SXML: Streaming XML

Boris Rogge[1], Dimitri Van De Ville[1], Rik Van de Walle[1], Wilfried Philips[2] and Ignace Lemahieu[1]

[1] University of Ghent
ELIS - MEDISIP - IBITECH
Sint-Pietersnieuwstraat 41
B-9000 Ghent, Belgium

[2] University of Ghent
TELIN
Sint-Pietersnieuwstraat 41
B-9000 Ghent, Belgium

Tel:[1] +32 9/264.89.11 Fax:[1] +32 9/264.35.94
E-mail: Boris.Rogge@rug.ac.be

Abstract— **When broadband networks will be implemented in the near future, a huge amount of bandwidth will be available and hence, a number of new applications will emerge. Application developers will need a framework that enables them to utilize the possibilities of these types of new networks. In this article we present a document type that will allow the addition of (meta-)information to data streams and the synchronization of different data streams. It is called SXML (Streaming XML) and is based on the eXtensible Markup Language (XML). The SXML grammar is defined in a document type definition (SXML-DTD). The content of an SXML document can be processed real time or can be retrieved from disk.**

XML is being used in a complete new manner and in a totally different environment in order to easily describe the structure of the stream. Finally, a preliminary implementation has been developed and is being tested.

Keywords— **XML, Internet, JAVA, Broadband Networks**

I. INTRODUCTION

The next big thing in the digital revolution might be the general use of broadband networks and digital television. The Internet has shown us the possibilities of global networks and has given birth to a huge number of new applications. Broadband networks will provide these applications with the bandwidth they need and thus become real multimedia transport channels [1,2]. Broadband networks are currently being deployed at a high pace. The amount of downstream bandwidth that these networks will provide us with will be huge compared to the downstream bandwidth currently available.

The availability of huge amounts of bandwidth will undoubtely stimulate the development of a number of new multimedia applications. Some of these applications will be inherited from the Internet and adapted to broadband networks, while others will be new and are still to be created. Some applications that are currently available are WebTV, tele-education [3] and online interactive games [4]. We strongly believe that one of the driving forces behind these new applications will be digital television and its derivatives.

In order to easily create and deploy such multimedia applications, developers will need a generic framework that can be used as a toolbox when creating applications that are to be used on broadband networks. Some of the services the framework will provide could be: easy distribution of data (advanced push technology) and synchronization of different data streams and media types. We will introduce a first building block for such a framework.

Today, a number of standards for multimedia content is widely used in the digital world. On the Internet, textbased documents are mainly marked up in HTML or XML while multimedia content is stored in a large number of different formats (Quicktime, MPEG, RealVideo and RealAudio, Flash, etc.). A standard that synchronizes multimedia content within a web-based environment is the W3C consortiums Synchronized Multimedia (SMIL).

In this paper we present a new document type that is based on the eXtensible Markup Language (XML). The new document type is used for adding information and references to existing data streams and for synchronizing different data streams. This article is structured as follows: in Sect. II a brief introduction to the XML standard is given. Section III provides a detailed overview of the Streaming XML protocol (SXML). Finally, Sect. IV summarizes the conclusions from this paper and Sect. V gives some directions for future work.

Dimitri Van De Ville is a Research Assistant with the FWO Flanders, Belgium.

Rik Van de Walle is a Post-Doctoral Fellow of the Fund for Scientific Research - Flanders (FWO, Brussels, Belgium)

ISBN: 90-73461-18-9

II. Extensible Markup Language

The eXtensible Markup Language (XML) is a direct descendant of the Standard Generalized Markup Language (SGML) [5]. Hundreds of optional features added to SGML over the course of years have been removed to produce XML[1] [6]. In order to introduce XML, we wil compare it to HTML (Hypertext Markup Language). Although they are both derivatives from SGML and hence being markup languages, HTML and XML are very different. HTMLs primary goal is to create a layout for a certain document whereas the primary goal of XML is giving structure to the content of documents in a formal manner. The XML-standard defines how to create an XML document by defining the tags your application needs. Thus, all tags that are used within an XML document are to be defined by the author of the document.

When writing a page in HTML, one knows that the document is formatted correctly if all tags are correct and it is correctly visualized. With XML there are two levels of correctness: a document can be *valid* or *well-formed*. A valid XML document is completely self-describing. It exists out of a base document and some optional auxiliary files. Everything the parser needs to know to process the documents structure and content is embedded in the document or in the auxiliary files that are refered to from the base document. A well-formed XML document contains the absolute minimum needed by the parser to process the documents structure and content. This means that the syntax has to be compliant with the XML standard. A valid XML document is always well-formed. Chief among the differences between valid and well-formed XML is that valid XML includes a document type definition (DTD). The DTD is the name given to the grammar that appears in the Document Type Declaration [7] once it is processed by the XML parser. The fact that both terms have the same acronym is regularly a source of confusion. A DTD includes [8], among other things:

- definitions of the tags used to mark up the document,
- description of the allowed structural relationship,
- specification of the sequence in which the tags can appear within the XML document,
- list of properties allowed or required for certain tags.

