



W3C Music Notation Community Group Meeting

4 April 2019



W3C Music Notation Community Group

- Founded in July 2015
- Develops and maintains format and language specifications for notated music used by web, mobile, and desktop applications
- Maintains and updates the MusicXML and Standard Music Font Layout (SMuFL) specifications
- Developing new MNX spec to handle new use cases and technologies
- Community group membership is free of charge and does not require W3C membership
- <https://www.w3.org/community/music-notation/>

- Introduction from our reception sponsor, capella-software
- Welcome to Adrian Holovaty as co-chair
- SMuFL 1.4 plans
- MusicXML 3.2 plans
- DAISY Braille Music Group
- MNX-Common and MNX-Generic
- Sounding and written pitch in MNX

capella and MusicXML

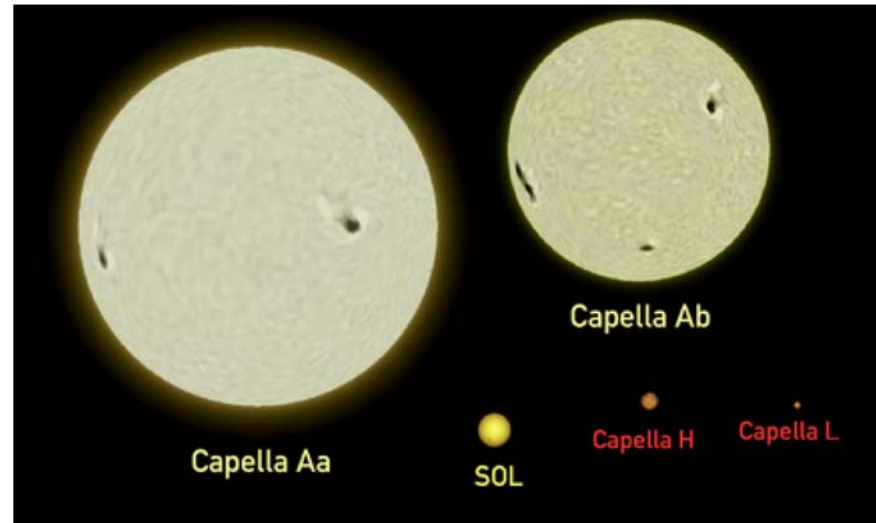
*W3C Music Notation Community
Group Meeting, 04.04.2019*

*Dominik Hörnel
capella-software AG*

What does capella stand for?

- capella-software has been creating software around music notation for almost 30 years
 - Easy to use and intuitive music software at moderate prices
 - Our software is aimed at individuals, music groups and associations in the amateur and professional area
-
- **capella** vs. „A **cappella** music“: A typo?

What does capella stand for?



- capella: brightly shining star in constellation Auriga; more exactly a star system

Our „capella stars“



capella melody trainer



capella-scan



capella wave kit



capella



capella reader



rondo



audite PLUS



tonica

capella software reading/writing MusicXML



capella melody trainer



capella-scan (w)



capella wave kit (w)



capella (r/w)



capella reader (r)



