Media Resources Adaptation for Limited Devices

Tayeb Lemlouma, Nabil Layaïda

MEDIA RESOURCES ADAPTATION FOR LIMITED DEVICES

TAYEB LEMLOUMA[1]; NABIL LAYAÏDA[1]

[1] WAM Project, INRIA Rhône Alpes
Zirst 655 Avenue de l'Europe – 38330, Montbonnot, Saint Martin, France
e-mail: Tayeb.Lemlouma@inrialpes.fr; Nabil.Layaida@inrialpes.fr

In this paper, we define a framework for media resources manipulation in an adaptive content delivery system. We discuss the media resources manipulation in an adaptation and negotiation architecture in terms of binary relationship definition using a so-called "related resources" approach, resources extraction, adaptation and delivery. We show how different resources can be related by predefined semantic relationships in order to help structural adaptation and content negotiation tasks. We give a simple recursive algorithm to extract media resources form an input tree structure. The media description is specified using an extensible CC/PP profile. In order to show the benefits of our approach, we give a practical application of media manipulation: an automatic adaptation of XHTML documents for mobile phones. We propose also a general mechanism for the content servers in order to deliver media resources according to client constraints or profile. The influence of media selection in the delivery of multimedia services is also described using evaluation formulae and measured with some experimentations.

Keywords: Media resources manipulation; Content adaptation; XML; CC/PP; Mobile devices; Universal Access

INTRODUCTION

Due to the growing number of exotic networks, hardware and software and the increased demand for using multimedia services comes a tremendous need for providing adaptation and negotiation techniques that end to deliver an understandable content to heterogeneous clients. These last are, generally, characterized by particular constraints and limitations which make the use of rich content difficult. Offering an adaptable content is simply similar to express the same information with two different vocabularies. The two vocabularies are not the same and may be different in the richness and the expression power. In multimedia contexts the vocabulary represents the set of supported functionalities by the environment. The problem of the adaptation becomes difficult when the target vocabulary is poor or limited.

Media resources manipulation represents a great deal of the work concerning the adaptation and the negotiation of multimedia content. Indeed, negotiating the service to be used is equivalent, at the end level of the delivery, to determine which media to deliver, to adapt or simply to remove from the original content. In this paper, we discuss the framework of media resources manipulation in developing an adaptive content delivery system. We focus on how media items will be processed in our multimedia content adaptation and negotiation architecture. The goal is to enable client terminals with special constraints (limited communication, storage, processing and displaying capabilities, etc.) to access to content servers in a heterogeneous multimedia environment.

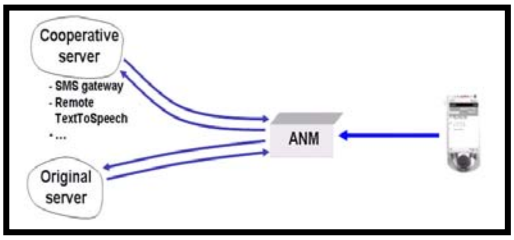
RELATED WORKS

Many recent efforts have focused on adapting multimedia environments to the actual situation. These efforts aim to enable the delivery of adapted content not only according to the client capabilities and limitations but also to the user preferences (language, level of the content detail, font size, etc.). Adaptation techniques can be classified in two main classes: document oriented adaptation and media oriented adaptation. In the first class, we find all the techniques that consider structural adaptations, such as XML-based adaptation [1][2] and those following other models. This kind of adaptation depends widely to the particularities of the handled multimedia document model [3]. Several tools are used to perform these transformations. XSLT [4] represents a declarative language that allows describing rules for transforming XML document into another type of document. Document transformation can also be achieved using other languages like Java or C++ and in many cases using document parsers such as SAX (Simple API for XML) for XML documents, and DOM (Document Object Model) for XML or HTML. Balise [5] and Omnimark [6] represent also transformation languages that allow manipulating and handling document trees. Other techniques are used to output adapted documents, such using JSP in web applications following different approaches [7].