A valid XML document includes a DTD within the document or makes a reference to a DTD. In both cases, the parser starts by processing the DTD before processing the content of the XML document. A well-formed XML document does not include or does not make any reference to a DTD. In the latter case, only the correctness of the XML syntax is checked by the parser.

To visualize the content of an XML document, style sheets are used. Currently there are two types of style sheets that can be applied to an XML document: Cascading Style Sheets (CSS) and eXtensible Style Sheets (XSL). Figure 1 shows the document structure necessary to display an XML document in a standard[2] Internet browser. The DTD can be included in the XML document (internal DTD) or can be referenced by the XML document (external DTD). An XSL style sheet is always referenced from the XML document.

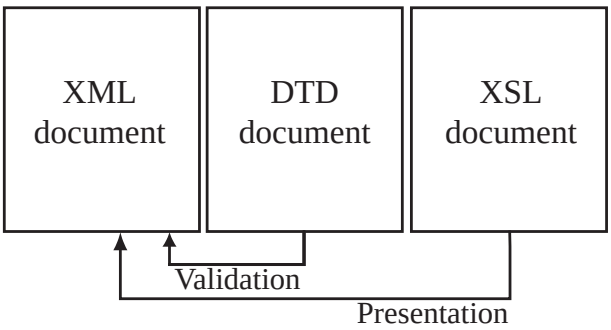

Fig. 1. Auxiliary documents linked to an XML document.

III. Streaming XML

Streaming XML (SXML) is an XML-based document type for adding (meta-)information to existing data streams and for synchronizing different kinds of media. An SXML document can be created and distributed on the fly or can be stored for retrieval at a later moment in time. An SXML document consists out of an undefined number of SXML frames (see figure 2 and table I). An SXML frame is a description of a moment or period in time. The properties that have to be assigned to this frame are included as attributes of the frame. Each frame also holds a data section and a reference section. Although the data section can include an unlimited amount of data, an SXML stream is not intended to carry large amounts of data. Instead, it is intended to carry control information and references to other objects or streams. The reference section holds references to other streams, Internet addresses or any type of reference that the client application can handle. The actual data streams are to be transported using standard network techniques.

[1]These are features from the SGML standard. An XML document can be as complex as the author is willing to make it.

[2]Currently only Microsoft Internet Explorer includes a non-validating XML parser.

TABLE I
An SXML stream

```
<?xml version="1.0"?>
<sxml>
  <frame id="2543" duration="15s" store="false" type="visible">
    <data>
      <![CDATA[Place actual data in here.]]>
    </data>
  </frame>
  <frame id="2544" duration="12s" store="false" type="visible">
    <data>
      <![CDATA[FF=+15.1;RW=+41.3]]>
    </data>
  </frame>
   ...
  <frame id="2611" duration="0s" store="false" type="hidden" title="Frame n">
    <data>
      <![CDATA[Reference to the RUG webserver]]>
    </data>
    <reference protocol="http://">
      <![CDATA[www.rug.ac.be/]]>
    </reference>
  </frame>
</sxml>
```

The SXML Document Type Definition (SXML-DTD) defines the grammar of an SXML document. A validating XML parser will check all SXML streams for their correctness using this SXML-DTD. A valid SXML document should start with an <sxml> tag and end with its closing equivalent </sxml>. Within these two tags an undefined number of frames can be defined by including the following two tags: <frame> and </frame>. Every frame holds a data section (<data> and </data>), and a reference section (<reference> and </reference>). Table II depicts the document type definition (DTD) for an SXML document.

Each frame tag can include a number of attributes. These attrubites are summarized and described in table III. The attributes for the reference section of a frame are described in table IV.

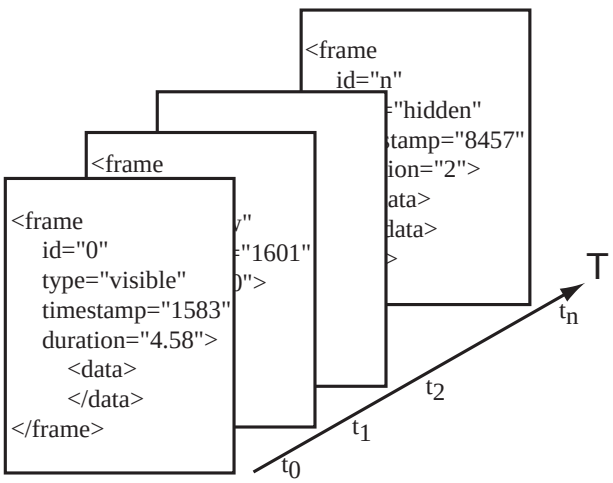

Fig. 2. An SXML stream consisting out of an undefined number of consecutive frames.