rondo

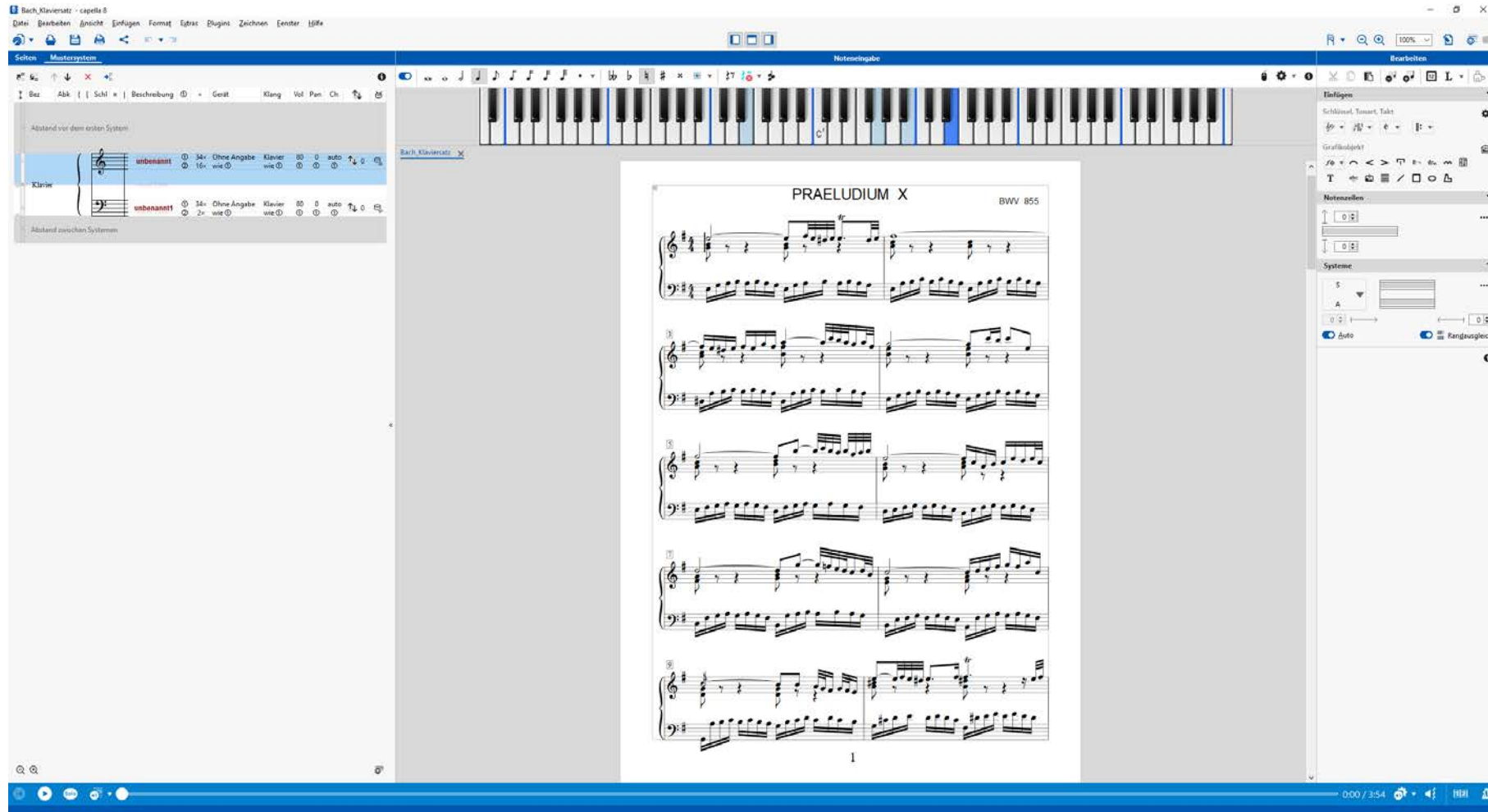


audite PLUS (w)



tonica (r/w)

Writing notes made easy with capella



[illegible]

Train intonation with **capella melody trainer**



Compose and analyze music with **tonica**

tonica fugata 13

Datei Bearbeiten Ansicht Dokument Extras Komponieren Analyse Harmonisierungsstile Fenster Hilfe

Einrichten **Komponieren**

Noteneingabe

Kompositionsassistent

Tonsatz Variation Begleitmuster Kanon Fuge **Prüfen** Generalbass H.

☒ alles
☐ aktuelle Phrase
☐ markierter Abschnitt

Analyisieren

Melodiestimme: S./A. oben

☐ Stimmumfänge prüfen

Analyse

Abstand für Harmonische Analyse: Viertel

Akkordschema: Klassische Musik

Stimmführung

Parallelen anzeigen

☐ Nicht anzeigen
☒ Nur echte Parallelen
☐ Verdeckte Parallelen in Außenstimmen anzeigen
☐ Verdeckte Parallelen in allen Stimmen anzeigen

☐ Harmonien neu analysieren

Ohne Titel

Sopran/Alt

Tenor/Bass

A: T T₃ T S D₃ D₃⁶ D₃⁷ D T

S./A.