The second class of adaptation techniques concerns media transformations. Most of related research work discusses this kind of adaptation in terms of transformation techniques applied to meet the target context [8][9][10]. Transformations include media compression, resizing, stream control, color depth reduction, etc. and can follow different schemes according to the entity that performs the adaptation: a server [11][12][13], a proxy [14][15][16] or a client [17]. Some media transformations apply textual transformation (e.g. postscript to PDF, etc.)[18][19], image transcoding [20][21], video and audio processing [22][13][23][24], etc. As we can note, efforts belonging to the two precedent classes are independent. Applications are based to only one axe of the content adaptation: the structure or the media. In heterogeneous environments, devices can request any kind of content that varies from poor textual content to rich and mixed complex multimedia documents or presentations. Ensuring more complete adapted delivery to all the community of users, requires efficient techniques that consider both the media and the document structure dimensions. The objective of this paper is to make automatic document adaptation more efficient by binding media and structure handling. The framework discusses basic sides for a complete content delivery: a) the definition of semantic relationships between resources, b) media extraction and profiles creation, c) content adaptation and negotiation that use predefined relationships and d) the final delivery of the adapted content.

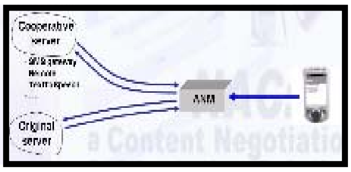
THE APPROACH OF RELATED MEDIA RESSOURCES

In the classical way, to adapt a media resource *M* (image, video, etc.) we should have methods to transform it from its original state to an adapted state. The approach of *related resources* that we introduce here enriches the adapting system by semantic substitution possibilities rather than transformations. Transformation techniques can not be applied in all the cases. For example if the server does not have the appropriate transformation method or simply when the transformation causes a semantic content distortion, such as in images resizing (Figure 1). The role of related resources is to complete the work of the existing transformation methods in order to achieve the adaptation task more efficiently. Related resources can be defined as a set of binary relationships defined between a pair of media resources. A media resource can be an image, a text file, an audio file, a video stream, etc. and can be used by more than one document or application. A resource can be authored locally or imported. It can be also obtained after applying some transformation techniques. A relationship gathers two resources that exist in a server, e.g. <../images/Computer.jpg> *Related-to* <../images/ ComputerScreen.gif>. It is also possible to define a relationship between resources that exist in different servers: e.g. <http://www.server1.com/images/Phone.jpg> *Related-to* <http:// www.server2.com/images/Phone.wbmp> (Figure 2).



(a) Original resource: image of 920x325 pixels

Image Resizing for a PDA (240x320)



(b) Adapted resource: image of 240x85 pixels

FIGURE 1 – CONTENT DISTORTION USING TRANSFORMATION

In a first step, we define such relationship as a value of the following set: {<equivalent-to>, <adapted-to>}. Two resources related by an *equivalent-to* relationship, means that in a multimedia context, they will be treated as alternative content. Generally, this can be found when the two resources keep the same characteristics from the end user point of view. For example, at the server side, the author can define an equivalent relationship between a JPG and a GIF images that represent the same information and use the same resolution. A resource *a* can be related by an *adapted-to* relationship to another resource *b*, if *a* can be substituted by *b* in the final document context. This happens when the two resources represent the same object but with different levels of detail (semantic or presentation level). For example, an *adapted-to* relationship can relate a 256-color video to a 2-color video or simply to a summary text file. The definition of resources relationships depends on two main considerations:

a) Semantic considerations: the definition of relationships is based on the semantic of the original resource, i.e. what does the resource give as understandable information. Related resources represent the same information but with different styles. For example, an original image can represent a title in English while the related image represents the same title in French. This kind of relationships is preferable when the adaptation concerns the user preferences independently to the device characteristics.

b) Presentation considerations: the definition of relationships between resources concern the final presentation of the resource. Such considerations can be related to the format, colors number, size, resolution, etc. So, it is possible to define a relationship between a AVI video and MPEG video, BMP image and SVG image, etc. Defined relationships will be used to deliver a content that respects the device capabilities. For example, the content author can define a relationship between a video that represents a speaker and another video, with a reduced size, that represents only the head of the same speaker. The defined relationship can be used to deliver the video for a device with limited displaying zone. Related resources relationships must be defined at the server side between common resources susceptible to be used by different clients. These definitions are used dynamically by the adapting task and allow accelerating its execution.

When the server starts to adapt a document *D*, the adaptation task is guided by related resources definitions used inside *D*. Resources exploited by *D*, that respect the target context, are directly sent. The definition of related resources is not required between all resources. Indeed, some resources can exist in a single variant and have not another related resource.

