Assessment

Author: Oriol Solé Molina

Director: Antoni Oller Arcas

Co-director: Alberto José González Cela

Date: November, 2nd 2009

Overview

This thesis presents three main contributions that helps to decide which the best multimedia codec is depending on the environment it will be used: network and device. To carry out this decision it is defined that a media codec is better than other if it presents the best perceptual quality and compression ratio relation. To perform this relation it is needed to know how to carry out the multimedia quality assessment.

In this document it is described the different techniques to perform multimedia quality assessment. From this study it has been designed a multimedia quality analyzer module that performs the multimedia quality assessment of different media types: image, video and audio.

The implementation of this multimedia analyzer module allows performing some test on the behaviour of multimedia codecs in multimedia streaming packet loss scenarios. The test results and conclusions are one of the main contributions to understand the importance of using a specific media codec depending on the environment it is going to be used.

Finally it is explained how the execution of this thesis allows a current multimedia project performing its main functionality: perform the best content adaptation decision depending on the network, device capabilities and user preferences.

INDEX

INTRODUCTION

With the broadband network consolidation (cable, ADSL2+, FTH, etc.) the IP services of multimedia contents were born. Services like Imagenio (Telefónica), OrangeTV (Orange) and JazzteliaTV (Jazztel) offer multimedia contents using the ADSL2+ technology. These services are well known as IPTV services. But these services are linked to the Public Switched Telephone Network (PSTN). This fact does not facilitate the user mobility.

Allowing a user consuming multimedia resources wherever and whenever he or she wanted, what is known as Universal Multimedia Access (UMA), is one of the added values of these services.

In this sense the i3media project was born. I3media is a Cénit project where pioneering audiovisual sector enterprises collaborate with it. Enterprises like MediaPro, Corporació Catalana de Ràdio i Televisió, Ativa Multimèdia, Telefònica, Havas media and others collaborate with the project.

The idea which i3media is based on is that to be able to compete in the multimedia market it must be provided the best technologies in the production, management, distribution and operation of the contents.
One of the i3media project activities is to develop technologies to automatically generate interactive and autoadaptable multimedia contents.
In this context it has been designed a multimedia content adaptation service.

The Content Adaptation Service is a service that allows adapting a multimedia content to the user’s device and network capabilities as well as user preferences and physical characteristics.

The problem

One of the main goals of the Adaptation Service is allowing the user mobility and the user context change. As a context change is understood a network and device change where the user is consuming a media resource through them.
This adaptation capacity is done through an Adaptation Module. This module contains a Logic Server that equips it of artificial intelligence. This Logic Server allows discerning, specifying the user context, which is the best content adaptation.

The main functionality of the Adaptation Module is to carry out a multimedia format recommendation. This format recommendation is a MPEG-21 Digital Item where multimedia format parameters are specified: codec, resolution, bits per sample, frame rate, sampling frequency, etc.
One of the most complex points is to determine which the best codec in a specific context is. This is the main goal of this thesis, to carry out multimedia quality assessment in order to determine which the best codec in a specific context is.

The Content Adaptation Service is growing to carry out context awareness: location, mobility and personalization. These functionalities will perform an improvement of the end user perceptual quality.

Thesis objectives

At the end of this thesis the i3media Adaptation Module will be able to decide which the best codec to be used in a specific context is. This is the mail goal. But in order to reach the mail goal it has been defined different objectives to be accomplished:

- To carry out a study about multimedia quality assessment techniques.
- Design and implement an application that carries out the quality assessment using the studied techniques.
- Study the quality behaviour of multimedia codecs if the coded resource is subject to data losses.
- Design and implement a multimedia analysis module to be integrated into the Content Adaptation Service.

In this document it is described the steps done to accomplish the thesis objectives and the results obtained from each stage.
The first step done is to know the environment where the problem is. This is what is described in CHAPTER 1. In CHAPTER 1 the i3media Content Adaptation Service architecture is described and it is focused on the main service component: the Adaptation Module. Understanding how the Adaptation Module works the problem, which this thesis works on solving, is understood.

Once the problem is known, in CHAPTER 2 it is described the different quality metrics used to perform multimedia quality assessment.

In CHAPTER 3 is described the design of an analyzer module to carry out the evaluation of which is the best media codec. This evaluation is done based on a mathematical parameter definition also described in this chapter.

To take advance of the implemented analyzer module, in CHAPTER 4 is described a study made on what is the media codecs resilience in a data loss environment network. The reason of this study is to see how affects the end user perceptual quality when a coded resource is subject to data losses.

As it has been mentioned the final thesis objective is to implement a multimedia analysis module to be integrated to the Content Adaptation Service. This integration will allow reaching the main thesis goal to allow the Adaptation Module to decide which the best codec is to be used in a specific context. The multimedia analysis module design and the integration process are described in CHAPTER 5.

Finally in CHAPTER 6 can be found some conclusions extracted from the overall work done and from the obtained results got in the final work.

CHAPTER 1. SERVICE ARCHITECTURE

This thesis is within the framework of the i3media project. In order to understand why is there a need to a multimedia quality assessment, in this chapter is explained the main ideas of the i3media project. And it is described the i3media Content Adaptation Service architecture where the quality assessment is required.

1.1 i3media project

i3media[1] is a Cénit project from the Ministerio de Ciéncia y Tecnología. I3media is based on the idea that to compete in the multimedia market it should has at one’s disposal the best technologies in production, management, distribution and content distribution.

i3media is divided in working groups. Each working group is responsible to develop a specific project area. In this thesis context, the working group is responsible in the development in automatic and autoadaptative interactive multimedia content creation.

One of the tasks of this working group is the automatic content adaptation due to user context (device, network, environment, etc.). This task is known as Content Adaptation Service.

What is a content adaptation?

A content adaptation is a process which alters the original resource characteristics in order to be able to obtain this resource whatever the environment is.

1.1.1 Content adaptation case of use

Margaret is coming back home after a hard working day. She is going by train and she is watching her favourite TV series in her mobile device. The train journey ends but her favourite series not. So, once she arrives at home, she decides to continue watching the series, from the scene she left, in her panoramic TV station.

1.2 Content Adaptation Service

The Content Adaptation Service is a service which main goal is to recommend a content adaptation to any user’s device through any network. This service provides to users a Universal Multimedia Access (UMA): consuming multimedia resources wherever he or she was and whenever he or she wanted.

The Content Adaptation Service is provided by a group of elements that provide different functionalities. In **Figure 1.1** it is shown the Content Adaptation Service architecture, where it can be seen the different elements that conforms the service.

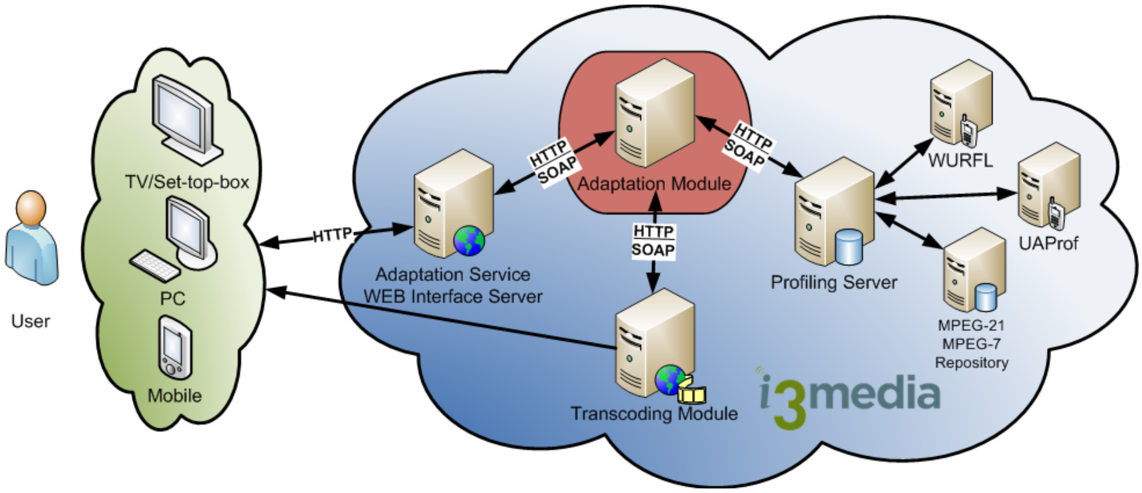


Figure 1.1 Content Adaptation Service architecture

As it can be seen in **Figure 1.1**, the Content Adaptation Service is performed by different components:

- The **Adaptation Module** is the core engine. It is the key component. It performs the best adaptation decision taking into account the user's device and the network and content characteristics.

- The **Transcoding Module** is the element which allows users to obtain the adapted content.

- The **Adaptation Service WEB Interface Server** contains the Adaptation Service WEB Interface. This interface allows users to access to the Content Adaptation Service and an administrator to realize management tasks.

- The **Profiling Server** contains a XML Database which contains the multimedia content and user environment (device, network and preferences) descriptors. These descriptors are coded in XML following the MPEG-21[2] and MPEG-7[3] standards. The Profiling Server functionality is to provide information to the Adaptation Module related to the content and the user environment (device, network and preferences)

Each element has been designed to be deployed as a web service [20]. This provides a distributed architecture that allows the decentralization of each service.

As each element is deployed as a web service, the communication between them is made over HTTP/SOAP [21].

As it has been mentioned, this thesis is focused on the Adaptation Module specifications. This is because this module needs some functionality that allows it to decide the best content adaptation. And one of the most important functionalities is studied, designed and implemented in this thesis. For this reason it is described deeply the Adaptation Module specifications.

1.2.1 Adaptation Module

The Adaptation Module is the main component of the Content Adaptation Service. It is the core engine which determines the best content adaptation taking into account the user's device, network and environment.

This module is composed by four components, as it can be seen in **Figure 1.2**:

- **Management Module**: it offers the capacity to control the adaptation state, manage the adaptation sessions and manage statistics information from the Content Adaptation Service use.
- **Information Module**: it is the component which manages the content, users and user context information. It communicates with the Profiling Server in order to get and/or to set this information.
- **Transcoding Management Module** manages the communication with the Transcoding Module.
- **Adaptor Module** is the component which makes the best content adaptation decision.

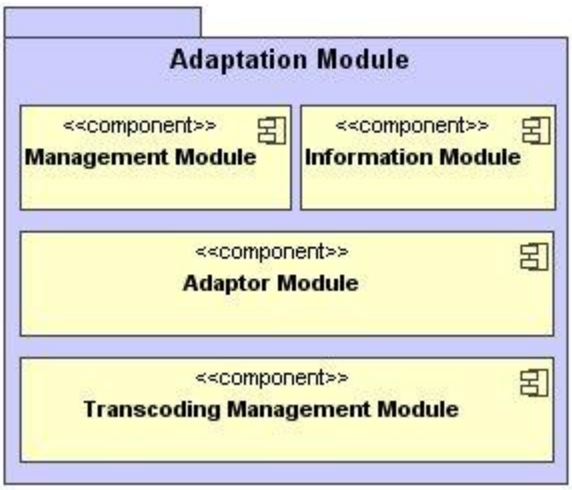


Figure 1.2 Adaptation Module component diagram

From these four components the most important one is the Adaptor Module. As it has been mentioned is the responsible for the best content adaptation recommendation.

In order to make the best content adaptation decision, the Adaptor Module needs the specific information extracted from three different contexts:

- From the network it is needed the network capacity to determine the best content bitrate adaptation.
- From the device it is needed its capabilities to determine the best adaptation of the media resource characteristics: resolution, supported codecs, number of audio channels, etc.

- User preferences to perform the best user adaptation.

In **Table 1.1** are listed the main parameters needed to perform the content adaptation recommendation extracted from the network and user device.

Table 1.1 User environment parameters needed to perform the content adaptation decision.

Device	**Network**
height	capacity
width	packet loss rate
media formats	jitter
video frame rate	
video codecs	
audio channels	
audio bits/sample	
audio frame rate	
audio codecs	
image formats	
image colour depth	

All these parameters are taken into account in the content adaptation process.

Parameters extracted from the network characteristics are used for different purposes:

- Capacity: determines the maximum media bitrate of the adapted content.
- Packet loss rate: used to emulate the user network in order to simulate and analyse a content adaptation profile.
- Jitter: used to determine the buffer size

Parameters extracted from the device capabilities are used for:

- Height and width: defines the maximum device screen resolution, and determines the maximum video content adaptation resolution.

- Video frame rate, audio channels, audio bits/sample, image colour depth: defines the characteristics that may accomplish the adapted content in order to be supported by the device.
- Media formats, video codecs, audio codecs and image formats: determine the content formats and codecs that the device supports.

It is easy to suppose that the best content adaptation for a specified device is the one that determines all these specifications. But, it is not as easy to determine the best content format and codec to be used. The previous specification list determines discrete values, but the content format and codec list does not determine a specific value, it determines different content formats and codecs. To choose one between them is not a trivial task.

So, it arise a need to make a multimedia formats study and objective analysis in order to decide which is the best media format and codec in a specific user context.

It is proposed to design a multimedia metric analyzer that carries out automatic objective multimedia quality assessment. With the analysis results a first approximation to decide which is the best content format and codec to be used will appear.
So, it is proposed to add to the Adaptor Module a multimedia metric analyzer to help it to decide the best content adaptation format.

CHAPTER 2. MULTIMEDIA QUALITY METRICS

As it has been mentioned in CHAPTER 1, to determine which the best media codec is for a specific context it is needed to make a media codec analysis. In this way it has been made a multimedia quality metrics study in order to determine which the best way is to carry out the multimedia quality assessment.

Following the ITU-T P.800 Recommendation [4], the best way to evaluate the perceptual quality of a multimedia content (image, audio or video) is to ask to human observers, as multimedia content is made by and for humans. But to obtain a representative Mean Opinion Score (MOS) is needed a great number of observers. So, for an industrial scenario it is not efficient due to it is time consuming and the observations must be done in normalized conditions.
As this assessment cannot be used it has been done a research on how to automatically extract this subjective quality perception without using any human observer. In this sense the multimedia quality metrics are specified. This is what is described in this chapter.

2.1 Multimedia quality metrics classification

Multimedia quality metrics can be classified into three groups taking into account in which way is needed a reference signal to extract the parameter value. Each of them is used depending on the availability of an undistorted signal (reference signal) or some characteristics of it.

2.1.1 Full Reference metrics (FR)

Full Reference metrics are those metrics which compare two entire signals: a reference signal, normally the original one, and a compared signal, normally the coded one.
Each signal is compared with each other. Higher the difference, higher the error in the compared signal.

This method is the most used one due to its low complexity. But it is needed the availability to obtain both signals: the original signal and the codec signal.

2.1.2 Reduced Reference metrics (RR)

Metrics based on a Reduced Reference are those metrics which only compare some signal characteristics. For instance, in multimedia domain these characteristics could be: blocking, blur, ringing, masking, etc.
These characteristics are detected in the reference and compared signal. Once detected are compared between them.

2.1.3 Non Reference metrics (NR)

Non Reference metrics, as its name indicates, are those metrics which do not use any reference signal to determine the quality of a signal.

These metrics tries to detect, using heuristic filters, effects like, in multimedia cases, blocking and blur. Depending on the power of these effects it is determined the quality of the signal.

For its low complexity and easier implementation it has been considered to study the Full Reference metrics. It is a good way to start analyzing media and extracting a first knowledge on this topic.

So, next are listed the Full Reference multimedia quality metrics studied in order to make a multimedia codec quality study.

2.2 MSE/PSNR

Any signal can be seen as the sum of a non distorted original signal and an error signal. So, the perceptual quality loss is related to the error signal perception.

How can be extracted the error signal from a multimedia content?
This is what the MSE or Mean Squared Error does.

In the Mean Squared Error calculation the error power signal is obtained by comparing two signals and extracting the differences between them. These differences are considered the error signal.

Next equation (**2.1**) represents the MSE calculation for a two dimensional signal, for example, an image.

$$MSE = \frac{1}{m*n}\sum_{i=1}^{m}\sum_{j=1}^{n}[x(i,j) - y(i,j)]^2 \qquad (\mathbf{2.1})$$

In (**2.1**), m and n corresponds to the height and width in the case of an image analysis.

But just knowing the error power signal from a multimedia content is not enough. A reference is needed to decide if this signal error is high or low.

The reference is the original power signal and the metric that relates both powers is the PSNR.

Peak-to-Signal Noise Ratio is the relation between the maximum power value of the original signal and the noise power signal (**2.2**).

$$PSNR = 10 * log_{10}\left(\frac{{MAX_x}^2}{MSE}\right) \quad (2.2)$$

As it is a relation between two powers the PSNR unit is decibels (dB).

For two identical signals the MSE value is equal to zero due to there is no differences between them, or what is the same, there is no error signal added to the analyzed signal. This implies that the PSNR value is infinite.
For two different signals the MSE trends to the maximum signal power value, so the PSNR trends to zero.

In conclusion, the lower the difference the higher the PSNR value, and the higher the difference the lower the PSNR value.

In order to reduce the computational complexity in image analysis it is recommended, in [15], to compute the PSNR in the Y image component or luma component. This is because a raw colour image is composed by three colour components: R, G and B. For this reason, it is mandatory computing the three components PSNR in order to obtain the overall image PSNR. This fact implies a higher computational load and, for this reason, it is recommended to compute image PSNR only on the Y or luma component. This specific PSNR is called Y-PSNR and its mathematical expression is shown in (**2.3**) as well as the Ymse (the Y component Mean Square Error) expression in (**2.4**).

$$YPSNR = 10 * log_{10}\left(\frac{{MAX_x}^2}{Ymse}\right) \quad (2.3)$$

$$YMSE = \frac{1}{m * n}\sum_{i=1}^{m}\sum_{j=1}^{n}\left[Y_x(i,j) - Y_y(i,j)\right]^2 \quad (2.4)$$

The Y or Luma image component corresponds to the image contrasts map and is on what the human eye is most sensitive, see [5]. For this reason it is considered this component in the PSNR computational reduction consideration.

This is the simplest and most used full-reference quality metric. And it can be used to analyze image and video either audio.
But it is not well matched with perceived visual quality [6]. There is no good correlation between the results and the Mean Opinion Score (MOS)

2.3 SSIM

Some studies ([7] and [8]) have concluded that the human visual system (HVS) is highly adapted to extract structural information from the viewing field. Taking into account this conclusion it has been thought that measuring a change in the structural information image can provide a good approximation on the perceived image distortion.

In this investigating line Zhou Wang [9] presented the Structural Similarity Index (SSIM). This metric tries to extract the structural information on an image in order to analyze the structural changes to get the perceived image distortion. He defines the structural information in an image as "those attributes that represent the structure of objects in the scene, independent of the average luminance and contrast."

The diagram of the SSIM measurement system is shown in **Figure 2.1**.

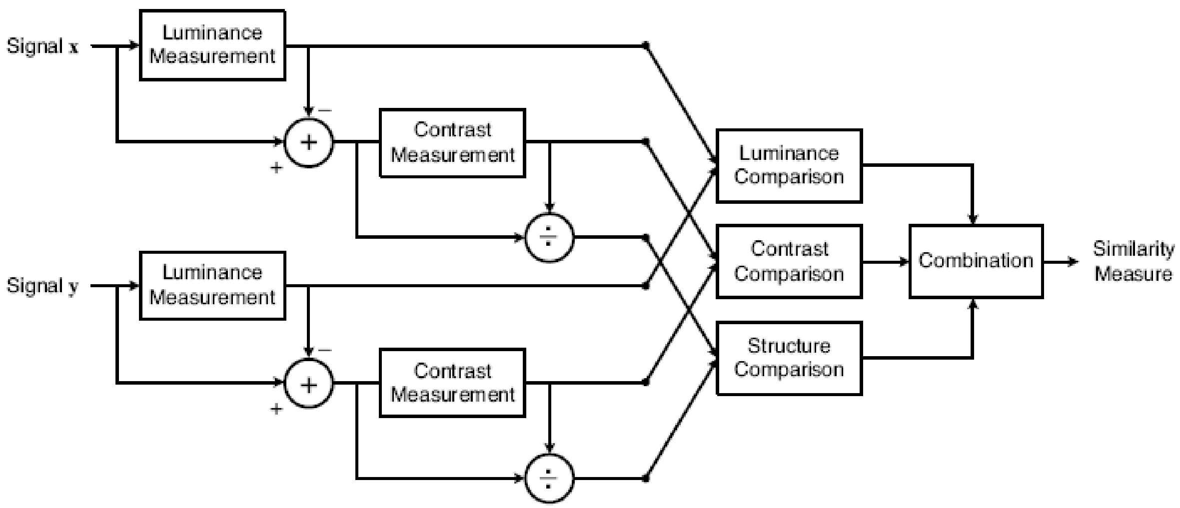


Figure 2.1 SSIM measurement system diagram

The system separates the tasks of analysis into three comparisons functions: luminance, contrast and structure.

The system takes into account a discrete signal and works like follows:

First, it is measured the luminance which is estimated as the mean intensity (**2.5**)

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{2.5}$$

Where, N is the number of pixels of the analyzed block.

So, the luminance comparison function *l(x,y)* is a function of μ_x and μ_y (**2.6**)

$$l(x,y) = \frac{2\mu_x\mu_y + C1}{\mu_x^2 + \mu_y^2 + C1} \quad (2.6)$$

$$C1 = (K_1L)^2 \quad (2.7)$$

Second, it is extracted the luminance from the image and measured the contrast. The contrast is estimated as the standard deviation (**2.8**)

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2} \quad (2.8)$$

So, the contrast comparison function *c(x,y)* is a function of σ_x and σ_y (**2.9**)

$$c(x,y) = \frac{2\sigma_x\sigma_y + C2}{\sigma_x^2 + \sigma_y^2 + C2} \quad (2.9)$$

$$C2 = (K_2L)^2 \quad (2.10)$$

Third, the structure is extracted. The structure is associated with the correlation between x and y. (**2.11**)

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) \quad (2.11)$$

So, the structure comparison function s(x, y) is as follows (**2.12**)

$$s(x,y) = \frac{\sigma_{xy} + C3}{\sigma_x\sigma_y + C3} \quad (2.12)$$

$$C3 = (K_3L)^2 \quad (2.13)$$

In (**1.7**), (**1.10**) and (**1.13**) K_1, K_2, K_3 << 1 are small constants, and L is the dynamic range of the pixel values (255 for 8-bit). C1, C2, and C3 are constants introduced to avoid unstable results when $(\mu_x^2 + \mu_y^2)$ or $(\sigma_x^2 + \sigma_y^2)$ are close to zero. In [10] is defined K_1 = 0.01 and K_2 = 0.03. They are arbitrary but is indicated that empirically the SSIM performance is fairly insensitive to these values. C_3 is also defined as $C_3=C_2/2$.

The result of combine (**2.6**), (**2.9**) and (**2.12**) is the SSIM index (**2.14**)

$$SSIM(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma} \qquad (2.14)$$

In (**2.14**) $\alpha > 0$, $\beta > 0$ and $\gamma > 0$ are parameters used to adjust the relevance of each component.

The SSIM function (**2.14**) has the next properties:

- Symmetry: SSIM(x,y) = SSIM(y,x)
- Bounded: 0 ≤ SSIM(x,y) ≤ 1
- Unique maximum: SSIM(x,y) = 1 if and only if x = y

In [10] it is analysed the SSIM index using an 11x11 circular-symmetric Gaussian weighting function with standard deviation of 1.5 samples. This means, that the image is analyzed in blocks of 11x11 pixels, assigning the corresponding weight in each pixel, according to the Gaussian window. In this case, the N parameter in (**2.5**), (**2.8**) and (**2.11**) has a value of 121, as the 11x11 window has a total of 121 pixels inside.
It is indicated that the SSIM analysis has to be made moving this Gaussian window pixel by pixel around the image. So, the final SSIM index value is the mean value of each block SSIM value as is indicated on (**2.15**)

$$mean\ SSIM(X,Y) = \frac{1}{M}\sum_{j=1}^{M} SSIM(x_j, y_j) \qquad (2.15)$$

Where M is the number of analyzed blocks and j is the analyzed block index.

As happens in PSNR, to reduce the computational complexity it is recommended to compute the SSIM on the Y (luma) component, as human visual system (HVS) is most sensitive to this component, [5].

2.4 SSIM Audio

The SSIM index was originally defined as an image/video quality metric. But in [12] it is analyzed in audio signals using two techniques and it is concluded that these techniques have good correlation to subjective perception.

