Content Interaction and Formatting for Mobile Devices

Tayeb Lemlouma
WAM Project, INRIA Rhône-Alpes
Zirst - 655 avenue de l'Europe
38334 Montbonnot, France
Tayeb.Lemlouma@inrialpes.fr

Nabil Layaïda
WAM Project, INRIA Rhône-Alpes
Zirst - 655 avenue de l'Europe
38334 Montbonnot, France
Nabil.Layaida@inrialpes.fr

ABSTRACT

In this paper we present an experimental content adaptation system for mobile devices. The system enables the presentation of multimedia content and considers the problem of small screen display of mobile terminals. The approach combines structural and media adaptation with the content formatting and proposes a system that handles the user interaction and the content navigation.

Categories and Subject Descriptors

I.7.2 [**Document and Text Processing**]: Document Preparation – *Multi/mixed media*, *Markup languages*, *Hypertext/hypermedia*, *Standards*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *interaction styles*, *graphical user interfaces*, Hypertext/Hypermedia – *navigation, architectures*.

General Terms

Modeling, Experimentation, Design.

Keywords

Content adaptation, content formatting, user interaction, mobile-devices, evaluation.

1. INTRODUCTION

Enabling the access and the presentation of multimedia content on limited environments represents a real challenge in currents mobile environments. Mobile devices platforms are subject to several and various constraints which make the design of adaptable multimedia architectures significantly complex. In addition to the physical limitations of the wireless devices (screen, memory, processor, etc.) and their various network access types, the complexity of multimedia content is also increasing. Among the languages defined for mobile devices we find: XHTML Basic, SVG Tiny and SVG Mobile, SMIL Basic, MMS, etc. [5]. These languages provide sufficient functionality to create compelling multimedia presentations for mobiles either by authoring or generation techniques. However, without advanced adaptation techniques, these languages fail to provide functions that consider the targeted client limitations. This limitation is addressed in adaptation systems which apply structural transformations using transformation languages such as XSLT [6] and media objects adaptation such as video and images transcoding [3]. Their goal is to transform the original content from its initial state to a new state (new structure with new media formats) that takes into account the constraints of limited devices. Unfortunately, such techniques are usually not sufficient to guarantee a correct handling and presentation of the adapted content in particular for very limited devices such as mobile phones. Structural adaptations can generate a content which is not adapted to the displaying limitations of the client especially when the original content contains a large amount of data. Media transcoding may also result in a severe degradation of the quality compared to the original content. Moreover, structural and media adaptation systems are usually based on a one pass adaptation process which does not offer any interaction between the user and the original content.

In [5] we developed a proxy-based architecture: NAC, that offers negotiation and adaptation services for heterogeneous environments. In this paper, we present an approach that enables the presentation and the access to rich multimedia content for mobile devices. Our goal is to overcome devices limitations and content complexity by providing an advanced adaptable system that combines content adaptation, formatting and user interactions. The proposed modules have been added to the architecture of NAC. The obtained content adaptation process allows a finer control on the presentation and an accurate formatting more suitable for small screen displays. At the user interface level, the adaptation involves an interaction in order to put the focus in the parts of the document of more importance to the end user.

2. SYSTEM ARCHITECTURE

The adaptation system is designed on top of the NAC architecture [5]. The main components of NAC are the proxy and the profiles repository. The proxy intercepts content access requests from heterogeneous clients, passes them to the corresponding server and adapts the original content according to some client properties. The properties of the context (hardware, software and user characteristics) are stored in the form of profiles written in UPS: a description model based on CC/PP and RDF [5]. The profiles are accessed, maintained and updated using a set of Web services exposed by the repository. Our adaptation system enriches NAC with two components: an adaptation component at the proxy side and a formatter module at the client side. The adaptation component processes the interaction requests of the user and applies a structural and media adaptation on the original content. The result of the adaptation is transmitted to the client formatter that presents the different parts of the adapted content. At any moment, the user can start an interaction with the adapted

DocEng'05, November 2–4, 2005, Bristol, United Kingdom.

content. The different interactions trigger navigation requests which are sent to the adaptation components.

2.1 Displaying layout

The main physical constraint addressed by the content presentation process is the rendering on small screen display of mobile devices. Presenting a complex content requires an efficient allocation of the display surface to the different media objects. In our system we use layout regions based on the region definition of the Synchronized Multimedia Integration Language (SMIL) [4]. In SMIL, a region element controls the position, size and scaling of media object elements. The position of a region, is specified by its *top*, *bottom*, *left*, and *right* attributes [4]. In order to provide more flexibility in the presentation process and the displaying size allocation, we extend the previous definition and define *variable regions* using two functions f_1 and f_2. The two functions are defined on a fixed interval, where f_1 represents the start left position and f_2 represents the region width for a given vertical position. For example, a rectangular region $R_{(w,h)}$ with the left top position of (0, 0) and having w as width and h as height can be defined on the interval [0, h] with $f_1 = 0$ and $f_2 = w$.