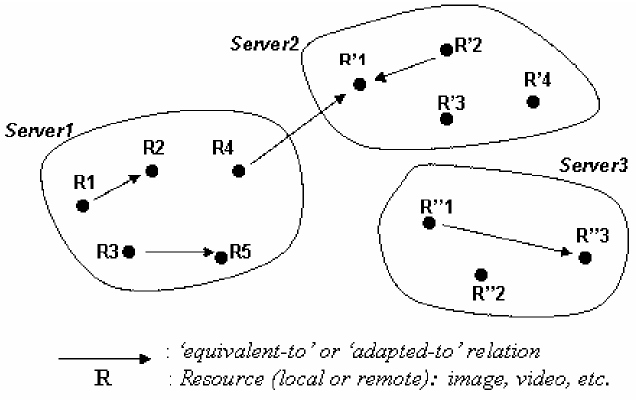


FIGURE 2 – RELATED RESOURCES APPROACH

DOCUMENT INSTANCES

Document profiles describe a class of documents. These last may have some differences from the side of required and used resources. The concept of *document instance* is different to the document class. An instance may use a limited sub set of particular resources compared to the high document class. For example, an instance belonging to the same document class that supports videos can contain just a particular format of video or do not contain videos at all. Considering document instances, at the low level of the content negotiation task, guides more the server delivery. The content delivery becomes more efficient to respond exactly to the need of the client according to the requested resources and documents. Figure 3 gives an overall view of the resources organization in the server side. The instance principle let us thinking about the abstraction level of functionalities and resources used in the definition of content organization. The lower bound of profiling definition is to consider each instance of document as an independent profile, the super bound is to make all existing instances belonging to the same profile class (Figure 3).

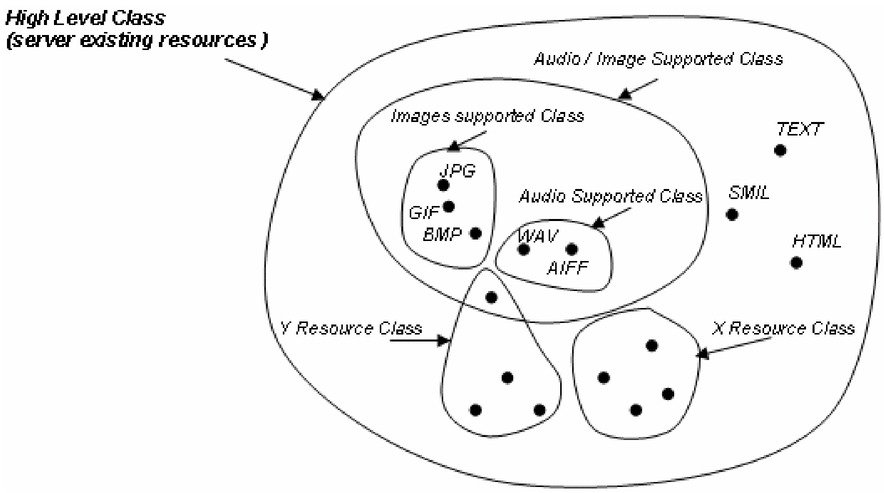


FIGURE 3 – SERVER RESOURCES ORGANIZATION

A good choice is the one that applies an intermediate organization according to the document functionalities and the context constraints. Thus from the profile point of view, a SMIL Basic [25] class for example, contains the list of modules defined is the SMIL Basic profile. A SMIL Basic instance is defined by this list of modules and the detailed set of used resources by the document instance.

THE CONTENT ADAPTATION

Ensuring content adaptation at the end level is equivalent to adapt the general structure of the content and the media resources that it contains. The related media resources approach does not represent a solution based on versioning selections. Its main goal is to complete the work of structural adaptations, and can also serve to build some new transformation processes. To illustrate this we give in the following a part of an XSLT [4] style sheet that adapts XHTML documents for mobile phones.

```
…<xsl:template match="html">
<wml>
<card id="main" title="{head/title/text()}" newcontext="true">
<xsl:apply-templates select="body/node()" />
</card></wml>
</xsl:template>
<xsl:template match="h1">
<big><b><xsl:value-of select="."/></b></big><br/>
</xsl:template>
<xsl:template match="h2">
<big><xsl:value-of select="."/></big><br/>
</xsl:template>
…<xsl:template match="h6">
<small><xsl:value-of select="."/></small><br/>
</xsl:template>
<xsl:template match="img">
<xsl:variable name="image_resource_name">
<xsl:value-of select="concat(substring-before( @src,'.'),'-adapted_to.wbmp')" />
</xsl:variable>
<img src='{$image_resource_name}' alt="[IMAGE: {@alt}]" />
</xsl:template>
<xsl:template match="br">
<br />
</xsl:template>
<xsl:template match="center">
<p align="center">
<xsl:apply-templates/>
</p>
</xsl:template>…
```

