Implementing the Enhancing Music Addressability API for MusicXML

Kevin Kuo, Raffaele Viglianti

INTRODUCTION
The ability to "address" areas of a musical score is useful in music scholarship such as analysis and historical research. In this project, we implement software that enables us to "select" regions of MusicXML files, in accordance with the Enhancing Music Addressability (EMA) specification.

EMA for MusicXML: [https://github.com/music-addressability/ema-for-musicxml](https://github.com/music-addressability/ema-for-musicxml)

PARSING EMA EXPRESSIONS
An "EMA expression" is a text sequence of the format

```
{measureRanges}/{stavesToMeasures}/{beatsToMeasures}
```

- **measureRanges**: Comma separated ranges of measures.
- **stavesToMeasures**: Staff ranges separated by + signs and mapped to measure ranges with commas.
- **beatsToMeasures**: Beat ranges marked by @ signs. Mapped to staff ranges by +, and mapped to measure ranges with commas.

XML SLICING
MusicXML is based on XML, a tree-based markup language. Given an EMA expression, we can traverse a music score (represented in XML) and check whether a measure/stave/beat should be selected.

EXAMPLE SELECTION

```
2,3 / 1 + 2,3 + 4 / @all
```

Figure 2. A sample score. The regions we want to extract are boxed in red.

Figure 3. The score output from our software after selection is complete.

ACKNOWLEDGEMENTS

MIT
Maryland Institute for Technology in the Humanities

Purdom Lindblad
Assistant Director of Innovation and Learning, MITH
INTRODUCTION

The ability to “address” areas of a musical score is useful in music scholarship such as analysis and/or historical research. In this project, we implement software that enables us to “select” regions of MusicXML files, in accordance with the Enhancing Music Addressability (EMA) specification.

EMA Homepage: http://music-addressability.github.io/ema/
EMA for MusicXML: https://github.com/music-addressability/ema-for-musicxml

EMA API

http://.../score.xml/2,3/1+2,3+4/@all

Extracted score portion

There are many different formats to computationally represent music notation, such as MEI, MusicXML, etc.

To address this limitation, the EMA standard provides a system for selecting music notation based on commonly understood primitives: measures, staves, and beats.

Implementations of EMA can run on a user’s local machine or on a remote server as a web service.
**PARSING EMA EXPRESSIONS**

An “EMA expression” is a text sequence of the format:

```
"{measureRanges}/{stavesToMeasures}/{beatsToMeasures}"
```

- **measureRanges**: Comma separated ranges of measures.
- **stavesToMeasures**: Staff ranges separated by + signs and mapped to measure ranges with commas.
- **beatsToMeasures**: Beat ranges marked by @ signs. Mapped to staff ranges by +, and mapped to measure ranges with commas.

**XML SLICING**

MusicXML is based on XML, a tree-based markup language.

Given an EMA expression, we can traverse a music score (represented in XML) and check whether a measure/stave/beat should be selected.

```
<measure number="2">
  <note>
    <pitch>
      <step>E</step>
      <octave>4</octave>
    </pitch>
    <duration>60480</duration>
    <type>whole</type>
    <lyric>
      <syllabic>end</syllabic>
      <text>-tez</text>
    </lyric>
  </note>
  ...
</measure>
```

**ACKNOWLEDGEMENTS**

MITH
Maryland Institute for Technology in the Humanities

Purdom Lindblad
Assistant Director of Innovation and Learning, MITH
Figure 1. An EMA expression divided into musical components.

Figure 2. A sample score. The regions we want to extract are boxed in red.

Figure 3. The score output from our software after selection is complete.