AN XML-BASED FORMAT FOR ADVANCED MUSIC FRUITION

Adriano Baratè
LIM – DICO
University of Milan
Via Comelico, 39
20135 Milano, ITALY
+39 02 50316382
barate@dico.unimi.it

Goffredo Haus
LIM – DICO
University of Milan
Via Comelico, 39
20135 Milano, ITALY
+39 02 50316222
haus@dico.unimi.it

Luca A. Ludovico
LIM – DICO
University of Milan
Via Comelico, 39
20135 Milano, ITALY
+39 02 50316382
ludovico@dico.unimi.it

ABSTRACT

This paper describes an XML-based format that allows an advanced fruition of music contents. Thanks to such format, namely MX (IEEE PAR1599), and to the implementation of *ad hoc* interfaces, users can enjoy music from different points of view: the same piece can be described through different scores, video and audio performances, mutually synchronized. The purpose of this paper is pointing out the basic concepts of our XML encoding and presenting the process required to create rich multimedia descriptions of a music piece in MX format. Finally, a working application to play and view MX files will be presented.

1. INTRODUCTION

Specific encoding formats to represent music features are already commonly accepted and used. They are aimed at a precise characterization of different music aspects. For example, AAC, MP3 and PCM formats encode audio recordings; MIDI represents a well known standard for computer-driven performance; JPEG and TIFF files can contain the results of a scanning process of scores; Finale, NIFF and Sibelius formats are aimed at score typing and publishing, and so on. But such formats are often characterized by an intrinsic limitation: they can describe music data or metadata for score, audio tracks, computer performances of music pieces, but they seldom encode all these aspects together.

On the contrary, we are interested in a "comprehensive" representation and fruition of music, addressed to musicologists, performers, students and people simply interested in music. The key characteristics that a comprehensive format should support can be summarized as follows:

- Richness in the multimedia descriptions related to the same music piece (graphical, audio, and video contents).
- Possibility to link and perform a number of media objects of the same type (for instance, many performances of the same piece or many score scans coming from different editions).
- Complete synchronization among time-based contents, meaning that audio and video contents are kept synchronized with score advancing, even when the user switches from a particular performance to another, or from a particular score edition to another.
- Interaction, that is the possibility for the users to click any point of the score and jump there also in the audio content currently performed, as well as the possibility to navigate the audio track moving the related slider control and highlighting the related portion of score.

Achieving these goals imply: 1) designing and adopting a suitable format to represent music, 2) implementing software applications to achieve synchronization and user interaction, and 3) encoding music pieces and all the related material in the aforementioned format. These subjects will be now treated in greater detail.

2. A COMPREHENSIVE DESCRIPTION OF MUSIC

In our opinion, it is necessary to conceive music description in a comprehensive way. In order to appreciate the richness of music communication, let us recall that such language is made up of many different and complementary aspects: music can be (and actually is) the idea that the composer translates to symbols as well as their performance, the printed score that musicians read as well as a number of other related contents. A complete analysis of music richness and complexity is provided in [3], where six different levels of music description are identified – namely General, Structural, Logical, Notational, Audio, and Performance layers. This multilayer structure, suitable for our concept of comprehensive description of music, is reflected by the encoding format we are developing at LIM.[1] Our goal is providing a highly integrated performance of music, where score, audio, video, and related graphical contents can be enjoyed together.

Nowadays we dispose of good and commonly accepted file formats to describe only one aspect (sometimes a few aspects) of music, but experts and technicians are more and more interested in ways to integrate different information sources and representation methods.

What does the adjective comprehensive mean, when applied to music field?

Music language is made up of many different and complementary aspects. One of the most complete analyses of music richness and complexity is provided in [2], where six different levels of description are identified: namely general, structural, logical, notational, audio and performance layers. This multi-layer structure

[1] Laboratorio di Informatica Musicale, Dipartimento di Informatica e Comunicazione, Università degli Studi di Milano.

could answer the request of completeness in the "horizontal dimension", as the layers we listed can be considered a good coverage of the different domains of music. They take into account the evidence that music is the composition itself as well as the sound a listener hears, and is the score that a performer reads as well as the execution provided by a computer system. In order to appreciate the richness of such a kind of communication, we can point out that music – in its most general meaning – can stimulate different senses: the sense of hearing, the sense of sight and even the sense of touch. In this sense, music is multimedia (as several different media are employed to convey information) and multi-layered (as information can be structured according to a multi-layered layout).