FIGURE 4 – AN XSLT ADAPTATION OF XHTML TO WML

The style sheet uses the related media approach in the creation of documents for mobile phones. In addition to the transformation of the document structure (XHTML to WML); the sheet uses the binary predefined relationships between media resources in the substitution of images which exist in the input document. The target of this substitution is the related WBMP images. For mobile phones which do not support images at all, images are substituted by their alternative text. The following figure gives the outputted WML document after adapting an XHTML document using our XSLT style sheet.

FIGURE 5 – THE GENERATED WML CONTENT AS IT RECEIVED BY A WML CLIENT

The following table gives binary relationships used in the adaptation of XHTML to WML:

TABLE 1 – PREDEFINED RELATIONSHIPS BETWEEN MEDIA RESOURCES

Initial resource	Relationship	Target resource
Document.html	adapted-to	document.wml
image.x	adapted-to	Image-adapted_to.wbmp
Text	equivalent-to	Text
H1, h2	adapted-to	Big
h3	adapted-to	none(normal)
H5, h6	adapted-to	Small

The presented principle of the style sheet can be used to cover other content adaptation types. For example, to substitute text to its summary or an audio format to another format such as WAVE to phone resolution voice, etc. Media substitution can also be obtained by real time transformations like text to voice transformations.

CLIENT PROFILE DECLARATION

From the client side, the determination of resources characteristics consists on the creation of a profile which must be conveyed to the content server. The profile must reflect the actual state of the client and depends widely to the requested content. This means that the client profile may change from an application to another (e.g. two browsers can send different profiles), and according to the content intended to be used by theses applications (cHTML, voice XML, etc.). Furthermore, profiles sending can also follow a predefined protocol. For instance, in the first step the client can send only its hardware description (the device description) -which is generally static- and then its software description in terms of media resources. In our approach, the client resources declaration is written in the form of a UPS-CC/PP profile [27]. A client profile contains three main components: two components concerned by the hardware and the software platforms, and a component concerned by the content intended to be used. The content component contains a detailed description about media resources that can exist inside a requested content. The description respects the client characteristics and the set of constraints presented in its environment.

THE CONTENT NEGOTIATION

Media resources should not be sent directly to clients if they do not respect the user context limitations. Resources can be substituted, removed or transformed into another acceptable format. A detailed view of the existing resources is required to achieve properly the content negotiation task. Since knowing the class of the requested document is not sufficient, media resources delivery must be guided by the client description [27]. For example, a client in the form of an Ericsson R320 phone [26] supports the WML-1.3 document class but does not support images alignment. A possible adaptation for such clients can be to avoid providing images with such attribute, or to include images in <*p*> elements with the corresponding value of the alignment attribute. The description form of medias supported by different user agents must follow a precise and extensible expressing way for all the heterogeneous components [27][28].

In the context of heterogeneous environments, HTTP negotiation principles [29] present many limits. The sending of accept headers in every request would be hugely inefficient. Moreover the syntax of user agents' preferences and capabilities is not large and extensible to cover all the diversity of the clients and the existing media resources of servers. In [28], an interaction protocol and basic tools are presented to give a more extensible negotiation solution which can be applied independently to the used protocol (for example, it can be applied using an XML based protocol within or without the HTTP protocol). In our approach, media resources are efficiently described using the CC/PP [30] and the RDF model [31]. CC/PP is simply an expressing tool for describing what the user agent can do. The expressing of media acceptance priorities is opened which makes the server process of content negotiation more efficient and easy to achieve. CC/PP uses RDF which gives a semantic to the CC/PP profiles. So, thanks to RDF a server can understand the profile content and consequently apply appropriate processes. The multimedia content delivery requires resources determination in two sides: a) The requested document instance, and b) The client requirements. To achieve the determination in the first side, the document instance is explored and a CC/PP profile (describing the used resources) is created. From the requirement side, the client conveys its CC/PP profile to the server in order to declare its capabilities and preferences about the resources to be used.