As it has been mentioned SSIM index is based on the idea that the measure of structural information changes is a good approximation to a quality perception change. But how can be understood structure in audio?
In this mentioned study ([12]) structure in audio is viewed in two ways: time domain and frequency domain.

By complexity it is considered the time domain description as its results are similar to frequency domain results.

So, in time domain the structure in audio is defined as the dependence of each time sample with its position with respect to a small temporal neighbourhood of samples around it. This is the same as happens in image structure analysis. The image structure is defined as the dependences between image samples (pixels) that are spatially proximate.

The audio time domain SSIM technique follows the next steps:

1. The samples are split into frames of length 128 with 50% overlap.
2. It is applied the SSIM to each frame
3. The final result is the mean SSIM value of each frame.

A 50% overlap means that the analyzer window is moved each 64 samples, like is indicated in section 2.3, where a Gaussian analyzer window is moved pixel by pixel.

In this way the temporal structure of the audio is analyzed.

2.5 PEAQ

Perceptual Evaluation of Audio Quality is a standardized algorithm to measure the perceived audio quality in an objective way. It is defined in the ITU-R BS.1387 [13].

The algorithm works like follows:
The system inputs are the audio reference and the audio signal to be tested.
Then the algorithm normalizes both signal levels and maps them in the time-frequency domain. Next, each signal is analyzed according to perceptual models, based on auditory critical bands (hearing threshold, hearing excitation, sharpness, masking …). From these analyses it is got a list of parameters called Model Output Variables (MOVs). (See **Table 2.1**)

Table 2.1 Description of the Model Ouput Variables

Model Output Variable	Description
WinModDiffB	Windowed averaged difference in modulation (envelopes) between Reference Signal and Signal Under Test
AvgModDiff1B	Averaged modulation difference
AvgModDiff2B	Averaged modulation difference with emphasis on introduced modulations and modulation changes where the reference contains little or no modulations
RmsModDiffA	Rms value of the modulation difference
RmsMissingComponentsA	Rms value of the noise loudness of missing frequency components, (used in RmsNoiseLoudAsymA)
RmsNoiseLoudB	Rms value of the averaged noise loudness with emphasis on introduced components
RmsNoiseLoudAsymA	RmsNoiseLoudA + 0.5RmsMissingComponentsA
AvgLinDistA	A measure for the average linear distortions
BandwidthRefB	Bandwidth of the Reference Signal
BandwidthTestB	Bandwidth of the output signal of the device under test
TotNMRB	logarithm of the averaged Total Noise to Mask Ratio
RelDistFramesB	Relative fraction of frames for which at least one frequency band contains a significant noise component
AvgSegmNMRB	the Segmentally Averaged logarithm of the Noise to Mask Ratio
MFPDB	Maximum of the Probability of Detection after low pass filtering
ADBB	Average Distorted Block (=Frame), taken as the logarithm of the ratio of the total distortion to the total number of severely distorted frames
EHSB	Harmonic structure of the error over time

From these parameters is determined the Objective Difference Grade (ODG) which is the objective measured parameter that corresponds to the subjective quality perceived. This parameter values goes from 0 (imperceptible) to -4 (very annoying).

In **Figure 2.2** is represented the block diagram of the PEAQ algorithm. There, it can be observed the different stages.

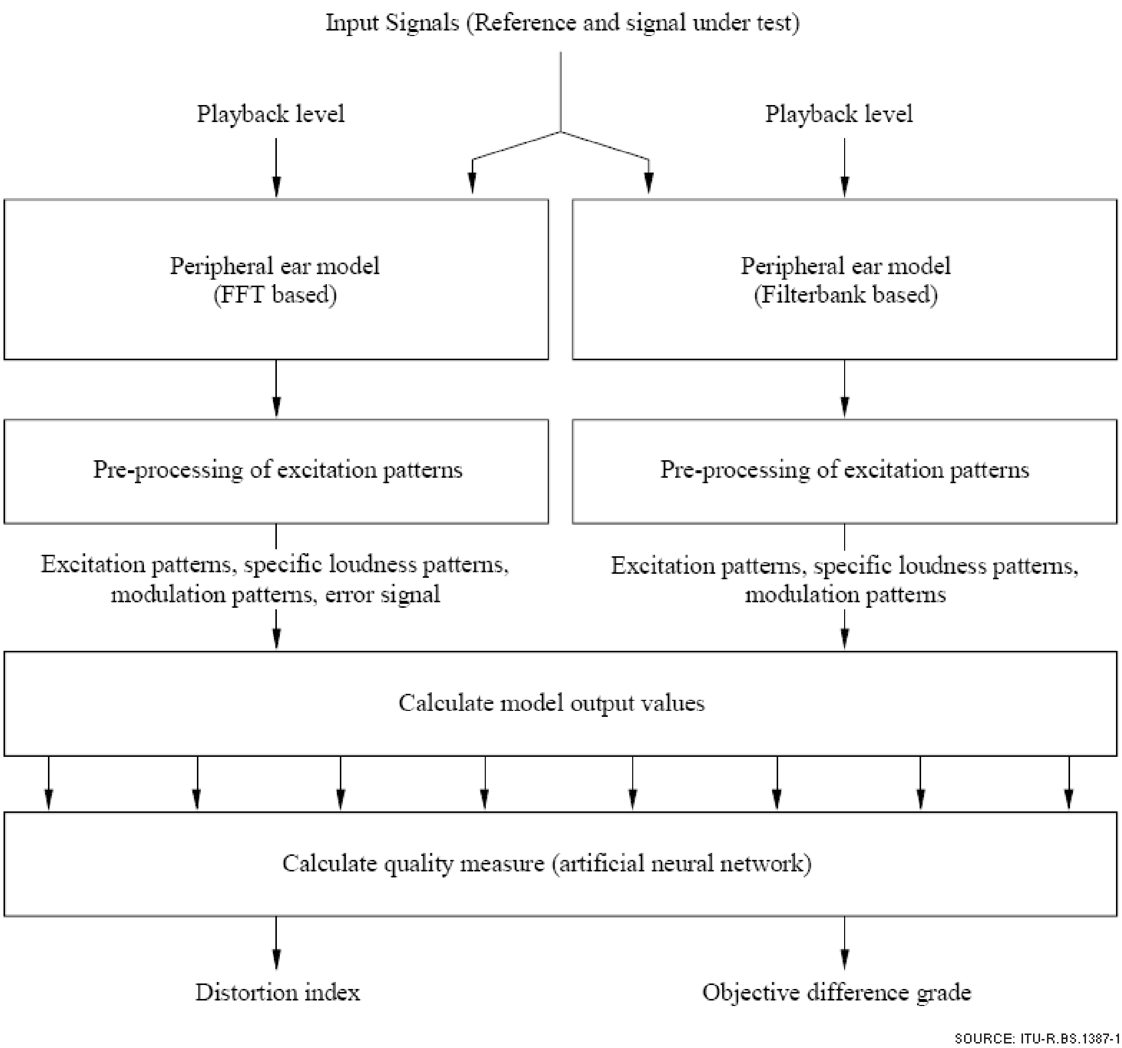


Figure 2.2 Block diagram of the PEAQ algorithm

The system also outputs another quality index, the Distorsion Index (DI). It is also a quality indicator like ODG but its difference is that it indicates the perception of very low signal qualities. Its values go, like ODG, from 0 (imperceptible) to -4 (very annoying).

CHAPTER 3. MULTIMEDIA QUALITY METRICS ANALYZER

In CHAPTER 1 is explained that there is a need in i3media project to introduce multimedia quality assessment in order to decide which the best media format in an adaptation process is.

Then, in CHAPTER 2 it is exposed a multimedia quality metrics study. This study it has been done in order to know what the best multimedia quality metrics are to be used in an automatic perceptual quality assessment.

From the multimedia quality assessment need and the multimedia quality metrics study, it has been design and developed a multimedia quality metrics analyzer module. The goal of this development is to determine the best multimedia formats taking into account the objective perceptual quality assessment and the codec compression ratio.

3.1 Multimedia quality metrics analyzer functionalities

The main goal of the multimedia quality metrics analyzer module is to analyze multimedia content using known multimedia quality metrics and get the perceptual quality as well as the compression ratio of different kind of multimedia coders for image, video and audio.

In order to manage the results, it has been specified some management functionalities.

Next are listed the main functionalities that the module supports:

- Given a media resource (image, video or audio) in RAW format and the same media resource coded and decoded in RAW format, obtain the media resource specific quality metric parameters.
- Analyze the next quality metrics for the different media types:
    - Image:
        - PSNR
        - SSIM
    - Video:
        - PSNR
        - SSIM
    - Audio:
        - PSNR
        - SSIM
        - PEAQ
- Given a media resource (image, video or audio) in RAW format scaled according to a coding parameters and the same media resource coded according the same coding parameters, obtain the compression ratio of a specific codec for a media resource coded with the specified codec.

- Manage the analyzed resource and codec lists.
- Determine which the best codec is due to a score parameter.

3.2 Score parameter definition

The use of coders allows the final compression of the resource size (in bits). But, intrinsically, it also reduces the user quality perception. So, it must be found a trade-off of the compression ratio and the perceptual quality.

A way to decide whether a codec is better than other is considering the perceptual quality and the compression ratio of a coded media resource. Thus, it can be said that a codec is better than other if this presents a better perceptual quality and compression ratio relation.

In this way it has been thought in the score parameter. This parameter indicates the perceptual quality and compression ratio relation of a coded media resource. So, a coded resource, coded with a specific codec, that presents a better score than the same resource, coded with other codec, is considered, for this specific resource, that the first used codec is better than the second one.

The mathematical expression that defines the score parameter is shown in (**3.1**)

$$score = A * perceptual\ quality + (1 - A) * compression\ ratio \qquad (\mathbf{3.1})$$

$$where\colon 0 \leq A \leq 1$$

The variable *A* determines the relevance of each parameter. This variable allows specifying the weight of each parameter on the score. This way, the relevance of each parameter can be changed. For instance, if it is wanted to consider that a codec is better than other giving priority to perceptual quality the A variable value must be increased.

The score that specifies an equitable relation between perceptual quality and compression ratio is those with an A variable equals to 0.5.

The score parameter is defined in the ℝeal set and can take values comprehended between -1 to 1 (**3.2**)

$$score \in \mathbb{R}, \qquad -1 \leq score \leq 1 \qquad (\mathbf{3.2})$$

- -1 indicates de worst perceptual quality and compression ratio relation.
- 1 indicates the best perceptual quality and compression ratio relation.

The compression ratio parameter is defined in the ℝeal set and it can also take values between -1 to 1 (**3.3**)

$$compression\ ratio \in \mathbb{R}, \qquad -1 \leq compression\ ratio \leq 1 \qquad (\mathbf{3.3})$$

- -1 indicates that there is no compression between the original and coded resources, but it has been an increment in the total number of bits.
- 1 indicates a 100% reduction of the total number of bits.

This way, a codec that increments the number of bits in the coded resource is penalized against those codecs that compress the original resource. Thus, two codecs that presents the same perceptual quality but one of them increments the coded resource total number of bits, it will get a lower score value.

The compression ratio parameter mathematical expression is defined in (**3.4**)

$$comp.ratio = \frac{original\ resource\ num.of\ bits - coded\ resource\ num.of\ bits}{original\ resource\ number\ of\ bits} \qquad (\mathbf{3.4})$$

The perceptual quality parameter can take values comprehended between 0 and 1 and it is also defined in the $\mathbb{R}$eal set (**3.5**)

$$quality \in \mathbb{R}, \qquad 0 \leq quality \leq 1 \qquad (\mathbf{3.5})$$

- 0 indicates, in perceptual quality terms defined by ITU-T in [4], very annoying perceptual quality.
- 1 indicates no perceptual differences between the original and coded media resource.

It can be understood as 0 indicates a 0% of similarity and 1 a 100% of similarity. As a perceptual quality parameter can be taken whatever studied quality metrics seen in CHAPTER 2: PSNR, SSIM and PEAQ.

Some of these quality metrics do not take values between the defined quality ranges. So, they must be normalized.
The quality metrics to be normalized are:

- PSNR which its values range is 0 ≤ PSNR ≤ ∞
- PEAQ which its values range is -4 ≤ PEAQ ≤ 0

It is not necessary to normalize SSIM quality metric as its values range is fitted into the perceptual quality parameter range.

3.2.1 PSNR normalization

As it has been mentioned before, the PSNR quality metric can take values comprehended between 0 and ∞. These values do not adjust to the quality parameter definition, so it must be normalized.

PSNR can take ∞ value. This fact complicates the parameter normalization. So, it must be taken into account some considerations for each kind of media, because there are some differences in the kind of samples between image and video resources and audio media resources.

3.2.1.1 PSNR normalization in image and video media resources

In section 2.2 is explained what PSNR is, and it is also explained that in image analysis is recommended to compute the PSNR analysis in the Y image component (Y-PSNR).

To compute Y-PSNR normalization, the Y-PSNR maximum value under ∞ must be known.

From (**1.3**) it is deduced that Y-PSNR is maximum when Ymse trends to 0 (**3.6**)

$$\lim_{Y_{mse}\to 0} YPSNR = \lim_{Y_{mse}\to 0} 10 * \log_{10}\left(\frac{255^2}{Y_{mse}}\right) = \infty \qquad (\mathbf{3.6})$$

So, the maximum Y-PSNR value just under ∞ is the minimum Ymse ($Ymse_{min}$) just above 0. This minimum value appears when there is a minimum difference in an image. This happens when there is a minimum difference in the Y component (in this particular case).

In [15] is defined the mathematical expression that matches the RGB image components with the Y or luma component. (**3.7**)

$$Y = 0.299 * R + 0.587 * G + 0.114 * B \qquad (\mathbf{3.7})$$

In (**3.7**) can be seen that B component is the least relevant component. So, a minimum difference in the B image pixel component gives the minimum difference in the Y pixel component.

The minimum variation in Y pixel component (ΔY_{min}) when there is a difference in the B component of a pixel is expressed in (**3.8**)

$$\Delta Y_{min} = Y_1 - Y_2 = 0.114 * (B_1 - B_2) = 0.114 * \Delta B_{min} \qquad (\mathbf{3.8})$$

B is a pixel component that increments its value in 1 integer. So, the minimum difference between two B pixel components is 1 (**3.9**)

$$\Delta B_{min} = 1 \qquad (\mathbf{3.9})$$

From (**3.8**) and (**3.9**) is obtained the ΔY_{min} value (**3.10**)

$$\Delta Y_{min} = 0.114 \quad (3.10)$$

From (**2.4**) and (**3.10**) the $Ymse_{min}$ is deduced (**3.11**)

$$Ymse_{min} = \frac{1}{height * width} * 0.114^2 \quad (3.11)$$

From (**2.3**) and (**3.11**) the Y-$PSNR_{max}$ is deduced (**3.12**)

$$YPSNR_{max} = 10 * log_{10}\left(\frac{255^2 * height * width}{0.114^2}\right) \quad (3.12)$$

Finally, the Y-PSNR normalized expression can be deduced (**3.13**).

$$YPSNR_{normalized} = \frac{YPSNR}{YPSNR_{max}} \quad (3.13)$$

3.2.1.2 PSNR normalization in audio media resources

In case of a sound signal, there is no Y component as there is only a sample component. So, the PSNR normalization has to be computed taking into an account the minimum sample variation.

Whatever the bits per sound sample defined in the sound signal, the minimum difference level between two signals is 1. So, the ΔY_{min} is equal to 1 and, in consequence, $Ymse_{min}$ is equal to 1.

Then, the $PSNR_{max}$ in audio is defined as in (**3.14**)

$$PSNR_{max} = 10 * log_{10}\left(\frac{\left(2^{bits/sample} - 1\right)^2 * samples}{1^2}\right) \quad (3.14)$$

Finally, the $PSNR_{normalized}$ in audio is expressed in (**3.15**)

$$PSNR_{normalized} = \frac{PSNR}{PSNR_{max}} \quad (3.15)$$

These normalizations, (**3.13**) and (**3.15**), allow obtaining PSNR values comprehended from 0 to 1.

3.2.2 PEAQ normalization

The PEAQ normalization is computed like follows.

As it has been mentioned before, PEAQ values are comprehended by values between -4 to 0:

$$-4 \leq PEAQ \leq 0$$

Where '-4' means a very annoying perceptual quality and 0 means a no perceptual difference between the compared media.

So, -4 PEAQ value must be fitted in a normalized 0 value, and 0 PEAQ value must be fitted in a normalized 1 value.

In (**3.16**) is expressed the mathematical expression used to normalize PEAQ values.

$$PEAQ_{normalized} = \frac{PEAQ + 4}{4} \qquad (\mathbf{3.16})$$

This way [(**3.16**)] the normalized PEAQ values comprehends the specification range values: from 0 to 1.

3.3 Multimedia quality metrics analyzer design

In section 3.1 it has been defined the functionalities that the quality metric analyzer module must accomplish. In this section it is described the module design and the functionalities of the components that perform the module.

In the module design process it has been taken into account a modular design. This way, it is easier to implement, test and add future modifications to the module.

The design it has been done taking into account the basic characteristics in the media content analysis and management. But, as it is well known, the media concept includes image, video and audio contents. These kinds of media have in common some attributes, as they are media contents, but they have other attributes that make them different from each other. For this reason it has been thought in the design of basic components that model the media. And the idea is to, later, extend these components in order to model the different kind of media: image, video and audio.
This concept is known, in software development, as inheritance. In **Figure 3.1** is shown the UML diagram that models this concept.

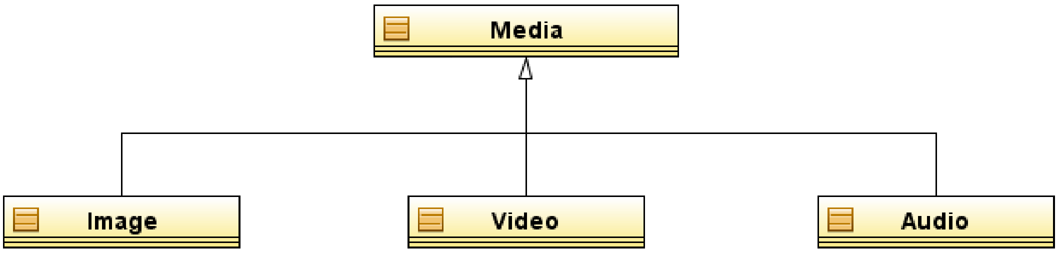


Figure 3.1 Inheritance UML Diagram

In the functionalities definition (see section 3.1) can be seen that there are two kinds of functionalities: analysis functionalities and management functionalities. In this way, it has been divided the module into two main components (**Figure 3.2**):

- A media codec manager, named MediaCodecManager
- A media quality metric analyzer, named MediaMetricAnalyzer.

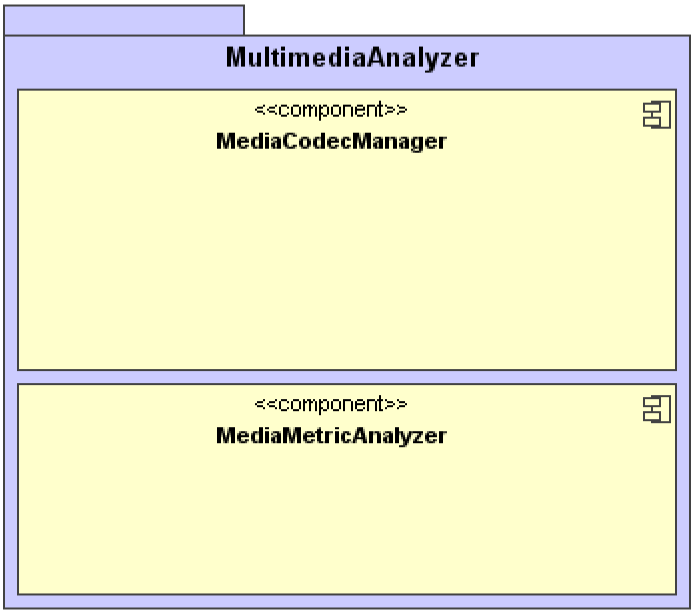


Figure 3.2 Multimedia Analyzer component diagram

3.3.1 Media Codec Manager

The MediaCodecManager is the responsible component to manage the different analyzed codecs.

Its main functionalities are:

- Allow adding coded resources.
- Classify coded resources in its correspondent media codec.

- Manage the coded resource analysis:
    - Analyze resources with different quality metrics
    - Allow sorting by score, ratio or quality metric.

It is talked about coded resources and media codecs. So, it is easy to think into two kinds of elements:

- CodedMedia is the element that represents a coded media resource.
- MediaCodec is the element that represents a media codec.

These two elements compose the MediaCodecManager. In **Figure 3.3** it is shown the MediaCodecManager dependences UML diagram.

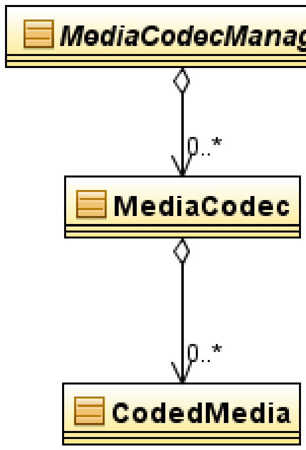


Figure 3.3 MediaCodecManager dependences diagram

In the dependences UML diagram (**Figure 3.3**) can be seen the dependences between elements:

- The MediaCodecManager is composed by several MediaCodec that represent the managed codecs.
- A MediaCodec is composed by several CodedMedia that represent the media resources coded with the codec modelled by the MediaCodec.

Figure 3.4 reflects clearly this composition concept.

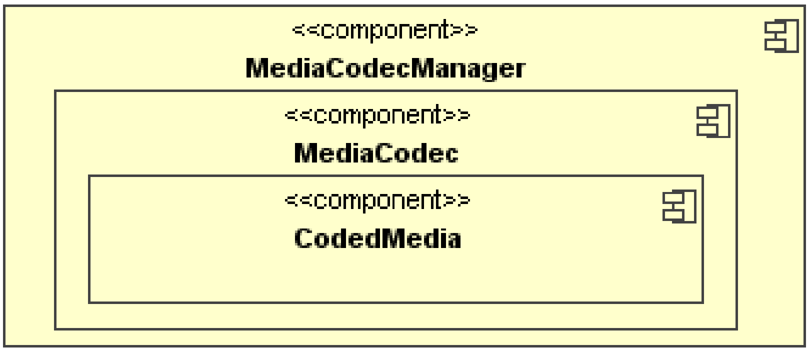


Figure 3.4 Media Codec Manager component diagram

In ANNEX 1 it is shown the MediaCodecManager class diagram.

It has been mentioned that there are 3 kind of media: image, video and audio. So, there will be 3 kind of MediaCodecManager:

- ImageCodecManager
- VideoCodecManager
- AudioCodecManager

Each of them manages the resources and codecs of the media type that they represent.
Each kind of codec manager extends the basic codec manager that is the MediaCodecManager. As it has been mentioned before, this property is known as inheritance. This way, each kind of codec manager will content its correspondent kind of media codec or, what is called, MediaCodec. And each kind of MediaCodec will content its correspondent CodedMedia.

In **Figure 3.5**, **Figure 3.6** and **Figure 3.7** it is shown the dependences diagram of each kind of media.