After identifying in general terms the layers music language can be divided in, we should face the problem of describing each of them in a comprehensive way.

We have listed the elements that can constitute the horizontal dimension; now let us move down the vertical dimension of the overall context.

Any layer presents different characteristics and requires a specific analysis of its peculiarities. As an example, we can consider the logical layer, where music events are listed and organised. It should provide a logical representation of the piece, made of clefs, time and key signatures, bars, notes, rests, horizontal symbols such as hairpins, and so on… The problems involved here mainly deal with music grammar and notation capabilities: a comprehensive format should support virtually all the different symbols, in all their versions, taken from any notation style. Even a small and well-defined subset of the whole problem is difficult to solve. In fact, considering only the Common Western Notation, it is rich in unusual notations such as the symbol named coulé,[2] nowadays almost disappeared from Music Theory texts. We have cited a form of embellishment, as many forms of ornament (in their name, graphical representation, and musical resolution) are related to a particular period, style and even author. Nevertheless, music grammar presents many other examples of variety in notation. As a matter of fact, we cannot oppose this phenomenon, which has historical justifications and represents an aspect of richness; we simply have to take it into account.

Leaving the "reassuring" field of Common Western Notation, and considering for instance contemporary scores, the problem of a comprehensive encoding format becomes hard to solve. And, even assuming that we are able to support all the symbols derived from the past and the present scores, what about future? Who could forecast the evolution of musical notation? Even best-selling notation software applications show serious gaps when they have to represent contemporary music or pieces beyond Common Western Notation.

[2] *Coulé* is a passing appoggiatura, namely a grace note that softens the line making it smoother by binding the notes together. This embellishment was a convention in both the XVII and XVIII centuries, often employed by W.A. Mozart himself.

In other words, it is not possible to identify all the instances of music symbols, from any language, notation style, historical period and geographical region. Probably, this is the reason why Perry Roland in [4] rejects the SMDL standard, saying that "it defines the term music much too broadly to effect a practical solution". However, from our perspective, a general definition of the concept of music is desirable.

Till now we focused on only one layer of music communication, but other description levels are affected by similar problems. For instance, despite its apparent simplicity, the general layer (devoted to metadata such as the author, the title, and the genre of the coded music work) hides challenging and insidious problems: Which is the comprehensive information a user expects to find in this section? Is it possible to treat the general metadata for a classical work like the ones for a rock or pop piece? Is it possible to create taxonomy about music genres? These are only some of the questions we should answer when we choose or design an encoding format for music.

A general idea, present both in IEEE and in ISO/IEC recent approaches to music description, is taking advantage of existing formats. As a consequence, the problem of a comprehensive representation for music can be articulated as follows.

- Which already existent descriptions should be supported?
- How to code the data and metadata that are not present in other kind of representation?
- How to combine such heterogeneous information within an integrated description framework?
- How to provide interoperability and synchronization in the multi-layer environment?

The following section will answer these questions.

3. MX FORMAT FOR MUSIC REPRESENTATION

In order to integrate all the aspects of music within a single description, we are developing a new XML-based format, called MX. Currently, MX is undergoing the IEEE standardization process (IEEE PAR1599), as described in [3]. Our approach is different from other kinds of music encodings, in particular because we represent music information according to a multi-layer structure and to the concept of space-time construct. Now we will explain these key concepts in greater detail.

The first key feature of MX format is the multi-layer structure. We can conceive each layer as a different degree of abstraction in music information. For a common and exhaustive description, in MX we distinguish the Structural, Music Logic, Notational, Performance and Audio layers. This multi-layered description allows MX to support a number of different formats aimed at music encoding without modifying such commonly accepted standards. For example, MusicXML can be integrated in our format to describe score symbolic information (e.g., notes and rests),

whereas other common file types such as JPEG and TIFF for notational layer, MP3 and WAV for audio layer can be linked to represent other facets of music.