Determining resources from a document instance can be done by exploring tags used in the source document. Only media resources tags are considered and all their attributes are examined. In order to facilitate the exploring process, it is preferable that the source document instance follows an XML structure. Otherwise, intermediate structure transformations can be applied to transform the initial structure to an XML one. The process of exploring the document instances is given by the next following algorithm. The presented recursive procedure shows how to treat a node *n* of the global document tree. All the nodes of the tree are treated in the same way. This process allows the creation of a CC/PP profile that gives a detailed description about the used media resources.

```
procedure Treates_node(n) {
  if (n represents a media resource){
        create an entry in the output CC/PP profile;
        explore n's attributes;
        create media output attributes;
  }else{
        if (n contains other child  nodes)
        for each child s {
              Treats_node(s); }
  }}
```

FIGURE 6 – THE CREATION OF THE MEDIA RESOURCES PROFILE

Exploring a node attributes means to retrieve the maximal information about the media resource, either the information included inside the document or outside it. The exploring algorithm requires the semantic knowledge of all the media nodes or tags that may exist in the source tree of the document instance. A tag semantic depends to the syntax definition used by the document instance. For example, in a SMIL document the <audio …> tag denotes audio clips, <video …> denotes video clips, etc.

We define the description of a media resource *D*, as the union of two sets: a *dynamic* set given by the document instance that uses the resource and a *static* description of the resource which is independent to the document (Figure 7). Formally this can be written as follows: $D = D_{\text{dynamic}} \cup D_{\text{static}}$. The dynamic description is given by the media resource properties included inside the multimedia document. This description can change from a document instance to another. An example of such descriptions: the attributes of an <img …> element used inside a HTML document, such as the height, the width, etc. The static description concerns the properties of the media resource itself independently to its use inside a document. Such descriptions can be the physical dimensions of the resource, its size, the color depth, the resolution, etc. (Figure 7)

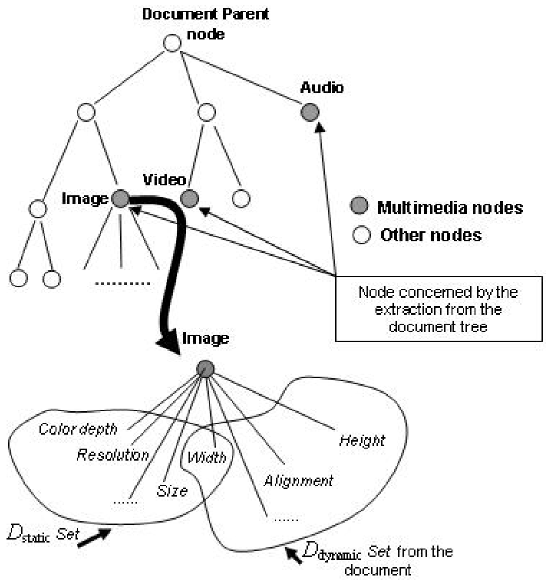


FIGURE 7 – THE DESCRIPTION OF A MEDIA RESOURCE

In the practical situation, client players (browsers) use a static resource property as default, when it is not given explicitly in the requested document. For example, according to the HTML 4.01 specification, a user agent displays an image with its natural dimensions if the <img …> element does not explicit the height and the width attributes. The algorithm of resource profiles creation is applied at the server side. This implies that the server must be able to extract the resource attributes either by storing them when the media resource is created or using real-time methods that allow retrieving resources characteristics.

THE FINAL MEDIA RESOURCES DELIVERY

The general form of the server algorithm applied for the media resources delivery is given as follows:

```
for each
(media resource category X existing in the profile of the  requested document) {
if
(all attributes and elements responds to the corresponding resource category X' (existing in the client profile)){
 deliver X;
}else{
 look for X-related resources (equivalent-to or adapted-to);
 evaluate the resources;
 if (an appropriate resource exist){
 deliver it;
 }else{
 look for available methods to adapt X according to X'
 constraints;
 if (such methods exist){
   apply the method on X;
   deliver the result;
   }else remove X;
  }}}
```

FIGURE 8 – THE ALGORITHM OF MEDIA RESOURCES DELIVERY

As we have already stated, media resources included in different documents should not be sent directly to clients if they do not correspond to their needs and do not satisfy their constraints. They can be substituted, removed or transformed to an acceptable format using available adaptation methods. The delivery of media services can depend to other parameters such as the network constraints. For example a media resource can be substituted by the *adapted-to* related resource, if this last consumes less the bandwidth. The extraction of media resources helps to evaluate locally the effort to apply before the final content delivery and consequently to choose the best resource and the best method to maintain a good QoS. To see better this, we try in the following to evaluate the relationship between media resources and time delivery.

