Proposals for MusicXML 3.2 and MNX Implementation from the DAISY Music Braille Project

11 March 2019

Author: Haipeng Hu, Matthias Leopold
With contributions from: Jostein Austvik Jacobsen, Lori Kernohan, Sarah Morley Wilkins.

This document provides general suggestions on the implementation of MusicXML 3.2 and MNX formats, hoping they can be more braille-transcription-friendly. But it doesn't limit to the requirement of music braille transcription. Some of the points may also benefit the common transformation of musical data to be more accurately and clearly.

Note: The word “text” in this document specifically means all non-lyrics texts, such as titles, composers, credit page information, as well as expression / technique / instructional texts applied to a staff or the whole system.

A. MusicXML 3.2 Suggestions

1. If possible, implement a flow to include all system-wide texts (such as tempo texts, stage indications), lines (such as a line with text “poco a poco rit.”) and symbols (such as a big fermata or breath mark above the whole system), like the "global" section of MNX, so that all things in it can affect the whole score instead of the topmost staff or (in orchestra) piccolo and first violin staves. This can also ease part extraction. The position of the elements can be placed using the forward command. (e.g. in SATB choir pieces: sometimes breathing marks, dynamics and wedges are only written once in Soprano but are meant to apply to all voices; and in orchestral scores, tempo texts and overall indications are written above piccolo and first violin; and some big fermata or breath marks appear above every instrument category).

Proposition:

```
<dynamics scope=“system“><fff/></dynamics>
<breath-mark scope=“staff“/>
<directive scope=”system”>cresc.</directive>
```

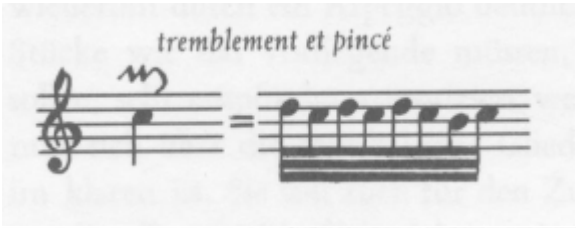
2. Create a category attribute for words and lines, to include both default and user-defined categories of texts and lines (the specific names of such texts and lines could be derived from notation softwares' internal definitions, such as technique, stage direction etc. which can be found in Sibelius), so that the braille transcription software can determine how to convert them in braille (correct equivalents and positions according to music braille convention).

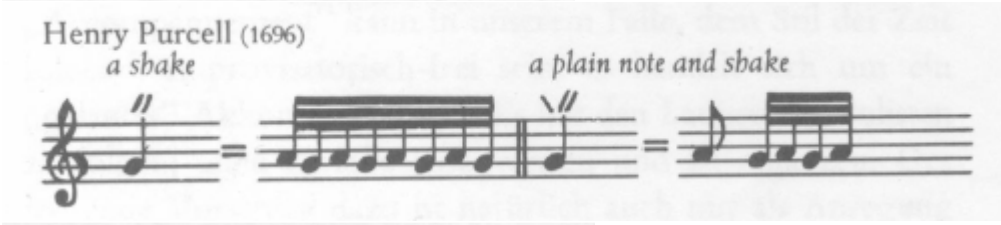
Propositions:

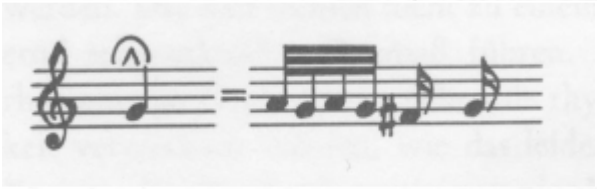
<words type="fingering">p</words>
<words type="flow-control-sign">D.C. al Fine</words>
<words type="tempo-indicating">faster</words>

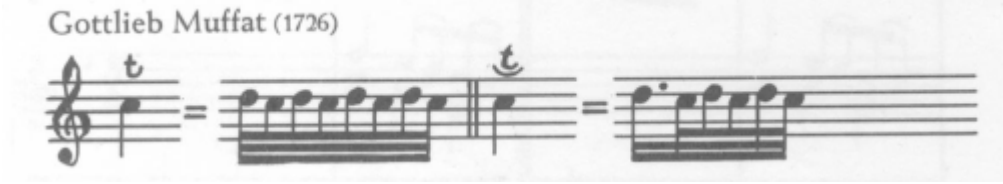
- directive
- dynamics
- finger
- normal
- rehearsal
- flow-control-sign
- post-flow-control-sign
- tempo-indicating