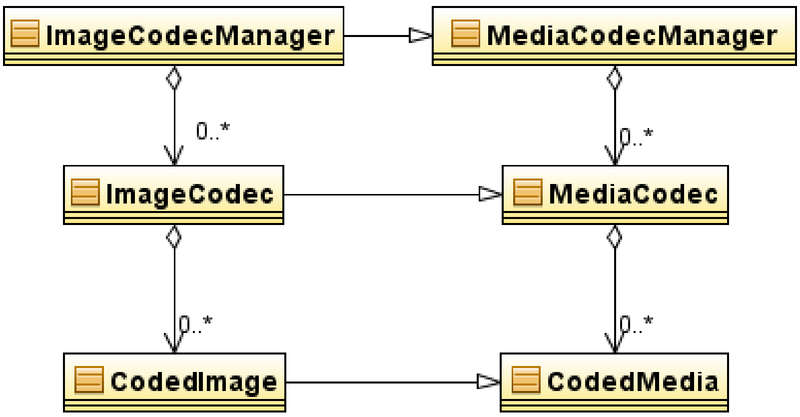


Figure 3.5 ImageCodecManager dependences diagram

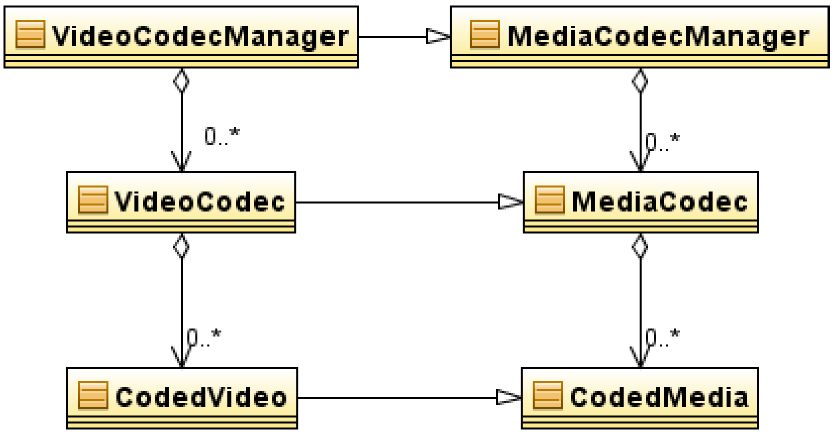


Figure 3.6 VideoCodecManager dependences diagram

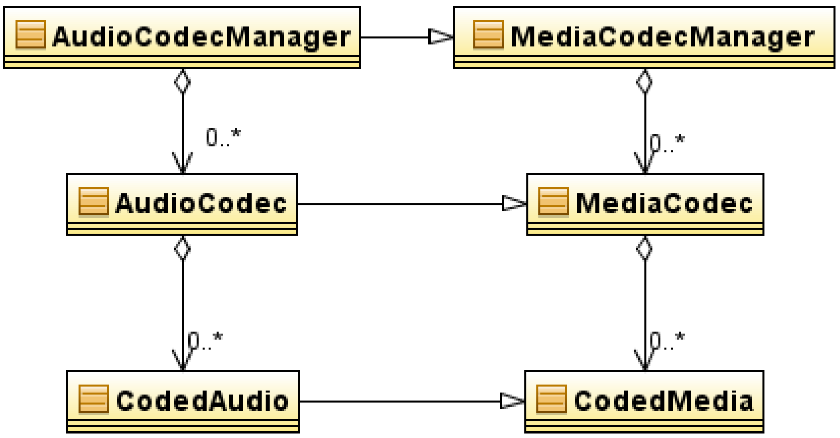


Figure 3.7 AudioCodecManager dependences diagram

In the MediaCodecManager class diagram (see ANNEX 1) can be observed that there are defined some abstract methods. This means that each kind of MediaCodecManager (image, video and audio) must implements these methods specifically for the media type they implement.

In ANNEX 1 it is shown the ImageCodecManager, the VideoCodecManager and the AudioCodecManager class diagrams.

3.3.1.1 MediaCodec

The MediaCodec is a MediaCodecManager element that models a media codec.
Its main functionalities are:

- To model a media codec
- To manage the coded media resources coded with the media codec represented.

The MediaCodec class diagram (see ANNEX 1) shows the attributes and methods that define its properties.

Each type of media codec (image, video and audio) is modelled by ImageCodec, VideoCodec and AudioCodec classes. These classes extend the basic functionalities of MediaCodec class and model specifically each type of media codec.
The class diagram of each type of MediaCodec is shown in ANNEX 1.

As it can be seen in **Figure 3.3** and as it has been mentioned before, MediaCodec is composed by a collection of media resources coded with the codec that MediaCodec defines. These elements are modelled by the CodedMedia class.

3.3.1.2 CodedMedia

The CodedMedia is the basic element of the management system of the codec analysis. It represents the analysis information related to a coded media resource.

The CodedMedia is who uses the MediaMetricAnalyzer to make the quality metric analysis.

As it happens in the MediaCodecManager and MediaCodec elements, there are 3 types of CodedMedia: CodedImage, CodedVideo and CodedAudio. These types represent the image, video and audio coded resources.
Each CodedMedia type has an attribute collection that makes the difference between them as it can be seen in the class diagrams (see ANNEX 1). And, also, each image, video and audio CodedMedia uses its correspondent MediaMetricAnalyzer: ImageMetricAnalyzer to analyze image metrics, VideoMetricAnalyzer to analyze video metrics and AudioMetricAnalyzer to analyze audio metrics.

3.3.2 MediaMetricAnalyzer

MediaMetricAnalyzer is the component who manages the quality metric analysis. This specific element is not which will manage all the media types analysis but the different types of MediaMetricAnalyzer will do it in a specific way due to the specific media type.

There are 3 types of MediaMetricAnalyzer:

- ImageMetricAnalyzer: image metric analysis manager
- VideoMetricAnalyzer: video metric analysis manager
- AudioMetricAnalyzer: audio metric analysis manager

Each type of MediaMetricAnalyzer refers to the implementation of each specific media metric quality.
The main inputs of this component are:

- The original media resource in raw format
- The original media resource scaled due the coding parameters of the coded media (resolution, frame rate, sampling frequency, bits/sample, etc.)
- The coded media resource
- The coded media resource in raw format

In order to perform the quality assessment the first and last inputs are needed. The second and third inputs are used to perform the compression ratio assessment.

3.3.2.1 ImageMetricAnalyzer

ImageMetricAnalyzer is the image quality metrics analysis manager.
The metrics that this element manages are those related to image in CHAPTER 2: PSNR and SSIM.
So, ImageMetricAnalyzer uses the specific PSNR and SSIM image implementation.

The ImageMetricAnalyzer class diagram is shown in ANNEX 1.

3.3.2.1.1 PSNR analysis in image

The image PSNR analyzer tool is implemented following the Y-PSNR mathematical formula shown in (**2.3**) where the minimum sample is the Y component of an image pixel.

3.3.2.1.2 SSIM analysis in image

The image SSIM analyzer tool is implemented taking as an implementation base an implemented SSIM tool [14]. This tool has been modified to be adapted to the specific Analyzer Module needs.

This implementation takes as a base the mathematical expression defined in (**2.15**). As in PSNR case, it is also taken the Y pixel component as a sample.
One of the main characteristics of this implementation, this makes that the final implementation differs from the basic mathematical formula, is that makes the image analysis splitting it in blocs of MxM pixels. Then is applied a MxM circular-symmetric Gaussian weighting function with standard deviation of 1.5 samples and normalized to unit sum (see **Figure 3.8**) as it is recommended in the SSIM author specification [11].

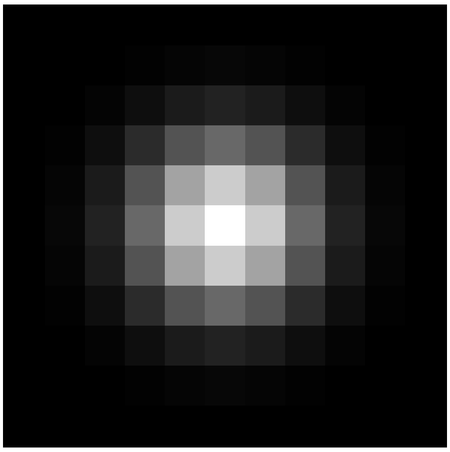


Figure 3.8 11x11 pixels circular-symmetric Gaussian weighting function image representation

The Gaussian weighting function assigns more weight relevance to some samples than others. This filter simulates the points where the eye takes more attention in.

In **Figure 3.8** is shown an image representation of an 11x11 pixels circular-symmetric Gaussian weighting function. It can be observed that there are brighter zones at the centre than at the corners. This indicates that brighter zones are regions where once the SSIM analysis is done the result is emphasized.

Once the weighting function is applied at each MxM bloc, the SSIM analysis is done in each bloc (see mathematical expression in (**2.14**)).
Finally it is done the average of each bloc SSIM value and the SSIM value of the image is obtained.

3.3.2.2 VideoMetricAnalyzer

The VideoMetricAnalyzer is the video quality metrics analysis manager.
Due to a video can be understood as an image sequence, the quality metrics applied to video are the same as for image: PSNR and SSIM.

To be able to apply the image analysis metrics in a video it has to be processed the video to extract the frames.
To realize this process it is used the FFMPEG transcoder [18]. The FFMPEG transcoder allows extracting the frames of videos.

Once the video frames are extracted the frame by frame analysis is done using the image analysis metrics.

As the frame by frame analysis can be computationally huge, the VideoMetricAnalyzer allows analyzing a specific percentage of the video frames.

The VideoMetricAnalyzer class diagram is shown in ANNEX 1.

3.3.2.3 AudioMetricAnalyzer

The AudioMetricAnalyzer is the audio quality metrics analysis manager. The metrics that it manages are those studied in CHAPTER 2 and applied in audio: PSNR, SSIM and PEAQ.

The AudioMetricAnalyzer class diagram is shown in ANNEX 1.

3.3.2.3.1 PSNR analysis in audio

The audio PSNR analyzer has been implemented following the mathematical expression of the PSNR shown in (**2.1**) and (**2.2**) with just the difference that in (**2.1**) instead of using a two-dimensional signal it is used a one-dimensional signal.
Each sample is an audio sample in the specific temporal instant.

3.3.2.3.2 SSIM analysis in audio

The audio SSIM analyser implementation has followed the technique defined in section 2.4. Unlike in image, in the audio SSIM analysis is not applied the Gaussian weighting function. This is because the audio analysis is done in the time domain and applying the weighting function does only result in a signal amplitude variation. And this amplitude variation does not give Human Hearing System emulation.

3.3.2.3.3 PEAQ analysis in audio

The PEAQ audio metric analysis is done using an analyzing tool named EAQUAL [19]. This tool is an implementation of the ITU-R recommendation BS.1387[13] that performs the PEAQ standard algorithm.
The EAQUAL tool allows getting the PEAQ analysis of an audio signal as is described in section 2.5.

CHAPTER 4. MEDIA CODECS ANALYSIS IN STREAMING PACKET LOSS SCENARIO

Not all media codecs have the same robustness in front data losses. This fact implies that the coded media perceptual quality varies depending on the codec used.

In this chapter is presented an objective quality study made over different coded media resources (video and audio) subjected to losses. It is also presented an objective quality study made over image resources without any data loss insertion.

To realise this study it has been made an automatic quality metrics analyzer which uses the analyzer module described on CHAPTER 3.

Along this chapter is described the realized setting-up to make the media codecs analysis and it is also presented results gotten from this analysis.

The different media codecs analyzed are:

- Image codecs:
  - JPG
  - GIF
  - PNG
- Video codecs:
  - MPEG 1
  - MPEG 2
  - MPEG 4 part 2
  - H.263
  - H.264
  - WMV1
  - WMV2
- Audio codecs:
  - MP3
  - AAC
  - AC3
  - VORBIS

It has been chosen these codecs because are the initial media codecs defined in the Transcoding Service of the i3media project. But the modularity of the transcoding module allows adding new media codecs to the service.

4.1 Testing scenario

The main goal of the tests is to see the media codecs robustness in front of losses on a video streaming over a network. And see how these losses are reflected on the media reception perceptual quality.

There are 3 basic elements in the testing scenario architecture:

- Streaming media server
- Streaming client where the media resource is analyzed
- A controlled network over which losses are introduced.

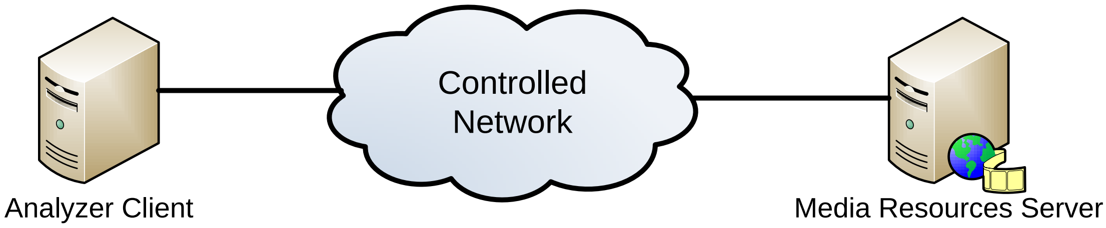


Figure 4.1 Scenario elements map

These are the 3 basic elements used in the tests. In **Figure 4.1** can be seen the scenario elements map.

4.1.1 Media Resource Server

As a video streaming server it has been used the FFMPEG transcoding software [18] with the codec library implementation libavcodec 52.10.0. This transcoder allows transcoding multimedia resources with whatever supported codec. FFMPEG also supports transcoding over a network interface and deliver a media stream through it over any IP transport protocol supported: tcp, udp, rtp and http.

In order to control remotely this transcoder it has been implemented a webservice [20] which publish the transcoding service. This service allows calling remotely to FFMPEG transcoding functions.

A java webservice implementation, like this, needs a web applications server. As a web applications server it has been used Apache Tomcat [22].

The public service interface is shown in the class diagram in **Figure 4.2**.

```
TranscodingModuleWS
Attributes
Operations
public String transcodingVideoRequest( int iwidth, int iheight, String ipath, String ocont, String vcodec, String acodec, int owidth, int oheight, float ovfr, int vbitrate, int abitrate, int oachann, int oasf, boolean sameq, String opath )
public String transcodingVideoRawRequest( int iwidth, int iheight, String ipath, String ocont, String vcodec, int owidth, int oheight, float ovfr, int vbitrate, boolean sameq, String opath )
public String transcodingAudioRequest( String ipath, String ocont, String acodec, int abitrate, int oachann, int oasf, boolean sameq, String opath )
public String transcodingAudioRawRequest( int iasf, String ipath, String ocont, String acodec, int abitrate, int oachann, int oasf, boolean sameq, String opath )
public String transcodingImageRequest( String ipath, String ocodec, int owidth, int oheight )
public String transcodingImageRawRequest( int iwidth, int iheight, String ipath, String ocodec, int owidth, int oheight )
public String transcodingUDPVideoRequest( int iwidth, int iheight, String ipath, String ocont, String vcodec, String acodec, int owidth, int oheight, float ovfr, int vbitrate, int abitrate, int oachann, int oasf, boolean sameq, String opath )
```

Figure 4.2 Transcoding WebService Interface

The only method used is the transcodingUDPVideoRequest to perform a video and audio transcoding using UDP as a delivering media transport protocol.
The method parameters are the parameters that the client will specify in the request. In **Table 4.1** are described the functionalities of these parameters.

Table 4.1 Transcoding UDP Video Request parameters

Parameter	Functionality
iwidth	In case of video resource transcoding, determines the original video resource width.
iheight	In case of video resource transcoding, determines, determines the original video resource height.
ipath	Specifies the original video/audio location
ocont	Output media container
vcodec	Video codec to be used
acodec	Audio codec to be used
owidth	In case of video transcoding, specifies the transcoded video width.
oheight	In case of video transcoding, specifies the transcoded video height.
ovfr	In case of video transcoding, specifies the transcoded video frame rate.
vbitrate	Specifies the coded video bitrate (bps)
abitrate	Specifies the coded audio bitrate (bps)
oachann	Determines the output coded audio channels (1-Mono, 2-Stereo)
oasf	Output audio sampling rate (Hz)
sameq	If true, the output media is tried to be coded with the same quality parameter as the original. This implies a variable output bitrate (VBR)
opath	Specifies the output media coded destination

The service descriptor (WDSL) can be found in ANNEX 2.

A call to the service makes that the application instantiates the FFMPEG transcoder and then, automatically, the media stream is sent to the client.

The sequence diagram (**Figure 4.3**) describes the process.

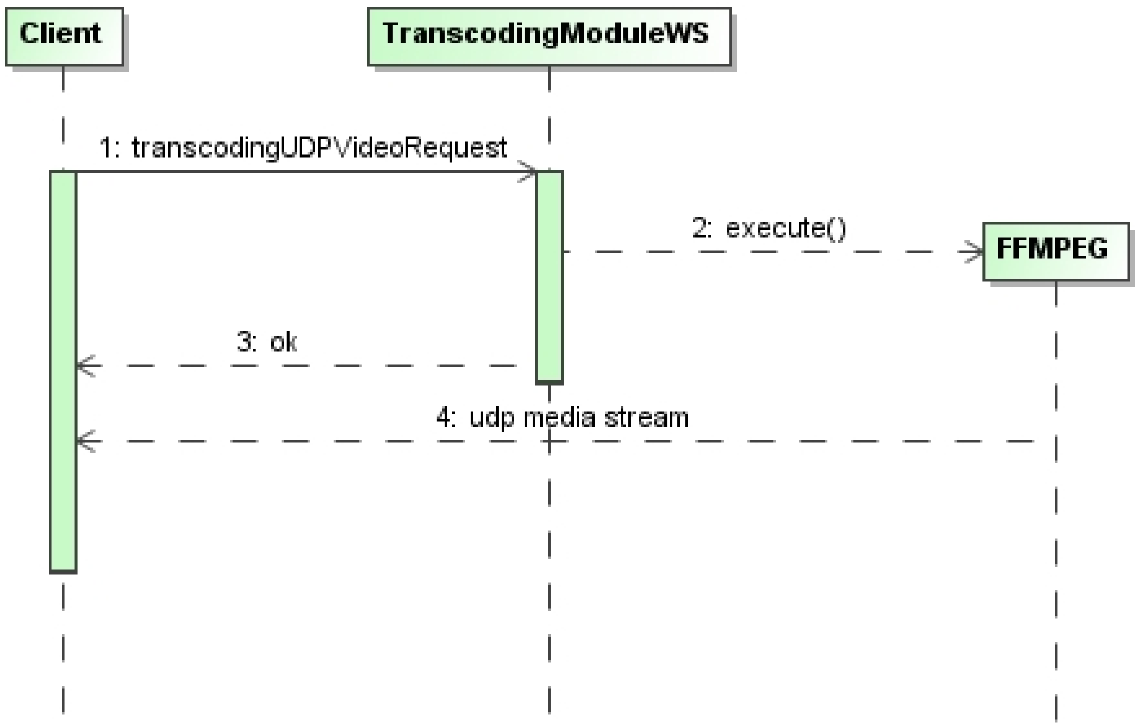


Figure 4.3 Transcoding calling sequence diagram

Once a client realizes the video transcoding request (1) the TranscodingServiceWS executes the FFMPEG transcoder (2). The FFMPEG transcoder transcodes the media and send it to the client (4). The transport protocol used is UDP as it allows introducing losses and receiving the media with these losses. This is specified in the *opath* parameter.

4.1.2 Analyzer Client

The client is a Java application that realizes requests to the transcoding webservice and receives the streaming sent by the server.
Once it receives the coded video resource, it decodes the video and analyzes its perceptual quality.

The decoding process is done using the FFMPEG transcoder. The client incorporates this element in order to realize this function.

The quality analysis is done by the Multimedia Quality Metrics Analyzer specified in CHAPTER 3.

It is mandatory that the client get the original resource in raw format in order that the analyzer module could process the resource analysis. This is because, as is described in CHAPTER 2, the analyzer uses Full Reference metrics and they require the reference resource in order to make the analysis.

The whole codification request and analysis processes are automated and follow the configuration specified into 2 configuration files:

- A file where the client configuration parameters are defined.
- A file where the parameters of the codecs (transcoding profiles) to be analyzed are defined.

An example of both files can be found in ANNEX 3.

The flow diagram shown in **Figure 4.4** describes the automated analysis process.

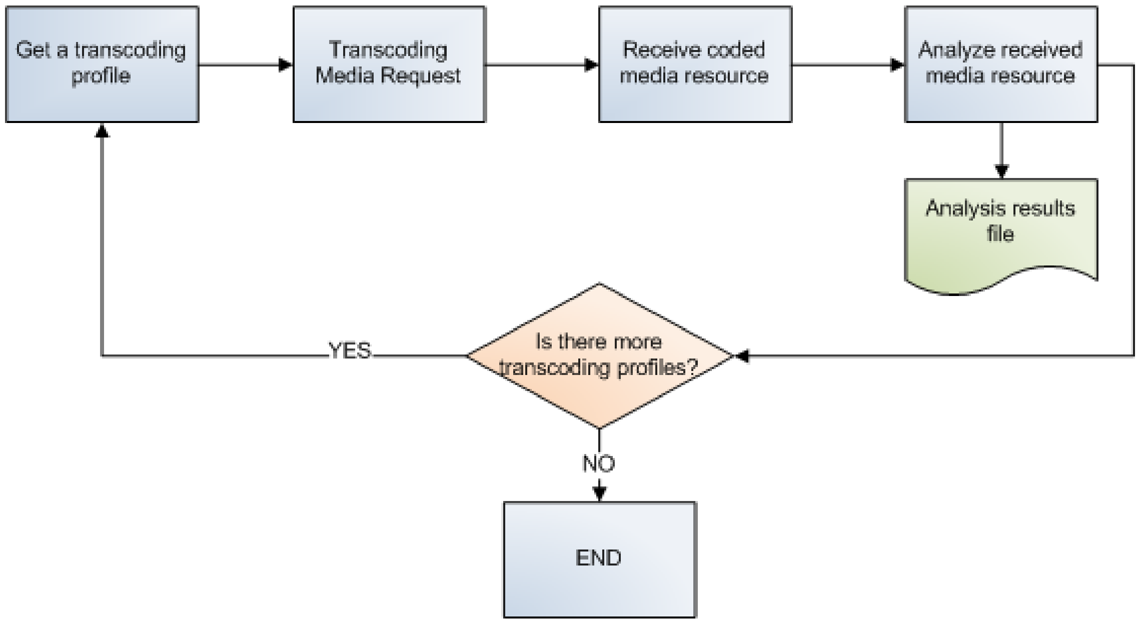


Figure 4.4 Automatic client process flow diagram

4.1.3 Controlled Network

It is good to have a controlled network where be able to determine the packet loss rate in order to introduce transmission losses.
To get through this task it has been used an Unix distribution called FreeBSD [23]. This Unix distribution incorporates a network emulator called DummyNet [24] which allows to emulate networks with a specific bandwidth and packet loss rate (PLR).
The commands used to configure the Dummynet network tool can be found in ANNEX 3.

The implemented testing scenario is shown in **Figure 4.5**

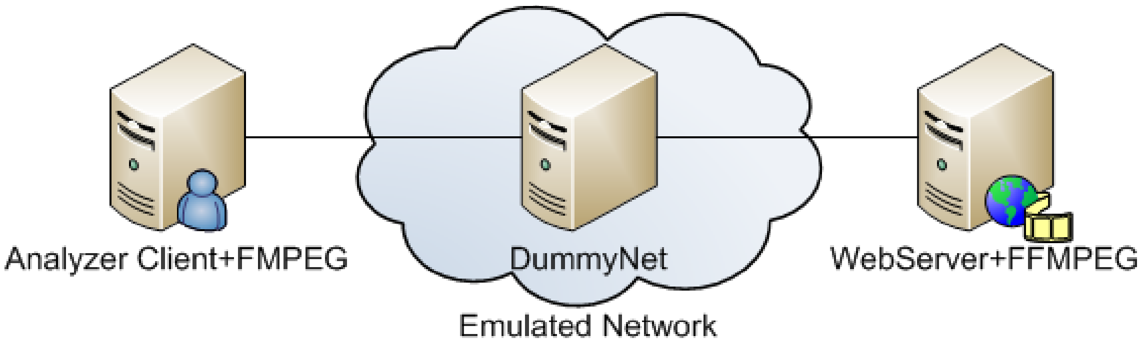


Figure 4.5 Implemented testing scenario

4.2 Realized tests description

The realized tests over the scenario defined in 4.1 have been done after defining the next parameters:

- Codecs to be analyzed
- Media resources to be coded
- Packet Loss Rate (PLR)

As it has been told at the top of this chapter, the codecs to be analyzed are:

- Image codecs:
  - JPG
  - GIF
  - PNG
- Video codecs:
  - MPEG 1
  - MPEG 2
  - MPEG 4 part 2
  - H.263
  - H.264
  - WMV1
  - WMV2
- Audio codecs:
  - MP3
  - AAC
  - AC3
  - VORBIS

The codecs configuration has been specified like follows:

- Image:
  - Quality paramenter (PNG and JPG): 80%
- Video:
  - Bitrate: 1024kbps
  - Frame rate: 25fps
  - GoP size: 12
  - Quantification scale variation: 0

- Audio:
    - Bitrate: 128kbps
    - Sampling frequency: 44100Hz
    - Bits/sample: 16bits
    - Coding quality parameter: codec default

All these codification parameters are defined for all the used codecs with the different media types, except the quality parameter in image codecs. This parameter is only compatible by PNG and JPG codecs.