A. Generic framework

In this paragraph we describe the final configuration in which SXML will be used. Although this paper only focuses on the document definition, a brief description of the framework we have in mind is given. Figure 3 shows the framework in which SXML is to be used. This will be a generic framework consisting of services for application developers. The final goal is to create an SXML-decoder that can be used in consumer devices such as set-top boxes or intelligent TVs. These devices will use SXML to combine different data streams into one interactive whole that is presented to the viewer or end user. The content

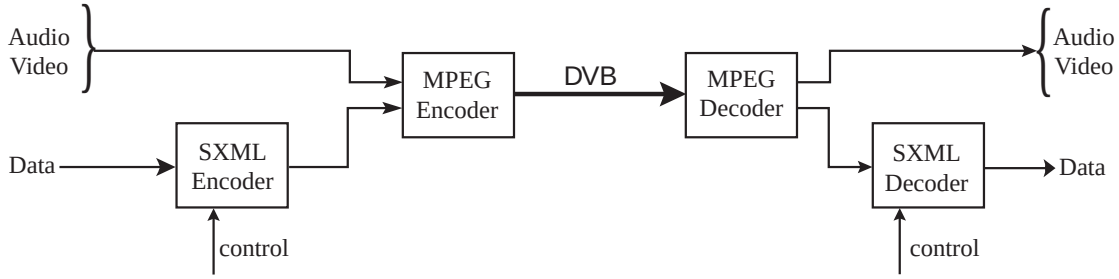

Fig. 3. Overview of the framework in which SXML will be used.

TABLE II
The SXML Document Type Definition (DTD)

```
<!ELEMENT sxml (frame)*>
<!ELEMENT frame (data?|reference?)>
<!ELEMENT data (#PCDATA)>

<!ELEMENT reference (#PCDATA)>
<!ATTLIST frame
    id          CDATA          #REQUIRED
    timestamp   CDATA          ""
    duration    CDATA          ""
    store       (true|false)   false
    type        (hidden|visible) visible
    layer       (true|false)   false
    title       CDATA          ""
    min         CDATA          ""
>
<!ATTLIST reference
    protocol CDATA #REQUIRED
>
```

TABLE III
Attributes Associated With An SXML Frame Tag

Attribute	**Explanation**
ID	A unique identifier is assigned to the frame at creation time
Timestamp	A timestamp is added to the frame at creation time
Duration	The amount of time a certain frame has to be shown at the client side
Type	The type of the frame. This can be one of the following types: visible, hidden
Store	True when the content of the frame has to be stored; false when the data is not to be stored
Description	A brief description of the frame

TABLE IV
Attributes Associated With an SXML Reference Tag

Attribute	**Explanation**
Protocol	Identification of the protocol that the reference uses

providers will use an SXML-encoder to add references to existing streams of data. Some functionality this framework will provide is listed below:

- adding (meta-)information to data streams,
- synchronization of different data streams (including audio and video),
- layering of different kinds of media,
- intelligent distribution of data over broadband networks.

B. Current implementation

Currently, a preliminary version of an SXML-parser has been implemeted in JAVA. This parser is based on the XP parser [9] written by James Clarck. A JAVA applet uses the SXML parser and processes the basic features from an SXML document. The applet processes all frames and shows all frame information, attributes and attribute values on screen. It also presents time information to the user (How much longer is a frame to be shown? What is the timestamp of a particular frame?). When a reference is made in the reference section and the browser supports the referred data type, this information is shown in the browser. Controlling instructions are currently discarded since no application is implemented yet.

IV. Conclusion

In this paper we presented a method for adding (meta-) information to data streams and for synchronizing several types of data using a new document type: SXML. This SXML stream is based on the XML standard and can be generated in real time or can be created in advance. A document type definition for SXML was introduced and a preliminary

implementation of an SXML parser based on the XP XML parser was presented. This parser has been integrated into a JAVA applet that interprets the SXML stream and shows the frame information on screen. Only well-known Internet protocols can be processed at this moment in time.

V. Future Work

Currently, an SXML document can only hold one information stream. This implies that all frames belong to the same stream. As a consequence, all frames are to be processed consecutively and no simultaneous tasks can be performed. The SXML document should be extended to hold several streams with each frame belonging to one of these streams. Therefore, the parser will have to manage all these streams and their relationships. Synchronization between different streams within an SXML document is a future research topic as well. Finally, the generic framework will be developed and tested in the near future.

References

[1] A. Dutta-Roy, "Bringing home the Internet," *IEEE Spectrum*, pp. 3338, march 1999.

[2] A. Dutta-Roy, "Cable, its not just for TV," *IEEE Spectrum*, pp. 5359, may 1999.

[3] D. A. Harris, "Online distance education in the United States," *IEEE Communications magazine*, pp. 8791, March 1999.

[4] C. Flerackers, F. Alsema, F. Van Reeth, and K. Coninx, "Typotoons: An interactive TV-application through a networked virtual environment," in *Proceedings of SCI 99*, pp. 349356, 1999.

[5] "ISO 8879 SGML," tech. rep., International Standards Organisation, 1988.

[6] J. E. Simpson, Just XML. Prentice Hall, 1999.

[7] S. McGrath, *XML by example*. The definitive XML series, Prentice Hall, 1998.

[8] C. F. Goldfarb and P. Prescod, *The XML handbook*. The definitive XML series, Prentice Hall, 1998.

[9] J. Clarck, "XP parser." Internet. `ftp://ftp.jclark.com/pub/xml/xp.zip`.

This page was intentionally left blank