DYNAMIC NOTATION – A SOLUTION TO THE CONUNDRUM OF NON-STANDARD MUSIC PRACTICE

Georg Hajdu
Center for Microtonal Music and Multimedia
Hamburg University of Music and Theater
georg.hajdu@hfmt-hamburg.de

ABSTRACT
This paper discusses dynamic notation—a method allowing, in a notation environment, instant switching between different staff views or notation styles, thus creating a common ground for practitioners of non-standard music, such as composers, performers, conductors and scholars. So far very few notation programs have explored this notion as much as it should have been. Therefore, we have implemented in the MaxScore Editor (a notation editor designed to run in Max or Ableton Live) a plugin structure for different notation styles based on a set of maps and queries executed during note entry and rendering—affecting music glyph choice and placement. We will give an in-depth analysis of the methods used for equidistant scales, non-octave tunings, music in just intonation as well as for instrument-specific layouts and will conclude with a description of a scenario in which dynamic notation was used for the transcription and performance of Alexander Scriabin’s piano poem Vers la Flamme op. 72 by an ensemble of acoustic Bohlen-Pierce instruments.

1. INTRODUCTION
In his 2001 book The Language of New Media [1] media theorist Lev Manovich points out that new media objects need to fulfill certain criteria among which variability is related to dynamic delivery of content. New media objects can exist in multiple versions such as, in case of an audio recording, a high-definition 192kHz 64-bit file, a standard CD-quality file, a lossless ALAC or a lossy compressed mp3 file.

Variability is key to solving the conundrum practitioners of non-standard music are facing when performing such music. The prerequisite is that the music is created and/or delivered by a computer-based system capable of switching between different views or representations in real time.

Rudimentary dynamic notation is common amongst notation environments. Most typically, a program will let the users switch between regular and percussion notation, or piano roll view. Bach [3], PWGL [4], OpenMusic [5] and InScore [6] as well as the lesser known Siren [7] and CMN [8] represent music in various measured and non-measured ways but lack a simple plugin structure for adding new views dynamically, hence requiring a higher degree of meddling with the code to achieve results comparable to the MaxScore Editor.

2. NON-STANDARD MUSIC PRACTICE
The practice of music in non-standard tunings has been hampered by, besides the lack of appropriate instruments, a “confusion of tongues” in respect to how this kind of music is supposed to be notated. Notation should cater to the needs of the people involved: A player performs best if the notation is close to the layout of the instrument played. For that matter, guitarists have been using tablature for centuries and Harry Partch’s notations for pitched percussion, for instance, are most often concerned with the topology of the instrument rather than actual pitch [9]. The notations for the Bohlen-Pierce clarinet which exist in two sizes (soprano and tenor) as well as the modified Bohlen-Pierce alto recorder also use fingering notation, taking advantage of the learned correspondence between finger position and sounding pitch on a traditional instrument. We can thus call this approach instrumental notation.

A composer or arranger works best when using a notation that represents the logic of a particular tuning. Sabat and von Schweinitz, for instance, have developed an elegant solution of representing frequency ratios in just intonation by designing a large set of accidentals [10]. Equidistant tunings in turn benefit from representing equal pitch distances as such. The diatonic Guidonian notation already poses difficulties when it comes to coherently representing whole-tone or atonal/12-tone music.

Copyright: © 2015 Georg Hajdu. This is an open-access article distributed under the terms of the Creative Commons Attribution License 3.0 Unported, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

1 We need to point out that our use of the term view differs from Dannenberg [2] who refers to “a data-structure that corresponds to a presentation.”
but fails bitterly at non-octave tunings such as the Bohlen-Pierce scale. Equidistant notations, such as the Hauer-Steffens notation [11] have the advantage that transpositions and transformations of tone gestalts become evident, but have been rejected in history because of cultural and economic implications, and most likely also because of the cognitive mismatch between notation and the piano layout with its black and white keys. There is no reason, though, to shy away from introducing equidistant notation for tunings other than 12EDO (and its related, circle-of-fifths-based tunings). We dub this approach logical notation.