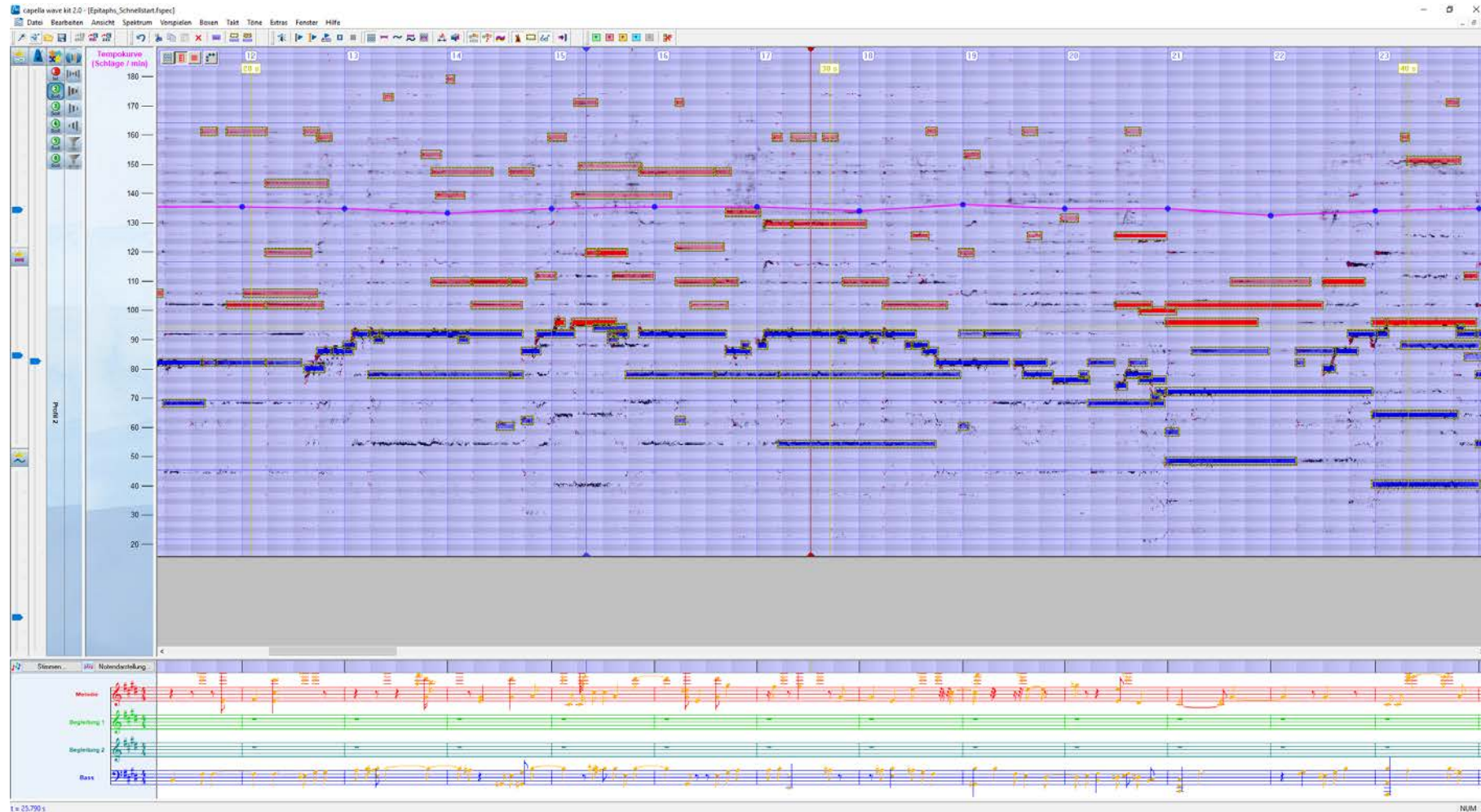
T./B.

A: T Tp D₃⁷ D D₃ D D₃ Tp Tp Sp d₃

Fehler (Parallelen-Regel): Antiquintparallele (S./A. oben, T./B. oben -> S./A. oben, T./B. oben) [erster Akkord]

0:00 / 0:16

Notate recordings with capella wave kit



Read your music with capella reader

The screenshot displays the Capella Reader application interface. The main window shows the musical score for "Ave verum KV 618" by W.A. Mozart, marked "Adagio". The score includes vocal parts (Soprano, Alto, Tenor, Bass) and instrumental parts (Violone 1, Violone 2). The lyrics are: "ve - rum cor - pus, na - tum de Ma - ri - a v". The interface features a sidebar on the left with page thumbnails, a main score area, and a right-hand pane showing a detailed view of a specific page. Below the main interface, a tablet and a smartphone are shown displaying the same application on their screens.