The second peculiarity of the MX format, directly related to its multi-layered structure, is the presence of a space-time construct called spine. In fact, considering music as a multi-layered information, we need a means to link and synchronize the heterogeneous facets that compose such information. To this end, we introduced the concept of spine, namely a structure that relates time and spatial information (see Figure 1). The light grey lines graphically represent the synchronization among different layers provided by the spine. In the example, three representations of the same piece are present: a score in TIFF format, a MIDI file and an audio track. Events are univocally identified by id attributes inside the spine structure. Each layer containing music data refers to the event identifiers listed in the spine, and this provides a global internal synchronization. Of course, some data intrinsically cannot be synchronized: it is the case of general metadata or related graphical files (photos, sketches, pictures) which have no relationships towards score representation and performance.

Through spine mapping, it is possible – for example –

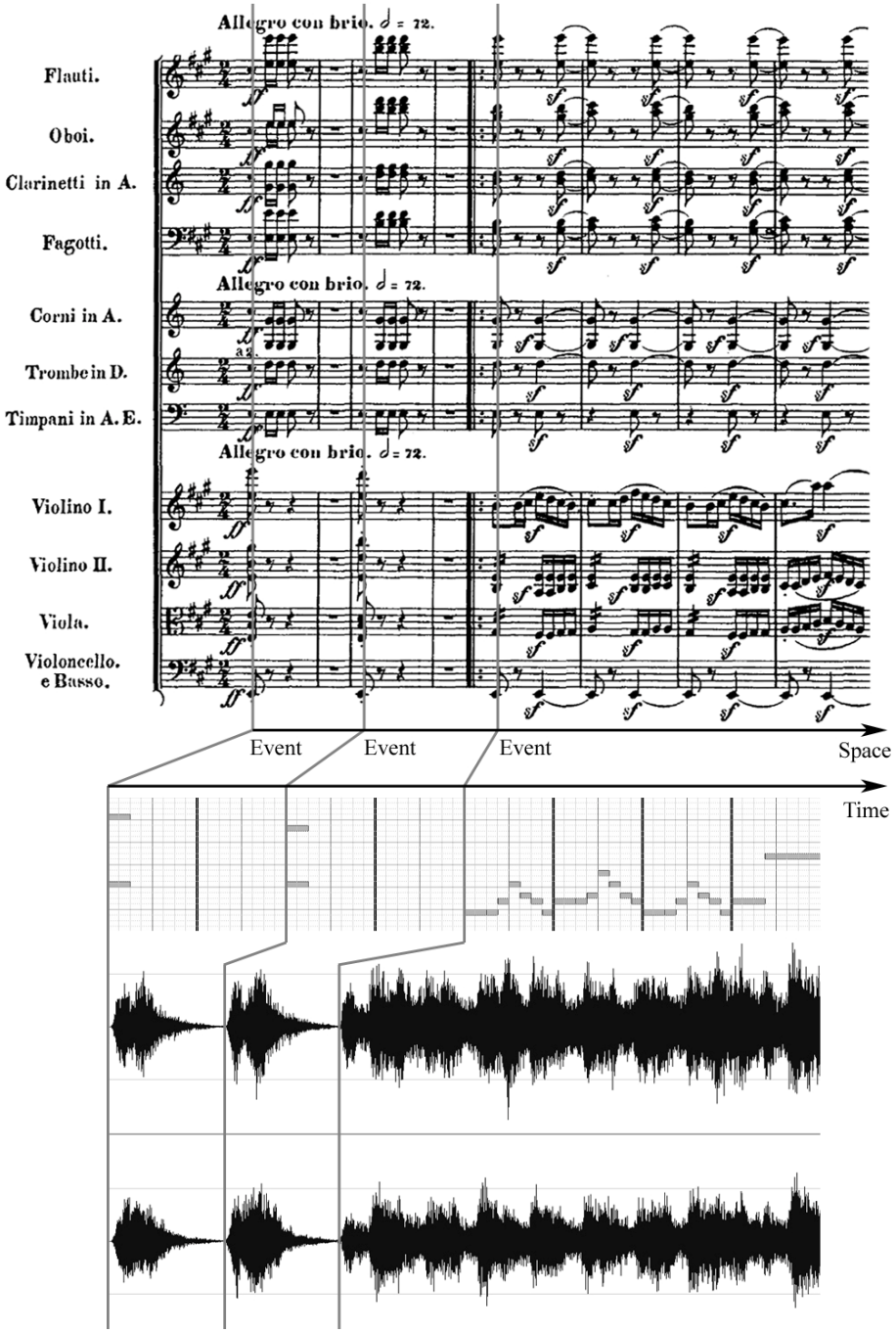

Figure 1. Synchronization among Notational, Performance and Audio layer provided by spine.

to fix a point in a layer instance (e.g. the notational one) and jump to the corresponding point in another one (e.g. the audio one). Besides, if many media objects of the same type are provided, a real-time switch from an object to another is allowed.