A conductor, finally, has different concerns as he/she needs to grasp the meaning of the different notation styles used in rehearsals and performances. A conductor needs to hear, identify and compare the sounding events to the score and to an internalized template—a feat facilitated by years of intensive training and practice. He/she may be best served by the representation of music in traditional Guidonian notation, enriched by an extended set of accidentals or indications of deviations written above the notes. This may also be the notation of choice for instruments such as standard string or wind instruments. We will name this approach conventional or cognitive notation as it depends on internalized templates.

As we are attempting to establish a taxonomy for notational approaches, we also need to concede that these distinctions are arbitrary to a certain extent. Instrumental and conventional notations have a common root originating from the logic of the music in practice when its notation was standardized.

2.1 Scenarios

One can conceive of the following scenarios in which dynamic notation may be welcome:

All musicians are reading from computer/tablet screens of either isolated devices or machines in a networked arrangement. Alternatively, only the person guiding the rehearsals/performance uses an electronic device while the other members of the ensemble read from paper-based print-outs. In the latter case, the responsibility lies in him/her to guide the communication on notational aspects. Finally, both scores and parts are paper-based, but they contain, on different staves, alternative representations of the music to be performed. Even in this case, a system capable of changing views in real-time can vastly simplify the process of creating scores and parts.

3. IMPLEMENTATION

A plugin structure for dynamic switching between notation styles has been implemented for the MaxScore Editor. MaxScore is a Max Java object designed and maintained by Nick Didkovsky since 2007 to bring music notation to the Max environment [12].

Since 2010 the author is developing an editor, which also interfaces with Ableton Live via Max for Live. As opposed to the Bach notation objects, which—are native Max externals—provide better integration in the Max environment, the MaxScore Editor is based on a hybrid approach consisting of the core mxj object and a periphery made of numerous C and Java externals, Max abstractions and JavaScript objects (forming the editor’s GUI, among other functions; see Figure 1). The advantage of a hybrid system is its high degree of adaptability and versatility. As the communication between the core and the periphery is based on messages, they can easily be intercepted and reinterpreted according to current demand.

The editor handles notation styles like plugins and loads them as Max patches dynamically from a folder in MaxScore package hierarchy. It is thus very straightforward to add new styles to the existing repertoire.

Every notation style defines 5 attributes:

- A name of the notation style appearing in the style menu of the Staff Manager.
- The name of the Max patch containing pitch maps,
- The number of staff lines employed by the style.
- The micromap to be used for the rendering of accidentals.
- The name of the clef appearing on the staff.
- If a non-standard clef is being specified such as Bohlen-Pierce T-clef, an additional definition needs to be given in the form of clef name, glyph, x and y offsets as well as font name and size.
3.1 Micromaps

Micromaps were introduced in 2011 to allow higher resolution representations of divisions of the semitone. While MaxScore’s pitch attribute is stored, processed and played back in 32-bit floating-point precision, the drawing messages generated by the object are limited to quarter tones. Hence, MaxScore would fail to accurately represent music in eighth tones (the standard among spectral composers) or twelfth tones (used by composers such as Ivan Wyschnegradsky and Ezra Sims). Micromaps are Max abstractions that intercept drawing messages and query the pitch and accidental preferences attribute of the corresponding note. Based on this information and the notation style chosen by the user, a micromap sends new, more fine-grained drawing messages to the MaxScore canvas. Currently, the maximum precision is sixteenth tones in Sagittal notation, taking advantage of the enormous set of accidentals in the Bravura font, just recently released by Steinberg [13].

How is this mapping performed? This is best explained by an example: After entering a middle c a sixteenth note sharp (pitch = 60.125) the MaxScore object sends out a drawing message with 9 items (accidental, x, y, zoom, parent object, measure, staff, track and note indexes) such as:

“no_accidental 75.555557 81. 0.5 Note 0. 0. 0. 0.”

The five last items are sliced off and a “getNoteInfo 0. 0. 0. 0.” query is sent to the core object. It returns a string in XML format which is being parsed and sent back to the micromap. Of the many note attributes, pitch and accidental information is being retained to calculate the pitch zone a particular accidental is applied to.