The resources used are:

- Image
    - Lena
- Video
    - Foreman
- Audio
    - Vocal quartet
    - Instrument flute

It has been chosen these multimedia resources because are the ones used in standard quality assessment studies.

The main characteristics of these resources can be found in ANNEX 4.

The Packet Loss Rates used in the video and audio analysis are: 1%, 3%, 5% and 10%. In image analysis it is not considered any loss insertion as a loss in data image is considered a 100% data loss as the decoder is not able to decode the data.

It has been chosen these PLR values because they are maximum values that determine the maximum PLR of some of the most common network technologies as it can be seen in **Table 4.2**.

Table 4.2 Maximum PLR in access networks technologies

Technology	Maximum PLR
xDSL	5%
ISDN	1%
ETH	3%
FR	1%
GPRS	5%
UMTS	3%

Once it has been defined the 3 basic parameters (codecs, resource and PLR) it has been realized 10 tests batteries. Each test result is stored in a log file.

4.3 Tests results

Once the testing process has been realized the results are processed. The different media testing results is shown and commented next.

It is good to be commented that all the test results are referenced to the specific codecs implementation, in this case, the FFMPEG codecs implementation. So, these results can vary depending on used implementation.

4.3.1 Video test results

From the video test results (see ANNEX 5) it can be concluded the following.
In lossless environments (PLR equals to 0%) the best video codec is H.264 with a 94% of similarity with the original resource, followed by MPEG-4 part 2 (92.6% of similarity). The other codecs have similar percentage values, around 91%, emphasising that MPEG-2 is located as the worst codec in terms of analyzed quality.

In packet loss environments the decoders behaviour change. So, these results will help to see which the most robust codec is in a packet loss scenario.

It is interesting to take a look in the MPEG-2 codec behaviour. MPEG-2 in a lossless environment it is not the best codec, due the results, in terms of analyzed quality. But in packet loss environments, MPEG-2 is the codec that performs the best perceptual quality, just the opposite of H.264. H.264 in lossless environments is the best codec, in terms of analyzed quality, while in packet loss environments, from a 3% of packet loss rate, H.264 is the worst codec. This is similar to MPEG-4 part 2, where in a lossless environment is the second best codec and in packet loss environments is the second worst codec.

Having a look at the compression ratio values, it can be seen that there are no great differences between the different analyzed codecs. All the codecs compression ratios are about 96%. But if it is considered the best compressor codec, this is the H.264 codec with a 96.6% of compression ratio.

It can be concluded that the MPEG-2 analyzed implementation is the most robust in front losses, while H.264 is less robust. So, in environment with a high data loss probability the best codec implementation to be used is MPEG-2.

If these results were extended to real scenarios, it could be understood why MPEG-2 is the used codec in the DVB-T standard. DVB-T standard is the standard used in the Terrestial Digital Television (TDT) and the signal is propagated through the air. This environment has a high data loss probability and it has been observed, from the test results, that the best video codec in a data loss environment is MPEG-2. But whatever extension to real scenarios it must be done taking into account that the results gotten from the tests are related to specific codecs implementation.

4.3.2 Audio test results

In ANNEX 5 are listed the audio tests results. From these results it can be concluded the following.
As it has been mentioned along this document the quality metrics used in audio analysis are: SSIM, PSNR and PEAQ.

SSIM and PSNR analyses the audio samples in the time domain. This fact implies that the results differ from the ones obtained from the PEAQ analysis. This difference is due to PEAQ performs a sampling processing in the frequency domain and the used codecs takes as a coding base the audio signal behaviour in the frequency domain.
The coding concept behind these codecs is that no matter the waving shape in the time domain if the final audio perception does not vary. For this reason the SSIM and PSNR analysis in audio does not give a good quality assessment.

Thanks to the tests results it has been corroborated the hypothesis that audio codec quality assessment has to be done taking into account the frequency domain but not time domain due the audio analyzed audio codecs behaviour.

For this reason the audio tests results are studied following the PEAQ analysis results.

From the PEAQ analysis results can be concluded that the best audio codec, whatever the scenario, is, in terms of quality, ac3 followed by aac. But in terms of compression ratio the best one is vorbis, although is the worst in terms of quality.
If the score parameter is considered, the best scored is, as if the quality assessment was only considered, the ac3 codec.

4.3.3 Image test results

From the image test results (see ANNEX 5) it can be concluded that the codec that performs the best perceptual quality is PNG (100% of similarity with the original resource). But against this fact, this codec is the codec that performs the worst compression ratio (an 8.6% compression ratio).

The image codec that performs the best quality-compression ratio relation is JPG. This codec performs a perceptual quality of 94.6% of similarity respects the original resource, and a compression ratio relation of 94.7%.

CHAPTER 5. CONTENT ADAPTATION SERVICE INTEGRATION

As it has been mentioned in CHAPTER 1 one of the handicaps of the Adaptation Module from the i3media project is to decide which the best media codec is to be used in the content adaptation for a specific user context.

It has been seen (CHAPTER 2) that exist some metrics that allow to determine the perceptual quality of a coded resource. It has been also explained that a way to determine that a codec is better than other is taking into account the perceptual quality versus compression ratio relation. For this reason, determining this relation for all the codecs helps to determine best one.

In CHAPTER 3 it is specified the design and implementation of a management and analysis module of coded resources. This module allows analyzing the perceptual quality of the coded resources, as well as the compression ratio of the used codec. This way the quality and compression ratio is got.

To exploit this analysis capacity to be used by the Content Adapatation Service it must be specified the integration process of this Multimedia Management Analyzer Module with the Adaptation Module. This is what is explained in this chapter.

5.1 Basic considerations

The Adaptation Module is ready to work with coding profiles. A coding profile is a set of specifications that determine how a resource has to be coded. These specifications can be: the codec to be used, image resolution (in case of an image or video resource), frames per second (in case of video resources), samples per second (in case of audio resources), etc.

In the content adaptation process a collection of coding profiles, supported by the user device and network, are chosen. From these profiles collection is chosen that profile that shows the best quality-compression ratio relation.

At this point is where the Multimedia Management Analyzer Module integration is needed.
The integration point is in the qualification due the score parameter (see 3.2) of the different coding profiles. From the user context compatible profiles the Adaptation Module will chose that profile that shows the best score.

So, it has to be analyzed the different coding profiles and qualify them due the score parameter.

5.2 Integration module: Profile Analysis Module

To carry out the integration it has been designed and implemented a Profile Analysis Module. This module has as a main component the Multimedia Analyzer Module described in CHAPTER 3.

The Profile Analysis Module functionality is to automate the transcoding profiles analysis.
As it can be seen in **Figure 5.1**, given a transcoding profile and multimedia resource the module carries out the analysis of the coded resource according to the indicated profile. The analysis result of each coded resource is the score (described in 3.2) that will be assigned to each coding profile.

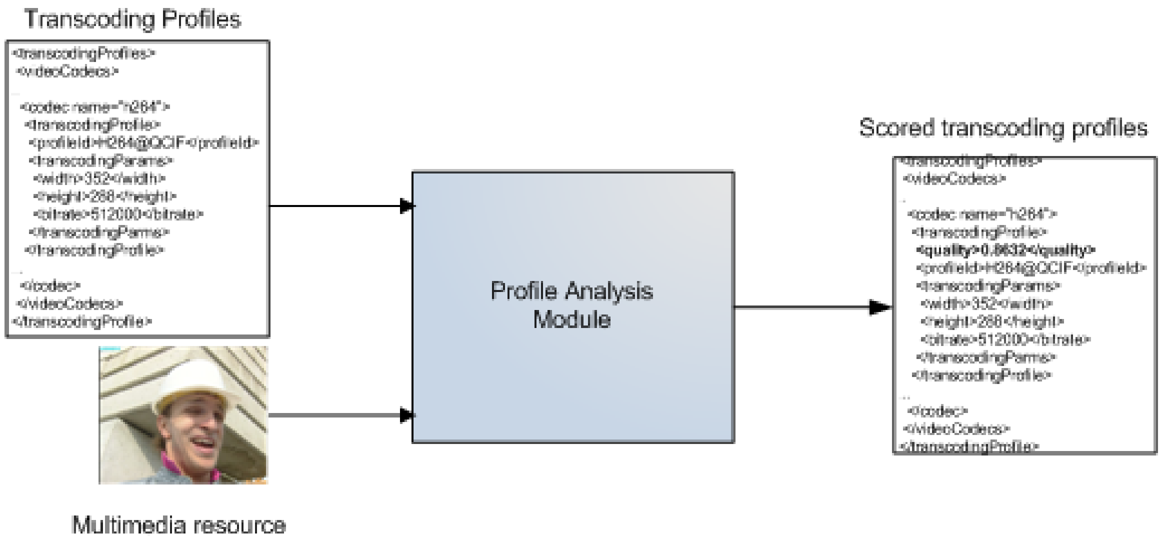


Figure 5.1 Input-output function diagram

The resource indicated as an input parameter is a resource that is not coded according to the transcoding profiles, also specified as an input parameter. So, it is needed a transcoder that carries out the resource coding process.

At this point has to be done integration with the Transcoding Module of the Content Adaptation Service.
The Transcoding Module is deployed as a web service and it can be acceded to it through web service calling.

In CHAPTER 3 is defined how the Multimedia Analysis Module works and one of the requirements to carry out the analysis of a resource is that are needed:

- The original media resource in raw format
- The original media resource in raw format scaled according to the coding parameters (resolution, frame rate, bitrate, bits/sample, sampling frequency, etc.)
- The coded media resource
- The coded media resource in raw format

Getting these requirements can be automated with the Transcoding Module integration.

Once these requirements are obtained, the Multimedia Analysis Module can analyze the resources and obtain the score parameter.

In **Figure 5.2** it is shown the Profile Analysis stream diagram. This diagram shows the profile analysis process inside the Profile Analysis Module explained above.

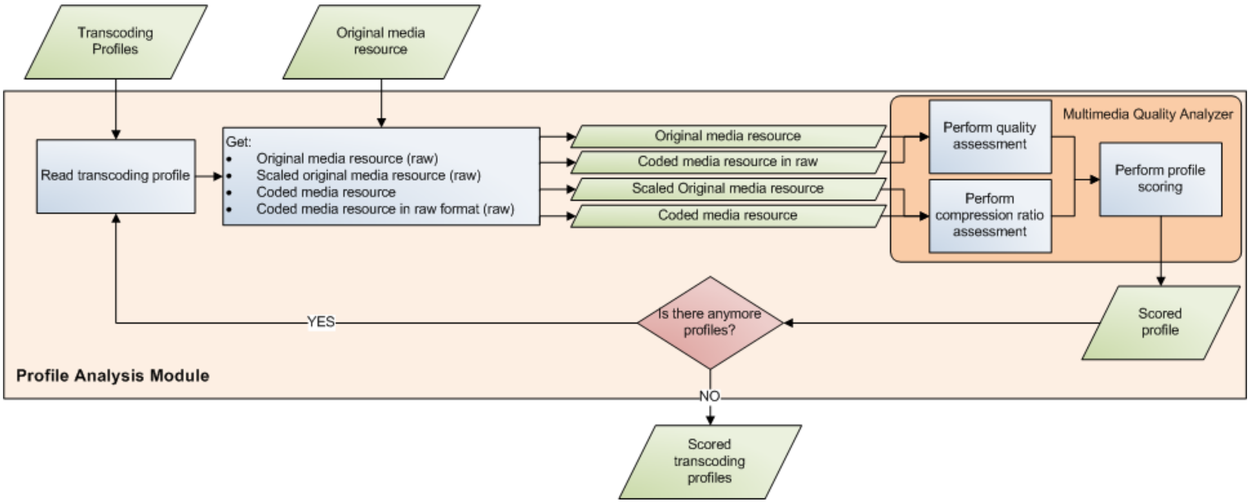


Figure 5.2 Profile Analysis stream diagram

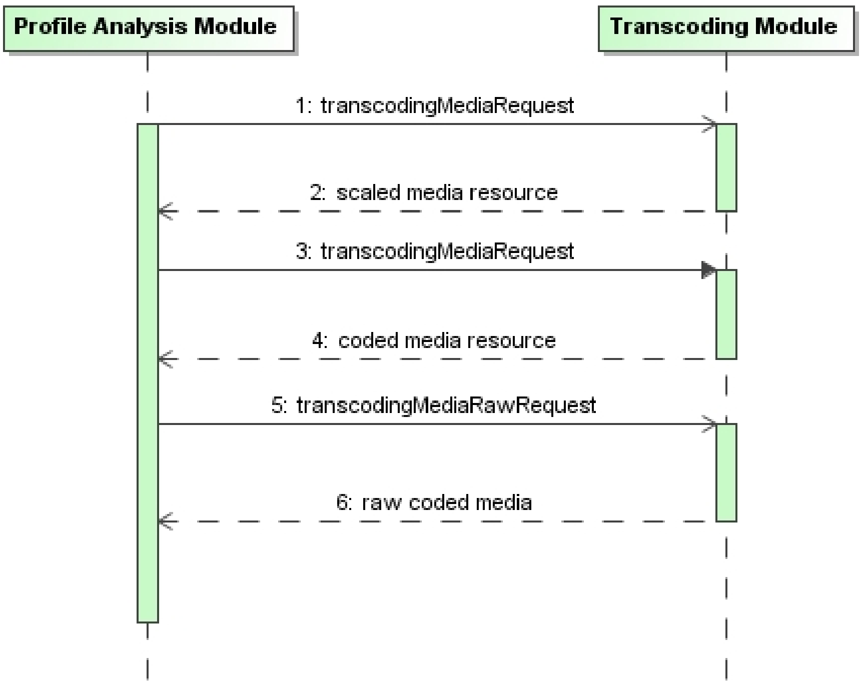


Figure 5.3 Resource obtaining process diagram

Once the Profile Analysis Module starts reading the transcoding profiles (as it is shown in **Figure 5.2**) the module starts getting the different manipulated media resources. This process is shown in **Figure 5.3**.

As the original media resource is one of the input parameters, there is no need of getting it as it is yet gotten. The other three resource requeriments are gotten from the Transcoding Module (**Figure 5.3**):

1. The original scaled media resource is requested. The Transcoding Module modifies the original media resource characteristics according to the specified parameters.
2. The Transcoding Module sends the scaled media resource to the Profile Analysis Module.
3. The third resource to be gotten is the coded media resource.
4. Once the Transcoding Module has done the transcoding process sends the coded resource to the Profile Analysis Module
5. The last resource to be gotten is the coded media in raw format. The Transcoding Module is able to perform the coding and raw decoding process.
6. Finally the Transcoding Module sends the raw coded media to the Profile Analysis Module.

5.3 Management web interface

It has been designed and implemented a management web interface. This interface completes the integration process as it is integrated in the Content Adaptation Service web interface.
The web interface allows a manager to manage the coding profiles of the Content Adaptation Service.

The management functionalities are:

- Add/Remove coding profiles
- Profiling analysis using available resources
- Select the weight parameter for image, video and audio resources (see 3.2 for more information on the weight parameter)

The way to integrate the profiling analysis web is through adding new menu items to the Content Adaptation Service web interface.

Figure 5.4 shows the welcome page of the Content Adaptation Service.

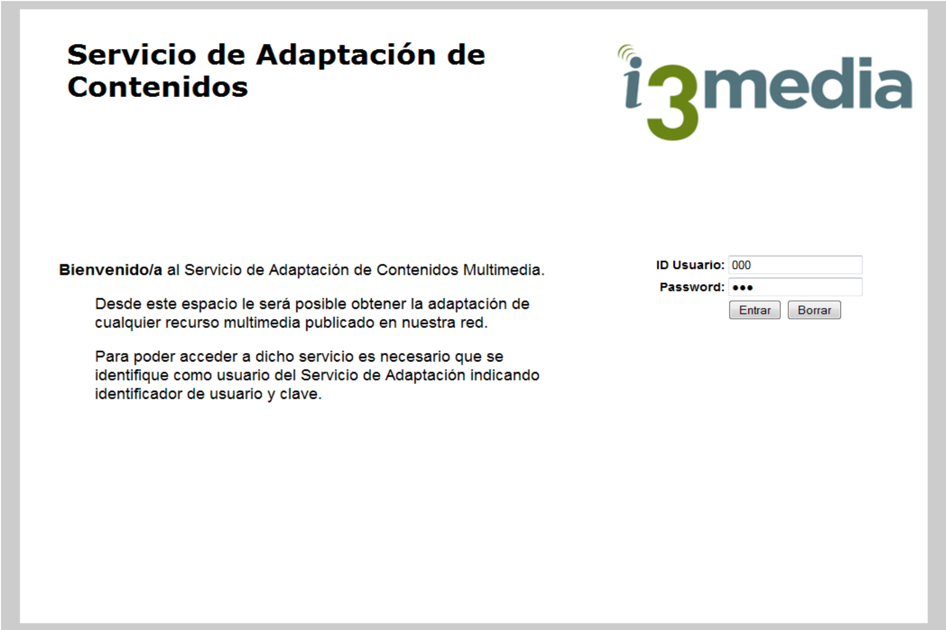


Figure 5.4 Content Adaptation Service welcome page

In this page (**Figure 5.4**) it is required the user name and password in order to be allowed to the Content Adaptation Service.

Once has been filled the entry form the user is able to see the main page (**Figure 5.5**). In this page can be reached to every Content Adaptation Service function:

- Automatic device type detection
- Location
- Automatic device capabilities detection.
- Content adaptation request
- Network capabilities detection
- Device capabilities management

Figure 5.5 Content Adaptation Service main page

In **Figure 5.5** can be seen the new menu item Profiling Management (Gestión de perfiles). This is the menu item added in order to integrate the profiling analysis with the Content Adaptation Service. If a user clicks this item will be redirected to the transcoding profiles management page (**Figure 5.6**)

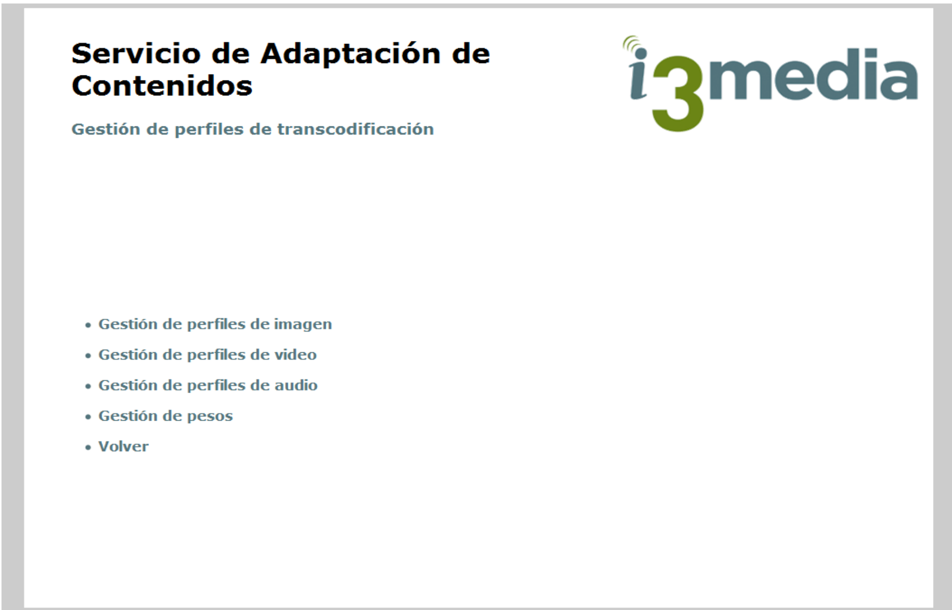


Figure 5.6 Transcoding profiles management page

5.3.1 Profiling management web interface

As it is shown in **Figure 5.6** the profiling management can be done by the different kind of media: image, video and audio.
The management of the media profiles is the same for each type of media and follows the next functionalities:

- Analyze profile
- Add and remove profile
- Show profile characteristics

These management functionalities are used through different menu items. In **Figure 5.7** can be seen a screenshot of the image profiles management page where the different image profiles are listed and the management menu items are shown.

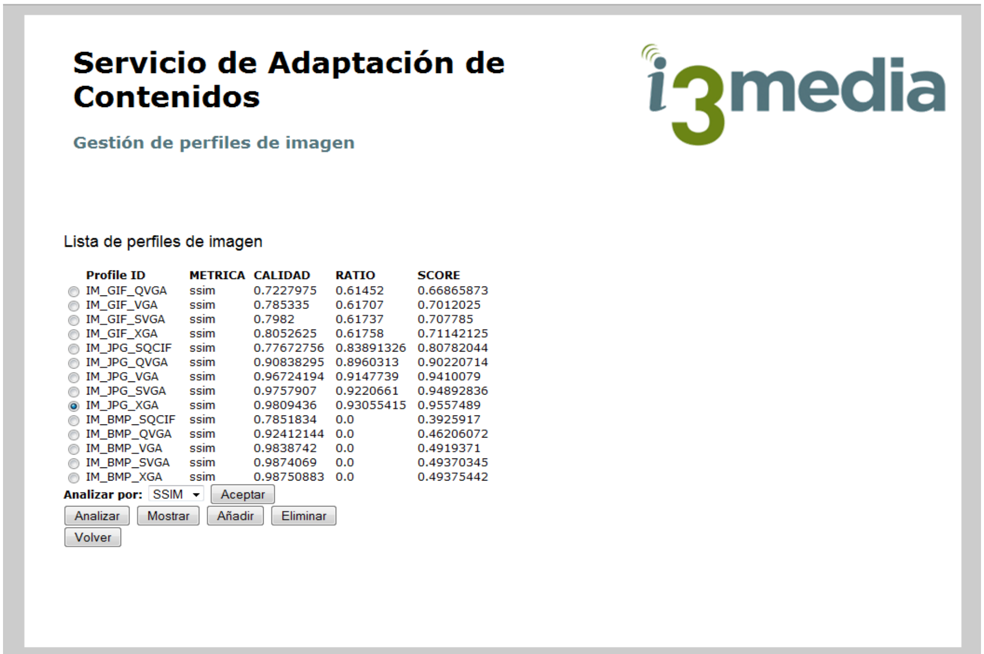


Figure 5.7 Image profile list

It is good to emphasize that in the media profile management are listed the different media profiles and it is shown the quality metric used to analyze the media profile as well as the quality metric value, the compression ratio and the score parameter gotten from the quality metric and the compression ratio (see 3.2).
In order to allow making use of the different quality metrics to analyze the media it is incorporated a list to allow a user selecting the quality metric to be used, as it can be seen in **Figure 5.7**.

5.3.1.1 Analyzing profiles

The main functionality of the profiling management interface is the analyzing one, as it is the integrating functionality between the Profile Analyzer Module and the Adaptation Module.

The web interface allows a user to choose the resource to be analyzed by the transcoding profile chosen.
There is a difference between analyzing an image and analyzing a video or an audio: in the case of video and audio the user can specify if the video or audio must be analyzed completely or only a portion of it. This is done because in case of a large time duration resource the media analysis process can be hard.

In **Figure 5.8** it is shown an example of video profile analysis management.

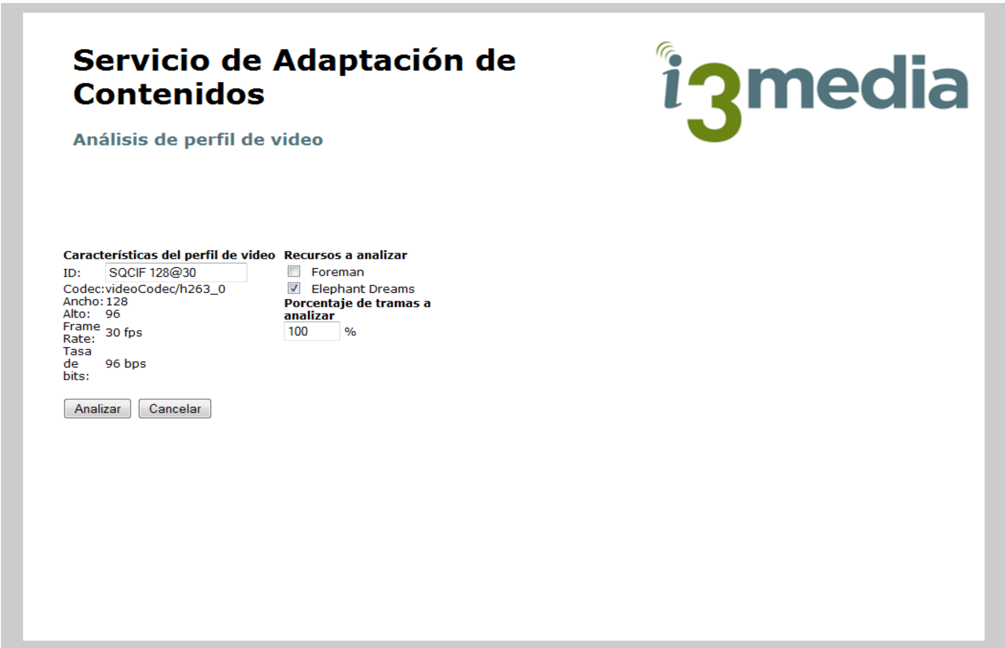


Figure 5.8 Video profile analysis form

5.3.1.2 Adding and removing profiles

In order to analyze profiles it is needed to create profiles. This is what the adding profile functionality does.
This functionality allows a user to add a new transcoding profile in order to add to the adaptation process a new profile to be chosen.