We suppose that the original content C contains n media resources MR and assume that fetching media time is negligible. A media resource can exist in a single or many versions. The document file which contains the global content (e.g. a HTML file) and which declares the used objects is considered as a media resource too.

Let MR_i^v be the v^{th} version of the i^{th} media resource, $Size(MR_i^v)$ be the size of MR_i^v, $Transformed_k(MR_i^v)$ be the media resource obtained after applying a transformation method k on MR_i^v, $T\_Transformation_k(MR_i^v)$ be the time necessary to transform MR_i^v using the transformation method k.

The size of the content C before applying the adaptation can be given by:

$$Size = \sum_{i=1}^{n} Size(MR_i^v)$$

The size of the adapted content equals to: $Size' = \sum_{i=1}^{l} Size(MR_i^v) + \sum_{i=l+1}^{n} Size(Transformed_k(MR_i^v))$

Where media resources from l+1 to n are adapted using transformation methods. Thus the difference of the size between the original and the adapted document is:

$$\Delta Size = \sum_{i=l+1}^{n} (Size(MR_i^v) - Size(Transformed_k(MR_i^v)))...(\mathrm{I})$$

If B_A is the average network bandwidth and D_{RTT} is the network roundtrip time, the delivery time of the adapted content of C is:

$$D\_Time = 2.D_{RTT} + \frac{Size'}{B_A} + \sum_{i=l+1}^{n} T\_Transfomation_k(MR_i^v)...(\mathrm{II})$$

The gain of the delivery time equals to:

$$G\_Time = \frac{\Delta Size}{B_A} - \sum_{i=l+1}^{n} T\_Transfomation_k(MR_i^v)...(\mathrm{III})$$

The following figure shows an example of different resources (images) that represent the same content. Resources b, c and d are created on the fly at the server side, by applying a JPEG compression of the resource a. Resources are delivered to the client using a Java Servlet.


Compression of 0%


Compression of 50%

Compression of 80% Compression of 90%

FIGURE 9 – THE SERVER MEDIA TRANSFORMATION (IMAGE COMPRESSION)

The formula (III) of the delivery time gain shows clearly the relationship between media resources adaptation and the delivery time of the final content. The result of this evaluation is useful for the content negotiation strategy applied by the server before the final content delivery. In order to respect the constraints of the heterogeneous environment (clients limitations, servers and network capabilities), the server can judge if resources adaptations or media substitution by existing related resources are necessary or not. In some cases, applying resources adaptation used inside a document can increase the time of documents delivery which decreases the system performances. Figure 10 shows the relationship between the delivery time gain and the network bandwidth, with a = $\Delta Size$ and b = $\sum_{i=l+1}^{n} T\_Transformation_k(MR_i^v)$.

