Implementing the Enhancing Music Addressability API for MusicXML

Kevin Kuo, University of Maryland
Raffaele Viglianti, University of Maryland

Introduction

The Enhancing Music Addressability API (EMA)[1] provides a specification for addressing exact portions of a digital score via a web URL. This work is the result of a now-completed project (Viglianti 2016) that created, in addition to the specification, an implementation for the Music Encoding Initiative (MEI) format[2] and a large Linked Open Data test collection based on the analytical work of the *Lost Voices: Digital Du Chemin* project.[3] Even though the only implementation of EMA so far can exclusively target MEI, the specification itself aims to be format-agnostic. This poster reports on a new implementation of EMA for the MusicXML format, which is the result of Kevin Kuo's independent study work at the Maryland Institute for Technology in the Humanities at the University of Maryland.

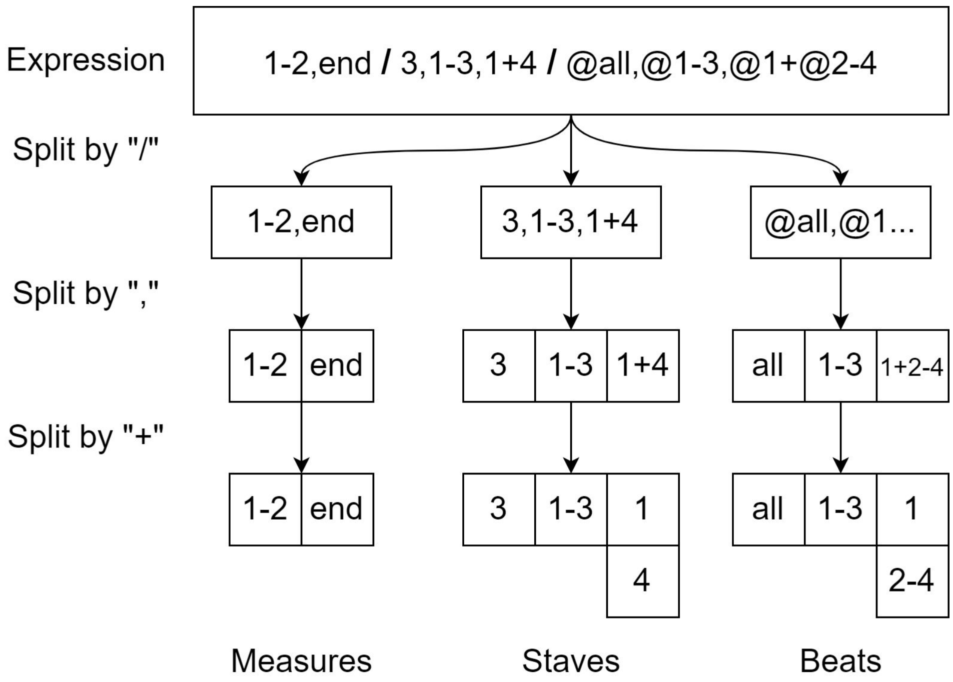
EMA originated from the lack of a standardized system to address areas of digital music notation. In order to create a generalizable specification, the project applied Michael Witmore's concept of text as a "massively addressable object" (Witmore 2010) to music notation. Texts are massively addressable because one can apply different levels of abstraction to them; with digital text, this idea of addressability expounds the selecting, rearranging, and manipulating of texts for research purposes. Music notation can be considered in very similar terms: one can identify, group and distinguish areas of a musical score to which to ascribe meaning. This is common both in preparation for a performance (e.g. by circling a passage) and in music scholarship based on the music score (such as many forms of analysis and historical research that will refer readers to the music notation). However, the concept of line and character that enables digital addressability for text does not apply to music notation as there are many different ways of representing music notation computationally. In response to this limitation, EMA provides a system for selecting music notation based on commonly understood primitives: measures, staves, and beats. This is not the only possible way of addressing digital music notation; for example formats expressed in XML such as both MEI and MusicXML can be targeted using XML-specific mechanisms. The Multimedia Access to Music Digital Libraries (MELD) (Page *et al*. 2019), for example, provides a rich linked data framework for "combin[ing] music-related materials including text, audio, video, images, facsimiles, and music scores'' where references to portions of MEI documents are handled through direct ID references or XPath selectors. In

[1] https://umd-mith.github.io/ema/api/.
[2] https://github.com/umd-mith/ema/tree/master/Omas.
[3] Project: http://digitalduchemin.org/; Linked Open Data collection: http://digitalduchemin.org:8080/nanopub-server/nanopubs.html.

comparison, the EMA specification stands out for its design based on musical concepts, which makes it format-agnostic, extensible to non-XML formats, and capable of making the same selection expression retrievable across multiple representations of the same score.

