A COMPUTER TOOL TO ENJOY AND UNDERSTAND MUSIC

D. L. Baggi*, A. Baratè†, L. A. Ludovico†, G. Haus†

*Scuola Universitaria Professionale della Svizzera Italiana, SUPSI, Manno, Switzerland, denis.baggi@supsi.ch
† Laboratorio di Informatica Musicale, LIM, Department of Information and Communication, University of Milan, Italy

Keywords: XML, Symbolic Music, Music Standards, Musicological Research.

Abstract

New technology for the *representation of music in symbolic form* (such as IEEE Standard P1599 developed by the authors) allows the realization of *computer tools* that transform the activity of *listening to music* in an *enjoyable learning experience.* This article gives a brief description of the standard and its features and of a simple browser realized to illustrate this technology.

1 Introduction

Music is much more than *audio.* For music lovers, the musical experience is not merely that of buying an audio CD and listening to it. It is, instead, the act of *entering a new world*, *living an experience*, *understanding a narration* and *recognizing images*, with the possibility of *investigating how the whole is built* – which is the object of *musicology*, the science of music.

To this end, music must be represented with something that goes beyond *unreadable, binary standards for audio,* such as WAV and MP3, which are not music standard. Musical *non-audio layers* have to be represented as well, in human-readable form such as *symbols* and *characters,* as it has always been for a *music score* (in classical music), or in *notation* such as the *harmonic grid* (as in jazz), or in anything else as it is the case in *non-Western music*.

In the notational standard ***IEEE P1599***, the language *XML* is used, which allows inherent *readability, extensibility* and *durability.* This technology represents a foundation for the development of applications that allow a full musical experience.

For these reasons, this article will be subdivided in the following paragraphs:

- Past and present standards for symbolic music
- Description of IEEE P1599
- Example of a music browser
- Other possible applications

2 Past and Present Standards For Symbolic Music

The idea of representing music with symbols is not new. If we consider music notation, it goes back several centuries in the West, and for computer applications, several decades, as shown by the *Plaine-And-Easie Code* [1] and *DARMS* [2].

Closer to us, attempts have been made to use the new technology brought about by SGML, a subset of which has been defined for music, namely, SMDL, or *Standard Music Description Language* [3]. Though well defined, it failed to attract much attention because of lack of applications.

Presently, there are some de-facto standards using XML, two of which are important for this work. *MusicXML* is a proprietary standard by company Recordare [4] and is used in dozens of existing applications on the market (including popular Finale). This standard has been in existence for over four years, and new Version 1.1 has just been released. The *Music Encoding Initiative,* or *MEI* [5], is a project by the Digital Library of the University of Virginia and it has been used in a few projects of music encoding.

3 Description of IEEE P1599

This new standard builds on existing efforts such as MusicXML and MEI, maintains compatibility with them and also with the core RM0 of *ISO/IEC JTC1/SC29/WG11* for moving pictures and audio in the making. This is represented in Fig. 1.

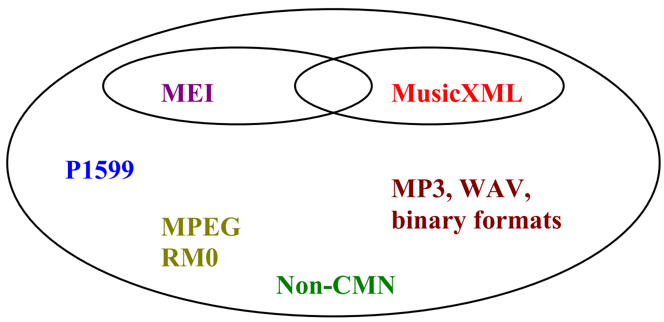

Fig. 1. P1599 Compatibility Diagram.

Fig. 1 explains that P1599 is compatible with MusicXML, MEI and MPEG RM0. Audio binary standards are used as *audio layers* [6].