4. MX LAYERS

After presenting the multi-layered structure of MX and the time-space construct that allows synchronization, let us briefly describe the meaning and the contents of each MX layer.

The Logic layer contains information referenced by all other layers, and it represents what the composer intended to put in the piece. It is composed of two elements: i) the Spine description, used to mark the significant events in order to reference them from the other layers and ii) the LOS (Logically Organized Symbols) element, that describes the score from a symbolic point of view (e.g., chords, rest).

The Structural layer contains explicit descriptions of music objects together with their causal relationships, from both the compositional and the musicological point of view. It represents how music objects can be described as a transformation of previously described music objects.

The Notational layer links all possible visual instances of a music piece. Representations can be grouped in two types: notational and graphical. A notational instance is often in a binary format, such as NIFF or Enigma, whereas a graphical instance contains images representing the score. Usually, the latter is in a binary format too (e.g., a JPEG image or a PDF file), but it can also be a vector image. The information contained in this layer is tied to the spatial part of the Spine structure, allowing its localization.

The Performance layer lies between notational and audio layers. File formats grouped in this level encode parameters of notes to be played and parameters of sounds to be created by a computer performance. This layer supports symbolic formats such as MIDI, Csound or SASL/SAOL files.

Finally, the Audio layer describes properties of the source material containing music audio information. It is the lowest level of the layered structure. The complete DTD of MX 1.5 format is available at http://www.lim.dico.unimi.it/mx/mx.zip, together with a complete example of music representation in MX format.

5. PREPARING AN MX FILE

As stated before, MX format allows a very rich and comprehensive description of music contents. Of course, richness is just a possibility: a piece could be described only in terms of its music symbols, without multimedia objects attached, and the MX file would be validated in any case. However, we are interested in a comprehensive description of music, so we will analyze the problems involved in the process of creation of a complex file.An MX file, like any other XML-based encoding, can be written and edited even by a simple text editor. It is virtually possible to encode all the score symbols and all the synchronization information by hand and in text format. Nevertheless, this approach would be very difficult and time-consuming.

First, for a musician the task of writing notes and rests according to XML formal rules (that are substantially different from score notation) is unacceptable. Besides, after obtaining a well-formed, valid, complete and semantically correct encoding of the piece, the author of the MX file should face other difficult tasks: e.g., manually finding values to map and synchronize heterogeneous media objects and entering those values in the MX file. This is the main reason why we implemented a number of utilities to support the creation and management of rich MX files.

6. MX UTILITIES

In order to simplify the assemblage process of heterogeneous contents within a single music description, we developed a suite of software applications. Unfortunately, the process required to map and synchronize heterogeneous media at the moment cannot be performed automatically. This would require, for instance, a good OMR[3] system in order to recognize musical symbols in scores, even when autographical. Besides, such symbols should be automatically put in relationship with the spine structure. As regards audio information, an effective application to extract automatically music events from a complex audio track should be employed. Some well known limits in automated music analysis techniques prevented us from reaching this ambitious goal.

On the other hand, acquiring music events by a completely hand-made process would be a terrible waste of time and energy: it would require, for instance, an accurate listening of the audio tracks in a sound editing environment, or the precise computation of the "bounding box" around each music event in a digital imaging software. Calculating milliseconds and pixels by hand is not effective nor efficient.

Our solution was designing and implementing some aiding applications to speed up the mapping process. In particular, two applications were released and employed to feed up MX Navigator with a rich MX file: MX Graphic Mapper and MX Audio Mapper.

