

Music Notation Markup for the Web Ecosystem

Joe Berkovitz (joe@noteflight.com)

President, Noteflight LLC; Co-chair W3C Web Audio WG



Why Music Markup at all?

- Enable music application ecology
- Document persistence and portability
- Dynamic rendering and synthesis
- Analysis, transformation, indexing...

A Bit of History

- Pre-ML: Humdrum, MuseData, ABC, LilyPond
- App-neutral: MusicXML, MEI
- New music font standard: SMuFL

Music Markup Challenges

- Default rendering decisions based on semantics alone are tricky and hard to codify
- Some human engraving decisions matter
- Ignore human decisions: poor rendering
- Respect them: rigid rendering
- Accommodating and balancing semantic, visual and auditory facets is necessary

Musical Markup Today

- Applications must exchange musical documents in some open format
- It's expensive to build any markup parser/generator, let alone an app
- MusicXML has broad developer support and is relatively mature
- MEI has some advantages but lacks broad support, chiefly used in academia
- A number of other standards attempts have failed

Why Do Anything Here?

- We want the Web to encompass, conserve and transform our global culture
- Music and its notational systems are part of it
- Web specs form an ecosystem that is more than a sum of parts (look at HTML, CSS, SVG, MathML...)
- Today's music notation markup chiefly reflects the concerns of print, though increasingly used online

A Starting Point

- Form a Music Notation W3C Community Group
- Capture present-day MusicXML as a CG report
- Begin work evolving a successor that is attuned to the Web ecosystem
- Base on MusicXML vocabulary and concepts to leverage existing technology investments
- Draw on wisdom from other standards

What might a “Web-oriented” music markup standard allow?

- Styling (both visual and auditory)
- Interactivity and Selection
- DOM Manipulation
- Anchors and Pointers
- Accessibility
- Annotations
- Web Fonts

What might a “Web-oriented” music markup standard allow?

- Mixing/Embedding Markup Languages
- Inclusion of embedded media
- Synchronizing with media
- Host environment hooks (e.g. copy/paste)
- Linkage to “Web of Data”