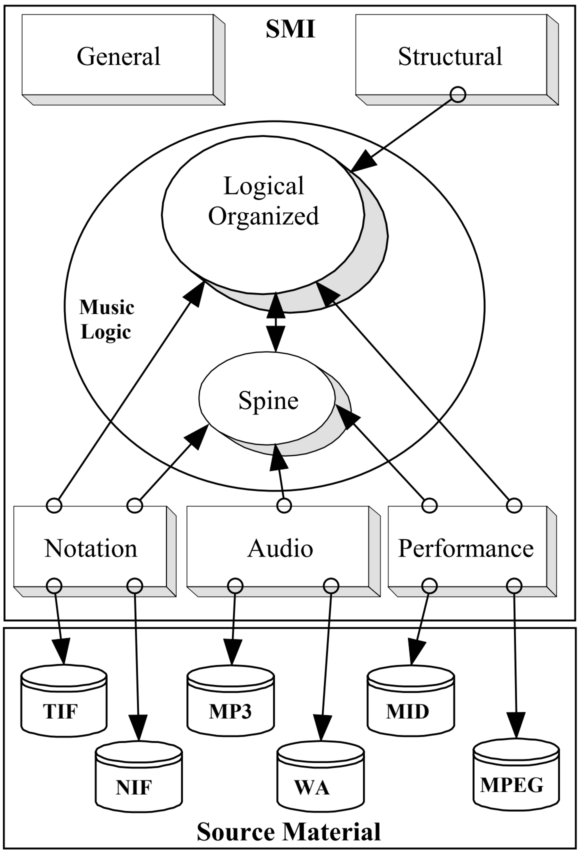

Fig. 2. Relationships between SMI layers and Source Material

Fig. 2 shows the relationship between the group constituted by *general*, *structural*, and *logic* layers and that embracing *notational* (e.g., TIFF, JPEG, GIF for a score), *audio* (e.g., MP3, WAV), and *performance* (e.g., MIDI) layers. This means that, in respect to the *representation of music*, the standard chosen to explicit, e.g., its audio contents, is *irrelevant*, since the *same logical symbol file* is used.

Fig. 3 shows the subdivision of layers to represent music in P1599. Though in reality not all layers are always present, the figure represents a typical example.

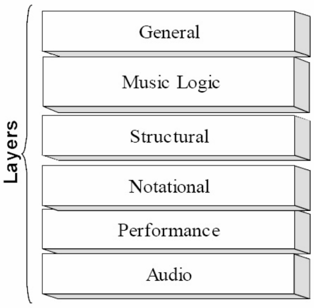

Fig. 3. Example of layers

The *logic layer* contains among others a data structure called *spine*, shown in Fig. 4, that relates *time* and *spatial information*. With such a structure, it is possible to move from a point in a notational instance, e.g., at the top of the figure, to the corresponding point in a performance or audio instance below.

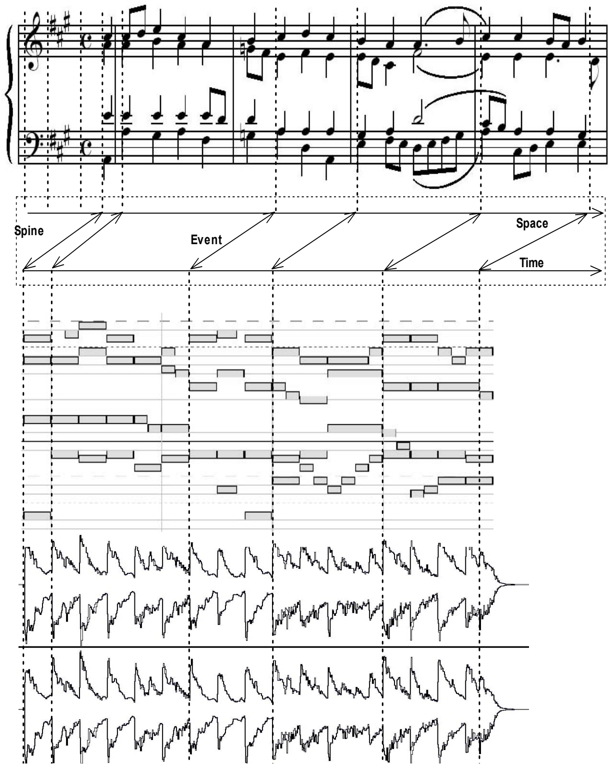

Fig. 4. Example of a *spine*, visualization of space–time relationships.

The existence, or shape, of layers depends on the type of music: a score would make no sense in e.g. *African drum music*, while a harmonic grid is senseless in *Gregorian Chant* or *Indian music*. The *spine* exhibits links among different representations, allowing navigation among layers. Applications could, in theory, reconstruct some layers from others (e.g., in some cases, build score from audio).