It has to be taken into account that adding new transcoding profiles are subjected to the transcoder capabilities. So, if the Transcoding Service does not support the new profile, this will not be added as a transcoding profile candidate to be chosen by the Adaptation Module.

It is also allowed to remove profiles in order to remove these profiles from the possible profiles chosen by an adaptation.

5.3.1.3 Showing profiles

If a user wants to see the main characteristics of a transcoding profile the web interface allows the user to see them.
Each transcoding profile type (image, video and audio) has different attributes as for example: in case of an image does not make sense to show the audio channels.

In case of image the attributes shown are:

- Width
- Height

In case of video the attributes shown are:

- Width
- Height
- Frame Rate
- Bitrate

In case of audio the attributes shown are:

- Number of channels
- Sampling frequency
- Bitrate

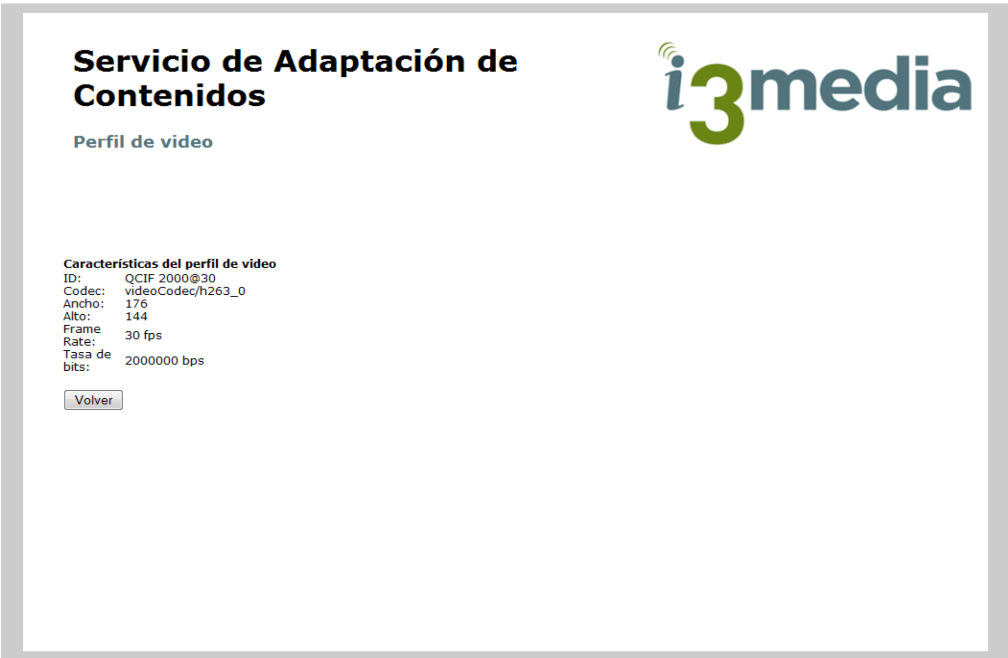


Figure 5.9 Video profile characteristics list

In **Figure 5.9** it is shown a screenshot of a video profile characteristics list where it can be seen the codec used, the resolution, the frame rate and bitrate.

5.3.2 Quality and compression relation parameter management

In section 3.2 is defined the score parameter. One of the main characteristics of this parameter is the availability of modifying the weight between the quality and the compression ratio relevance.

The way to allow a user to modify this weighting parameter is allowing him to do it using a user interface like a web interface.

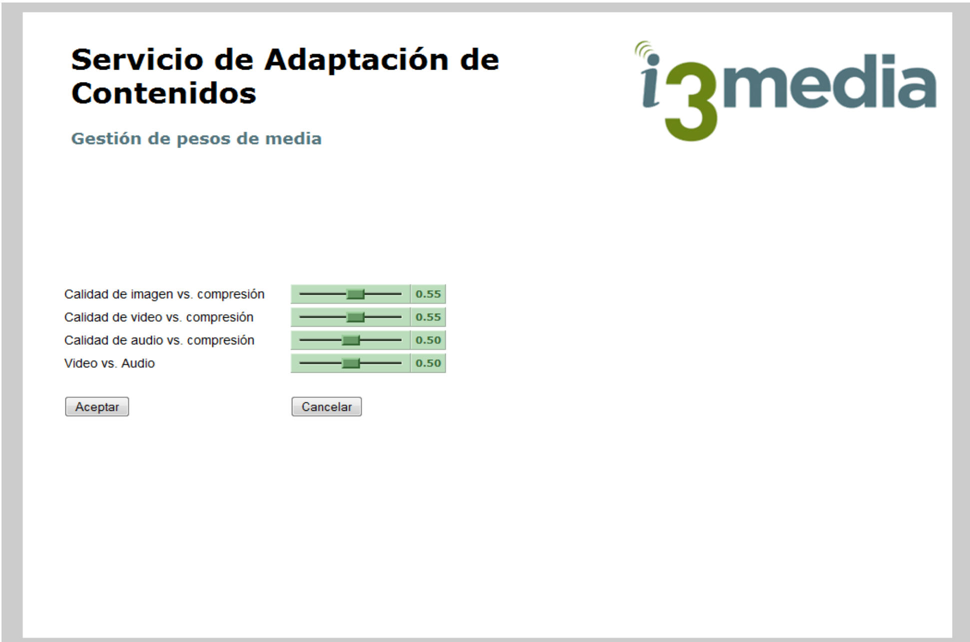


Figure 5.10 Quality and compression ratio weight management

Figure 5.10 shows the weighting parameter management web interface. Through this web interface a user can modify the weighting parameter for the different media types: image, video and audio. Using the sliders the user can assign different values, from 0 to 1:

- 1 indicates that the parameter that must predominate is the perceptual quality value.
- 0 indicates that the compression ratio is the parameter that must predominate.

There is also a fourth slider. This slider allows a user to determines, in the case of an audiovisual content, which of the two media score types (video and audio) must predominate.
This parameter is used by the Adaptation Module to choose an adequate video and audio combination in case of an audiovisual adaptation.

CHAPTER 6. CONCLUSIONS

The use of media codecs helps to reduce the final resource size (in bits) and, consequently, the use of network bandwidth. But intrinsically they also reduce the final perceptual quality of the media resource.
But, in a multimedia streaming scenario which is the best codec that performs the best compression ratio without decreasing too much the perceptual quality?

The objectives presented at the beginning of this thesis have been realized in order to be able to response to this question.

1. It has been carried out a study about multimedia quality assessment techniques.
2. It has been designed and implemented an application that performs the multimedia quality assessment using the studied techniques.
3. It has been studied the behaviour of different media codecs when the coded resources are subjected to data losses
4. This project has been one of the key pieces in the decision process of best content adaptation of the i3media project's multimedia Content Adaptation Service. For this reason it has been done the integration of the multimedia analyzer module with the Adaptation Module of the Content Adaptation Service.

At the end of this thesis it can be concluded that codec behaviour depends on the content type of the coded resource. There are differences in the compression ratio and also in the perceptual quality depending on the coded content type.
It has been found that the H.264 video codec, implemented in the FFMPEG tanscoder, is the best video codec in lossless environments. While in data loss environments the best FFMPEG codec implementation is MPEG-2.
In audio resources, the best audio codec implemented by FFMPEG is ac3, whatever the kind of environment. And in image, the best codec is JPG as it is the codec that performs the best quality and compression ratio relation.

These results, obtained from the codec tests done in CHAPTER 4, are related to a specific codec implementation done in the FFMPEG transcoding software in its version r16573 (codec library version 52.10.0). These results do not necessary imply that a different codec implementation will perform the same results. It has to be taken into account also that the differences between the content types (static video scenes, action video scenes, rock music content, etc.) also implies differences in the codecs behaviour.

6.1 Environmental impact

The development of this project does not imply an important environmental impact.

For the project development it is only needed three PC's and a deployed network infrastructure. So, the environmental impact is due to these elements: their manufacture process and power consumption, and, at the end of their life, their recycling process.
One of the main points that this project helps in the environment protection is that this project is related to the optimization of multimedia transmissions. This project has allowed a transcoding system to use the best codec in terms of quality and also in terms of compression ratio. This optimization allows performing a correct network use. So, the network capacity is well spent and this fact helps to reduce the need of deploying new network infrastructures.

As this project is related to transcoding processes, it also helps to reduce the use of storage supports like: hard disks, CD-ROMs, DVD-ROMs and any kind of media support. This is because as a multimedia resource is transcoded depending on the user needs it is not necessary to replicate the same multimedia resource in different formats.

6.2 Future work

The near future work is yet a present work. The media analyzer module designed and described in this thesis has been, at the end of its development process, one of the main components of the Multimedia Adaptation Service of the i3media project.

At the present time it is working in a web based media manager that allows a better management of the coded media and the analysis results.

In the near future, one of the working lines is the use of the content classification in MPEG-7, developed in the i3media project, to make a codec classification depending on the content type they codify. A codec that codifies a static content does not behave the same way that the same codec codifying motion content. This way it can be analyzed codecs behaviours, not only depending on the network where the coded content is transmited on, but also depending on the content type that the codec is coding. Like this, the final user experience will be improved.

It is planned to publish an article about the Content Adaptation Service project, which this thesis has apported one of the main modules: the Multimedia Profile Analyzer. A draft of this article can be found in ANNEX.

CHAPTER 7. BIBLIOGRAPHY

In this chapter is listed the consulted reference where some knowledge has been acquired.

[1] i3media: Tecnologías para la creación y gestión automatizada de contenidos audiovisuales inteligentes [Looked up: 15th July 2009]. Available on: <https://i3media.barcelonamedia.org>

[2] MPEG-21: the multimedia framework. [Looked up: 26th June 2009]. Available on: <http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm>

[3] MPEG-7: the standard for description and search of audio and visual content. [Looked up: 26th June 2009]. Available on: <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>

[4] ITU-T, "Recommendation P.800: Methods for subjective determination of transmission quality", International Telecomunications Union – Telecomunication Standarizarion Sector, 1996

[5] Iain E.G. Richardson, "Video Formats and Quality" Cap. 2 in H.264 and MPEG-4 video compression, ed. Wiley pp.15 (2003)

[6] B. Girod, "What's wrong with mean-squared error" in *Digital Images and Human Vision*, A.B. Watson, Ed. Cambridge, pp 207-220 (1993)

[7] Furht and O.Marqure, "Objective Video Quality Assessment" Cap.41 in *The Handbook of Video Databases: Design and Applications* ,ed. CRC Press, pp. 1041-1078 (2003)

[8] S. Daly, "A visual model for optimizing the design of image processing algorithms", *IEEE ICIP*, pp 16-20 (1994)

[9] Z.Wang, L. Lu, A. C. Bovic. "Video Quality Assessment based on Structural Distorsion Measurement", *Signal Processing: Image Comunication*, vol. 19, no. 2, 121-132 (2004)

[10] Z.Wang, A.C.Bovik, Eero P. Simoncelli. "Image Quality Assessment: From Error Visibility to Structural Similarity", IEEE Transactions on Image Processing, vol. 13, no. 4 (2004)

[11] Z.Wang, A.C.Bovik, Eero P. Simoncelli. "Image Quality Assessment Using SSIM Index" in *Image Quality Assessment: From Error Visibility to Structural Similarity*, IEEE Transactions on Image Processing, vol. 13, no. 4, pp 605-606 (2004)

[12] Srivatsan Kandadai, Joseph Hardin, Charles D. Creusere. "Audio Quality Assessment using the mean Structural Similarity measure", IEEE (2008)

[13] ITU-R, “Recommendation BS.1387-1, Method for Objective Measurements of Perceived Audio Quality”, International Telecommunication Union – Radiocommunication Sector, 2001

[14] Gabriel Prieto, Margarita Chevalier, Eduardo Guibelalde, “SSIM index as a Java plugin for ImageJ”, Departament of Radiology, Faculty of Medicine. Universidad Complutense de Madrid. [Looked up: 15th April 2009] Available on:< http://rsb.info.nih.gov/ij/plugins/download/SSIM_index.java>

[15] David Salomon, “Error Metrics” Cap 5.3.1 in *Coding for Data and Compution Communications* ed. Springer, pp.143 (2005)

[16] A.A. de Lima, F.P.Freeland, R.A. de Jesus, “On the Quality Assessment of Sound Signals”, IEEE (2008)

[17] ITU-R, “Recommendation 601, Encoding Parameters of Digital Television for Studios”, International Telecommunication Union, 1982

[18] FFMPEG transcoder [Looked up: 28th July 2009]. Available on: <http://ffmpeg.org>

[19] EAQUAL, ITU-R recommendation BS.1387 implementation [Looked up: 24th May 2009]. Available on: <http://wiki.hydrogenaudio.org/index.php?title=EAQUAL>

[20] W3C, WebServices Activity official web page. [Looked up: 20th June 2009]. Available on: <http://www.w3.org/2002/ws/>

[21] SOAP W3C Recomendation [Looked up: 20th June 2009]. Available on: <http://www.w3.org/TR/soap12-part0/>

[22] Apache Tomcat, Web Applications Server [Looked up: 10th June 2009]. Available on: <http://tomcat.apache.org>

[23] FreeBSD Unix distribution. [Looked up: 25th June 2009] Available on: <http://www.freebsd.org/>

[24] Dummynet, a tool used for testing networks. [Looked up: 25th June 2009] Available on: <http://www.dummynet.com/>

[25] EBU.CH “Tech3253-Sound Quality Assessment Material” [Looked up: 15th July 2009]. Available on: <http://www.ebu.ch/en/technical/publications/tech3000_series/tech3253/index.php>

Escola Politècnica Superior de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

ANNEX

TITLE: Multimedia Quality Assessment

MASTER DEGREE: Master in Science in Telecommunication Engineering & Management

AUTHOR: Oriol Solé Molina

DIRECTOR: Antoni Oller Arcas

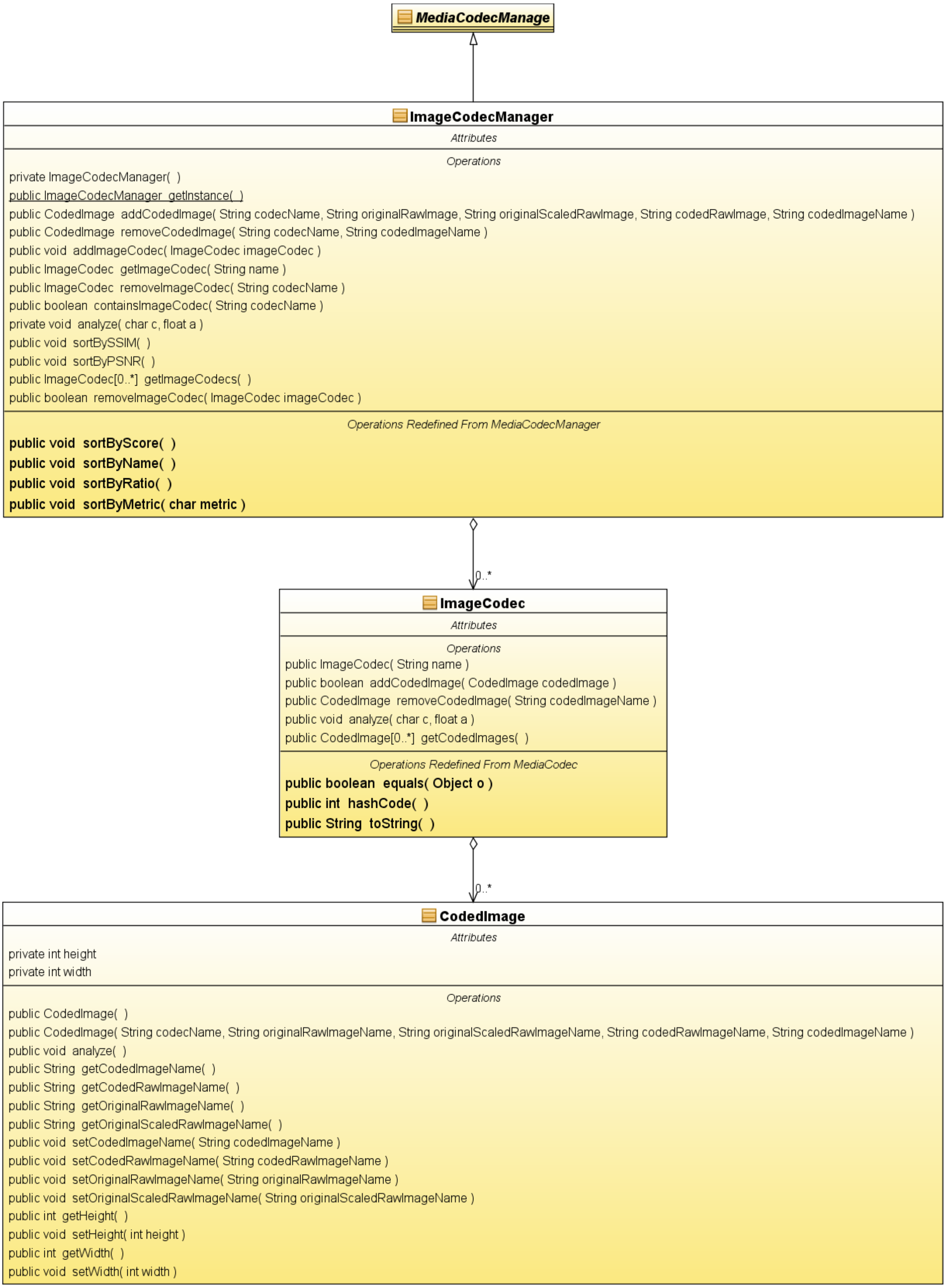
CO-DIRECTOR: Alberto José González Cela

DATE: November 2nd 2009

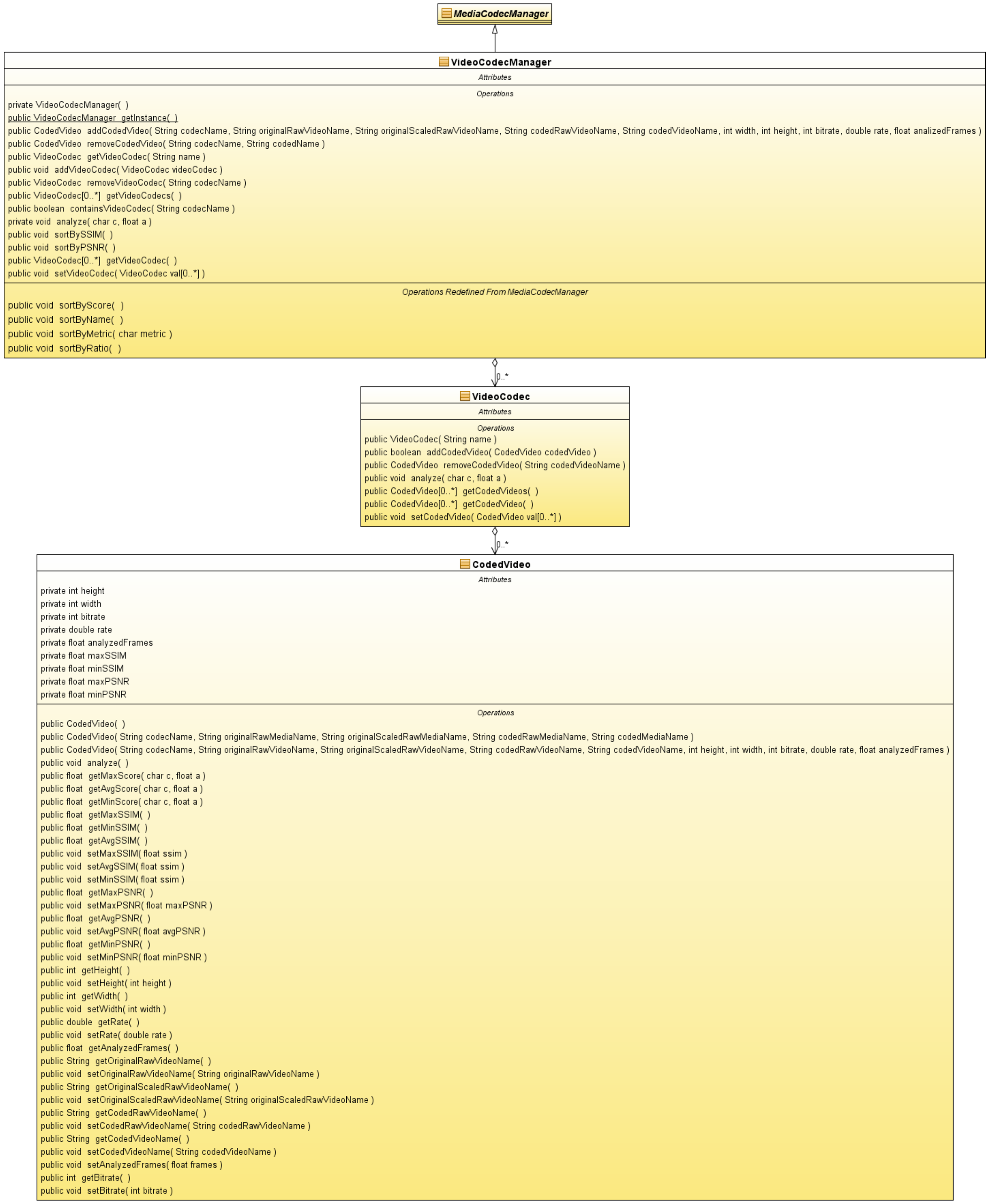
ANNEX 1. Multimedia Analyzer Module class diagrams

MediaCodecManager dependencies class diagram

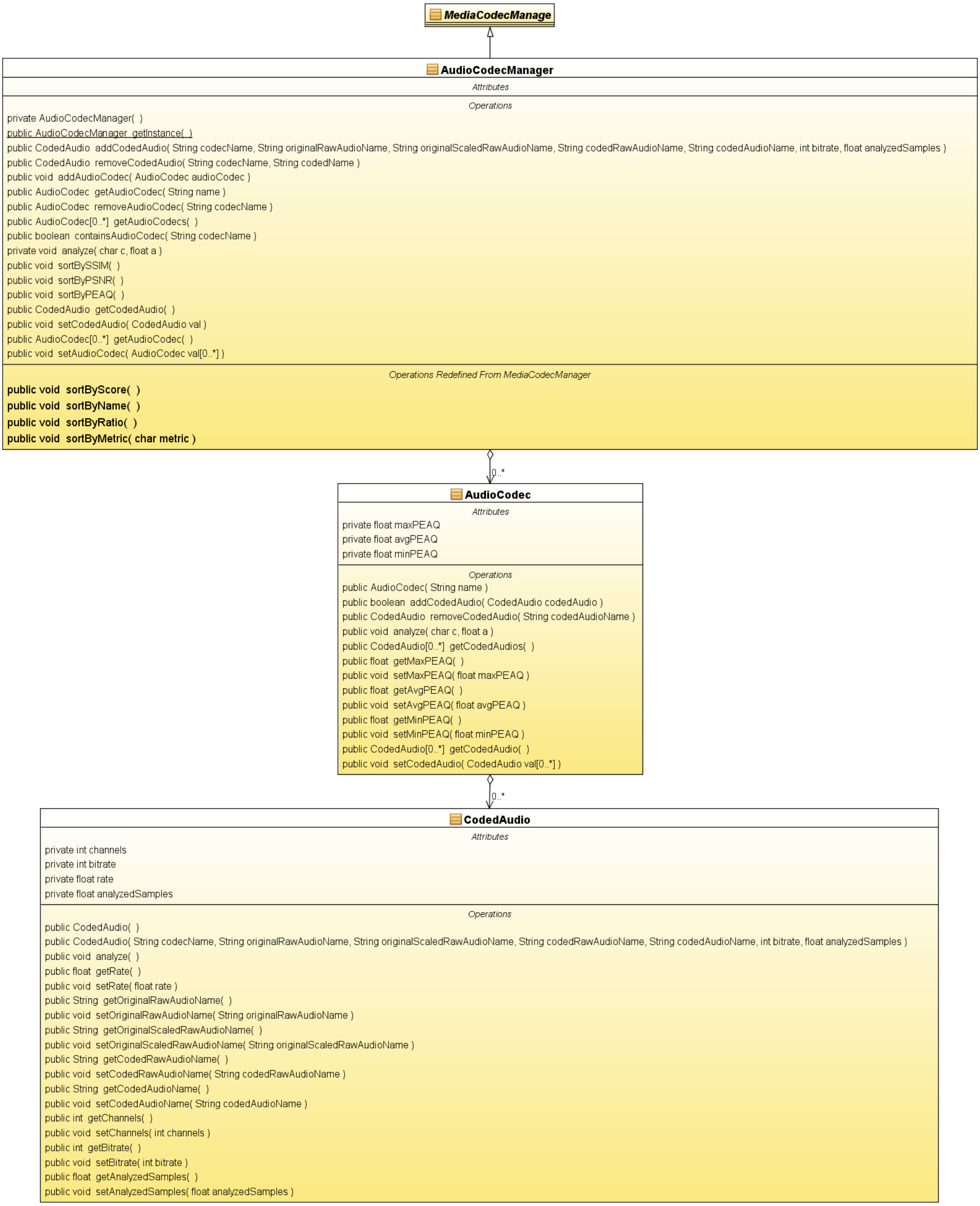
ImageCodecManager dependencies class diagram