Seiten

Ave verum KV 618

1

2

3

Sopran

Alt

Tenor

Bass

Violone 1

Violone 2

Ave verum KV 618

W.A. Mozart

Adagio

sotto voce

A - ve, a - ve

sotto voce

A - ve, a - ve

sotto voce

A - ve, a - ve

sotto voce

sotto voce

sotto voce

5

S

A

T

B

VI 1

ve - rum cor - pus, na - tum de Ma - ri - a v

ve - rum cor - pus, na - tum de Ma - ri - a v

ve - rum cor - pus, na - tum de Ma - ri - a v

ve - rum cor - pus, na - tum de Ma - ri - a v

ve - rum cor - pus, na - tum de Ma - ri - a v

0:00 / 2:05

0:00 / 2:05

0:00 / 2:05

From MusicXML to MNX

- We are following (and occasionally contributing to) the MNX discussions with great interest
- Thanks to Joe for his great pioneering work on MNX!
- A big welcome to Adrian for taking up and continuing Joe's work!
- Looking forward to discussions on this topic...

Thank you for your interest!



capella melody trainer



capella-scan



capella wave kit



capella



capella reader



rondo



audite PLUS



tonica



- SMuFL 1.3 Community Report published on 8 March
- Some new character ranges and glyphs added since 1.2 draft published in April 2016, including:
 - New ranges for Kahnotation (dance notation) and German organ tablature
 - Extensions to existing ranges for fingering, clefs, vocal techniques, percussion beaters
 - New ranges of optional stylistic alternates added to Bravura for time signatures and chord symbols
- Some minor specification clarifications for specific metadata key/value pairs



- Aim to complete SMuFL 1.4 before the end of 2019
- Around 30 issues in total currently outstanding: 16 currently in scope
 - Define complementary text font family in SMuFL font metadata
 - Expand triplets in “Beamed Groups of Notes” range
 - Add noteheads for chromatic solfège
 - Support for numbered notation (e.g. Chinese *jianpu*)
- New proposals are also always welcome – just open an issue



Why MusicXML 3.2?

- There is still room for improvement in MusicXML's core use cases of music notation exchange and archiving
- In particular, MusicXML needs better support for parts
- Working on MNX does not make MusicXML obsolete
- Issues that cannot be resolved effectively given MusicXML's design are postponed to MNX-Common



Current MusicXML 3.2 Plans

- About 20 open issues in the [V3.2 milestone](#), focused on 3 themes
 - Better support for parts
 - Better XML tool support
 - Clarify documentation of several features
- Some common suggestions currently postponed to MNX-Common
 - Improve semantics for text
 - Improve semantics for brackets and lines
 - Anything that would break compatibility with older MusicXML versions
- No firm release date planned yet, but likely sometime in 2020



Improved Part Support

- Many applications allow editing of score and parts in a single file
- MusicXML files encode either a score or a part, with no standard way to represent relationships
- [Issue 278](#) has the current main proposal: optionally include a full score and full copy of all parts within the zip archive of an .mxl file
- Also provide better support for creating parts when just importing a score as a .musicxml text file
 - Specify transpositions for parts within concert scores ([Issue 279](#))
 - Specify top staff and bottom staff directions ([Issue 37](#))



MusicXML 3.2 Next Steps

- Does proceeding with MusicXML 3.2 along with MNX-Common seem like a good idea?
- If so, now is the time to write up MusicXML issues for possible inclusion in 3.2
- Any other major areas to work on for this release?
- Plan to create an MNX milestone or tag in GitHub for issues targeted for MNX-Common



DAISY Consortium Braille Music

- CG is working on Music Scores in Braille for the Blind
- Goal: Make more music Braille scores available to more blind musicians
- Cross-sector project between different partners
 - Musicians, libraries, producers, etc.



DAISY: Background

- Make Braille music notation save for future
- Producers will change regarding budget and expertise
- Accessible developments in notation or music software are highly welcome
- But: this project concentrates on Braille sheet music which will be of much importance for a long time



DAISY: Key People

- Arne Kyrkjebø from National Library for the Blind, Norway
- Dr. Sarah Morley Wilkins
- Haipeng Hu from China (blind composer)

1. Input files need to be as good as they can be at the start
2. Conversion and mark-up tools need to be accurate & reliable, for agencies and end-users
3. Good access is needed to existing intermediary files
4. Good teaching, learning and promotional materials are needed



MNX update

An introduction to MNX-Common [↗](#)

MNX-Common is a new, open standard for representing music notation as machine-readable data. It's still being created, but this document gives context on our goals and plans.

Conceptually, MNX-Common is a way to represent this image:

Image of music notation with middle C

...as something like this:



MNX-Common vs. MNX-Generic

§ 2.2. MNX score types

This section is non-normative.