The zone index Z is given by this formula ($rnpc = $ floating point remainder of natural pitch class, $n =$ division of the semitone, $sgn = -1$ for ACCPREF = flat and $sgn = 1$ for ACCPREF = sharp):

$$Z = \left\lfloor sgn \cdot \frac{rnpc}{n} + \frac{1}{n+2} \right\rfloor \quad (1)$$

In our case, the result would be

$$[1 \cdot 0.125 \cdot 8 + 0.0625] = 1$$

This value is fed into a zone-to-glyph-name lookup table (a Maxcoll object), which sends out accSagittal5CommaUp, the name of the Sagittal accidental in the Standard Music Font Layout specification on which the Bravura font is based [13].

This message is combined with the rest of the message into

“accSagittal5CommaUp 75.555557 81. 0.5 Note 0. 0. 0. 0.”

of which the first four items are further processed:

The zoom value scales the font size as well as x and y offsets. The accidental name is sent to another instance of a Maxcoll object which returns

“\texttt{\textbackslash 4 \ -4 \ 0 \ Bravura \ 24}” (glyph, x offset, y offset, font name, font size).

---

2 no_accidental messages will be ignored by the drawing engine, but are a prerequisite for mapping.
This information is then translated into three separate Max lcd messages:

1. “font Bravura 24.”
2. “moveto 71.555557 82.” (due to various reasons a y offset of 1 is applied to all glyphs)
3. “write 4”

3.2 Notation Styles

For the representation of music in the Extended Helmholtz-Ellis JI Pitch Notation created by Sabat and von Schweinitz we had to go a step further. A problem arises when the MaxScore object is no longer capable of representing the correct mpq in case of complex harmonic relationships such as 75/49 (3⁻¹⁻²⁻⁷). According to the logic of the Helmholtz-Ellis notation the interval size of 736.9 cents—when applied to a middle C—would have to be represented by an f with two accidentals, a double-sharp with two arrows down and a raise by two septimal commas (see Figure 4) The diatonic pitch class is calculated by moving along the circle of fifths, which in our case would be moving 13 ticks in clock-wise direction, thus amounting to an f double-sharp. This conflicts with the pitch class natively assigned by MaxScore which is g³. The solution was found by creating a specific Just Intonation notation style. Kuuskankare and Laurson [14] use a similar term (notational style) denoting changes in notation that include not only pitch but other aspects such as measured/non-measured notation.

A notation style is basically defined by two maps (map and inverse map) but also requires the definition of an additional attribute. This can be easily achieved in MaxScore where an unlimited number of dimensions can be added to notes and intervals first by applying the “setInstrumentDimension” message to a particular staff and then setting note dimensions values individually. Considering this, we have defined an additional originalPitch dimension which holds the pitch of a note regardless of how it is represented graphically in the score and is also used for playback.

After choosing a notation style from a menu in the Staff Manager, all notes for a given staff are passed to the inverse map of the abstraction of the current notation style (default for new scores) to restore the originalPitch attribute. Then all events are routed to the map of the abstraction of the selected notation style⁴. Here the pitch attribute is set to the position of the note it is supposed to occupy in the given notation style.

3.3 Examples

We will now more closely examine some of the notation styles implemented in the Max Score Editor.

3.3.1 Just Intonation

Definition: “Just Intonation” justintonation 5 mM-JI default

As pointed out above, certain ratios lead to a situation where the actual pitch is more than 225 cents off its natural “anchor” tone. Having introduced originalPitch as an attribute for correct playback, the pitch attribute can now be “arbitrarily” moved to a tone above or below—thus displaying the correct pitch/accidental combination. The calculations involved are fairly complex and have been described in [12].

---

③ MaxScore stops considering enharmonic spellings for ranges outside of double flats and sharps (maximum considered deviation = 225 cents; in case of 75/49 the deviation is 236.9 cents).

④ NB.: Instead of current and selected, we could also use the terms old and new.
3.3.2 19EDO

Definition: “19EDO 19EDO 5 mM-none default”

Russian-American musicologist Joseph Yasser argued in his 1932 book *Theory of Evolving Tonality* that 19-tone music, in its just or equal tempered forms, constitutes the next logical step in the development of music [14]. While we can no longer subscribe to this claim, this tuning remains one of the popular ones, having been investigated by composers such as Easley Blackwood and Joel Mandelbaum. Its 19 tones form a closed circle of fifths and, thus, the scale possesses a diatonic subset and enharmonic alternatives for each black key, in addition to an e# and a b# between e and f and b and c, resp. The mapping is performed by:

1. Calculating the 19EDO scale step index with µUtil.PitchToStep abstraction, which is part of the author’s µUtilities package bundled with MaxScore.
2. Extracting octave index and pitch class by dividing the index by 19 and passing the remainder through a lookup table yielding the 12EDO pitch class, accidental preference (sharp or flat) and enharmonic spelling for any of its 19 pitch classes.
3. Calculating pitch by multiplying octave index by 12 and adding the respective 12EDO pitch class and an offset.