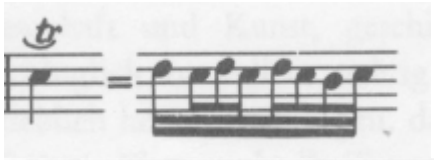
3. Create more ornaments elements met in Bach's scores, like various cadences, tremblements, bebungs etc. Taken from Hans-Martin Linde, "Kleine Anleitung zum Verzieren alter Musik", Schott

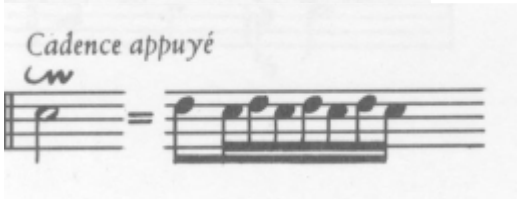


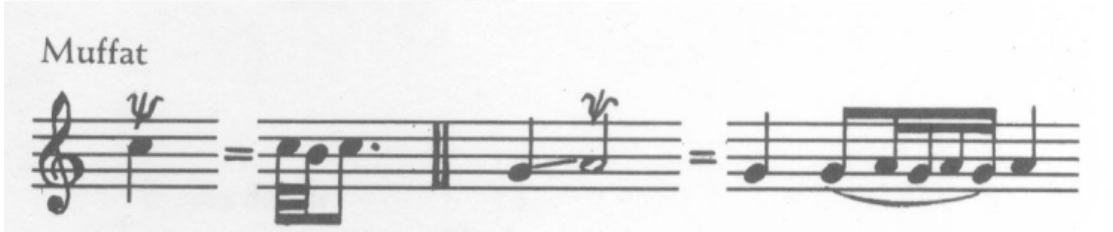


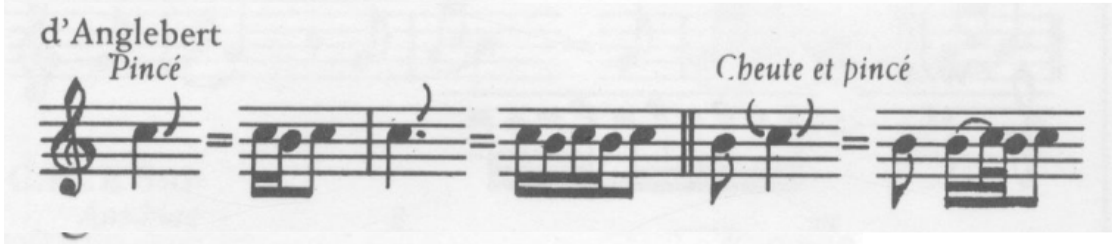


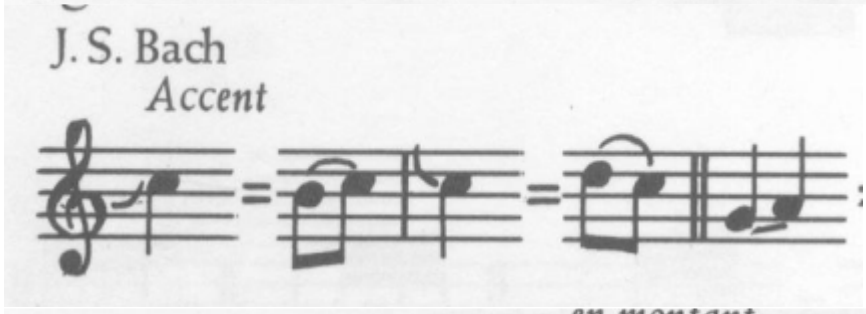


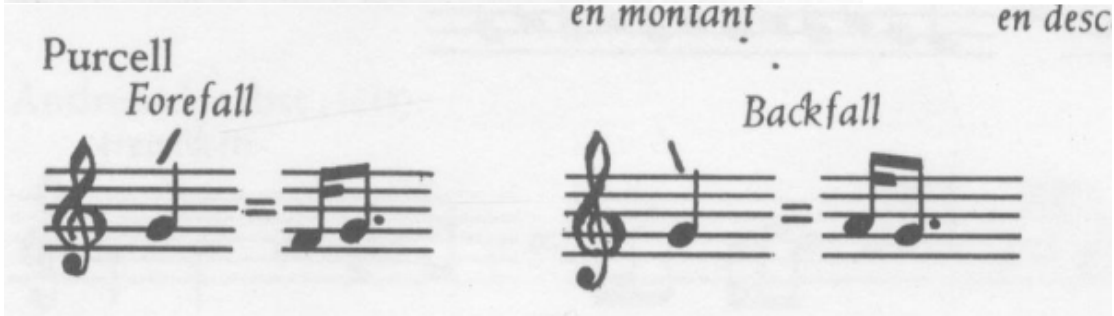


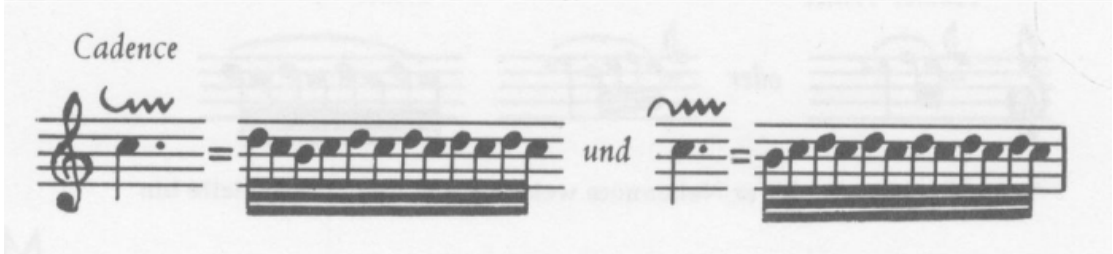

Cadence appuyé

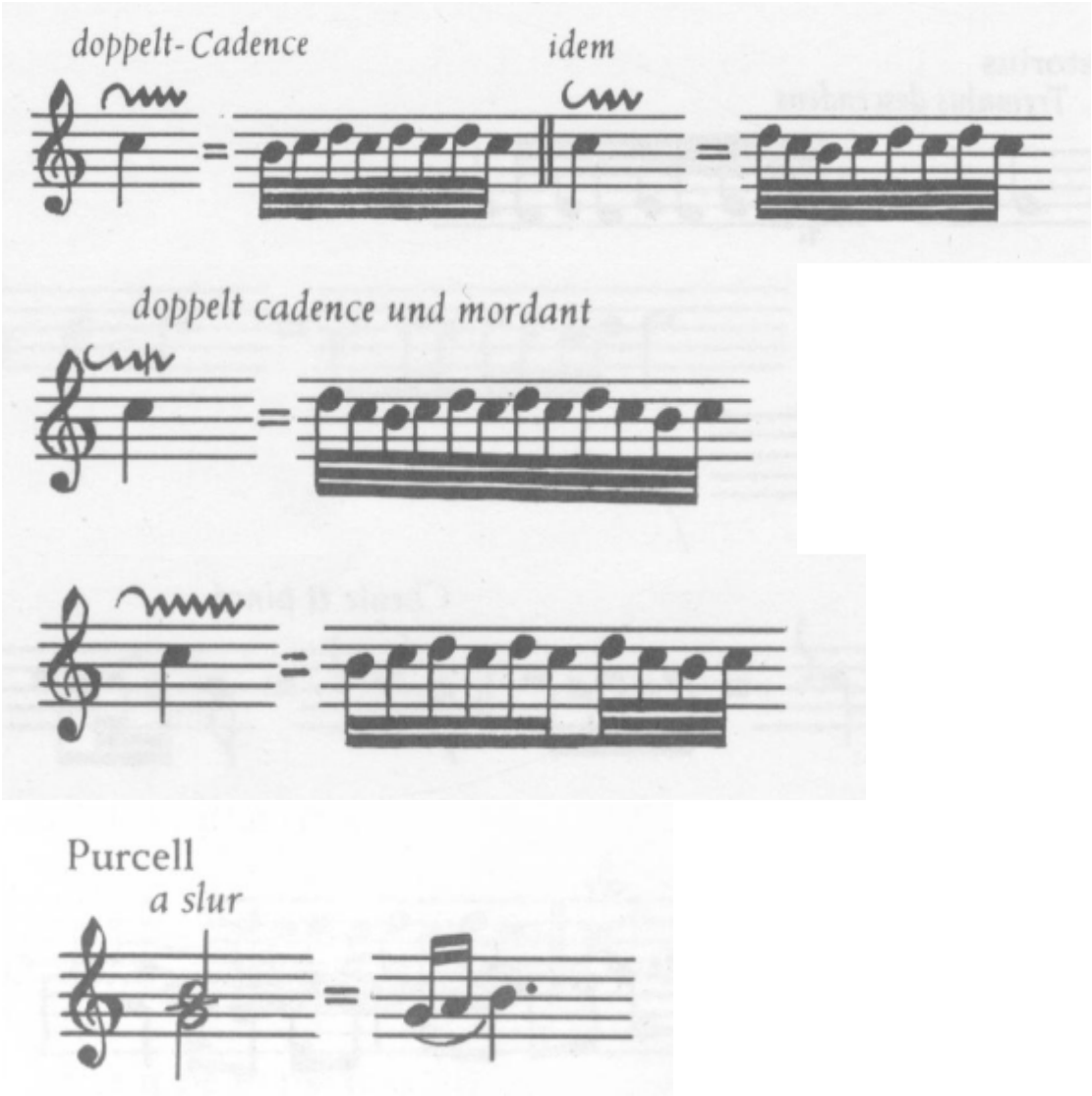