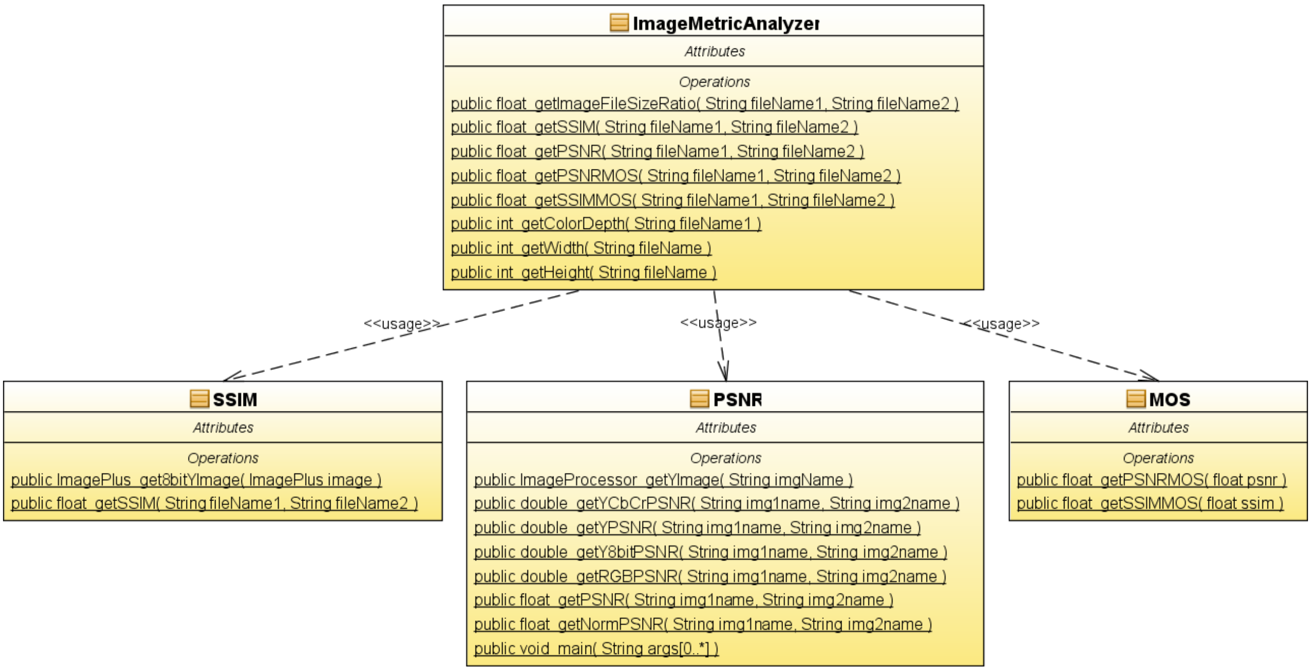
VideoCodecManager dependencies class diagram



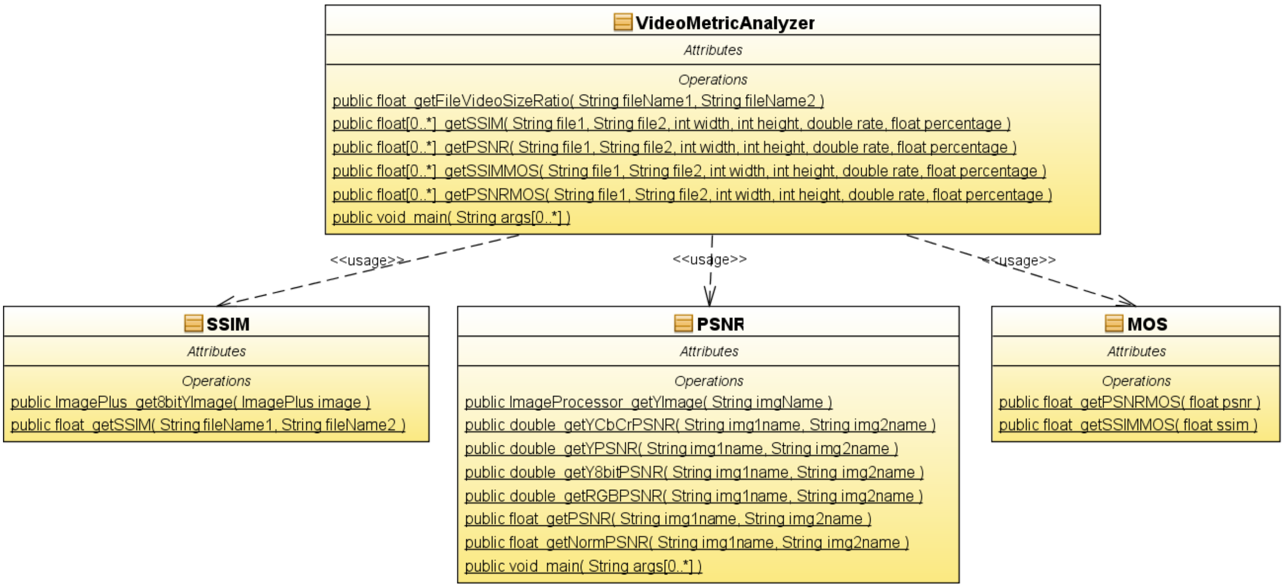
AudioCodecManager dependencies class diagram



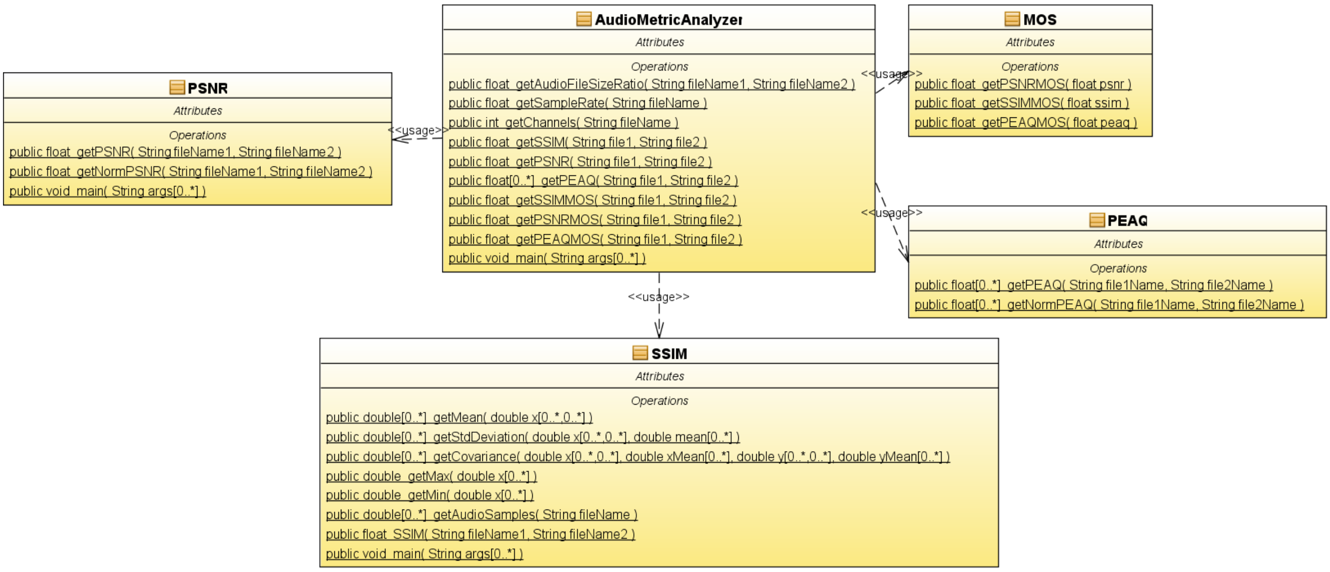
ImageMetricAnalyzer dependencies class diagram



VideoMetricAnalyzer dependencies class diagram



AudioMetricAnalyzer dependencies class diagram



ANNEX 2. Transcoding Service WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
targetNamespace="http://webservice.transcodingmodule.i3media.i2cat">
<wsdl:documentation>TranscodingModuleWS</wsdl:documentation>
<wsdl:types>
<xs:schema attributeFormDefault="qualified"
elementFormDefault="qualified"
targetNamespace="http://webservice.transcodingmodule.i3media.i2cat">

<xs:element name="transcodingVideoRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="iwidth" type="xs:int"/>
<xs:element minOccurs="0" name="iheight" type="xs:int"/>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="ocont" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="vcodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="acodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="owidth" type="xs:int"/>
<xs:element minOccurs="0" name="oheight" type="xs:int"/>
<xs:element minOccurs="0" name="ovfr" type="xs:float"/>
<xs:element minOccurs="0" name="vbitrate" type="xs:int"/>
<xs:element minOccurs="0" name="abitrate" type="xs:int"/>
<xs:element minOccurs="0" name="oachann" type="xs:int"/>
<xs:element minOccurs="0" name="oasf" type="xs:int"/>
<xs:element minOccurs="0" name="sameq" type="xs:boolean"/>
<xs:element minOccurs="0" name="opath" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingVideoRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingVideoRawRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="iwidth" type="xs:int"/>
<xs:element minOccurs="0" name="iheight" type="xs:int"/>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="ocont" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="vcodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="owidth" type="xs:int"/>
```

```
<xs:element minOccurs="0" name="oheight" type="xs:int"/>
<xs:element minOccurs="0" name="ovfr" type="xs:float"/>
<xs:element minOccurs="0" name="vbitrate" type="xs:int"/>
<xs:element minOccurs="0" name="sameq" type="xs:boolean"/>
<xs:element minOccurs="0" name="opath" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingVideoRawRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingUDPVideoRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="iwidth" type="xs:int"/>
<xs:element minOccurs="0" name="iheight" type="xs:int"/>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="ocont" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="vcodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="acodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="owidth" type="xs:int"/>
<xs:element minOccurs="0" name="oheight" type="xs:int"/>
<xs:element minOccurs="0" name="ovfr" type="xs:float"/>
<xs:element minOccurs="0" name="vbitrate" type="xs:int"/>
<xs:element minOccurs="0" name="abitrate" type="xs:int"/>
<xs:element minOccurs="0" name="oachann" type="xs:int"/>
<xs:element minOccurs="0" name="oasf" type="xs:int"/>
<xs:element minOccurs="0" name="sameq" type="xs:boolean"/>
<xs:element minOccurs="0" name="opath" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingUDPVideoRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingImageRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
```

```
<xs:element minOccurs="0" name="ocodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="owidth" type="xs:int"/>
<xs:element minOccurs="0" name="oheight" type="xs:int"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingImageRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingImageRawRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="iwidth" type="xs:int"/>
<xs:element minOccurs="0" name="iheight" type="xs:int"/>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="ocodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="owidth" type="xs:int"/>
<xs:element minOccurs="0" name="oheight" type="xs:int"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingImageRawRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingAudioRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="ocont" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="acodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="abitrate" type="xs:int"/>
<xs:element minOccurs="0" name="oachann" type="xs:int"/>
<xs:element minOccurs="0" name="oasf" type="xs:int"/>
<xs:element minOccurs="0" name="sameq" type="xs:boolean"/>
<xs:element minOccurs="0" name="opath" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>
```

```
<xs:element name="transcodingAudioRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingAudioRawRequest">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="iasf" type="xs:int"/>
<xs:element minOccurs="0" name="ipath" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="ocont" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="acodec" nillable="true"
type="xs:string"/>
<xs:element minOccurs="0" name="abitrate" type="xs:int"/>
<xs:element minOccurs="0" name="oachann" type="xs:int"/>
<xs:element minOccurs="0" name="oasf" type="xs:int"/>
<xs:element minOccurs="0" name="sameq" type="xs:boolean"/>
<xs:element minOccurs="0" name="opath" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:element name="transcodingAudioRawRequestResponse">
<xs:complexType>
<xs:sequence>
<xs:element minOccurs="0" name="return" nillable="true"
type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
</wsdl:types>

<wsdl:message name="transcodingImageRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingImageRequest"/>
</wsdl:message>

<wsdl:message name="transcodingImageRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingImageRequestResponse"/>
</wsdl:message>

<wsdl:message name="transcodingAudioRawRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingAudioRawRequest"/>
</wsdl:message>

<wsdl:message name="transcodingAudioRawRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingAudioRawRequestResponse"/>
</wsdl:message>

<wsdl:message name="transcodingImageRawRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingImageRawRequest"/>
</wsdl:message>
```

```
<wsdl:message name="transcodingImageRawRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingImageRawRequestResponse"/>
</wsdl:message>

<wsdl:message name="transcodingAudioRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingAudioRequest"/>
</wsdl:message>

<wsdl:message name="transcodingAudioRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingAudioRequestResponse"/>
</wsdl:message>

<wsdl:message name="transcodingVideoRawRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingVideoRawRequest"/>
</wsdl:message>

<wsdl:message name="transcodingVideoRawRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingVideoRawRequestResponse"/>
</wsdl:message>

<wsdl:message name="transcodingUDPVideoRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingUDPVideoRequest"/>
</wsdl:message>

<wsdl:message name="transcodingUDPVideoRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingUDPVideoRequestResponse"/>
</wsdl:message>

<wsdl:message name="transcodingVideoRequestRequest">
<wsdl:part name="parameters" element="ns:transcodingVideoRequest"/>
</wsdl:message>

<wsdl:message name="transcodingVideoRequestResponse">
<wsdl:part name="parameters"
element="ns:transcodingVideoRequestResponse"/>
</wsdl:message>

<wsdl:portType name="TranscodingModuleWSPortType">
<wsdl:operation name="transcodingImageRequest">
<wsdl:input message="ns:transcodingImageRequestRequest"
wsaw:Action="urn:transcodingImageRequest"/>
<wsdl:output message="ns:transcodingImageRequestResponse"
wsaw:Action="urn:transcodingImageRequestResponse"/>
</wsdl:operation>

<wsdl:operation name="transcodingAudioRawRequest">
<wsdl:input message="ns:transcodingAudioRawRequestRequest"
wsaw:Action="urn:transcodingAudioRawRequest"/>
<wsdl:output message="ns:transcodingAudioRawRequestResponse"
wsaw:Action="urn:transcodingAudioRawRequestResponse"/>
</wsdl:operation>

<wsdl:operation name="transcodingImageRawRequest">
<wsdl:input message="ns:transcodingImageRawRequestRequest"
wsaw:Action="urn:transcodingImageRawRequest"/>
<wsdl:output message="ns:transcodingImageRawRequestResponse"
wsaw:Action="urn:transcodingImageRawRequestResponse"/>
```

```
</wsdl:operation>

<wsdl:operation name="transcodingAudioRequest">
<wsdl:input message="ns:transcodingAudioRequestRequest"
wsaw:Action="urn:transcodingAudioRequest"/>
<wsdl:output message="ns:transcodingAudioRequestResponse"
wsaw:Action="urn:transcodingAudioRequestResponse"/>
</wsdl:operation>

<wsdl:operation name="transcodingVideoRawRequest">
<wsdl:input message="ns:transcodingVideoRawRequestRequest"
wsaw:Action="urn:transcodingVideoRawRequest"/>
<wsdl:output message="ns:transcodingVideoRawRequestResponse"
wsaw:Action="urn:transcodingVideoRawRequestResponse"/>
</wsdl:operation>

<wsdl:operation name="transcodingUDPVideoRequest">
<wsdl:input message="ns:transcodingUDPVideoRequestRequest"
wsaw:Action="urn:transcodingUDPVideoRequest"/>
<wsdl:output message="ns:transcodingUDPVideoRequestResponse"
wsaw:Action="urn:transcodingUDPVideoRequestResponse"/>
</wsdl:operation>

<wsdl:operation name="transcodingVideoRequest">
<wsdl:input message="ns:transcodingVideoRequestRequest"
wsaw:Action="urn:transcodingVideoRequest"/>
<wsdl:output message="ns:transcodingVideoRequestResponse"
wsaw:Action="urn:transcodingVideoRequestResponse"/>
</wsdl:operation>
</wsdl:portType>

<wsdl:binding name="TranscodingModuleWSSoap11Binding"
type="ns:TranscodingModuleWSPortType">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http"
style="document"/>

<wsdl:operation name="transcodingImageRequest">
<soap:operation soapAction="urn:transcodingImageRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="transcodingAudioRawRequest">
<soap:operation soapAction="urn:transcodingAudioRawRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="transcodingImageRawRequest">
<soap:operation soapAction="urn:transcodingImageRawRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="transcodingAudioRequest">
<soap:operation soapAction="urn:transcodingAudioRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="transcodingVideoRawRequest">
<soap:operation soapAction="urn:transcodingVideoRawRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="transcodingUDPVideoRequest">
<soap:operation soapAction="urn:transcodingUDPVideoRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>

<wsdl:operation name="transcodingVideoRequest">
<soap:operation soapAction="urn:transcodingVideoRequest"
style="document"/>

<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>

<wsdl:output>
```

```
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>

<wsdl:service name="TranscodingModuleWS">
<wsdl:port name="TranscodingModuleWSHttpSoap11Endpoint"
binding="ns:TranscodingModuleWSSoap11Binding">
<soap:address
location="http://demo.i3media.i2cat.net:80/TranscodingModule/services/
TranscodingModuleWS.TranscodingModuleWSHttpSoap11Endpoint/"/>
</wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

ANNEX 3. Automatic media metric analyzer configuration files

Automatic analyzer configuration file

```
#The port where the client will receive the coded media stream
clientPort=1234
#The server IP
serverIP=192.168.48.122
#The webservice descriptor location
wsdl=http://192.168.48.122:8080/TranscodingModule/services/TranscodingModuleWS
?wsdl
#The path where the media will be stored and where they will be analyzed
workingPath=C:\\Documents and Settings\\Oriol\\Escritorio\\proves
```

Media coding parameters configuration file

```
#Original YUV resource URI
resource=http://demo.i3media.i2cat.net/mediaResources/video/foreman_352x288.yuv
#Original YUV resource parameters
width=352
height=288
frameRate=25
type=video
#Percentage of frames to be analyzed. 1 = 100%, 0.5 = 50%, 0 = 0%, etc.
percentage=1
#Number of tests to be done
simulations=2
#Transcoding parameters list
#Format: NUMBER=media container-mediacodec
1=mpeg-mpeg1video
2=mpeg-mpeg2video
3=h263-h263
4=h264-libx264
5=mpeg-mpeg4
6=asf-wmv1
7=asf-wmv2
```

Dummynet configuration commands

```
#destroys all pipes
ipfw pipe flush
#add a new pipe
ipfw add pipe1 ip from any to any
#configure the new pipe with a 100Mbit/s bandwidth and 1% of PLR
ipfw pipe1 config bw 100Mbit/s plr 0.01
```

ANNEX 4. Analyzed multimedia resources characteristics

Lena image resource

Name: Lena or Lenna image

Resolution: 512x512 pixels

Color depth: 24 bits per pixel

Size: 768kbytes

Description:
The image contains smooth color changes, flat regions, shading, and textures

Available on [Looked up:15th July 2009]:
SC-SIPI Viterbi-School of Engineering
http://sipi.usc.edu/database/misc/4.2.04.tiff

Foreman video resource

Name: Foreman

Resolution: 352x288 pixels

Pixel format: 4:2:0 YUV format

Size: 44500 kbytes

Number of frames: 400

Description:
Foreman talks, while rapidly swings around his head. Slight camera movements.

Available on[Looked up:15th July 2009]:
Stanford Center for Image Systems Engineering
http://scien.stanford.edu/Video/foreman.qcif.gz

Vocal quartet audio resource

Name: Quartet
Time duration: 0:28
Channels: stereo
Bits/sample: 16bits
Sampling frequency: 44100Hz

Description:
A vocal quartet singing a melodious phrase.

Available on [Looked up:15th July 2009]:
European Broadcasting Union. Tech 3253-Sound Quality Assessment Material
http://www.ebu.ch/CMSimages/en/tec_sqam_48_bwf_tcm6-12471.wav

Flute instrumental audio resource

Name: Flute melodious phrase
Time duration: 0:17
Channels: stereo
Bits/sample: 16bits
Sampling frequency: 44100Hz

Description:
A flute melodious phrase.