E.g. for 7136 MIDI cents, the scale step index is 113. Divmod 19 yields 5 and 18. Feeding 18 into the coll returns “11 1 1”, thus setting setAccPref to sharp and setAltEnharmonicSpelling to true. Pitch is 12 \* 5 + 11 + 1 = 72, displayed as b#.

3.3.3 Percussion

Definition: “Percussion percussion 5 mM-none percussion”

Most notation programs implement the percussion notation style in which MIDI notes (range 35 - 81) are mapped to the white keys between d4 and a5. Percussion notation uses certain positions redundantly, yet differentiates between classes of instruments by assigning various notehead shapes to the notes.

When switching to the percussion notation style the originalPitch attribute is sent to a Max coll (percussionMap) containing:

- The name of the instrument,
- Its (notated) pitch in percussion notation
- The corresponding notehead shape.

Figure 6. The 19EDO notation style takes pitch, accidental and enharmonic spelling preference into consideration.
The map will now send three messages to the MaxScore core object: “setPitch value”, “noteheadTransform shape” and “setAltEnharmonicSpelling false”, the latter message to clear double sharps or flats should they have been set previously.

When switching from the percussion notation style the pitch and notehead attributes are evaluated and sent to another coll (inversePercussionMap) in order to clear notehead shapes and reconstruct the originalPitch attribute. The messages to MaxScore are “noteheadTransform NOTEHEAD_STANDARD” and “setNoteDimension originalPitch value”.

Figure 7. The percussion notation style consists of lookup table setting pitch and notehead shape.

### 3.3.4 Bohlen-Pierce T-Clef

Definition: “BP chromatic T clef” BP-chromatic-T 6 mM-BP BP-T-clef

Clef definition: “BP-T-clef T 2 -1 "Greifswaler Deutsche Schrift” 28”

Figure 8. The Bohlen-Pierce scale in T clef maps the range of the tritave d3-a4 onto the six lines of Müller-Hajdu notation. Note that the lowest and highest notes look like d and a; this is a desirable, albeit coincidental trait.

We have described the design of a new notation system for the non-octave Bohlen-Pierce scale with 13 steps [16]. This scale, which was independently discovered by three people (Heinz Bohlen [17], John Pierce and Kees van Prooijen) in the 1970’s to 1980’s, is probably the most common and best-investigated scale of its kind. A number of acoustic instruments have been built since 2007 and a group devoted to the practice of this kind of music has been founded a year later in Northern Germany. It was shown by Pierce, Mathews et al. [18], Loui [18] and us [16] that this scale, substituting the octave (2:1) by a tritave (3:1), exhibits characteristics analogous to the 12-tone chromatic scale and its diatonic subsets and whose inherent relationships can be learned through repeated exposure. To allow for a new theory, we came up with a six-line staff, new note names, interval designations and clefs, which we call the Müller-Hajdu notation. There are three clefs, N, T and Z, for which we created corresponding chromatic notation styles (with notes either written without accidentals on a line or between two)

Figure 9. The Bohlen-Pierce T clef also allows for the microtonal subdivision of the BP base interval into 5 steps. It uses the mM-BP micromap for its single and double-shaft accidentals.

### 3.3.5 Bohlen-Pierce Clarinet Fingering Notation

Definition: “BP Soprano Clarinet” BP-soprano-clarinet 5 mM-none default

Figure 10. The Bohlen-Pierce scale in soprano clarinet fingering notation

Music written in N, T or Z clef can easily converted into Bohlen-Pierce clarinet fingering notation. There are two sizes of clarinets built by Canadian clarinet builder Stephen Fox [20], a soprano clarinet and a tenor clarinet, the former having the same size as a Bb clarinet and the latter having the size and shape of a basset horn. As the Bohlen-Pierce scale is based on the tritave, or just twelfth, the interval to which members of the clarinet instrument family will overblow, Fox was able to simplify the mechanics and thus proposed a fingering notation where certain notes are omitted in comparison to the Böhm and German systems.