MX Graphic Mapper is the application devoted to link the logic events of a MX file to their corresponding graphic counterparts within the score image (see Figure 2). A MX file is opened, scanned and a list of all notes/rests events is created. The user opens one or more images that contain the score and begins mapping all the events by drawing rectangles in the central window. The representation of the note/rest event is graphically shown in the "Event Graphic Parser" and the XML fragment to be added is visible in the upper part of the interface. When a rectangle is created, the application generates

[3] OMR stands for Optical Music Recognition.

the XML line by reading the current event in the "Spine Elements" listbox, and by computing the coordinates of the rectangle. When a new line is added, the current element indicator is moved forward and the mapping process continues.

The MX Audio Mapper is the application used to compute the indexes of events in audio/video clips (see Figure 3). This application is similar to the previous one, but instead of mapping the graphic representations of a piece (scores) it maps its audio/video performances (clips). To map a clip, the MX and the audio/video file are opened, and some parameters are defined: the timing unit per quarter used in the logic layer and the rhythm figure to be processed (quarter, quaver …). The mapping process is achieved by "tapping" the rhythm on a button: when this process is completed, the MX spine is processed and all the timings of the events are computed (in seconds), interpolating figures between two consecutive "taps". After this preliminary procedure, the computed maps can be fine-tuned by hearing the selected map position in the clip, and by adjusting the wrong timings.

As a consequence, our way to compile an MX file is not completely automated nor completely hand-made: it is a compromise that could be defined as a "semi-automated solution", as human intervention is still fundamental, but computer plays an important role.

7. MX NAVIGATOR

MX Navigator is the name of the application that allows an integrated and evolved navigation of music contents. It presents all the aspects we cited before: richness in multimedia description of music, synchronization among heterogeneous contents and easy user interaction.

MX Navigator was installed at the exhibition "Tema con Variazioni. Musica e innovazione tecnologica" [Theme with variations. Music and technological innovation], a voyage into the Italian musical heritage through the rooms of Rome's Music Park Auditorium. One of the purposes of the exhibition was making music tangible and visible bringing together the five senses, not just hearing. In this context we have designed a simple user interface, conceived for not cultured people, in order to listen to and visualize a track alongside variously interpreted scores.

MX Navigator represents the natural evolution of MX Demo [1], an experimental prototype presented in a scientific research context at AXMEDIS 2005 conference. The main differences between MX Demo and MX Navigator are two:

- The latter is a generalized version of the former, as MX Demo was designed only to demonstrate MX format possibilities and worked on the limited number of pieces consequently chosen, whereas MX Navigator is virtually able to open any MX file. It is

