XML and XSLT Modeling for Multimedia Bitstream Manipulation

Sylvain Devillers
Laboratoires d'Electronique Philips
22, avenue Descartes
94453 Limeil-Brevannes, France
+33 (0)1 45 10 67 55
Sylvain.Devillers@philips.com

ABSTRACT

New devices gaining access to the Internet need to obtain multimedia content adapted to their limited capacities. Scalable formats allow to retrieve different versions of a single file by simple manipulations of the bitstream. However, prior knowledge of the data structure is required to perform these operations. In this paper, XML is used to describe the structure of a bitstream. The resulting document is transformed via XSLT Style Sheets into a new XML document from which an adapted bitstream is generated. This method is generic and applicable to any coding format. An application to the emerging JPEG2000 image coding standard is given.

Keywords

XML, XSLT, Web Publishing, Scalable Media, JPEG2000.

1. INTRODUCTION

The multiplication of new devices gaining access to the Internet makes crucial to be able to propose different versions of the same multimedia content (audio, video or still image) adapted to the client resources in terms of bandwidth, display or computing capabilities. The use of scalable media allows to retrieve different versions of a content from a single file and hence saves the burden of generating and handling one file for each required version. This paper proposes a framework to fully exploit these scalable features for multimedia content adaptation. Section 2 explains the principle of scalable multimedia coding and shows how XML (Extensible Markup Language) can be used to describe the structure of a bitstream. Then in section 3, we extend the web publishing framework to multimedia content by using XSLT (Extensible Stylesheet Language Transformations). Finally, we apply this framework to the emerging JPEG2000 image coding standard in section 4.

2. SCALABLE MULTIMEDIA CONTENT

In scalable multimedia coding standards such as MPEG-4 for video or JPEG2000 for still image, data are organized in a way such that, when retrieving a bitstream, it is possible to first render a degraded version of the content, and then progressively improve it by loading additional data retrieved from the source. With a multimedia content coded in a relevant scalable way, it is hence possible to retrieve an adapted version from a single bitstream by performing simple operations such as data truncation.

For example, the 3D subband video codec proposed by Bottreau *et al.* [1] provides several progressive modes. Successive spatial, temporal and quality levels are indicated by flags in the bitstream. If the required spatial resolution, frame-rate or quality are inferior to the ones provided by the encoder, the decoder or server must skip some parts of the bitstream. For this purpose, a dedicated software parses it and cuts the irrelevant part off. A server providing several contents coded in different standards therefore needs as many software modules as offered formats to manipulate them.

To solve this problem, we propose to use a common structuring language such as XML [2] to describe the structure of a multimedia bitstream. A multimedia file generally consists of headers containing the parameters needed to decode the content and the actual payloads containing the compressed data. Header parameters may be included in the XML document as attributes or elements values. The payload is usually meant to be interpreted on a bit per bit basis and its compactness should therefore be preserved. Furthermore, being pure binary data, it cannot be directly included into the XML document. A classical solution is to first encode it in a parsable format such as base64 [3]. Another solution is to use pointers in the XML document to indicate the data segments in the original binary file.

In this way, it is possible to generate an XML document describing the structure of a bitstream. The obtained document may then be parsed with a generic software (XML parser) and accessed via a standard interface such as DOM (Document Object Model) [4]. Other XML technologies may then be used as described in the following section.

3. WEB PUBLISHING FRAMEWORK

The problem of content adaptation is well-known for XML documents on the web and is addressed by the so-called web publishing framework. This framework is designed to publish the same text document in different versions adapted to the capabilities of the rendering devices. For this purpose, the structure of the document must be separated from its presentation: the source document is structured with XML and then dynamically processed to generate a presentation adapted to the available resources, e.g. in HTML or WML for respectively a web or WAP browser. This processing may be performed by XSLT style sheets [5], the W3C language specifying XML-to-XML transformations. The overall framework principle is described hereafter: following a client request for a resource, the server first exchanges its capabilities with the client in a content negotiation stage (not described here) to determine the adapted version to be published. It chooses the relevant XSLT transformation, applies it to the source XML document, and returns the resulting XML document.

We propose here to extend this framework to multimedia content. XSLT style sheets are created to specify transformations on the XML document describing the bitstream. These transformations modify attributes and remove elements. From the transformed XML description, it is then possible to generate a new compliant bitstream sent back to the requesting client (see Figure 1).