MNX can support multiple ***score types***.



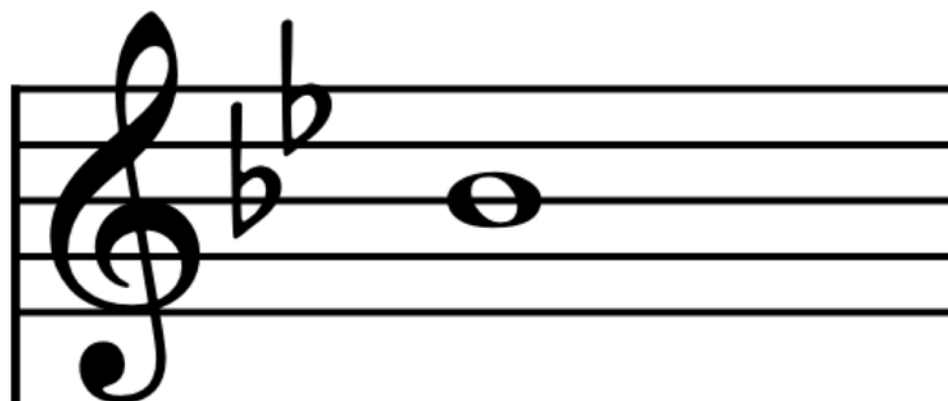
MNX-Common vs. MNX-Generic

1. Confusing for end users
2. Confusing for developers
3. No huge benefit in combining them

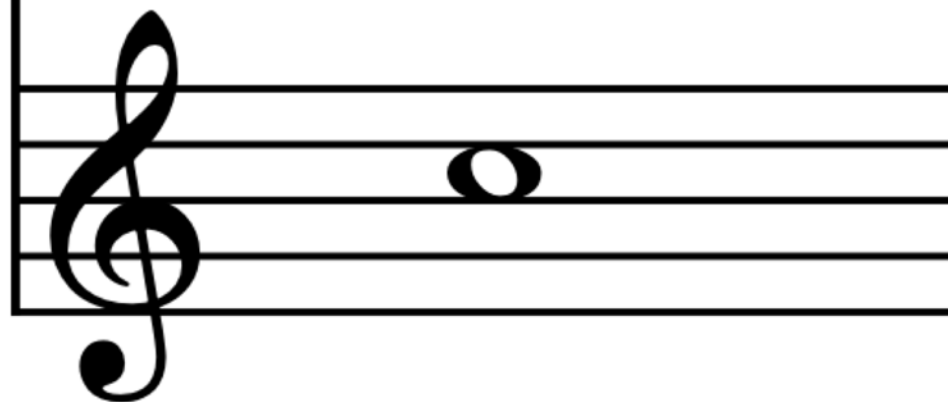


Written vs. sounding pitch

Piano



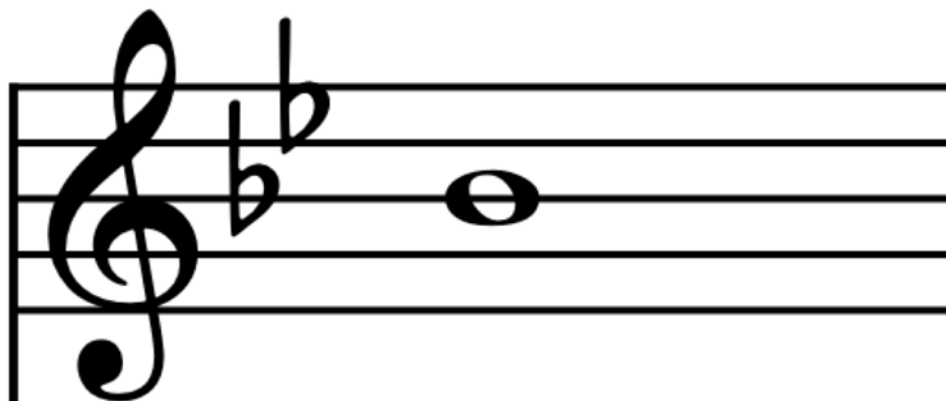
Clarinet





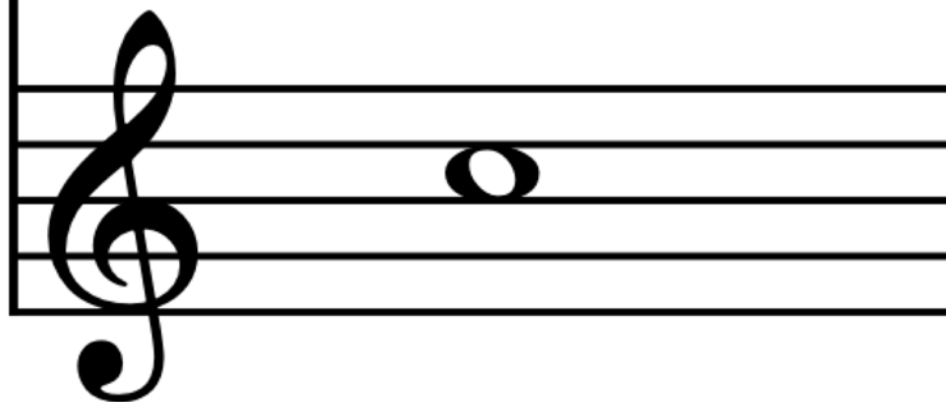
Written vs. sounding pitch

Piano



`<note pitch="Bb4">`

Clarinet

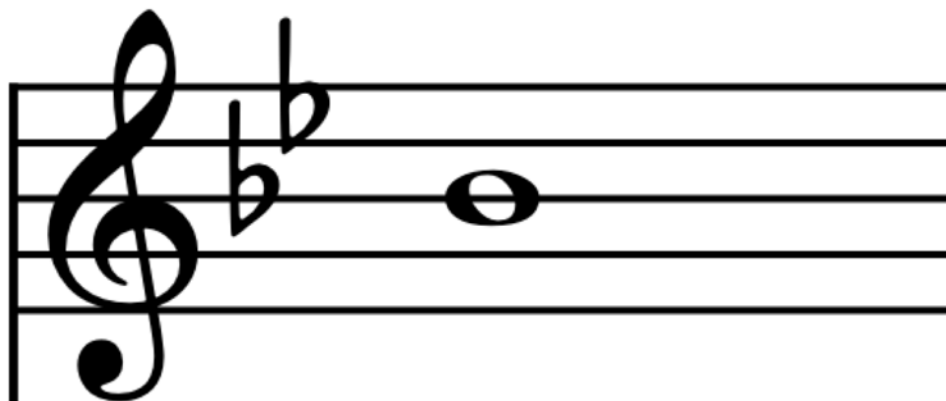


??