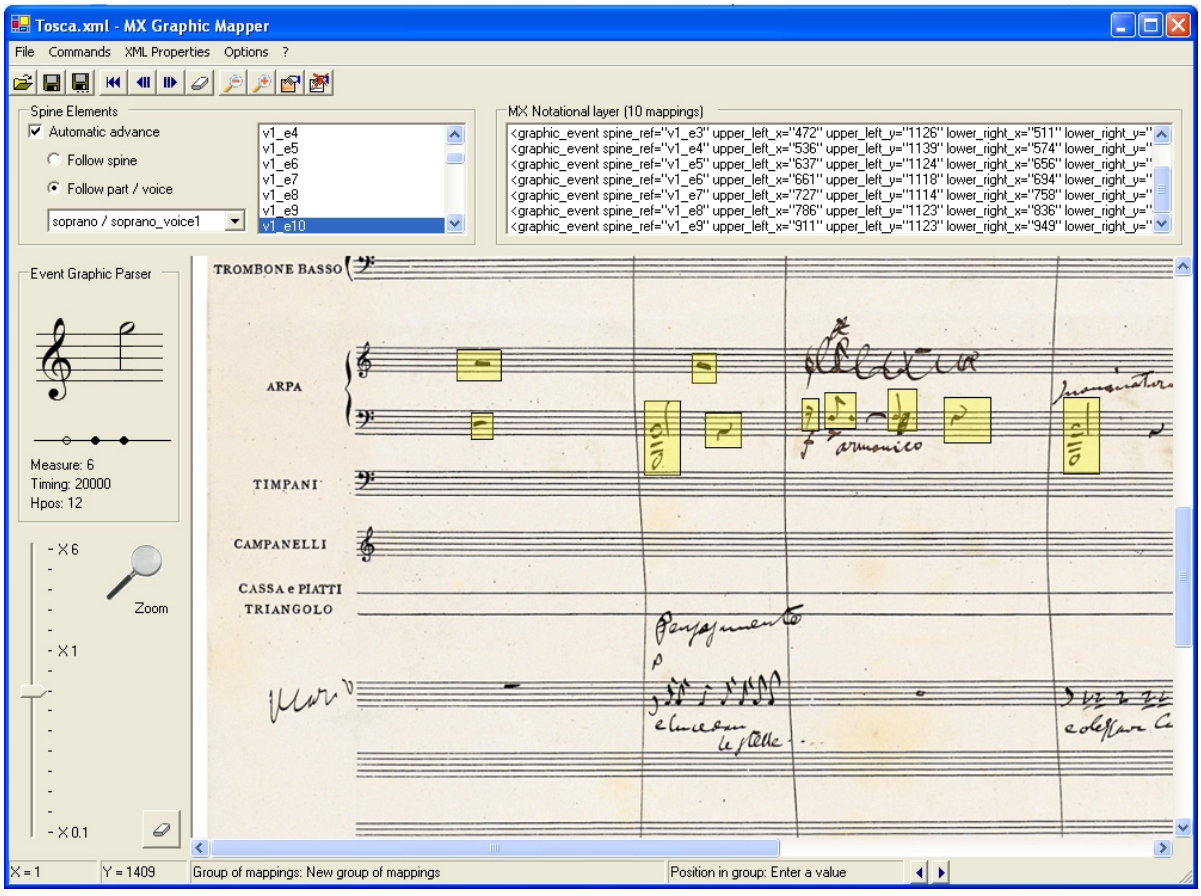

Figure 2. Screenshot of the MX Graphic Mapper application.

able to adapt itself in real time to the different kind and number of media contained.

- MX Navigator is necessarily characterized by an improved usability and by a simpler user interface. In fact, MX Demo had to be presented to a scientific audience by its authors, while MX Navigator is at common users' disposal in a public exhibition.

MX Navigator allows an integrated fruition of different media contents related to the same music piece. For the exhibition held in Rome, we translated into MX the aria "E lucevan le stelle…" from Puccini's Tosca – III Act. The choice of that piece was imposed by the leitmotiv of the exhibition "Tema con Variazioni", nevertheless MX Navigator could open and play any MX file.

Apart from a number of multimedia objects such as greeting cards, playbills, historical photos and sketches, the software mainly shows two versions of the score (an autographical and a printed one), a video and two audio performances of Puccini's aria.

The central part of the interface contains the score, since this is the main media in terms of interaction and even because of its visual extent.

The upper left part of the window contains controls related to audio/video interaction. In this application there are three different clips: an audio clip of a 1953 version performed by the Italian tenor Di Stefano and both an audio and video clip of the 1984 version performed by Aragall at the Verona's Arena. When a version is chosen and played, the music and/or video is executed, and in the score window a red rectangle indicates the event being played by the clip (thanks to the synchronization achieved by the MX Audio Mapper and the MX Graphic Mapper). Since there are many parts in the score (all coded into the MX file), it must be possible to select the current part to follow. This application is intended to be used even by non-musicians, so this choice is simplified, and only two parts can be selected. To do this in the top of the interface there are two buttons that control which part is to be followed by the rectangle in the score, the Clarinetto or the Tenor (Cavaradossi) one.

In the top of the interface, the controls are devoted to manage the score visualization. In this MX file there are two types of mapped scores: the autographical Puccini version and the Ricordi published version. The upper left buttons switch (even when running an audio/video clip) between this two scores. The upper right buttons control the zoom (50% or 100%) and the current page displayed (only in the Pause state, otherwise the current page is automatically selected to follow the audio/video clip execution).

The left bottom part of the interface is simply used to open many graphical elements related to the piece.