A new parser

In order to provide support to the MusicXML implementation, we have developed a new generic parser for EMA expressions.[4] The parser checks if an expression adheres to the EMA specification; if so, the expression is split into the nested list structure illustrated below.



Because EMA allows users to input non-numeric tokens (“start”, “end”, and “all”), the expression cannot be fully evaluated without information about the targeted digital score. Variables such as the starting and ending measure numbers, the number of parts, and the placement of time signature changes, are all needed in order to address a section of music.

The testing suite for the parser utilizes the Linked Open Data collection from the *Digital Du Chemin* project. During testing, we scrape all the EMA expressions listed in the database and attempt to parse each expression as shown above.

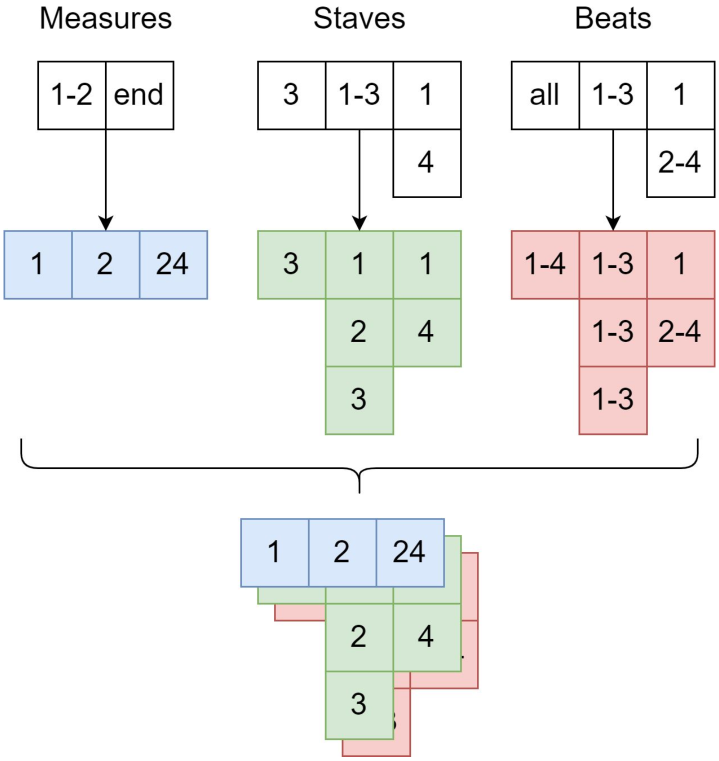
EMA for MusicXML

Because of MusicXML’s prominent role as an interchange format between music notation software, there is a large number of freely accessible MusicXML scores on the web, such as the

[4] The code is provisionally located at https://github.com/imkevinkuo/ema2.

Petrucci Music Library (IMSLP)[5] or the collection of scores by the MuseScore community of users.[6] Extending the functionality of EMA to MusicXML data is therefore a priority, given the potential of expanding the applicability of EMA to a wider number of use cases.

The implementation for MusicXML first evaluates and expands the data structure created by the parser. We expand ranges denoted by a dash (e.g. 1-3) into enumerations (1,2,3) and replace non-integer tokens based on structural attributes of the score such as measure, staff, and beat counts. Finally, to make the MusicXML selection, we traverse a copy of the XML file and remove notes that have not been selected.



Based on the example in the previous image. The ending measure number is 24 and there are 4 staves.

The evaluation of this tool is still in progress. We plan to use freely available MusicXML scores and to craft specific examples. Additionally, like for the parser, we plan to use the *Digital Du Chemin* database for testing. The MEI scores and selections will be converted to MusicXML documents, which we will then use to evaluate the accuracy of the slicer. However, lossy conversion from MEI to MusicXML may hinder performance on these documents.

Conclusions

The work presented in this poster has two major aims: to demonstrate the applicability of EMA to other formats besides MEI and to provide a functioning implementation for music scores in

[5] https://imslp.org/.
[6] https://musescore.com/

MusicXML, a popular interchange and archival format. The new EMA parser, moreover, paves the road for future improvements to existing implementation for MEI and enables future work on extending EMA's addressability to other music notation formats.

References

Kevin R. Page, D. Lewis and D. M. Weigl. 2019. "MELD: A Linked Data Framework for Multimedia Access to Music Digital Libraries," *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, Champaign, IL, USA, 2019, pp. 434-435. DOI: 10.1109/JCDL.2019.00106

Raffaele Viglianti. 2016. "The Music Addressability API: A draft specification for addressing portions of music notation on the web". In *Proceedings of the 3rd International workshop on Digital Libraries for Musicology (DLfM 2016)*. Association for Computing Machinery, New York, NY, USA, 57–60. DOI:https://doi.org/10.1145/2970044.2970056

Michael Witmore. 2010. "Text: a massively addressable object". Blog post in *Wine Dark Sea: literary and cultural history at the level of the sentence*. URL: http://winedarksea.org/?p=926