2.2 User interaction and navigation

In order to add user interaction to the layout, a region is associated with a set of events that triggers the interaction with the content presented within the region. The user can select a particular region, and, via the proxy, navigate in the different parts of the content (linear navigation). He can also request the current part of the document in more details (hierarchical navigation).

Linear navigation enables the access to the entire content while respecting the displaying limitations of the user device. A media object (text, image, etc.) can be navigated and displayed in a region of the device screen. According to the user interactions, the different parts of the content are adapted and transmitted or simply ignored depending on the defined user preferences. The hierarchical navigation avoids content distortion especially for images due to the adaptation of media objects for small screen displays. This issue is important in many applications where the fidelity between the original and the adapted content should be maintained. An example of such applications is found in the medical sector which involves the access to radiography images. The hierarchical navigation enables to receive an original image object adapted to a given region. Depending to the user interaction, the user can navigate into the different areas of the image. The selected partial area is adapted and presented in the same region which allows navigating in depth into the media object.

After a client request, the adaptation module (located on the proxy side) parses the different media objects nodes from the original document. The original document can be a SMIL or an XHTML document. A text node is decomposed into a set of sub-nodes (Figure 1) according to the text length and the client's device characteristics stored in the profiles repository of NAC [5]. This decomposition is achieved in streaming. This means that a sub-node is created using a textual extraction and sent directly to the client's formatter (Figure 2). For the first sub-node, the textual extraction is applied from the beginning of the text content; the extraction length depends to the client's region. The next extractions are calculated depending on the user interactions. The user can navigate to the different parts of the original text using the direction keys of the device (*Left* and *Right* keys). The content of an image node is resized according to the corresponding region. The user can select an image region and navigate in the corresponding image. After the user interaction (using the direction keys: *Left*, *Right*, *Up* and *Down*) only a partial area of the original image is resized and rendered in the selected region (Figure 3). *Left* and *Right* direction keys enable linear navigation between sibling content blocks whereas *Up* and *Down* keys allow a hierarchical navigation (Figure 1).

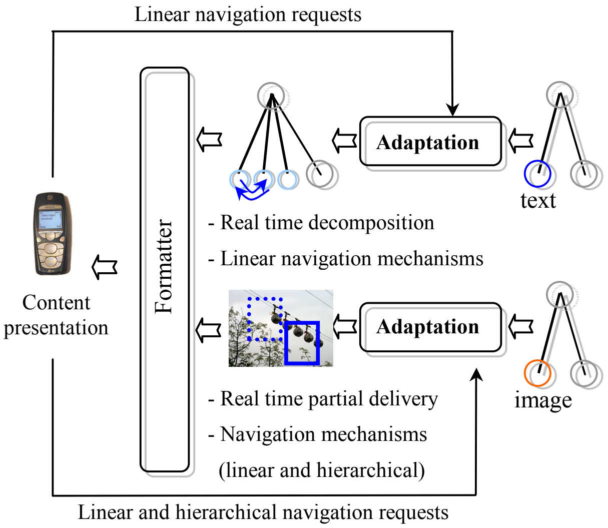
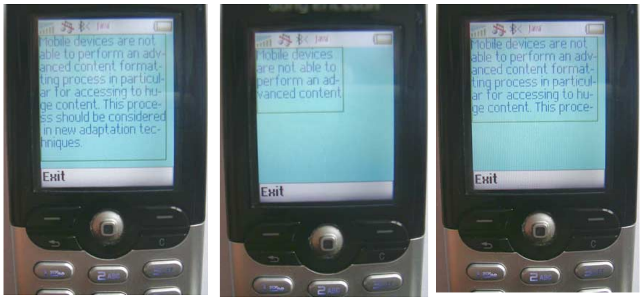

Linear navigation requests
Formatter
Adaptation
text
- Real time decomposition
- Linear navigation mechanisms
Content presentation
Adaptation
image
- Real time partial delivery
- Navigation mechanisms
(linear and hierarchical)
Linear and hierarchical navigation requests

Figure 1. Content decomposition and navigation

2.3 Content formatting

The formatter (Figure 1) presents the content of the different regions and adds listeners in order to intercept the user interactions. Each user interaction triggers a request sent to the adaptation module of the proxy. A request includes a set of parameters such as: the selected region and the part of the content currently displayed together with the user interaction event (Figure 1). The proxy replies to the client request by an adapted content portion extracted from the original content. Consequently, the different parts of the content can be navigated progressively



Mobile devices are not able to perform an advanced content formatting process in particular for accessing to huge content. This process should be considered in new adaptation techniques.

Exit

Mobile devices are not able to perform an advanced content

Exit

Mobile devices are not able to perform an advanced content formatting process in particular for accessing to huge content. This proce-

Exit

Figure 2. Formatting regions on mobile platforms.