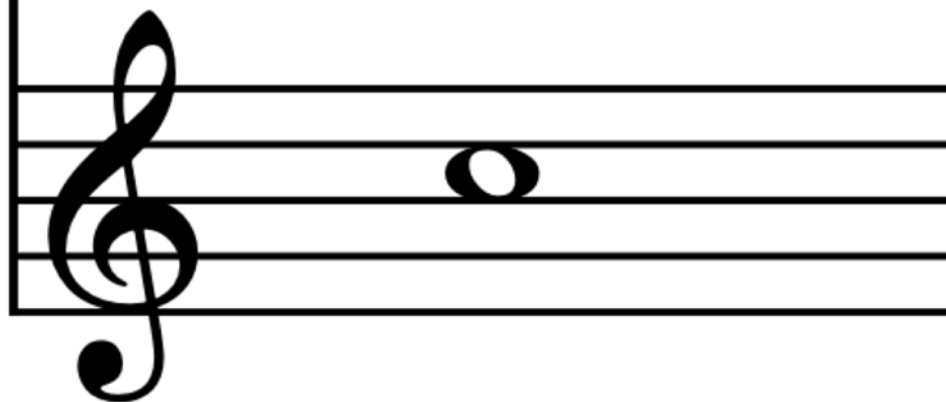
Option: sounding pitch

Piano



```
<note pitch="Bb4">
```

Clarinet



```
<note pitch="Bb4">
```

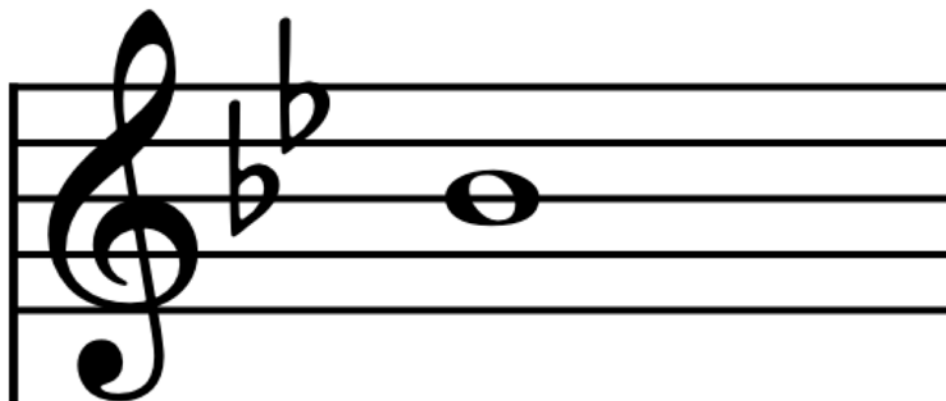
...

```
<part transpose="-2">
```



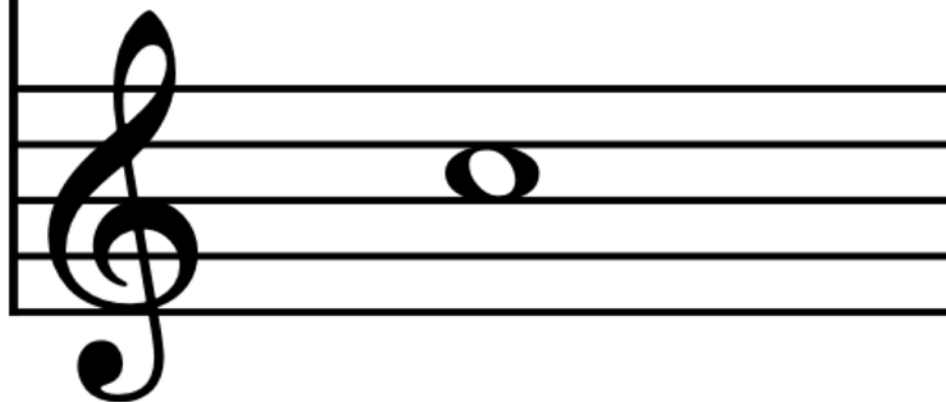

Option: written pitch