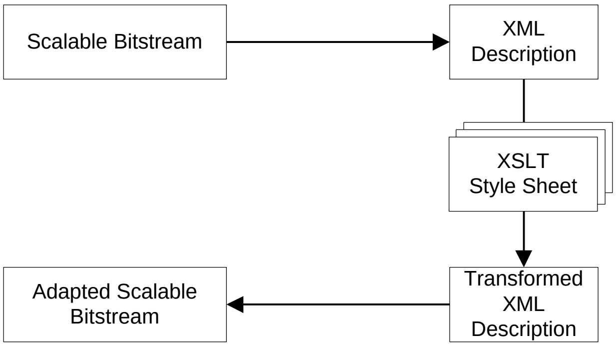

Figure 1: Edition of a multimedia bitstream with XSLT

This framework is generic and may be applied to any kind of multimedia bitstream. The following section shows an application to JPEG2000.

4. APPLICATION TO JPEG2000 IMAGES

JPEG2000 [6] is an emerging image coding standard, meant to replace the widely used JPEG format. It is based on the wavelet transform, an inherently scalable method which allows the progressive encoding driven by SNR quality, color component or resolution, i.e. image size.

In the first case (progression by SNR quality), the bitstream is organized in layers, each layer carrying a quality increment. In the second case (progression by color component), data is organized by color components. For an input color image, the JPEG2000 encoder first transforms it into the YCrCb space. By removing the last components, one therefore gets the luminance (i.e. gray-level) image. In the last case (progression by resolution), data are organized following the successive image size decompositions starting from the lowest one. For example, considering a 512x512 image, the bitstream first yields a 32x32 image, then a 64x64 image and so forth up to the full original resolution. All these image transformations can be achieved by editing the bitstream in a relevant way.

A JPEG2000 bitstream is structured with 2-byte markers. We use these markers to parse the bitstream and generate an XML description of its structure. Then three XSLT style sheets corresponding to the progressive schemes listed above are defined. Each style sheet may modify some attributes and remove XML elements corresponding to the segments of data to be truncated. The style sheets may use input parameters indicating for instance the desired number of quality layers for an image encoded by layers.

The web publishing framework described in section 3 can therefore be extended to multimedia content in the following way: along with a JPEG2000 image, the server stores an XML document describing the structure of its bitstream. Upon request, the server chooses the relevant version to be published and applies the corresponding XSLT transformation to the XML document. Based on the resulting transformed XML description, a new JPEG2000 compliant bitstream is generated from the source image and returned to the client.

Note that the possible transformations are restricted to the chosen progressive scheme. For example, to generate a gray-level image, the source image must have been encoded by components.

5. CONCLUSION

XML and related technologies have gained an overwhelming success for structuring, editing and exchanging electronic data, but their use has been limited so far to text documents. We show here how to apply the same powerful languages and tools to binary multimedia formats. In this paper, XML is used to describe the structure of a scalable bitstream and XSLT style sheets are used to transform this description, hence generating a new, adapted bitstream. This novel method is generic and thus applicable to any coding format. Application of XML technologies to binary multimedia data is therefore a flexible and powerful solution and opens extremely promising prospects for the exchange and the manipulation of multimedia content.

6. REFERENCES

[1] V. Bottreau, M. Bénetière, B. Felts and B. Pesquet "A fully scalable 3D subband video codec", submitted to *ICIP'2001, IEEE International Conference on Image Processing*, Thessaloniki,Greece, Oct. 7-10, 2001

[2] *Extensible Markup Language (XML) 1.0 (Second Edition)* W3C Recommendation, October 6th, 2000 Available on W3C site at http://www.w3.org/TR/2000/REC-xml-20001006

[3] *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies* IETF RFC2045 Available at http://www.rfc-editor.org/

[4] *Document Object Model (DOM) Level 1, Version 1.0* W3C Recommendation, October 1st, 1998 Available on W3C site http://www.w3.org/DOM/

[5] *XSL Transformations (XSLT), Version 1.0* W3C Recommendation, November 16th, 1999 Available on W3C site at http://www.w3.org/TR/xslt

[6] *JPEG 2000 Part I Final Draft International Standard* ISO/IEC JTC1/SC29 WG1, JPEG 2000 Available on official JPEG web site, http://www.jpeg.org/