Available on [Looked up:15th July 2009]:
European Broadcasting Union. Tech 3253-Sound Quality Assessment Material
http://www.ebu.ch/CMSimages/en/tec_sqam_13m_bwf_tcm6-12497.wav

ANNEX 5. Multimedia testing results

Video test results

Table A5.1 Video Y-SSIM tests results

Y-SSIM	PLR				
	0,00	0,01	0,03	0,05	0,10
mpeg1	0,91695780	0,70417039	0,69368936	0,67617430	0,59717346
mpeg2	0,91458213	0,74831834	0,71179852	0,70036044	0,65797126
mpeg4	0,92615380	0,66885068	0,52830978	0,45459677	0,39929270
h263	0,91729265	0,69379583	0,54520465	0,49219939	0,40757870
h264	0,94010633	0,70766501	0,50664808	0,43637913	0,37068742
wmv1	0,91991780	0,71669284	0,68402358	0,63692696	0,54020464
wmv2	0,91747560	0,73304226	0,65403923	0,61389979	0,50809010

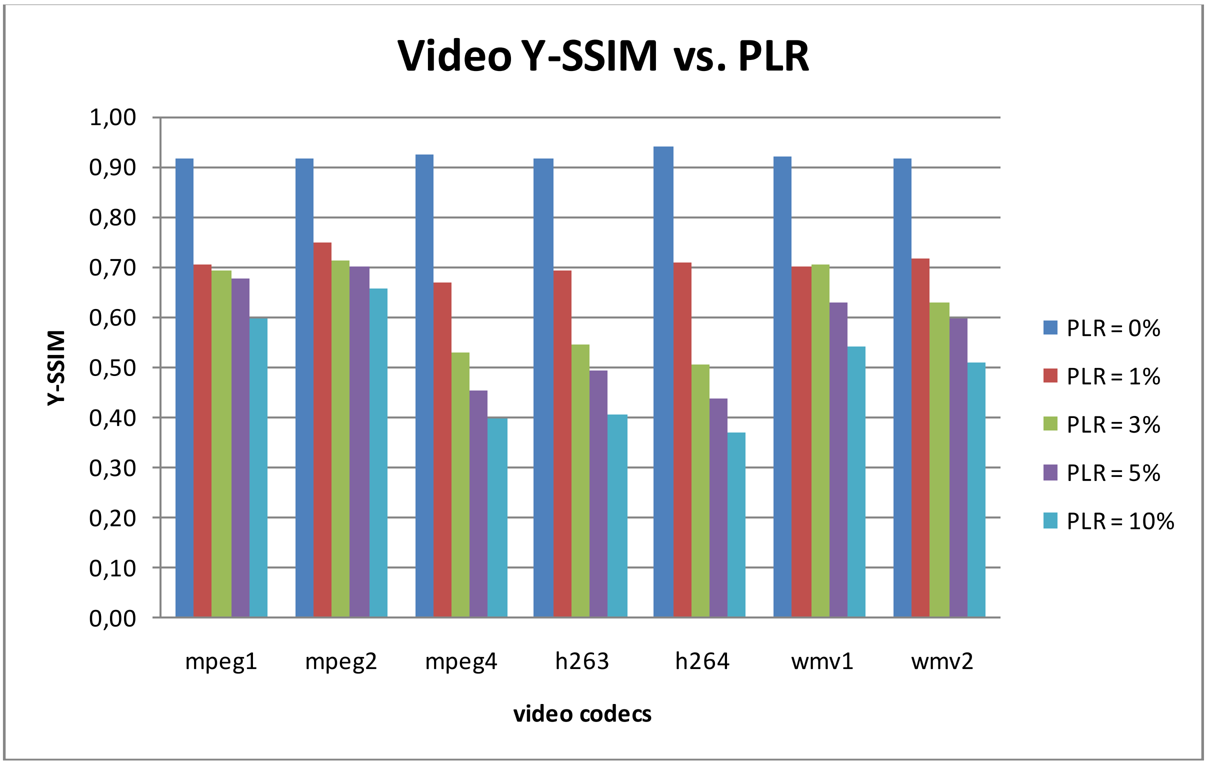


Figure A5.1 Video Y-SSIM versus PLR

Table A5.2 Normalized video Y-PSNR tests results

Normalized Y-PSNR	PLR				
	0,00	0,01	0,03	0,05	0,10
mpeg1	0,30041100	0,21170355	0,19943492	0,19201410	0,16486058
mpeg2	0,29905580	0,22421746	0,20497427	0,20055216	0,18647785
mpeg4	0,30561180	0,20453473	0,15624270	0,13629801	0,12097577
h263	0,29988712	0,21075902	0,16632075	0,14945478	0,12378376
h264	0,30892277	0,21938185	0,15063453	0,12751075	0,10492217
wmv1	0,30142444	0,20336278	0,19611192	0,17689763	0,15362355
wmv2	0,29995555	0,21117605	0,17879258	0,17061182	0,14374800

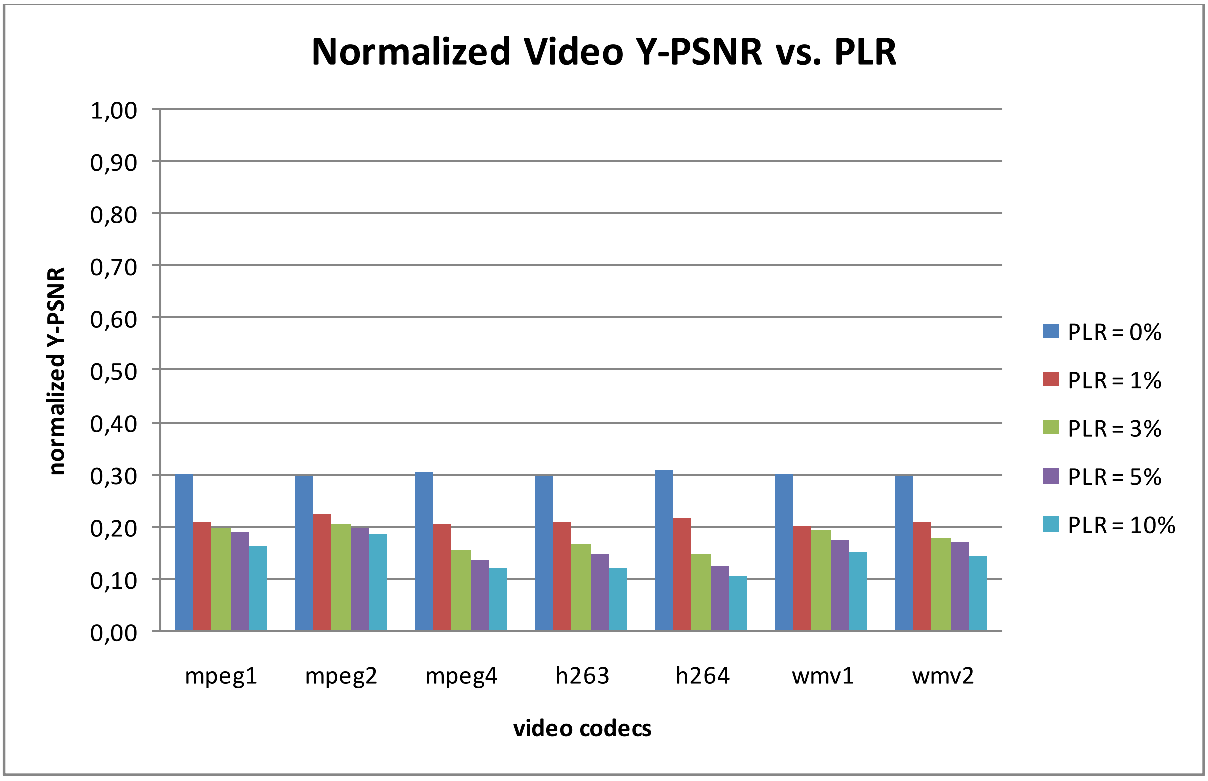


Figure A5.2 Normalized video Y-PSNR versus PLR

Table A5.3 Video codecs compression ratios

	RATIO
mpeg1	0,96552189
mpeg2	0,96552189
mpeg4	0,96581369
h263	0,96583614
h264	0,96650954
wmv1	0,96545455
wmv2	0,96538721

Audio test results

Vocal

Table A5.4 Audio SSIM tests results

SSIM	PLR				
	0	0,01	0,03	0,05	0,1
mp3	0,63471425	0,627628032	0,594227591	0,574190743	0,549333606
aac	0,98695344	0,662909136	0,618937351	0,589474558	0,561671044
ac3	0,66528463	0,644220948	0,618001531	0,605271179	0,577448303
vorbis	0,61935186	0,610675384	0,596813324	0,589165393	0,5651805

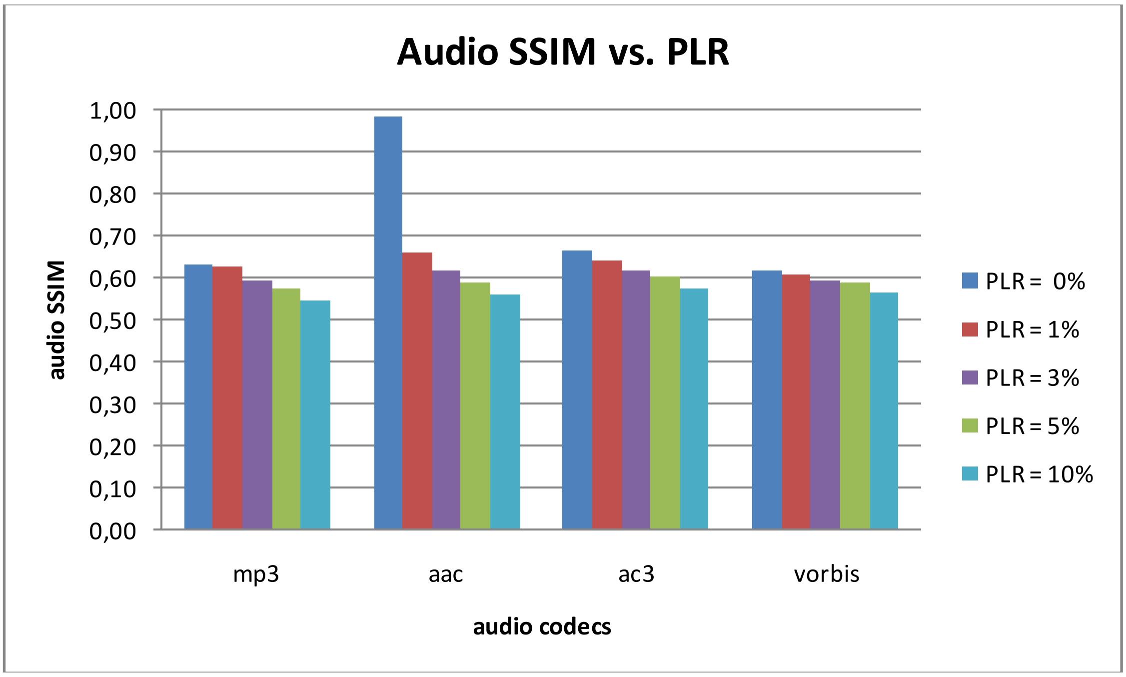


Figure A5.3 Audio SSIM versus PLR

Table A5.5 Normalized Audio PSNR tests results

Normalized PSNR	PLR				
	0	0,01	0,03	0,05	0,1
mp3	0,16935383	0,167534464	0,162427858	0,161477333	0,159916348
aac	0,302007081	0,167303818	0,161751342	0,160690139	0,15913168
ac3	0,162813741	0,161127934	0,160437482	0,159513959	0,159359627
vorbis	0,13366968	0,15830038	0,160155594	0,160824759	0,159241369

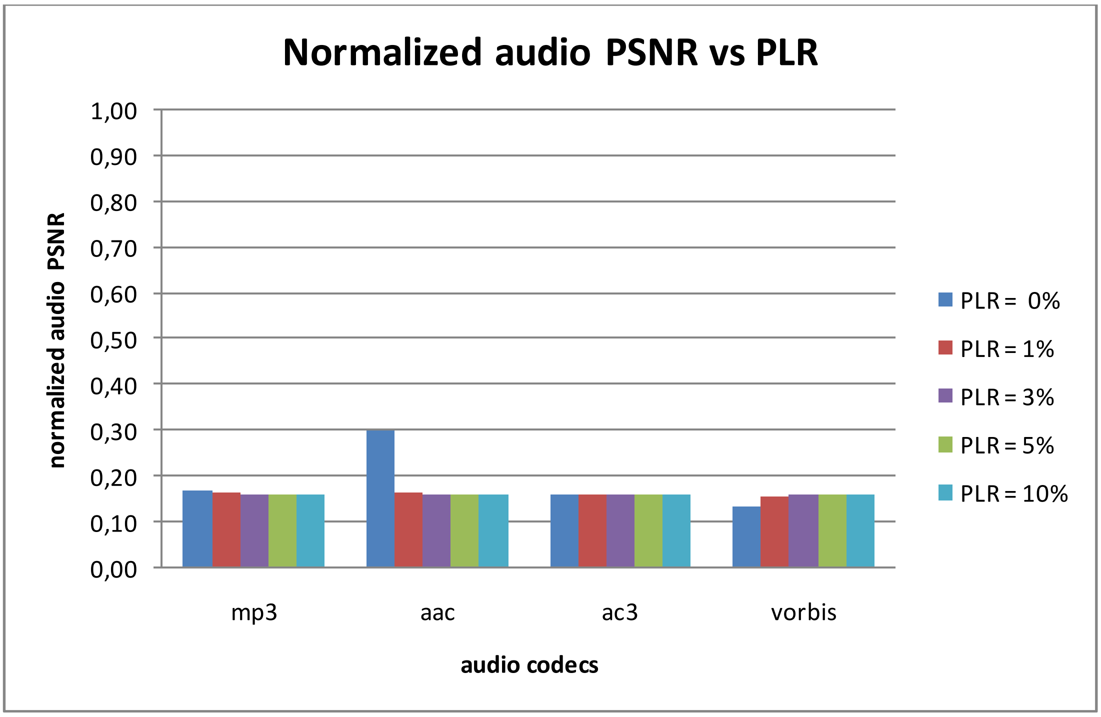


Figure A5.4 Normalized audio PSNR versus PLR

Table A5.6 Normalized audio PEAQ tests results

Normalized PEAQ	PLR				
	0	0,01	0,03	0,05	0,1
mp3	0,268	0,238	0,226	0,225	0,222
aac	0,482	0,250	0,228	0,223	0,220
ac3	0,614	0,355	0,244	0,235	0,226
vorbis	0,294	0,232	0,226	0,223	0,221

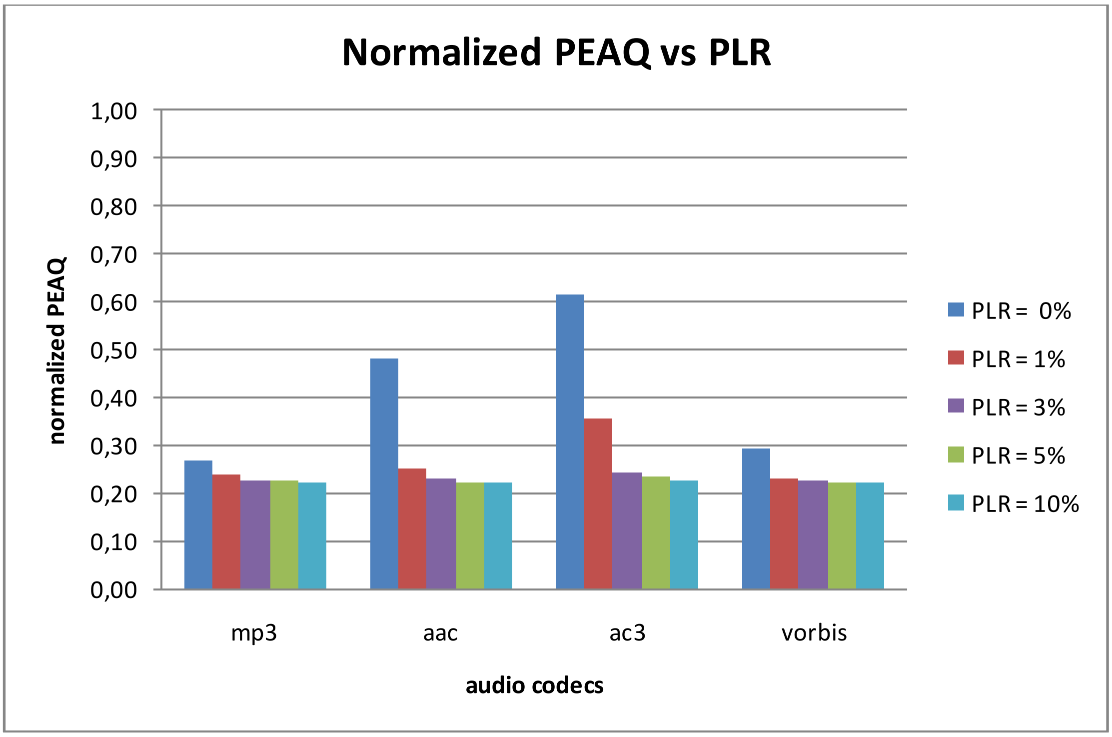


Figure A5.5 Normalized PEAQ versus PLR

Table A5.7 Audio codecs compression ratio (Vocal)

CODEC	RATIO
mp3	0,90870185
aac	0,91107941
ac3	0,90870185
vorbis	0,95815502

Flute

Table A5.8 Audio SSIM tests results

SSIM	PLR				
	0	0,01	0,03	0,05	0,1
mp3	0,7068741	0,766097	0,70013799	0,67330183	0,6046324
aac	0,9878853	0,76709557	0,7045588	0,69046786	0,65949953
ac3	0,7477257	0,7472916	0,70945597	0,71133908	0,68643705
vorbis	0,7179505	0,73568707	0,71689187	0,69224423	0,65185897

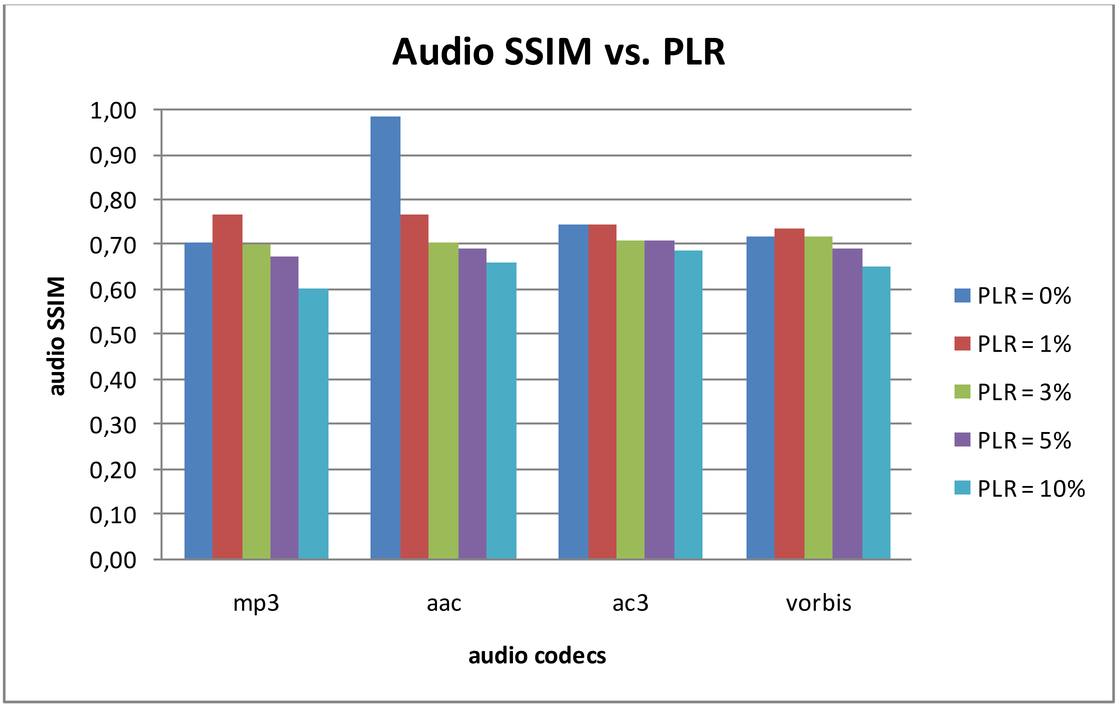


Figure A5.6 Audio SSIM versus PLR

Table A5.9 Normalized audio PSNR tests results

Normalized PSNR	PLR				
	0	0,01	0,03	0,05	0,1
mp3	0,15444623	0,15406328	0,14432742	0,14213587	0,13970882
aac	0,27838562	0,1577229	0,14071731	0,14140693	0,14190276
ac3	0,15268407	0,15390309	0,14493515	0,14180361	0,14184349
vorbis	0,11474599	0,13637053	0,14647104	0,14243502	0,14092426

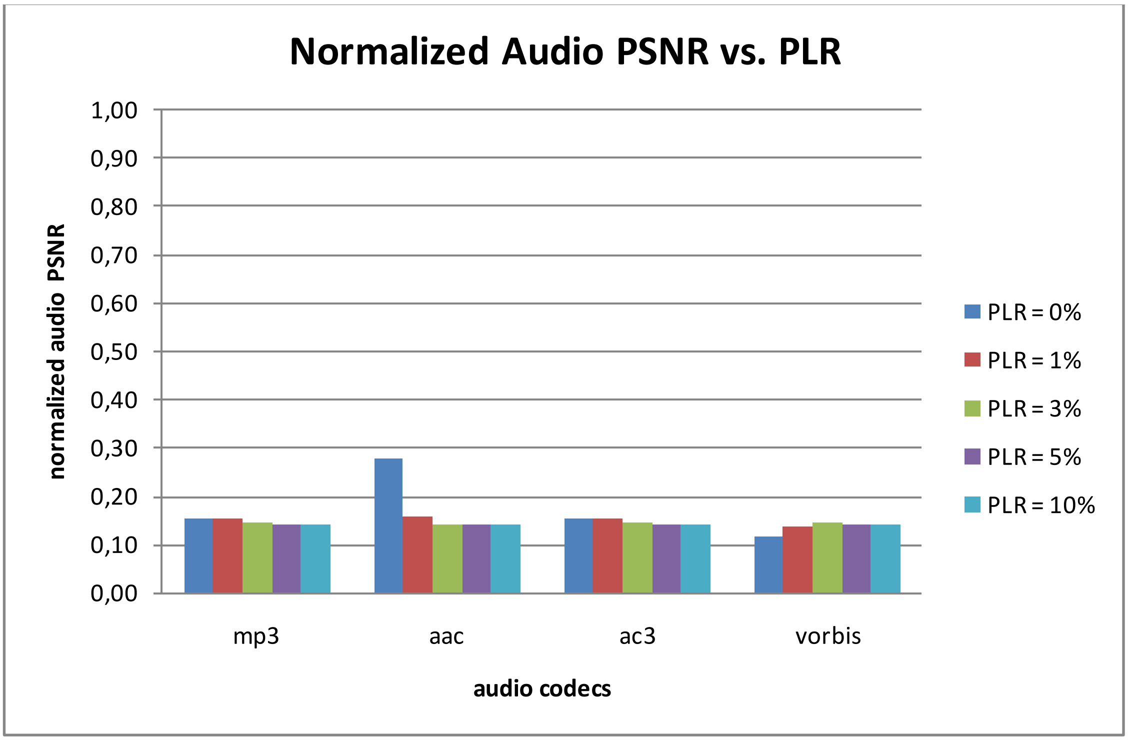


Figure A5.7 Normalized audio PSNR versus PLR

Table A5.10 Normalized PEAQ tests results

Normalized PEAQ	PLR				
	0	0,01	0,03	0,05	0,1
mp3	0,264	0,234	0,222	0,222	0,218
aac	0,504	0,249	0,224	0,221	0,218
ac3	0,582	0,308	0,241	0,236	0,222
vorbis	0,308	0,242	0,223	0,220	0,218

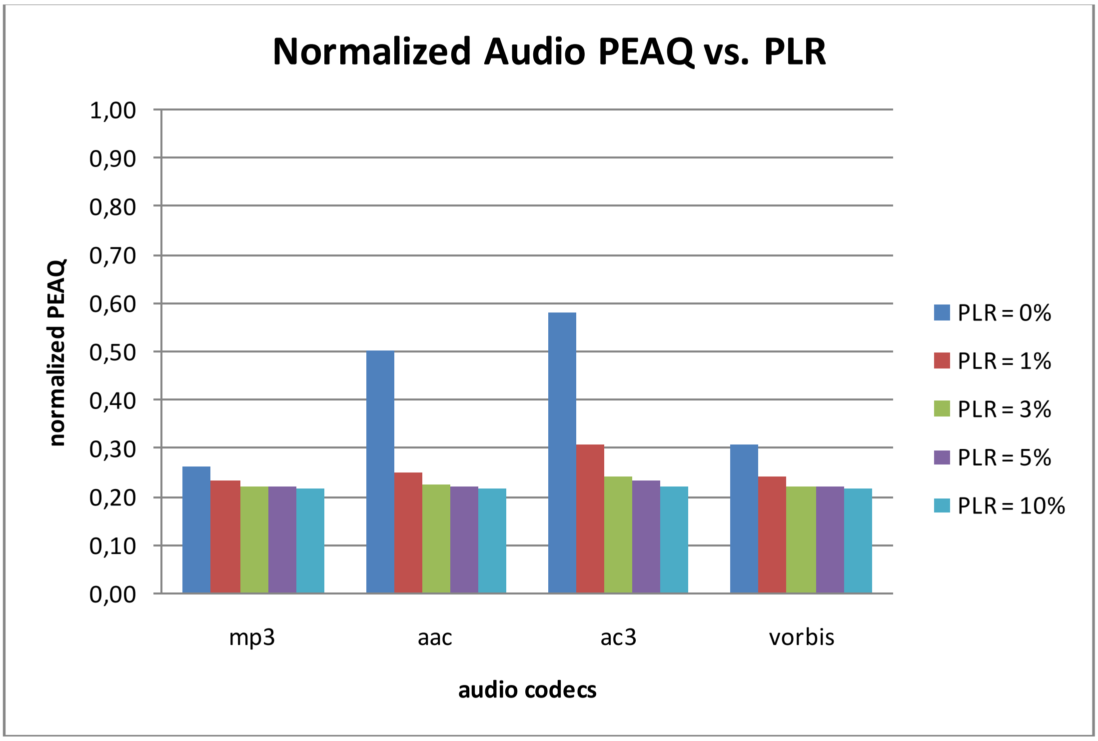


Figure A5.8 Normalized audio PEAQ versus PLR

Table A5.11 Audio codecs compression ratio (Flute)