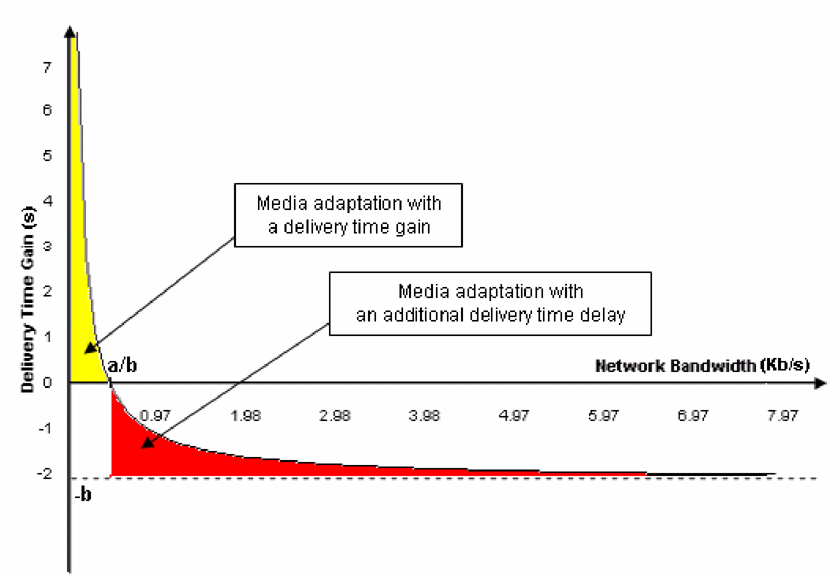


FIGURE 10 – CONTENT DELIVERY AND NETWORK BANDWIDTH

As we can see in the figure above, the delivery time gain remains positive while the network bandwidth value is less than a/b. This result can be interpreted in our context as follows: resources transformation offers a gain of time when the bandwidth is less than a/b. Otherwise the delivery of the adapted resources used inside a document will take more of time than the delivery of the original content. Simply, this result shows that when the bandwidth decreases, it will be more efficient to apply adaptation methods to deal with the network situation.

CONCLUSIONS

Media adaptation has a vital importance in the global architecture of content adaptation and negotiation. We have presented in this paper an extensible framework for media resources manipulation in heterogeneous environments. Media manipulation was discussed in terms of semantic relationships definition and adaptation techniques. In order to show how media resources can be processed in practical cases using these concepts; a useful application was given: XHTML adaptation for mobile phones. We have shown also the influence of the media selection and transformation in the content delivery. This was done using an evaluation formula that we have given and that evaluates the delivery time of the multimedia content. This evaluation is benefic because it shows when the content adaptation starts to decrease the system efficiency and when it remains utile concerning to the delivery performance.

The framework that we have defined should facilitate the achievement of the two main tasks of an adapting multimedia system, which are the multimedia content adaptation and negotiation. The definition of relationships between media resources requires sometimes a human knowledge about the semantic of the resource, but makes the content adaptation automatic and more efficient for heterogeneous devices access.

NOTES AND REFERENCES

1 BOLL S., KLAS W., and WANDEN J. A Cross-Media Adaptation Strategy for Multimedia Presentations. *Proc. Of the ACM Multimedia'99*, Orlando, Florida, USA, 30 October - 5 November 1999.

2 VILLARD L., ROISIN C. and LAYAÏDA N. An XML-Based Multimedia Document Processing Model for Content Adaptation. Opera Project, INRIA Rhône-Alpes, Digital *Documents and Electronic publishing (DDEP00)*, ed, LNCS, September 2000.

3 BOLL S., KLAS W., and WESTERMANN U. A Comparaison of Multimedia Document Models Concerning Advanced Requirements. *Technical report, University of Ulm*, Germany, http://www.informatik.uni-ulm.de/dbis/Cardio-OP/publication/TR99-01.ps.gz, February 1999.

4 W3C. XSL Transformations (XSLT) Version 1.0. *W3C Recommendation*, http://www.w3.org/TR/1999/REC-xslt-19991116, November 1999.

5 BALISE. Balise 4. http://www.us.balise.com/products/balise/index.htm, 2000.

6 OMNIMARK. Guide to Omnimark 5. http://www.omnimark.com/develop/om5/docs/, 2000.

7 SUN MICROSYSTEMS. Java, JSP and XML. http://java.sun.com/products/jsp/html/JSPXML.html

8 FOX A., GRIBBLE S.D., BREWER E.A. and AMIR E. Adapting to Network and Client Variability via On-demand dynamic Distillation. *Proc. 7th Intl. Conf. On Arch. Support for Prog. Lang. and Op. Sys. (ASPLOS-VII)*, Cambridge MA, October 1996.

9 FOX A. and BREWER E.A. Reducing WWW Latency and Bandwidth Requirements by Real-Time Distillation. In *Proc. of the 5th International WWW Conference*, 1996.

10 CHANG S-F., ELEFTHERIADIS A., ANASTASSIOU D., JACOBS S., KALVA H. and ZAMORA J. Coulmbia's VOD and Multimedia Research Testbed with Heterogeneous Network Support. *International Journal of Multimedia Tools and Application*, Kulwer Academic Publishers, September 1997.

11 INOUYE J., CEN S., PU C. and WALPOLE J. System Support for Mobile Multimedia Applications. *Proc. of the 7th NOSSDAV*; May 1997.