Muffat


d'Anglebert
Pincé
Cheute et pincé


J. S. Bach
Accent

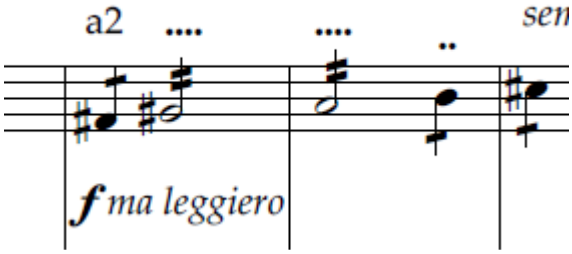

en montant
en desc
Purcell
Forefall
Backfall


Cadence
und



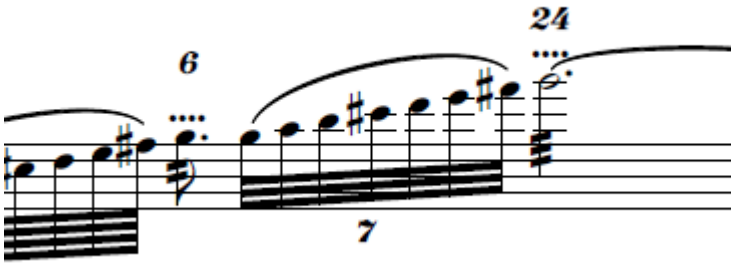
4. Add way to express multiple staccato dot symbols. Such double, triple and quadruple dots usually appear above a tremolo, which indicate an abbreviation of staccato'd multi-tonguing or ricochet bowing of repeated short notes. These can be done using symbols in Sibelius, but there are no MusicXML equivalents for them. (insert music examples)