Each layer has a role in the context of music integration, since XML organizes information in a hierarchical structure. A layer is represented in a sub-tree of the global representation, where relationships among sub-trees are implemented by means of XML references, while relationships between SMI and Source Material are implemented as descriptions of extracted parameters.

Fig. 2 to 4 show details describing a particular implementation as follows. The *General Information Layer* does not link directly to any layer, since it describes information about the whole Musical Work. Its XML representation is straightforward and does not need any particular description. The *Structural Layer* represents music objects – parts of the *Music Logic Symbols* layer – and relations among them. The *Notational Layer* is where visual information can be represented. From a logical perspective, it can be divided in two sub-layers, *layout description* and *linking information*: the former provides tools to describe a screen layout that links directly to the Logical Organized Symbols and the Spine events, while the latter consists of descriptors for mapping score images and binary notation file formats (in various encoding, such as NIFF and Enigma) to the Music Logic Information layer.

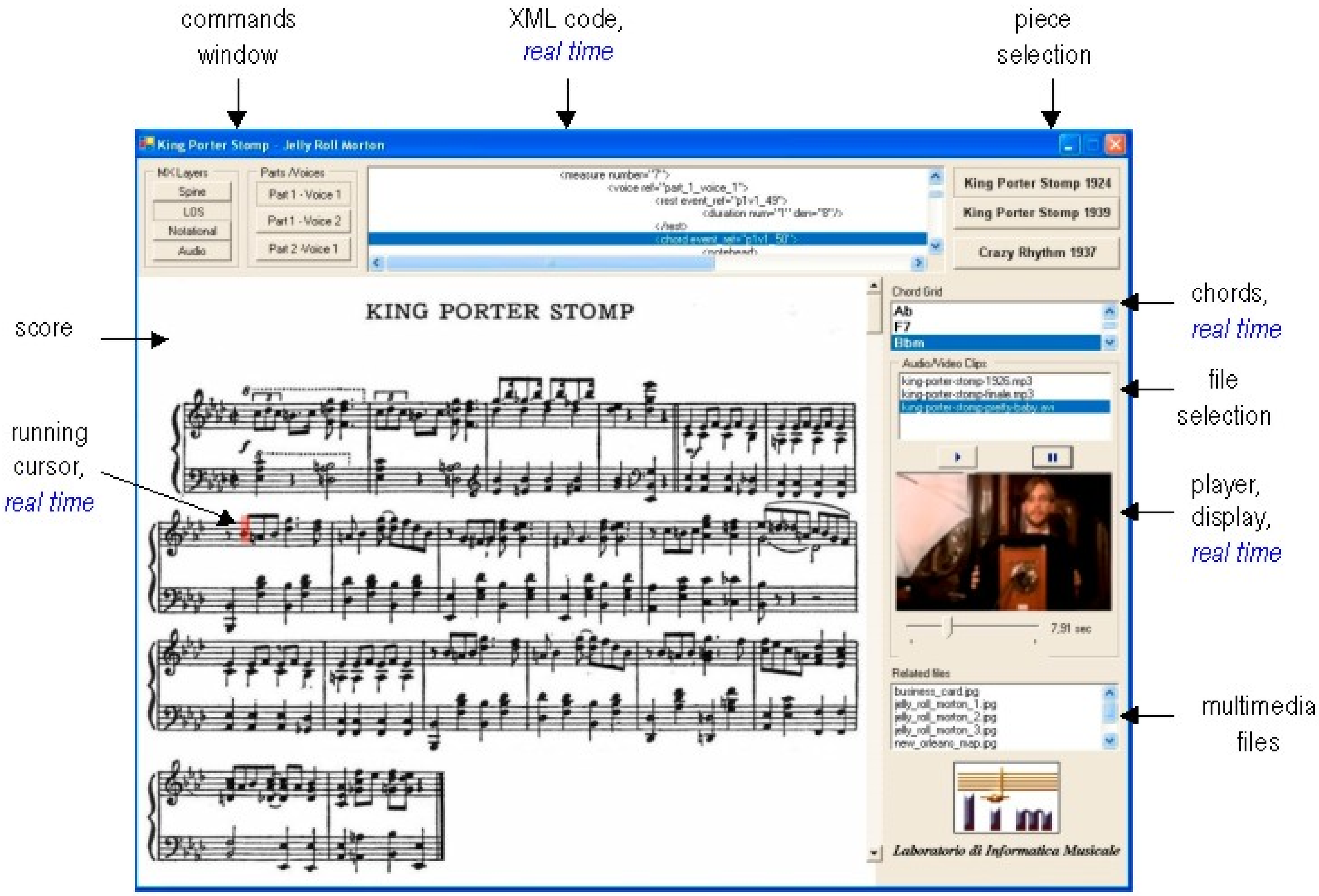

Fig. 5. Screenshot of a music browser

4 Example of a Music Browser

This is an application built at the *Laboratory for Musical Informatics* at the University of Milan, Italy, to illustrate how the standard for symbolic music works, and its power. The screenshot of Figure 5 contains different windows, of which those with the extra caption *real time* operate in synchronism while the music is being played.

The user starts with the *piece selection* window: in this case, there are three choices, *King Porter Stomp 1924, King Porter Stomp 1939,* and *Crazy Rhythm.* The first two refer to two published scores of that piece, once famous, by American composer and pianist *Jelly Roll Morton*, *or Ferdinand Joseph La Motte*, 1889-1941, while the latter is an improvised jazz piece for which there is no score.

In the *file selection* window, the user can choose among alternate multimedia files and renditions, in this case a recording from 1926, transformed in MP3 format, a MIDI rendition of the 1924 score encoded in MP3, and an excerpt from Louis Malle's movie *Pretty Baby* of 1977, in which a character patterned after Morton is heard composing this very piece in the background. The latter, i.e. the movie, is the one shown here in the window *player, display*, which for plain music shows instead a common player.