Piano



`<note pitch="Bb4">`

Clarinet



`<note pitch="C5">`

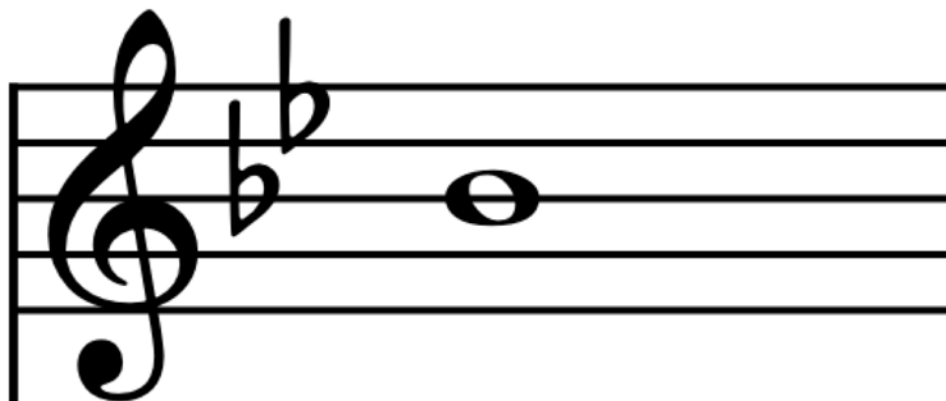
...

`<part transpose="-2">`

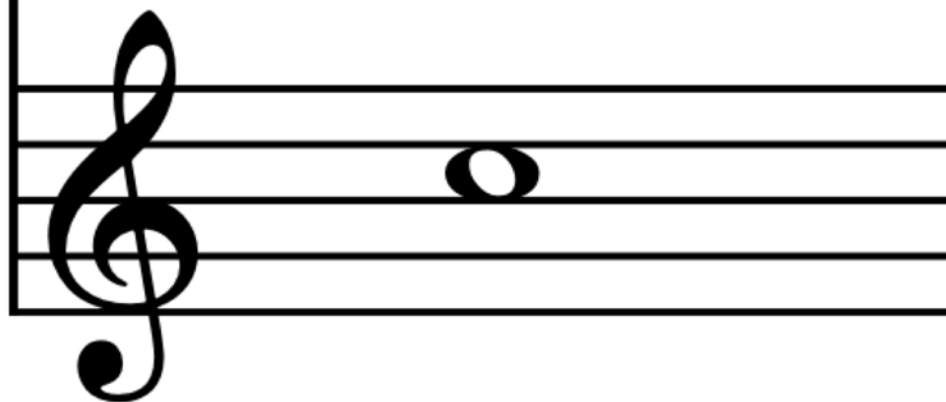


Option: “realizations,” issue 138

Piano



Clarinet



`<note pitch="Bb4">`

(Score and part duplicated)