This visual interface allows a number of different ways to enjoy music. First, it is possible to select a score version, an audio track, a leading instrument and simply follow the instrument part evolution. This is a first original degree in music fruition, as music can be listened and watched in a synchronized fashion. But a second way to enjoy music through MX Navigator is even more interesting: it consists in switching from an aural/visual representation to another. In other words, it is possible to compare in real time different versions of the score (the hand-made and the printed one) or

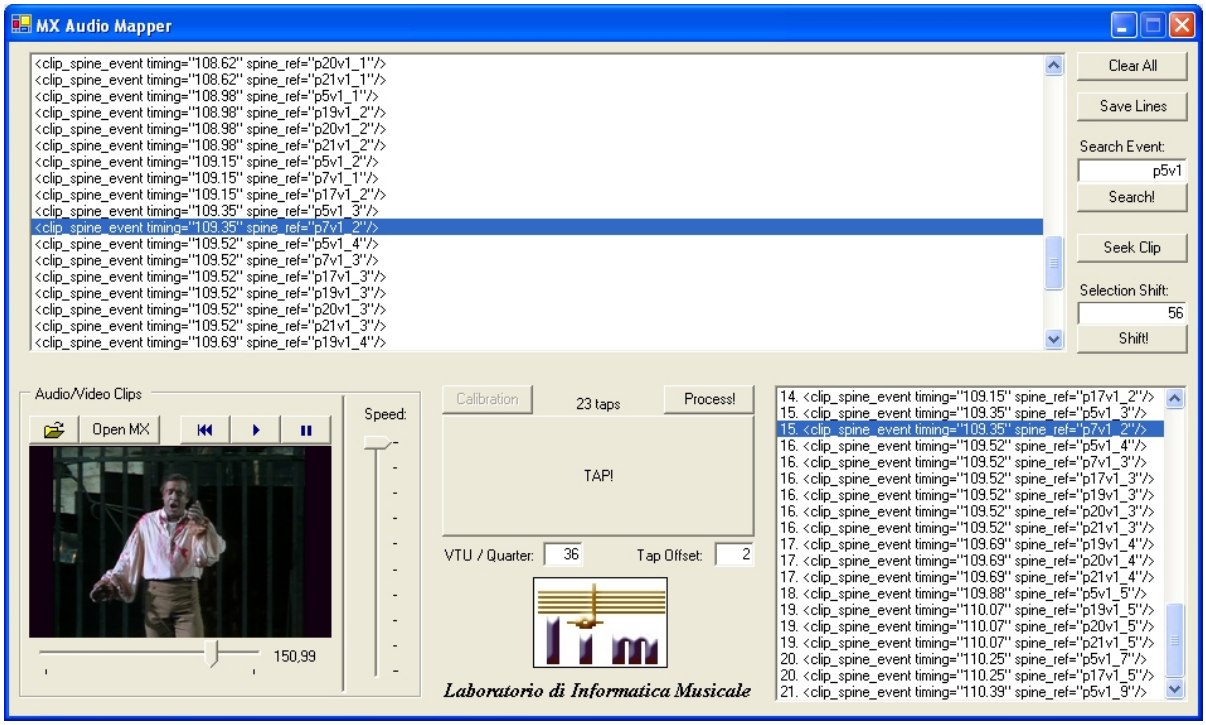

Figure 3. Screenshot of the MX Audio Mapper application.