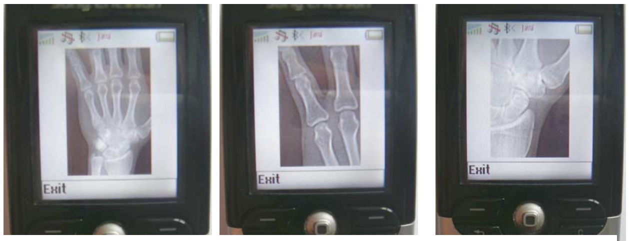


Figure 3. Hierarchical navigation within regions.

and are only sent when necessary.

To adapt an area (w_o,h_o) of an original image to a region (w_r,h_r) while maintaining the image aspect ratio, the image area is resized to the new adapted width and height: $w_{adapted} = \alpha.w_o$ and $h_{adapted} = \alpha.\ h_o$, where $\alpha = Min(w_r/w_o, h_r/h_o)$. For the textual adaptation, the text is extracted from the original document according to the region dimensions and the width and height of the destination font (Figure 2). In a given textual region, the width of each presented line must not exceed the region's width (the f_2 function in Section 2.1) and similarly for the region height.

Table 1. Average regions formatting time

Regions (w, h) *pixels*	**Rich platform – Emulator (*ms*)**	**Mobile platform (*ms*)**
(80,60)	54	765
(120,80)	124.4	1767
(100,100)	132.2	1913
(120,120)	158	2342

3. EXPERIMENTAL RESULTS

A prototype system of this adaptation framework has been implemented. An adaptation module, implemented in Java, is integrated to the negotiation and adaptation module of the NAC architecture [5]. The module is dedicated to content decomposition in streaming, linear and hierarchical navigation and user interactions processing. A formatting module has been implemented for mobile phones using the common part of J2ME: version 1.0 of Mobile Information Device Profile (MIDP) [2] and Connected Limited Device Configuration (CLDC) [1] supported by a majority of mobile phones. The formatter uses the Portable Network Graphics (PNG) format -the only format that is mandatory for MIDP- and various text fonts (the combination of the three supported faces, styles and sizes of MIDP 1.0 |2]).

The results of our experimentations show that a mobile device has very high latency when it receives content which is not adapted to its displaying dimensions, for example when it receives a text with a width that exceeds its screen width. The implemented adaptation prototype enhances significantly the performance by presenting the content in adapted regions. Table 1 reports some measures of the average formatting time for 4 different regions. The average time is measured and compared between two different platforms: a desktop PC Pentium III processor of 850 MHz and 256 MByte of memory and a poorly capable mobile phone SE T610 with a Java processor of 0,4 MHz and 255 KByte of memory. Table 2 gives a summary of the display time (formatting and rendering) of the previous regions with the same content parts. The important difference in time between the desktop and the mobile platform is as expected due to the Java processor of the mobile phone. The adaptation prototype succeeds in presenting most of the content that was initially authored for rich platforms and was unusable for this type of terminals. Moreover, the system considers the user interaction and provides navigation capabilities. Linear and hierarchical navigation enables to access to the content even with a limited displaying size. Hierarchical navigation allows the presentation of a content that can reach the same level of detail as in a rich platform. Combining the user interaction with the quality and the transmission of the adapted content improves the adaptation system and optimizes the use of the environments resources such as the network bandwidth and the processing time.

Table 2. Complete displaying time

Regions (w, h) *pixels*	**Rich platform – Emulator (*ms*)**	**Mobile platform (*ms*)**
(80,60)	60.2	810
(120,80)	136	1855
(100,100)	142.4	2001
(120,120)	164.2	2530

4. CONCLUSION

In this paper, we have presented an experimental content adaptation system that enables the presentation of multimedia content for resource limited mobile terminal. In particular, the system tackles the limitations related to small screen displays. The presented system is integrated in a complete adaptation and negotiation architecture called NAC [5]. A stream-based text formatter together with an interaction-based access to adapted content enables the presentation of rich multimedia content on lower end terminal. The application has a small footprint on the terminals since most of the processing is achieved on the proxy.

5. REFERENCES

[1] J2ME, Connected Limited Device Configuration (CLDC), JSR 30, JSR 139, Sun Microsystems, http://java.sun.com/products/cldc/.

[2] J2ME, Mobile Information Device Profile (MIDP), JSR 37, JSR 118, Sun Microsystems, http://java.sun.com/products/midp/.

[3] K. Curran and S. Annesley. Transcoding media for bandwidth constrained mobile devices, International Journal of Network Management, 2005. Vol. 15, pp. 75-88.

[4] Synchronized Multimedia Integration Language (SMIL 2.0), W3C Recommendation. http://www.w3.org/TR/2005/REC-SMIL2-20050107/, Januar 2005.

[5] T. Lemlouma. Multimedia Services Negotiation and Adaptation Architecture in Heterogeneous Environments, PhD Thesis, 9 June 2004 INPG, Grenoble, France.

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