Upon that selection, several synchronized activities start and execute in real time. The music starts playing, and in this case the movie segment starts, with its sound. On the *score* windows, the *running cursor* indicates what is being played, here the beginning of the 7th bar. The user can click on and move the red cursor with the mouse and initiate playing from another point in the score while the other real time windows instantly adjust synchronously, and of course move the player cursor.

The *XML code* window shows the encoded events, in this case those of the LOS, Logical Organized Symbols of Figure 2, scrolling with the music. In the *command window,* the user can select which XML code is displayed: *spine, LOS, notational* and *audio,* again those of Figure 2, and in the same window he can choose which *voice* the running cursor will follow – there are three available voices in this case.

The *chords* window displays the elements of the music harmony of the piece, again synchronously with the playing, and the window for the *multimedia files* allows selection of pictures or other, portraits of Morton, of his band, and curiosities.

The screen for jazz piece *Crazy Rhythm,* not shown here, displays the harmonic grid, pointed to by the running cursor, and lets picture and name of each soloist pop up at the appropriate moment. There are four saxophonists taking solos, among others top Afro-American jazz musicians Coleman Hawkins and Benny Carter, who lived in Europe in the mid-1930's.

The browser shows how music represented with *readable symbols* can be accessed and manipulated; and that only one single XML file is needed for *several renditions of the same piece*, hence the format used for reproducing the audio is irrelevant.

5 Other Possible Applications

The same methodology can be applied to several other musical contexts, such as:

I. **An opera**. A DVD of an opera encoded with the standard allows the user to: *see the play* on the screen; *hear the music*; *see the score*; *read the libretto*; all in a linked and *synchronized* way; see *excerpts of alternative renditions*, e.g. arias sung by another interpreter, or segments by another conductor.

II. **A piece by a jazz Big Bang**. Again the orchestra can be seen, and the *harmonic grid* followed (rather than the score); the *name of the soloist* pops at the beginning of his solo (no more arguing, as people did in the early 30's, whether it is Joe Smith or Tommy Ladnier taking the trumpet solo in Fletcher Henderson's orchestra); moving the mouse allows exploration of all members of the orchestra, including the rhythm section; *alternate renditions* of the same piece can be examined [7].

III. **A fugue.** The ability of the interpreter consists, among others, in *extracting* and *making explicit* the *theme* as it gets passed among the different voices. This could be done both aurally (different timbre, loudness) and graphically (different colour in the score.)