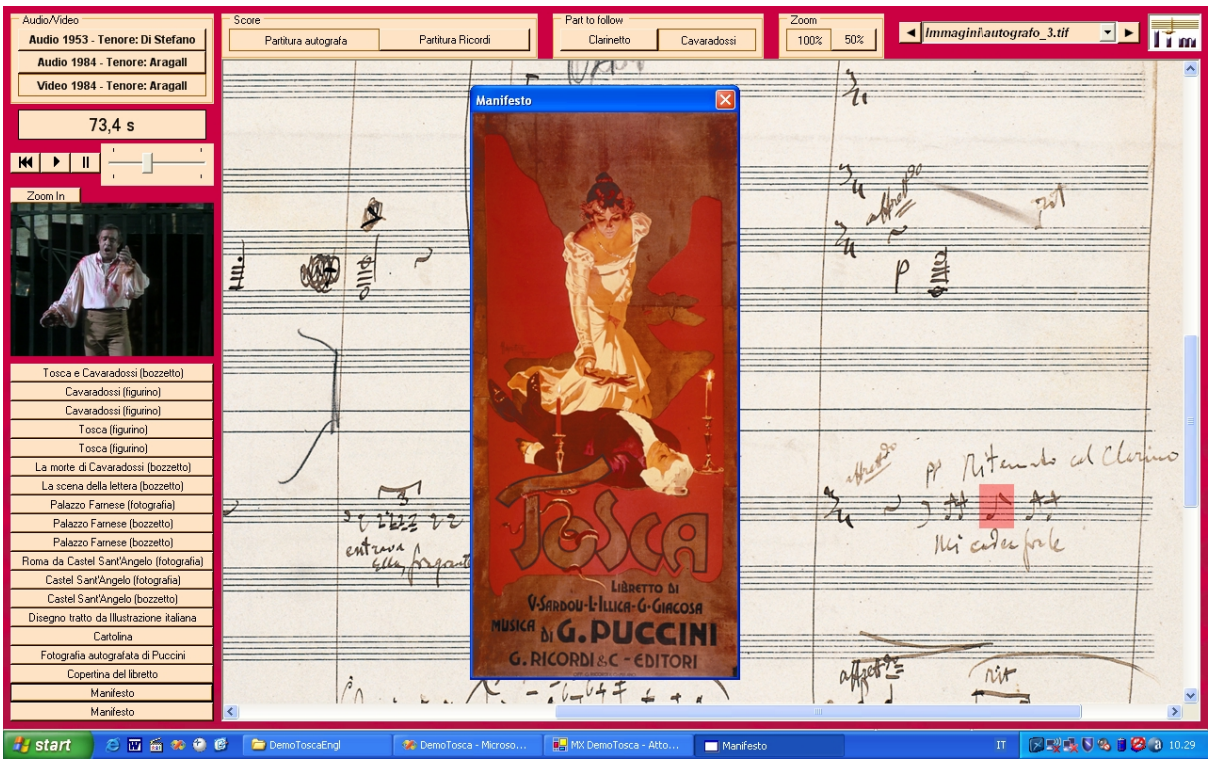

Figure 4. Screenshot of MX Navigator, installed at the exhibition "Tema con Variazioni. Musica e innovazione tecnologica" in Rome.

different performances. When the user decides to switch from a representation to another, MX Navigator continues from the point previously reached. Finally, the application suggests a third way to enjoy music, that consists in altering the original sequence of music events. It is possible to jump – forward or back – from a point to another point of the score, both in its visual and aural representations; of course, the effect will be the same as the former and the latter are synchronized.

8. CONCLUDING REMARKS

The MX Navigator version we presented at the exhibition "Tema con Variazioni" is an interface conceived for the spread of art and music among people not necessarily cultured. The application is characterized by a comprehensive multimedia description of music, synchronization among heterogeneous contents and easy user interaction.

Since our tool is working on XML data, a possible future application will be to have it as a client-side script of a standard browser.

We think that MX Navigator represents an important experiment in the field of cultural heritages, as users can enjoy music also in non-traditional ways and they can intuitively interact with musical contents.

9. ACKNOWLEDGMENTS

The authors want to acknowledge researchers and graduate students at LIM, and the members of the IEEE Standards Association Working Group on Music Application of XML (PAR1599) for their cooperation and efforts. Special acknowledgments are due to: Denis Baggi for his invaluable work as working group chair of the IEEE Standard Association WG on MX (PAR1599); Maria Pia Ferraris and Cristiano Ostinelli (Ricordi) for their fundamental contributions as regards the original material (autographic and printed scores, sketches, pictures, photos) used in MX Navigator.

10. REFERENCES

[1] Baratè, A.., Haus, G., Ludovico, L. A., and Vercellesi, G., "MXDemo: a Case Study about Audio, Video, and Score Synchronization", *AXMEDIS 2005 Proceedings*, Firenze, Italy, 2005.

[2] Haus, G., "Recommended Practice for the Definition of a Commonly Acceptable Musical Application Using the XML Language", *IEEE SA 1599*, PAR approval date 09/27/2001, 2001.

[3] Haus, G. and Longari, M., "A Multi-Layered Time-Based Music Description Approach based on XML", *Computer Music Journal*, MIT Press, Spring 2005

[4] Roland, P., "The Music Encoding Initiative (MEI)", *Proceedings of IEEE 1st International Conference MAX 2002 - Musical Application Using XML*, IEEE, Milan, Italy, 2002