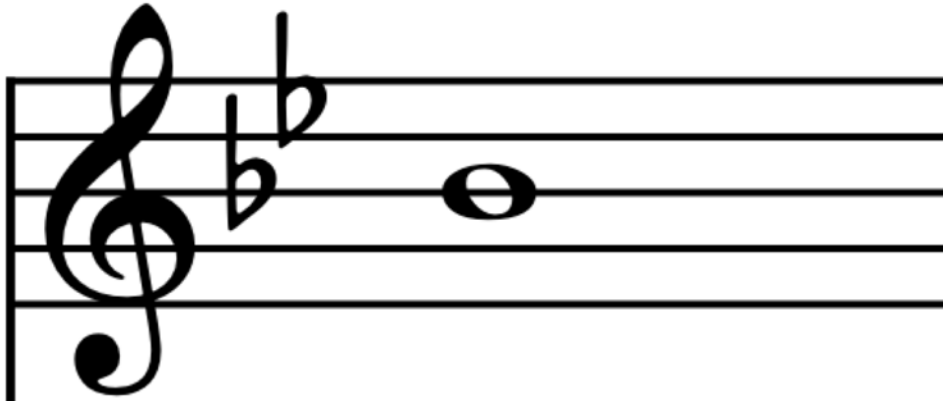
```
<score>  
  <note pitch="Bb4">  
</score>
```

```
<part transposition="-2">  
  <note pitch="C5">  
</part>
```

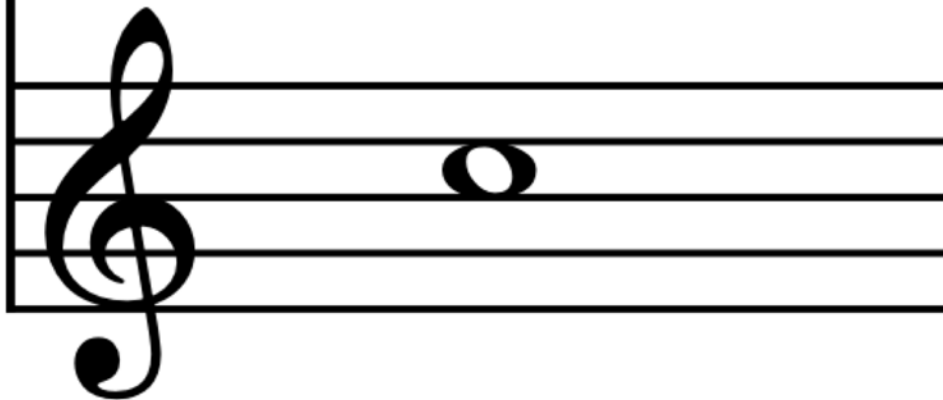


Option: both, via alternates

Piano



Clarinet



```
<note pitch="Bb4">
```

```
<note pitch="C5">  
  <sounding pitch="Bb4" transpose="-2"/>  
</note>
```

Or...

```
<note pitch="Bb4">  
  <written pitch="C5" transpose="-2"/>  
</note>
```



More complex example

Flute



Key: B
Note: C#



Key: B
Note: C#

Concert score

Parts



More complex example

Flute



Key: B
Note: C#

Clarinet
in Bb



Key: B
Note: C#

Concert score



Key: B
Note: C#

Key: C#
Note: D#

Parts



More complex example

Flute



Key: B
Note: C#

Clarinet
in Bb



Key: B
Note: C#

Tenor Sax



Key: B
Note: C#

Concert score



Key: B
Note: C#

Key: C#
Note: D#

Key: Db
Note: Eb

Parts



The Big Questions

- Are we encoding musical ideas or musical documents?
- Do we prioritize the performer's perspective or listener's perspective?
- What is the ground truth: the sound or the visual display?
- How important is XML readability vs. a reference implementation?
- Is redundancy between scores/parts inevitable, or is every part-specific notation derivable from a score given the right hints?