12 BHARGHAVAN V., KANG-WON L., SONGWU L., SUNGWON H., JIN-RU L., and DWYER D. The TIMELY Adaptive Resource Management Architecture. *IEEE Personal Communications*; August 1998; 5(4):20-31.

13 BOLOT J.C. and TURLETTI T. Experience with Control Mechanisms for Packet Video in the Internet. *Computer Communication Review*; ACM, January 1998; 28(1):4-15.

14 MARGARITIDIS M. and POLYZOS G.C. Adaptation Techniques for Ubiquitous Internet Multimedia. *Wireless Communications and Mobile Computing Journal*, no. 1, vol. 1, John Wiley & Sons, 2001.

15 SATYANARAYANAN M., NOBLE B., NARAYANAN D., TILTON JE., FLINN J., WALKER KR. Agile Application-Aware Adaptation for Mobility. *Proc. of the 16th ACM Symposium On Operating Systems Principles*; December 1997; 31(5):276-287.
16 BHARGHAVAN V., GUPTA V. A Framework for Application Adaptation in Mobile Computing Environments. *Proceedings of the IEEE Compsoc'97*; November 1997.
17 MARGARITIDIS M. AND POLYZOS G.C. Adaptation Techniques for Ubiquitous Internet Multimedia. *Wireless Communications and Mobile Computing Journal*, no. 1, vol. 1, John Wiley & Sons, 2001.
18 HOUSEL BC., LINDQUIST DB. WebExpress: A System for Optimizing Web Browsing in a Wireless Environment. *MOBICOM'96 Demonstration Session*; November 1996.
19 LILJEBERG M., ALANKO T., KOJO M., LAAMANEN H. and RAATIKAINEN K. Optimizing World Wide Web for Weakly Connected Mobile Workstations: An Indirect Approach. *Proc. of the 2nd International Workshop on Services in Distributed and Networked Environments* 1995; 132-139.
20 ORTEGA A., CARIGNANO F., AYER S. and VETTERLI M. Soft Caching: Web Cache Management for Images. *In IEEE Signal Processing Society Workshop on Multimedia*, Princeton, NJ, June 1997.
21 SMITH J. R., MOHAN R. and Li C.-S. Content-based transcoding of images in the Internet. In *IEEE Proc. Int. Conf. Image Processing (ICIP)*, Chicago, Il, October 1998.
22 COWAN C., CEN S., WALPOLE J., PU C. Adaptive Methods for Distributed Video Presentation. *ACM Computing Surveys*, 27(4), pp. 580-583, Symposium on Multimedia, December 1995.
23 JACOBS S. and ELEFTHERIADIS A. Real-time dynamic rate shaping and control for internet video applications. *Workshop on Multimedia Signal Processing*, pp. 23–25, June 1997.
24 CHRISTIANSON L. and BROWN K. Rate Adaptation for Improved Audio Quality in Wireless Networks. *The 6th IEEE International Workshop on Mobile Multimedia Communications MOMUC'99*, San Diego, California, USA, 15 - 17 November 1999.
25 KUBOTA K. and COHEN A. SMIL Basic Language Profile. http:// www.w3.org/TR/2000/WD-smil-boston-20000622/smil-basic-profile.html, 22 June 2001.
26 Ericsson mobile communications. Mobile Phone R320 Design Guidelines for WAP Services. *Ericsson mobile communications AB*, January 2000.
27 LEMLOUMA T. and LAYAÏDA N. Universal Profiling for Content Negotiation and Adaptation in Heterogeneous Environments. *W3C Workshop on Delivery Context*, W3C/INRIA Sophia-Antipolis, France, 4-5 March 2002.
28 LEMLOUMA T. and LAYAÏDA N. The Negotiation of Multimedia Content Services in Heterogeneous Environments. In *the 8th International Conference on Multimedia Modeling*, CWI, Amsterdam, The Netherlands, 5-7 November 2001.
29 HOLTMAN K., TUE and MUTZ A. Transparent Content Negotiation in HTTP. RFC 2295, *Network Working Group*, March 1998.
30 W3C. Composite Capabilities / Preferences Profiles. http://www.w3.org/TR/CCPP-struct-vocab/, 15 March 2001.
31 W3C. Resource Description Framework (RDF) Model and Syntax Specification. *W3C Recommendation*, http://www.w3.org/TR/1999/REC-rdf-syntax, 22 February 1999.