Examples:



Proposition:

<staccato display-count=”4” />
<staccato display-count=”2” />



Proposition:

<tremolo tremolo-type=”single” display-number=”yes” />

Attention: There are elements double-tongue (two dots arranged horizontally) and triple-tongue (three dots arranged horizontally) but this seems to be something different?

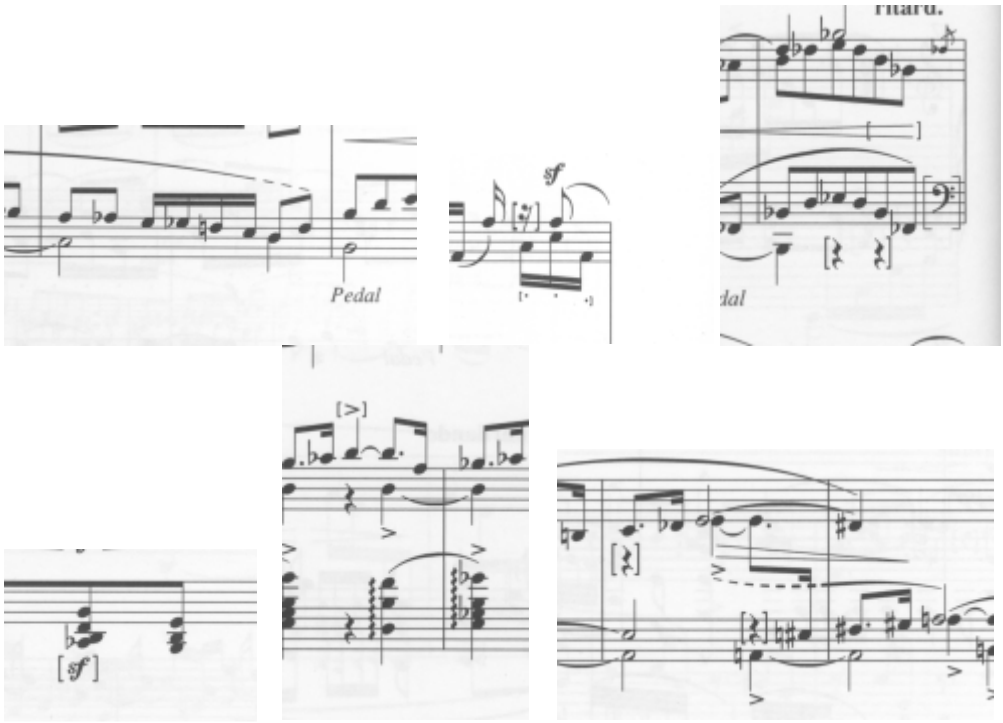
5. Support conductor symbols (e.g. baton symbols), which are currently not supported in MusicXML.

https://w3c.github.io/smufl/gitbook/tables/conductor-symbols.html

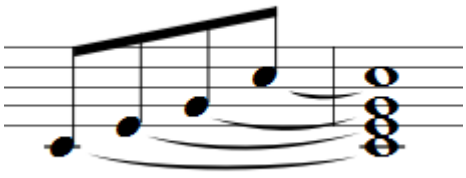
Glyph	Description	Glyph	Description
	U+E890 *conductorStrongBeat* Strong beat or cue		**U+E891** *conductorLeftBeat* Left-hand beat or cue
	U+E892 *conductorRightBeat* Right-hand beat or cue		**U+E893** *conductorWeakBeat* Weak beat or cue
	U+E894 *conductorBeat2Simple* Beat 2, simple time		**U+E895** *conductorBeat3Simple* Beat 3, simple time
	U+E896 *conductorBeat4Simple* Beat 4, simple time		**U+E897** *conductorBeat2Compound* Beat 2, compound time
	U+E898 *conductorBeat3Compound* Beat 3, compound time		**U+E899** *conductorBeat4Compound* Beat 4, compound time
	U+E89A *conductorUnconducted* Unconducted/free passages		

6. For the rare cases when we need to include MusicXML inside some other XML grammar, for instance inside a XHTML document, it would be useful to have a standardized namespace to put the MusicXML elements in. It would not be required for applications to support this namespace. It would only be for the cases where a namespace is needed as part of a bigger system/workflow.
In MNX I see that "http://www.w3.org/mnx" is suggested.
For MusicXML, we could for instance use "http://www.musicxml.org/".
See also: https://github.com/w3c/musicxml/issues/266
And https://github.com/w3c/publ-epub-revision/issues/1227

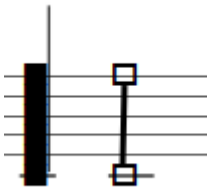
7. Left open / right open brackets and parentheses. Editor signs.



8. Indication of arpeggiating ties



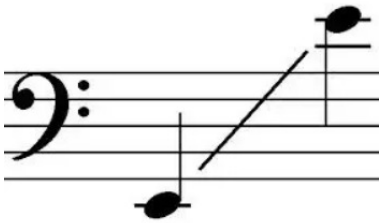
9. Clusters



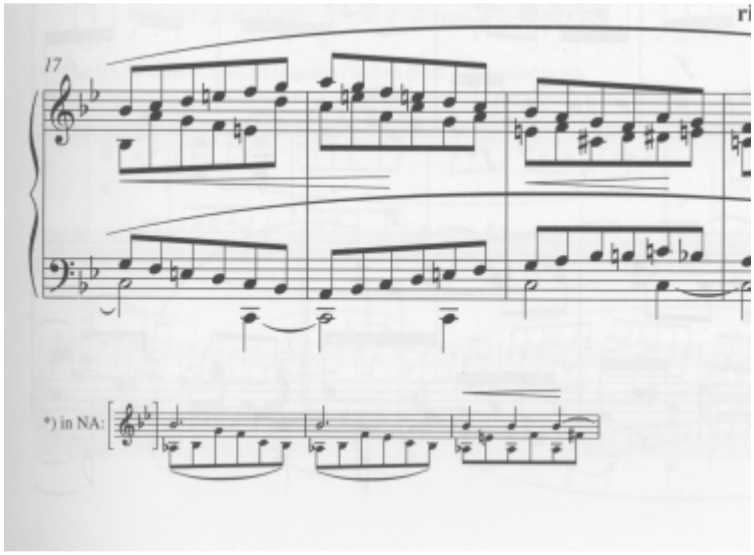
10. Half pedal

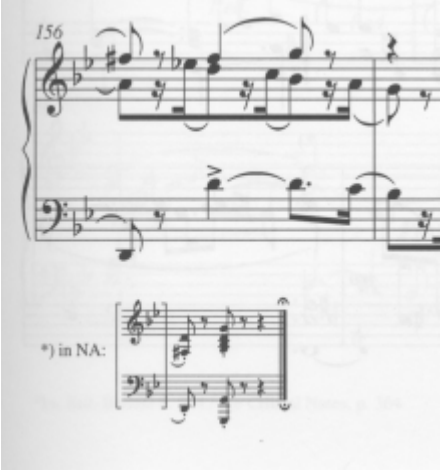
SMUFL?

11. Ambitus and/or Vocal range



12. Indicate variants
    1. Alternatives within one measure indicated by voices
    2. Smaller execution staffs to indicate how a trill-mark is expressed etc.





13. How to handle repeating content of creator and credit words

---

B. MNX Suggestions

Note: The following suggestions to MNX will only focus on CWMNX, since GMNX scores are graphical, and are impossible to be converted into braille. Sometimes there are questions left for the developers to think of a better solution.

1. Generally, include all symbols, articulations, ornaments, chords, diagram expressions, pictograms and other existing features from MusicXML 3.1.

2. Have a good strategy to include both default and user-defined categories of texts and lines, so that the braille transcription software can determine how to convert them in braille (correct equivalents and positions according to music braille convention). If a line contains text or symbol, it's better to include the text or symbol as attributes directly in the line element itself, rather than creating elements apart from the line.

3. Support more widely defined dynamic marks such as sfff, not limited by SMUFL-defined ones. These characters are usually entered as dynamic texts, so simply include them as dynamic marks other than using the unrecognized tags like other-dynamics in MusicXML.

4. Since MNX has no division values like MusicXML, when different time signatures are applied to different parts and they share the same bar length with different

scaled durations, every individual <global> element should specify appropriate duration scaling to make the affected parts take the correct bar length. This can avoid using large number of time-modifications or hidden tuplets throughout the music, and makes the output much cleaner.

5. Besides common time signatures including numerator(s) and denominator(s), also support special time signatures represented by note(s) such as a single dotted quarter, or like O Fortuna of Carmina Burana, 3 whole notes.

6. Besides common metronome marks of tempo bpm, also support unusual metronome marks. Inherit the already-complete specification from MusicXML.

7. Create more ornaments elements met in Bach's scores, like various cadences, tremblements, bebungs etc.

8. If an ornament contains accidentals, they should be put together with the ornament. The accidentals should have position indications to show whether they are above or below the sign. This is extremely important for all turns. Although we can show their positions visually using the y values, these written-out attributes will be much more clear, especially for braille transcription, and also for playback implementation.

9. Add way to express multiple staccato dot symbols. Such double, triple and quadruple dots usually appear above a tremolo, which indicate an abbreviation of staccato'd multi-tonguing or ricochet bowing of repeated short notes. These can be done using symbols in Sibelius, but there are no MusicXML equivalents for them.

10. For symbol element, it's better not to limit the symbol values to SMUFL glyphs. Some user-defined symbols and symbols from Sibelius will have their own names. Should we create another kind of symbol element to include these?

11. For lines, simple "bracket" or "dashes" is not enough. it's better to define an exclusive line element for staff and system lines, with dashed or solid as attributes. If a user-drawn line is not a straight line (e.g. a free curve), more intermediate points should be inserted into the sequence to describe the explicit curve points, such as a curved free glissando line in modern scores. If possible, marking the points with optional staff position (approximate pitch location) may be useful for braille transcription, since braille is one-dimension only, and we need a pitch position to know where the line goes to or passes. This is useful for representing curve lines and glissando lines without destination note but just lines pointing to a place on the staff

(this may need corresponding algorithm on the notation software side, but we should reserve this feature for possible use).

12. Support conductor symbols (e.g. baton symbols), which are currently not supported in MusicXML.