IV. **Music with a "program" or story.** Examples are Prokofiev's *Peter and the Wolf,* Saint-Saëns' *Carnival of the Animals.* For his *Four Seasons,* composer *Antonio Vivaldi* wrote poems with lines that specifically refer to segments of the music. A music browser would make these connections immediate, showing what is happening.

V. **A piece of Indian classical music**. The particular *scale* of the raga is made available in graphic form to the user, and every time the player "refreshes" the musical memory by playing it, a signal is issued. The melodic development is highlighted with a two-dimensional diagram, meant to underline the creation of the *climax* and the *changes of mood*, to enable the listener to follow the *unfolding story* the musician is conveying.

VI. **A piece of several drums,** as in **African Drumming**. The user sees, for e.g. five drums, a five-line *graph* illustrating the playing and accents of the drums, and he is made aware that the various hits *do not fall exactly on the beat,* a technique that could be called the *ancestor of swing.* This has been researched in detail with a Cuban Guaguanco group at the Massachusetts Institute of Technology [8].

VII. **Preservation of the music heritage from the past**. Previous work by one author of this writing [9] has consisted of the creation of a database with records of variable formats that contain a representation of the music piece, several ways for accessing it, or other performances of the same piece, in the usual way with SQL, or more spectacularly by simply whistling a segment of the main leitmotiv.

VIII. **Musicological study**. The symbolic representation of this standard makes it easy to start a query to find pieces of music with given characteristics, notes, or other – such as, find all piano pieces containing the lowest note of a grand piano.

Needless to say, these are only examples. Ideas for new possible applications arise at any moment, in particular in contexts for kinds of music, such as those studied by *ethnomusicology*, where there has traditionally been little investigation and methodological research.

6 Conclusion

The need for representing music with symbols, a tradition that goes back several centuries, cannot be stressed enough. It is high time to progress from *closed binary audio standards* to *open symbolic representations for music.*

This writing attempts to show that the standard sponsored by the IEEE is not merely a project. It is, instead, the development of a new *enabling technology* that serves at the base for the realization of unending new applications. The natural way to represent music is with *symbols*, a tradition that goes back several centuries. It is time to progress from *closed binary audio standards* to *open symbolic representations for music.* IEEE P1599 is the development of an *enabling technology* that serves as the base for the realization of unending new applications.

While music can be appreciated by just listening to it, the attentive listener senses that there is whole world beyond the sound just waiting to be explored. While this world can be discovered thanks to several years of formal, disciplined study, this kind of applications make its discovery available to every listener, raising interest even among those with no love for music, thus taking the listener *from music enjoyment to music education*. Thus, they represent an example of how technology can allow both users and specialists to discover the deep meaning of music, and generate *connections to all aspects of culture*, from art enjoyment to contemporary life.

Acknowledgement

We thank the ***IEEE CS Standard Activity Board*** for sponsoring this project, the ***Swiss IMS (Intelligent Manufacturing Systems) Secretariat*** for its financial support, and ***LIM,*** the ***Laboratory for Musical Informatics*** of the University of Milan, for its work.

References

[1] B.S. Brook, "The Plaine and Easie Code", in *Musicology and the Computer*, ed. Barry S. Brook. New York: City University of New York Press, 1970 pp53-56.

[2] DARMS, Digital Alternate Realization of Musical Symbols: R.F. Erickson, "The Darms Project: A Status Report", *Computers and the Humanities* 9(6):291-298 (June 1975).

[3] S.R Newcomb, "Standard Music Description Language complies with hypermedia standard", *IEEE COMPUTER*, July 1991, pp. 76-79.

[4] http://www.recordare.com/xml.html .

[5] P. Roland, "The Music Encoding Initiative (MEI) DTD and the Online Chopin Variorum Edition", http://www.lib.virginia.edu/digital/resndev/mei/mei_ocve.pdf

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

[7] D. Baggi, "Capire il Jazz: le strutture dello Swing", *Surveys of the institute CIMSI of SUPSI*, Manno, Switzerland, 2001.

[8] J. Bilmes, "Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm", *Master's Thesis*, Massachusetts Institute of Technology, Cambridge, MA, 1993.

[9] G. Haus, "Rescuing La Scala's Audio Archives", *IEEE COMPUTER*, 31(3), pp.88-89, IEEE CS Press, Washington, 1998.