---

\[ In case of the T clef we have even allowed for a microtonal division of the tempered Bohlen-Pierce scale step (146.3) into 5, virtually identical with the division of the octave into 41 steps (a well known tuning with a diatonic subset and a 24th scale degree being just the tiniest fraction higher than 3:2). \]
3.3.6 Special Applications: Bohlen-Pierce Alto Kalimba

Definition “BP Alto Kalimba” BP-alto-kalimba 5 mM-none percussion

As mentioned before, new notation styles can easily be added such as in the case of the Hugh Tracey alto kalimba [21] whose 15 tines I tuned to the Bohlen-Pierce scale. For a percussionist, its pitches are best represented by notating the tines according to their alternating “centrifugal” layout with the longest tine in the middle being represented by a note in the middle of the staff.

This feat was accomplished by feeding originalPitch values through a µUtil.PitchToStep abstraction (to calculate the Bohlen-Pierce scale step index) and a lookup table, yielding the pitch to be displayed.

4. PRACTICAL APPLICATIONS AND FUTURE PLANS

One of my recent musical activities was the arrangement of the piano poem Vers la Flamme op. 72 by Alexander Scriabin [22] for 3 Bohlen-Pierce clarinets, Bohlen-Pierce guitar, double bass in Bohlen-Pierce scordatura, keyboard in Bohlen-Pierce layout, Bohlen-Pierce kalimba and tam-tam.

During the arrangement I:

1. imported a MIDI file found on www.kunstderfuge.com into the MaxScore Editor
2. mapped the tracks to the Bohlen-Pierce N, T and Z clefs
3. checked for motivic inconsistencies created by the automatic mapping and changed pitches where necessary
4. mapped the voices to various instrumental notations styles
5. extracted the parts for the musicians using the editor’s pdf generation capabilities.

Melle Weijters, the Amsterdam-based guitarist involved in the performance of the arrangement actually plays a 10-string guitar in 41EDO tuning. As the Bohlen-Pierce scale is actually a subset of this tuning, he only needed to find the correct positions on the fretboard. He therefore requested his part in T-clef with an additional empty 10-line tablature staff to manually note finger positions. We are planning to automate this process and make it generally applicable to instruments with standard and non-standard numbers of frets and tunings. A number of papers have already dealt with the intricacies of automatic tablature transcription for guitar using genetic algorithms, neural networks and hill-climbing algorithms [23] [24].

Another interesting path to take would be exploration of graphical notation, such as the one employed by Karlheinz Stockhausen in his Elektronische Studie II [25]. The technological prerequisites have already been implemented in MaxScore, but it remains to be seen whether the effort of creating such a notation style or set of styles, for that matter, is justified in the light of the many individual solutions created by composers over the last few decades, or whether users would be best served by a separate specialized application. Sara Adhitya and Mika Kuuskankare have demonstrated a possible solution using macro-events in PWGL [26] for a piece by Logothetis.

Currently, our dynamic notations system is somewhat hampered by efficiency issues found in the Max JavaScript object. An effort will be spent to streamline the code and to replace it with Max C externals, if necessary.

5. CONCLUSIONS

We have developed for the MaxScore editor a plugin structure for dynamic notation that greatly facilitates the creation and practice of microtonal music in scenarios where composers, conductors and performers can no
longer rely on a common notational reference internalized by years of training such as with the 12-tone system. Applying various styles in an arrangement for Bohlen-Pierce instruments proved to be a viable approach for editing, printing and rehearsing. More notation styles will be added as we further develop this version of the software, which currently is in a beta state and can be downloaded from http://www.computermusicnotation.com.

Acknowledgments

I would like to thank the Behörde für Forschung und Wissenschaft Hamburg for supporting our research in the framework of its Landesforschungsförderung. I would also like to thank Benedict Carey and Todd Harrop for proof reading.

6. REFERENCES


Webpages all accessed on January 28, 2015.