CODEC	RATIO
mp3	0,90842961
aac	0,91613179
ac3	0,90885751
vorbis	0,96277279

Image test results

Table A5.12 Image test results

	Y-SSIM	RATIO	SCORE
JPG	0,945557	0,9471	0,9463285
GIF	0,840963	0,7656	0,8032815
PNG	1	0,086	0,543
BMP	1	0	0,5

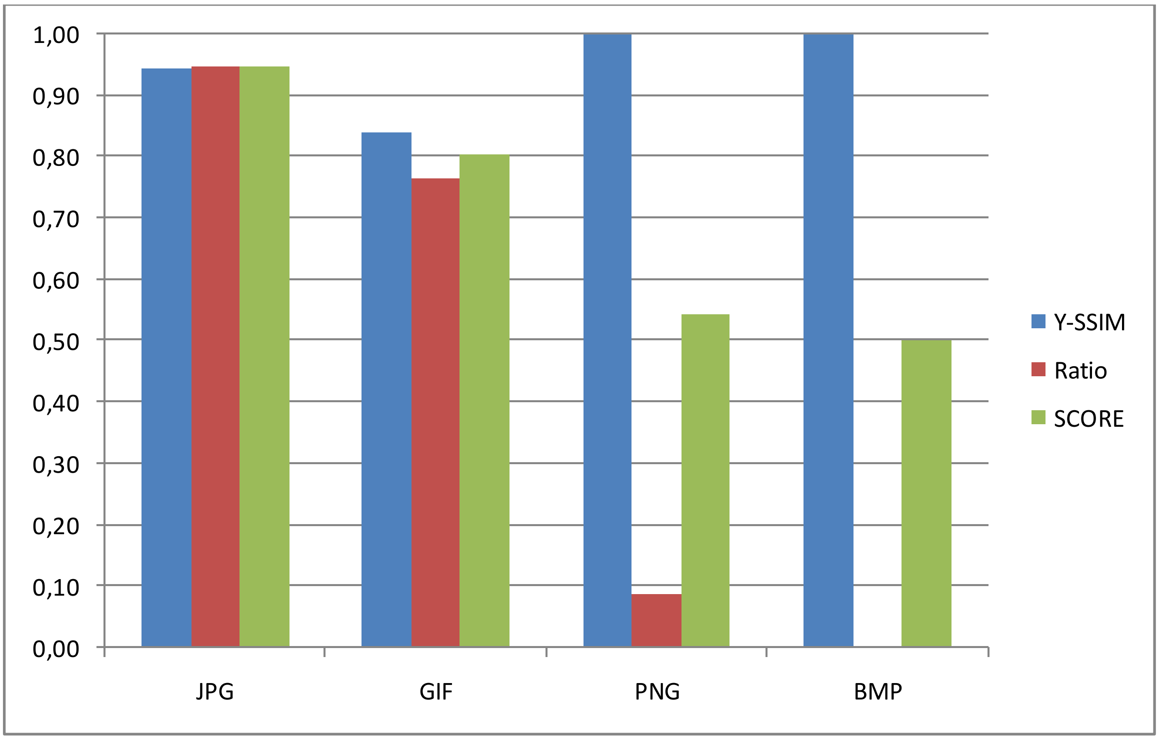


Figure A5.9 Image test results

Context-Aware Multimedia Services In Horizontal Convergent Hosted Service Layer Architecture

Abstract—**In this paper, a convergent Hosted Context-Aware architecture based on NGN is presented. With such Hosted Architecture, the Context-Aware applications are globally deployed and accessible from ubiquitous agents being connected to different access networks through a convergent core control layer. A multimedia Content Adaptation Service is presented as one a Context-Aware application. The multimedia Content Adaptation Service recommends the best multimedia adaptation due to the user context information. A whole architecture is presented and validated, including all logic elements involved and the contextual protocol used to exchange the contextual information among the components. In the last part a validation use case is presented, where the architectural principles described in the first part of the paper are implemented.**

Index Terms— **Next Generation Networking (NGN), IP Multimedia Subsystem (IMS), Session Initiation Protocol (SIP), Multimedia Content Adaptation, Context-Aware.**

I INTRODUCTION

Context-aware applications are one of the coming paradigms in telecommunication commercial environments. Today is possible to obtain a great deal of parameters about the user's context due to the enhanced both hardware and software functionalities of the communication terminals.

The context includes the characteristics of the user's device, the type of network technology being used, the user preferences and the system resources, among others. All this captured contextual information, if processed in an appropriate way, may become the key to deploy, select, manage or enrich different added value applications or services.

So in order to obtain a context-aware system that enables global access to the services, a hosted architecture is proposed. In addition, to process this contextual information and adapt the services to it, a content adaptation service will be also proposed in this paper.

Based on the previous facts, the captured contextual information can be reused at the service layer in a horizontal way. The way to fulfil these requirements is the Next Generation Network (NGN) convergent architecture, in which the service layer is shared among different access network technologies and follows a horizontal structure that allows the mentioned information aggregation and reuse.

This paper is organized as follows. Section II presents foundations of the context and context-aware applications, in section III the application to new multimedia content adaptation service is presented and in section IV a prototype is presented. In section V are presented the results of the multimedia content adaptation service integration with the context enabler architecture, and in Section VI the conclusions are presented.

II. USER CONTEXT AND CONTEXT-AWARE APPLICATIONS

IP Multimedia Subsystem is a new technological option deployed by 3GPP [1] in order to evolve from a vertical network model, designed for a specific range of services, to a horizontal model of unified network capable of supporting the full range of multimedia services imaginable. The IMS architecture gives service providers the opportunity to deliver new and better services, with reduced operating costs, across wireless, wireline, and broadband networks.

One of the most relevant aspects of the Horizontal Service Layer is the concept of Enabler. The Enabler is an entity whose objective is to provide additional information or capabilities to the existing Applications at the Horizontal Service Layer. The Enabler is not a service itself, but it provides added value to the service delivery.

So based on the previous sections, the contextual processing is understood not as a service itself, but as a way to significantly enhance the added value that each individual end-user service may provide through contextual service selection, recommendation, triggering ,and personalization.

Its architecture is shown in picture below (Fig. 1).

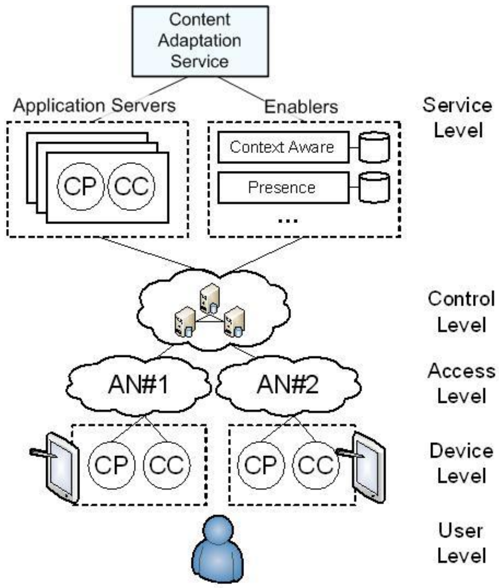


Fig. 1. Contextual Enabler Architecture

This proposed Architecture provides an optimized way to turn all the traditional services into context-aware applications, by defining a Service Layer entity that will centralize the contextual processing. Several contextual entities can be identified in this diagram:

- Context Providers (CP). These entities are capturing and/or reporting contextual information from Device level to Service level. These entities will be logical software agents that have direct interfaces with sensing hardware capabilities in the same or different device in which these agents are running. CPs may also be located at Service Level, located at service platforms that may act also as sources of contextual information. These CP elements will implement the appropriate protocols to progress the captured information to the context aware application.

- Context Consumers (CC). These entities are requesting contextual information in order to provide specific contextual behaviour. They will implement the appropriate interfaces to receive the context information, namely request, receive and process notifications of contextual changes of a specific user or set of users. Those Context Consumers can be associated to SW agents located in contextual services, located either at Device Level or at Service Level. Through these CC, the context-aware services can capture the contextual information and process that to get a specific contextual behaviour.

In order to carry the information among the different contextual entities (CC, CP, CE) is proposed to use a protocol based on IMS, since it shall support the most suitable transport protocol. So, as a signalling protocol is recommended Session Initiation Protocol (SIP)[2] and HTTP as a transport layer. This is advisable due to well known interoperability reasons.

Given this global architectural framework, remarkable research efforts are currently being oriented to obtain optimised services that behave based on user's context. As can be seen, Future Networks shall be aware of context. In this sense, is proposed a new service responsible of the context adaptation. This service can be located in the Service Level of the proposed architecture (Fig. 1).

Three important aspects of context are: where you are; with whom you are; and what resources you are nearby. So, the context includes, but not limited, the following parameters:

- The user context can include user characteristics, user's location, user's preference, and environmental constraint of user (e.g. public are where silence is required, working place, home, etc.).

- The device context can include type and capability of the device.

- The service context can include service availability, required QoS level, and service performance.
- The system resource context can include CPU, memory, processor, disk, I/O devices, and storage.

- The network context can include bandwidth, traffic, topology, and network performance.

So, it is important that Future Networks support the context management to provide customized and optimized context-based services.

III. APPLICATION TO NEW SERVICE: MULTIMEDIA CONTENT ADAPTATION SERVICE

With the broadband network consolidation (cable, ADSL2+, FTH, etc.) the IP services of multimedia contents were born. IPTV and Video on Demand (VoD) are multimedia services in extension process over the network.

New device capabilities brought by new device solutions allows users to access to any available network and to any available service over these networks.

New devices allow users to access to any available broadband network. Broadband networks consolidation allows new multimedia services to be published. Allowing a user consuming multimedia resources wherever and whenever he or she wanted, what is known as Universal Multimedia Access (UMA)[3]. This is one of the added values of these multimedia services.

In this sense the proposed Contextual Enabler Architecture enables this universal service access. As this architecture allows users to access any service in this chapter is described a service architecture that allows a user to consume any available multimedia content.

A multimedia resource can be defined as a multimedia content coded due to specific attributes. This multimedia coded content cannot be decoded by whatever device. A multimedia coded content can be decoded by a device able to understand the multimedia codification process; it means that the device supports the used codec.

As a multimedia resource cannot be consumed by whatever device, because each device has specific capabilities; a multimedia content adaptation service is proposed.

The multimedia content adaptation service is a multimedia service that allows whatever device to consume whatever available multimedia resource accessible in whatever network. This is because the content adaptation service adapts whatever multimedia resource due to the user context: user device, user accessing network, user preferences and user environment.

A. MULTIMEDIA CONTENT ADAPTATION SERVICE ARCHITECTURE

The Content Adaptation Service is a service which main goal is to recommend a content adaptation to any user's device through any network. This service provides to users a Universal Multimedia Access (UMA): consuming multimedia resources wherever he or she was and whenever he or she wanted.

The Content Adaptation Service is provided by a group of elements that provide different functionalities. In Fig. 2 is shown the Content Adaptation Service architecture, where it can be seen the different elements that conforms the service.

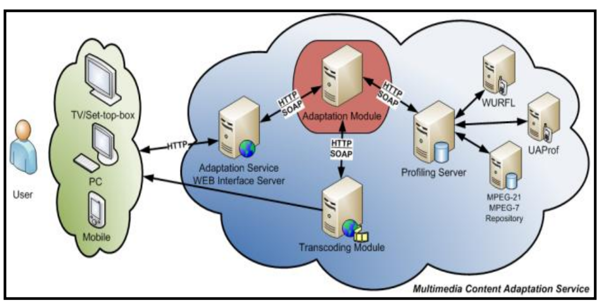


Fig. 2 Multimedia Content Adaptation Service Architecture

As it can be seen in Figure 2, the Content Adaptation Service is performed by different components:

- The Adaptation Module is the core engine. It is the key component. It performs the best adaptation decision taking into account the user's device and the network and content characteristics.

- The Transcoding Module is the element which allows users to obtain the adapted content.

- The Adaptation Service WEB Interface Server contains the Adaptation Service WEB Interface. This interface allows users to access to the Content Adaptation Service and an administrator to realize management tasks.

- The Profiling Server contains a XML Database which contains the multimedia content and user environment (device, network and preferences) descriptors. These descriptors are coded in XML following the MPEG-21 [4] and MPEG-7 [5] standards. The Profiling Server functionality is to provide information to the Adaptation Module related to the content and the user environment (device, network and preferences)

The Adaptation Module is the main element. This element needs the user context information and the multimedia content characteristics. The user context information is obtained from the Contextual Enabler Architecture and the multimedia content characteristics are obtained from Profiling Server.

IV. ARCHITECTURE VALIDATION: USE CASE

In order to validate the integration of the Multimedia Content Adaptation Service with the Contextual Enable Architecture IMS based, it is described a use case.

Margaret is coming back home after a hard working day. She is going by train and she decides to watch her favourite TV series in her mobile device. The train journey ends but her favourite series not. So, once she arrives at home, she decides to continue watching the series, from the scene she left, in her panoramic TV station.

In this situation there are two Context Providers (CP): mobile device and the TV station. Both devices contain an application client that knows some user context information: device location, networks where the device can be connected to, the network where the device is connected, and other device capabilities. The application client works as a CP and sends a notification with the contextual information to the Context Enabler (CE)

The process since Margaret choose the multimedia content in her mobile device until she arrives at home and continue seeing the multimedia content in her TV station is like follows:

1. Margaret starts the mobile application client that works as a CP.
2. The CP sends to the Context Enabler a notification of the contextual information of the device: location, network where is connected to, networks around it and device capabilities. This notification is send through a SIP PUBLISH.
3. While the user is on the train, the CP running on the mobile device periodically generates a notification of the position and sends that information to the Context

Enabler through a SIP PUBLISH with the format presented in Figure 3.


```
PUBLISH sip:CEnabler@coreims2.hi.inet:6070;transport=tcp
SIP/2.0
Call-ID: 16366dbd90fe21f23f80dabf052ebeaa@10.95.29.237
CSeq: 1 PUBLISH
From: <sip:jose@coreims2.hi.inet:6070>;tag=1
To: <sip: CEnabler@coreims2.hi.inet:6070>;tag=567
Via: SIP/2.0/TCP
10.95.29.237:5060;branch=z9hG4bK2c64853cd3a0457bf2bbceb
bfdf8d448
Max-Forwards: 70
Contact: <sip:jose@10.95.29.237:5060;transport=tcp>
Event: presence
Content-Type: text/plain
Content-Length: 134
<?xml version="1.0" encoding="UTF-8"?>
<emergency_data>
<msisdn>15555218135</msisdn>
<imsi>310995000000000</imsi>
<g>1.01</g>
<position>
   <lat>40.869636890588225</lat>
   <lng>-3.822334846993267</lng>
   <cid>44503</cid>
</position>
<speed>0.0</speed>
<status>OK</status>
</emergency_data>
```


Fig. 3. Contextual notification from the Context Provider

4. Every change or novelty in the user context is notified to the Content Adaptation Service with a SIP NOTIFY.
5. When Margaret choose the multimedia content the CP sends a notification to the Content Adaptation Service (SIP SUBSCRIBE).
6. The Content Adaptation Service knows the contextual information of the user device so it adapts the desired content resource due the user device capabilities, access network characteristics and user preferences. The adapted multimedia content stream is sent to the user device. This way Margaret can watch the desired multimedia content adapted to her user context.
7. When Margaret arrives at home the mobile CP notify the context change to the CE. Once Margaret switches on her TV station this CP indicates its contextual information to the CE. The CE indicates these new situations to the Content Adaptation Service (CAS). The CAS realizes the new user context situation, so it readapts the multimedia content resource to the new online user device.
8. Margaret, now at home, can switch off her mobile multimedia application as she is able to continue watching her favourite series through her TV station.

V. RESULTS

In this section, we evaluate the performance of the Content Adaptation Service. We provide efficiency measurements of the motion of the adaptation service over three different scenarios:

- Streaming download by a mobile terminal
- The behavior of the Content Adaptation Service ongoing
- An screen shot of the web service ongoing

The purpose of these tests is to evaluate the resource consumption and analyze the average loading delay of the transcoding profiles and context description, and also, the average response delay of the adaptation process.

A. ADAPTATION SERVICE: MEASUREMENTS OF EFFICIENCY IN AN ENVIRONMENT WITH MULTIPLE USERS

The scenario for this test corresponds to a different number of users making adaptation requests to the Adaptation Service at different intervals of time. For greater realism, the requests for adaptation were made by a random timer that follows a realist patron similar to that made by the users.

The next table (Table 1) shows the four different scenarios used for realize these proofs:

Scenario	Users	Average time between requests
1	1	1 s
2	10	1s
3	20	1s
4	1000	5minutes

Table 1 - Characteristics of the proof scenarios

The purpose of this proof is determining the average response delay of the Adaptation Service when requests for adaptation is received, and quantify the load of the server (RAM and CPU resources).

The next graphic (Fig. 4) shows the response delay of the Adaptation Service in the presented scenarios:

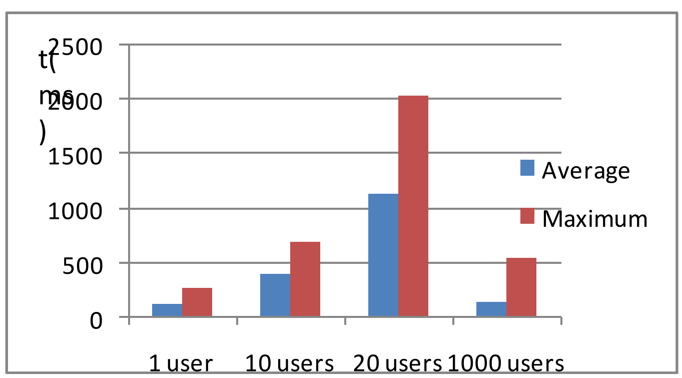


Fig. 4 Time measurements results

The fourth scenario is the most realistic of all. We can see that the average response delay of a request for adaptation is about milliseconds, so is an appropriate value in order to attend the user requirements in sense of its delay perception.

The more relevant processes that increases this delay value are the descriptors acquisition and the loading of the transcoder.

We measured also the consumed resources of the Content Adaptation Service server. The next graphic (Fig. 5) shows these values in an interval of 40 seconds.

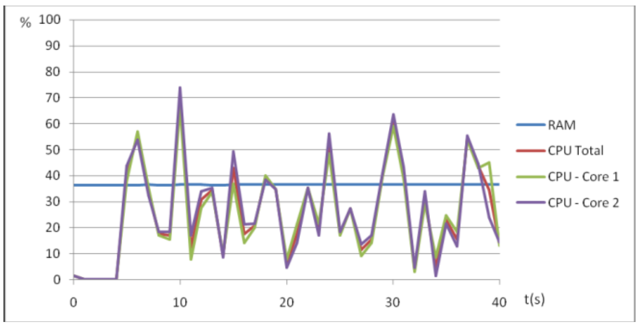


Fig. 5 Resources consume in scenario 4

The RAM consumption gets always the same value, about 35%, regardless the number of requests done to the server. In the other side, it is determined that the CPU consumption is a restrictive factor due that it increases when the number of requests also increases, so it is directly proportional.

B. VALIDATION TESTS OF THE INTERACTION BETWEEN A MOBILE TERMINAL AND THE ADAPTATION SERVICE

In order to validate the performance of the Content Adaptation Service, some test were done. Specifically, the test consisted in requesting adaptation (Fig. 6) and obtaining location parameters (Fig. 7) by a mobile terminal.

The first step is to log in the web service of the Content Adaptation Service. At the welcome page there are five options available:

- Terminal capabilities detection
- Network test
- Content adaptation request
- Media management
- Exit

The third option, content adaptation request, is used to validate the adaptation performance. As it is showed in picture below (Fig. 6), the mobile terminal receives all the parameters of the adapted content that will be sent.

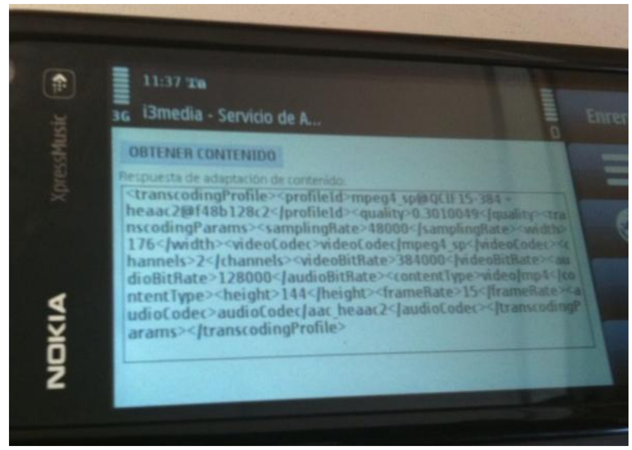


Fig. 6 Adaptation reply on a mobile terminal

The picture below (Fig. 7) shows the location parameters received on the mobile that have been sent by the Adaptation Service web page.

Fig. 7 Location parameters received on a mobile terminal

C. WEB SERVICE PERFORMANCE

Through the web interface the user can directly interact with the Content Adaptation Service. In this web page, once the user is authenticated by user name and password, he or she is able to chose any of these options:

- Show/modify the terminal capabilities
- Request a content adaptation
- Manage the media parameters
- Exit

The possibility of modify the terminal capabilities is offered in order to allow the user to chose what he or she consider better in any case (i.e. if he or she prefers more quality of audio than video).

The picture below (Fig. 8) shows this web interface with the detected parameters of a mobile terminal.

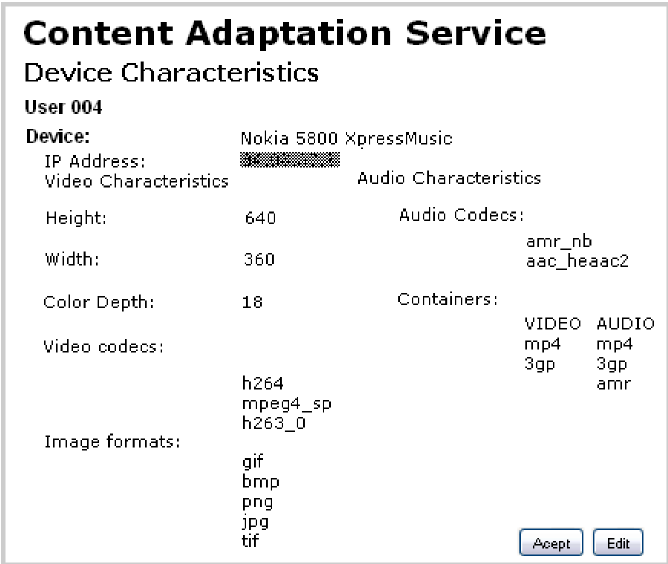


Fig. 8 Screen shot of the Web Service ongoing

As can be seen, it is listed all the necessary characteristics that allows to take a decision about the best multimedia content adaptation due to the terminal characteristics (e.g. screen parameters, audio/video codecs, containers and image formats supported). In the bottom right corner is located the Edit button that allows the user change any of these parameters.

As it has been demonstrated by these tests, the Content Adaptation Service is operating as expected. The passive detection module of terminal capabilities interacts as it expected with the Content Adaptation Module allowing taking the best decision on the content adaptation.

VI. CONCLUSIONS

In this paper a new emerging research and development area is described. Its purpose is to make it technically viable to provide hosted context aware processing for services both new as well as for legacy or traditional ones.

A convergent architecture based on IMS is proposed, justified and demonstrated through a use case implementation. An horizontal service layer scheme is followed, where a Contextual Enabler is proposed, centralizing at the Service Layer the processing of the contextual notifications generated by different Context Providers globally distributed at sensor networks, personal devices or even Service layer. The architecture proposed optimizes the contextual information sharing among the different entities involved and proposes a model to enhance all existing services with context-aware behavior. In this line a demonstrator based on the contextual information captured by a mobile terminal and a fixed terminal is developed: a Multimedia Content Adaptation Service.

REFERENCES

[1] "IP Multimedia Subsystem (IMS), Stage "2, 3GPP Technical Specification TS 23.228.
[2] SIP, RFC 3261 [Looked up: 29th June 2009]. Available on: <http://tools.ietf.org/html/rfc3261>
[3] Universal Multimedia Access (UMA) [Looked up: 29th June 2009]. Available on: <http://uma.epfl.ch/index.html>
[4] MPEG-21: the multimedia framework. [Looked up: 26th June 2009]. Available on: <http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm>
[5] MPEG-7: the standard for description and search of audio and visual content. [Looked up: 26th June 2009]. Available on: http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm