Unofficial MusicXML test suite

1 Introduction

Why a MusicXML test suite?

This test suite of sample MusicXML (`http://www.musicxml.org/`) files is supposed to fill a severe gap for all developers implementing MusicXML support in their application: There is no complete test suite of MusicXML files available for testing purposes.

Downloading the test suite

The complete set of MusicXML test files contained in this suite can be downloaded here (`MusicXML-TestSuite-0.1.zip`) as a ZIP archive.

License of the test suite

This collection of MusicXML test files is distributed under the MIT license (`http://www.opensource.org/licenses/mit-license.php`), which means that you can use the files for any purpose, as long as you leave the copyright notice (or the LICENSE file) intact.

Connection with LilyPond (`https://lilypond.org/`)

At the same time as providing a generic test suite for MusicXML document, this test suite also serves as proofs for the musicxml2ly script provided with LilyPond dev. The images shown in the Chapter 2 [Test cases], page 4, chapter were generated by running `musicxml2ly` and `lilypond` on the MusicXML files. As `musicxml2ly` does not yet perfectly support every single aspect of MusicXML, the output is not supposed to be used as a definitive reference rendering, but rather as an indication how one particular application supports and interprets each of the test files.

If something does not seem right in the output, it might either be that this feature has not been implemented yet, has been wrongly implemented, or a regression has crept in recently...

In the web version of this document, you can click on the file name or figure for each example to see the corresponding .ly intermediary file.

Structure of this test suite

Each test file (typically hand-crafted from the MusicXML "specification") checks one particular aspect of MusicXML. A short description of the particular feature for a file is given element inside the file in a comment element of the form:

```
<identification><miscellaneous>
  <miscellaneous-field name="description"> .... </miscellaneous-field>
</miscellaneous></identification>
```

The files are categorized by their first two digits with the following meaning:

- 01-03 ... Basics: Pitches, Rests, Rhythm
- 11-13 ... Staff attributes: Time signatures, Clefs, Key signatures
- 21-24 ... Note settings: Chorded notes, note heads, tuplets, grace notes
- 31-33 ... Notations and articulations: Dynamics (staff-attached), Notations (note-attached), Spanners
- 41-44 ... Parts: Multiple parts, multi-voice parts, multi-staff parts
- 45-46 ... Measure issues and repeats
- 51-52 ... Page issues: Header fields, page layout
- 55-59 ... Exact positioning of items, offsets, etc.

- 61-69 ... Vocal music
- 71-75 ... Instrument-specific: Guitar (Chord, fretboards), Transposing instruments, Percussion, Figured Bass, Others
- 81-89 ... MIDI generation (all sound-related issues)
- 90-99 ... Various Other: Compressed MusicXML files, compatibility with broken MusicXML files exported by other applications

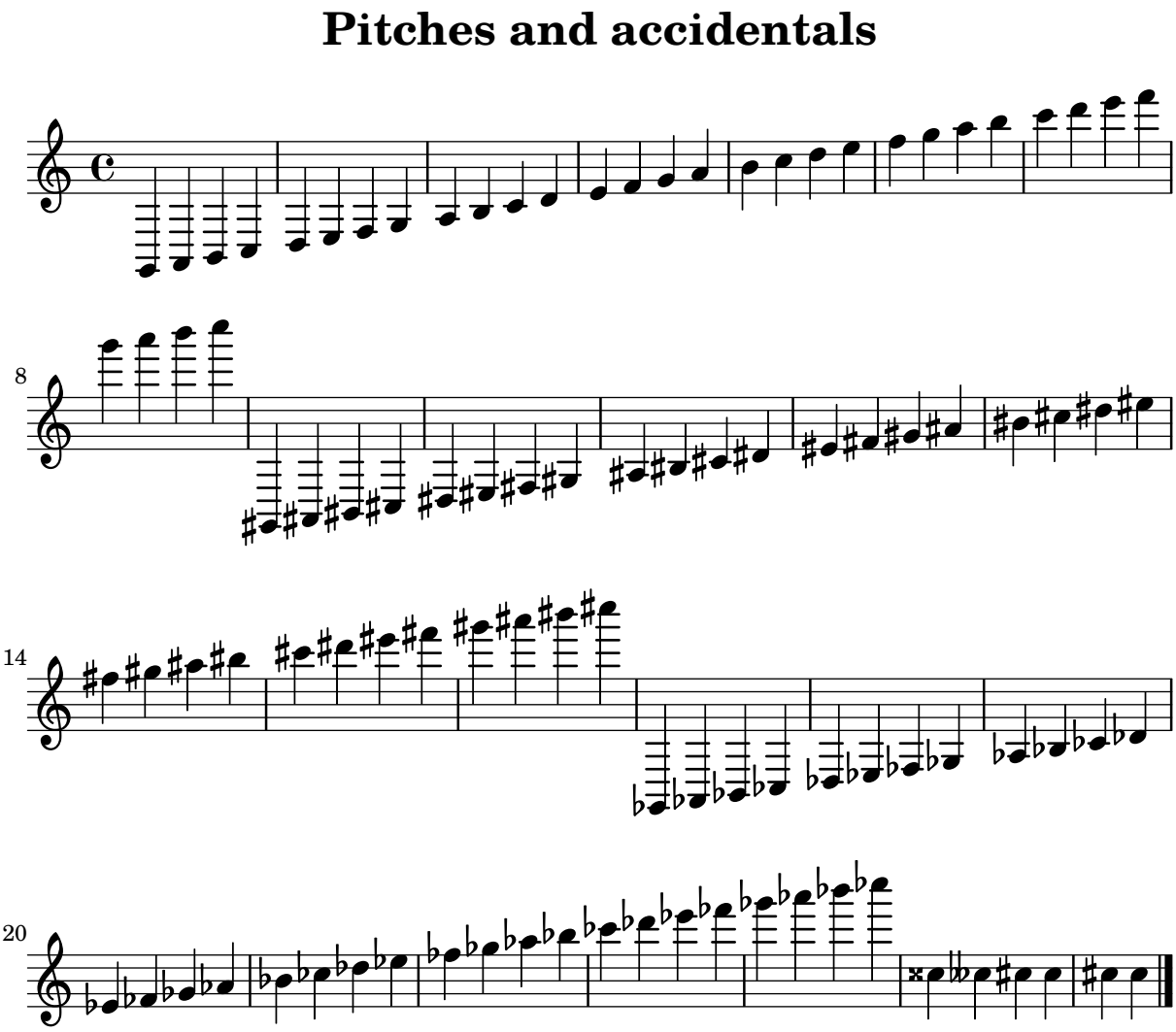
Some of the catetories (in particular the exact item positioning and the MIDI generation don'thave any test cases yet.

2 Test cases

01 ... Pitches

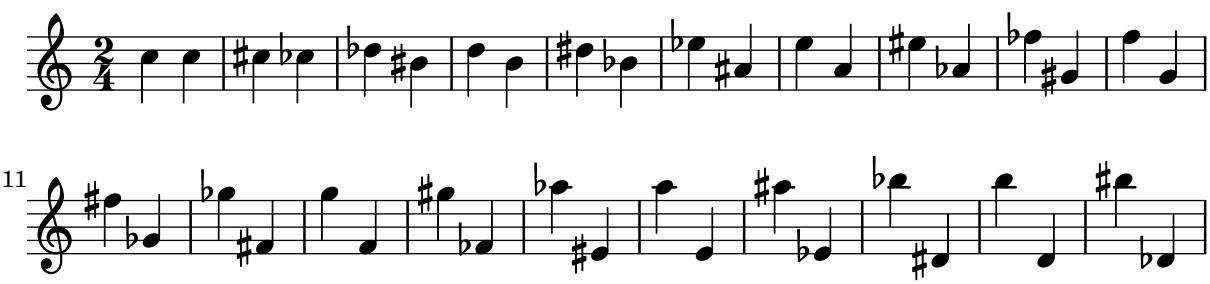
All pitches from G to c”” in ascending steps; First without accidentals, then with a sharp and then with a flat accidental. Double alterations and cautionary accidentals are tested at the end.

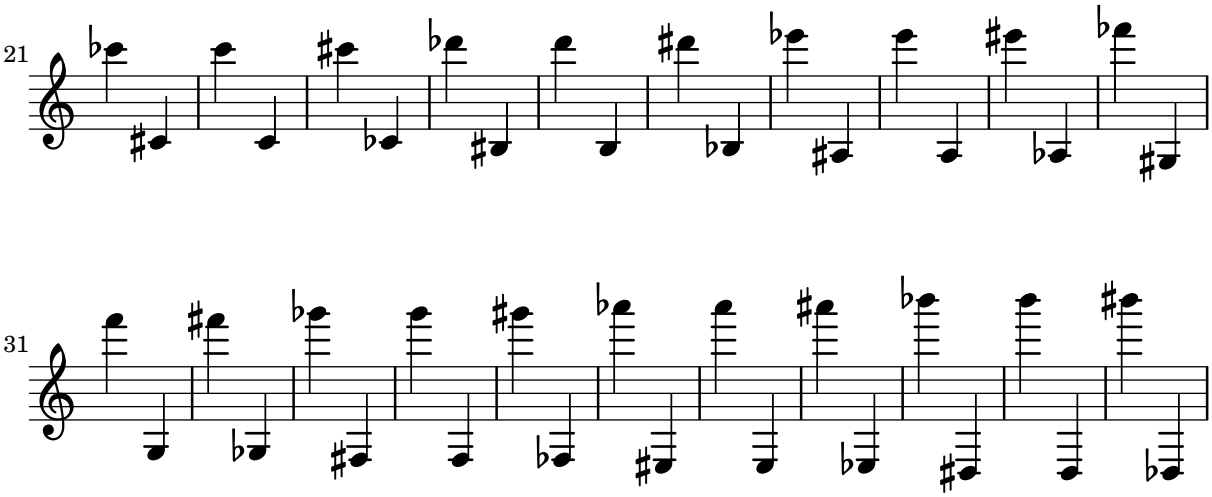
`01a-Pitches-Pitches.xml`



All pitch intervals in ascending jump size.

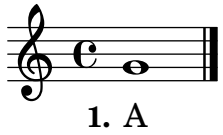
`01b-Pitches-Intervals.xml`
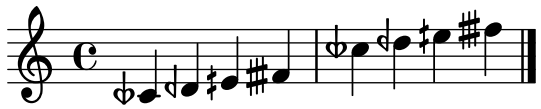




The <voice> element of notes is optional in MusicXML (although Dolet always writes it out). Here, there is one note with lyrics, but without a voice assigned. It should still be correctly converted.

`01c-Pitches-NoVoiceElement.xml`
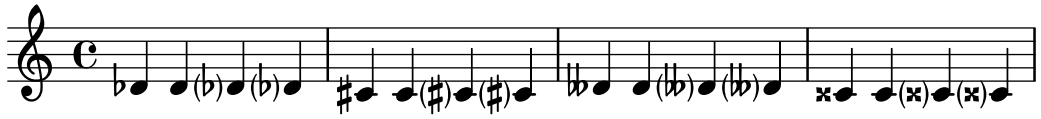


Some microtones: c flat-and-a-half, d half-flat, e half-sharp, f sharp-and-a half. Once in the lower and once in the upper region of the staff.

`01d-Pitches-Microtones.xml`
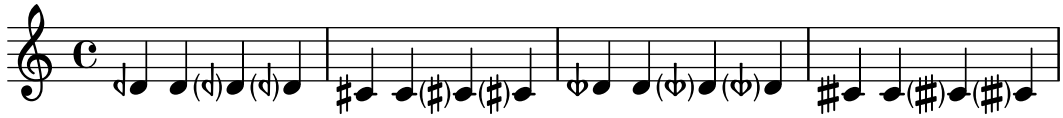


Accidentals can be cautionary or editorial. Each measure has a normal accidental, an editorial, a cautionary and an editioal and cautionary accidental.

`01e-Pitches-ParenthesizedAccidentals.xml`



Microtone accidentals can be cautionary or editorial. Each measure has a normal accidental, an editorial, a cautionary and an editioal and cautionary accidental.

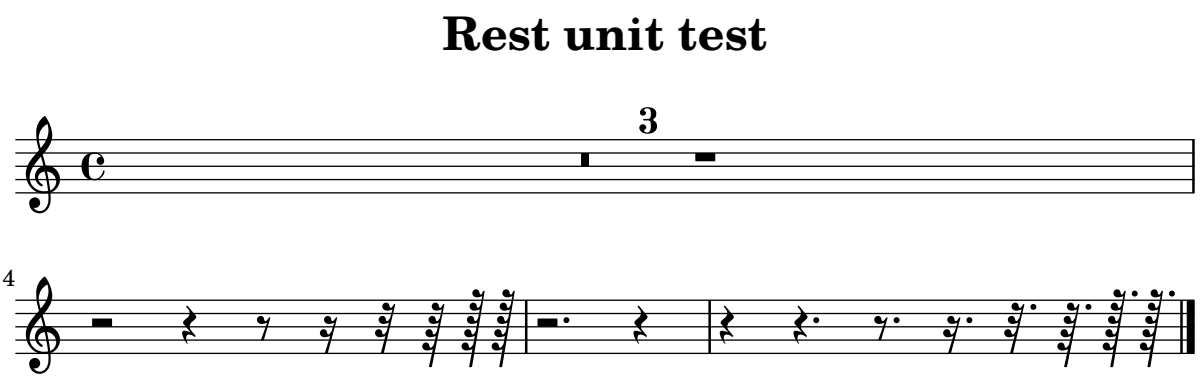
`01f-Pitches-ParenthesizedMicrotoneAccidentals.xml`



02 ... Rests
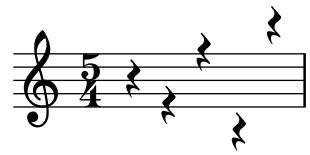
All different rest lengths: A two-bar multi-measure rest, a whole rest, a half, etc. until a 128th-rest; Then the same with dotted durations.

`02a-Rests-Durations.xml`



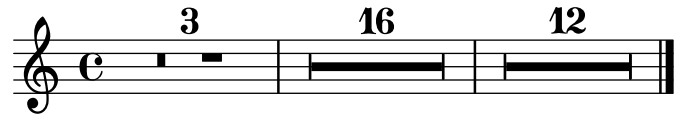
Rests can have explicit pitches, where they are displayed. The first rest uses no explicit position and should use the default position, all others are explicitly positioned somewhere else.

`02b-Rests-PitchedRests.xml`



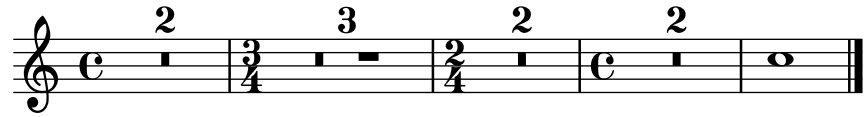
Four multi-measure rests: 3 measures, 15 measures, 1 measure, and 12 measures.

`02c-Rests-MultiMeasureRests.xml`
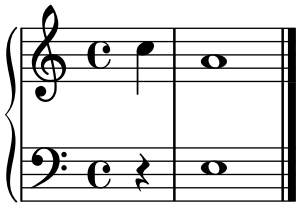


Multi-Measure rests should always be converted into durations that are a multiple of the time signature.

`02d-Rests-Multimeasure-TimeSignatures.xml`



In some cases, a rest might not have its type attribute set (this happens, for example, with voices in Finale, where you don't manually insert a rest).
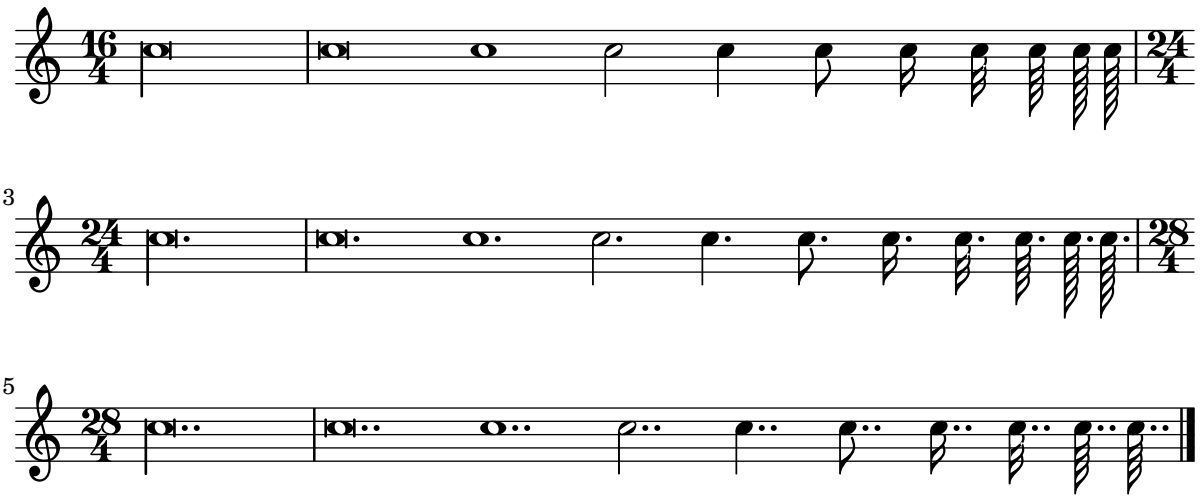
`02e-Rests-NoType.xml`



03 ... Rhythm

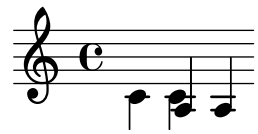
All note durations, from long, brevis, whole until 128th; First with their plain values, then dotted and finally doubly-dotted.

`03a-Rhythm-Durations.xml`
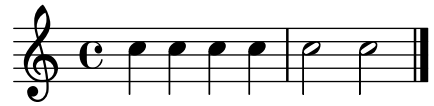


Two voices with a backup, that does not jump to the beginning for the measure for voice 2, but somewhere in the middle. Voice 2 thus won't have any notes or rests for the first beat of the measures.

`03b-Rhythm-Backup.xml`



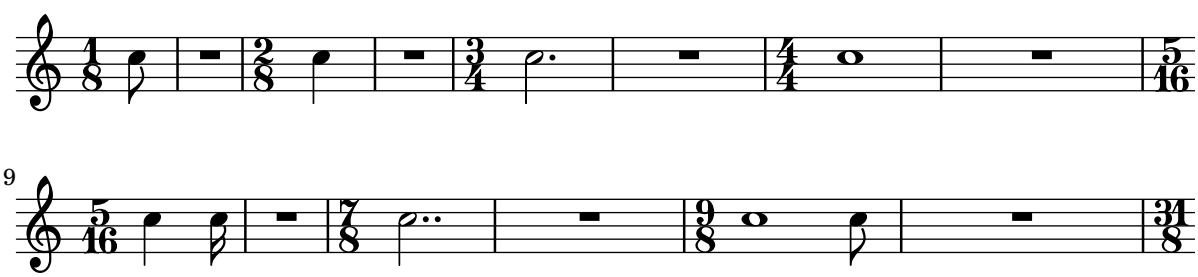
Although uncommon, the divisions of a quarter note can change somewhere in the middle of a MusicXML file. Here, the first half measure uses a division of 1, which then changes to 8 in the middle of the first measure and to 38 in the middle of the second measure.

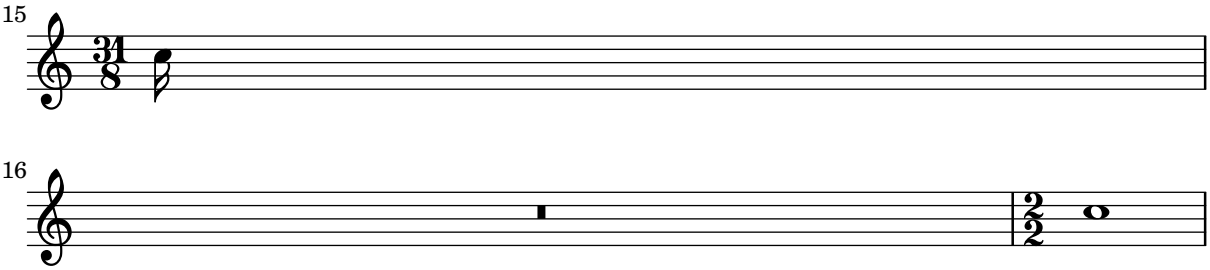
`03c-Rhythm-DivisionChange.xml`



Several durations can be written with dots. For multimeasure rests, we can also have durations that cannot be expressed with dotted notes (like 5/8).

`03d-Rhythm-DottedDurations-Factors.xml`

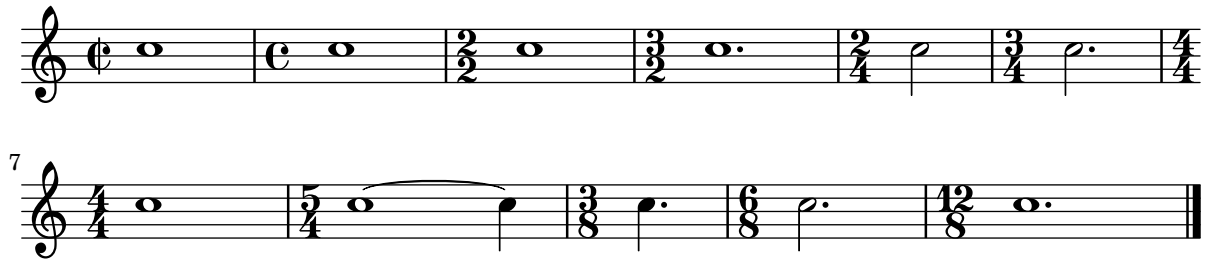




11 ... Time signatures

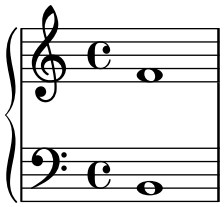
Various time signatures: 2/2 (alla breve), 4/4 (C), 2/2, 3/2, 2/4, 3/4, 4/4, 5/4, 3/8, 6/8, 12/8

`11a-TimeSignatures.xml`
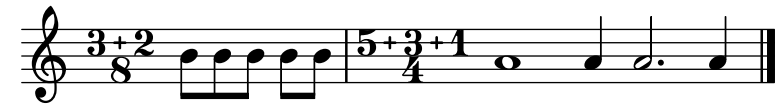


A score without a time signature (but with a key and clefs)

`11b-TimeSignatures-NoTime.xml`
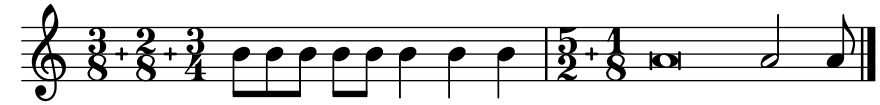


Compound time signatures with same denominator: (3+2)/8 and (5+3+1)/4.

`11c-TimeSignatures-CompoundSimple.xml`
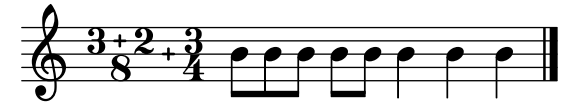


Compound time signatures with separate fractions displayed: 3/8+2/8+3/4 and 5/2+1/8.

`11d-TimeSignatures-CompoundMultiple.xml`
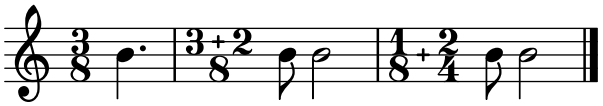


Compound time signatures of mixed type: (3+2)/8+3/4.

`11e-TimeSignatures-CompoundMixed.xml`
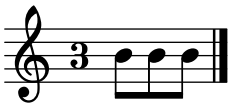


A time signature of 3/8 with the symbol=" cut" attribute and two symbol="single-number" attributes with compound time signatures. Shall the symbol be ignored in this case?

11f-TimeSignatures-SymbolMeaning.xml
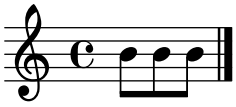


Time signature displayed as a single number.

11g-TimeSignatures-SingleNumber.xml



Senza-misura time signature

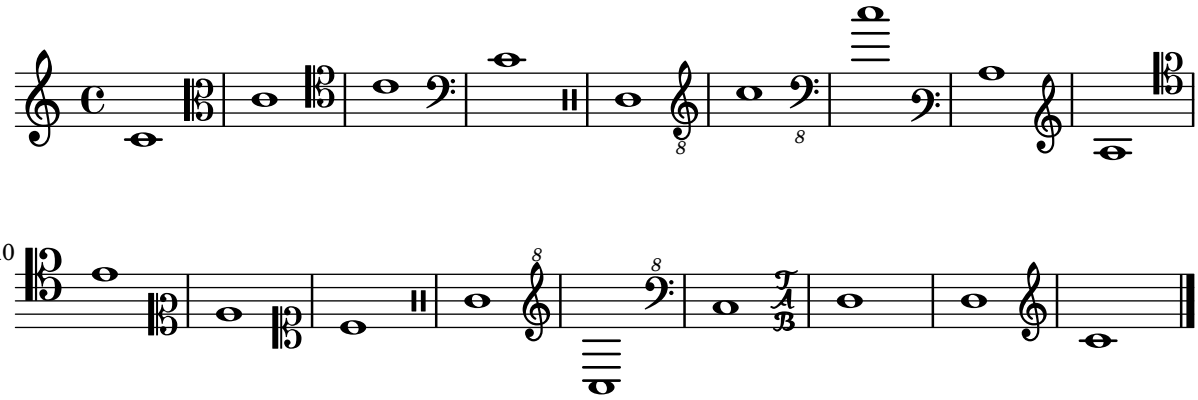
11h-TimeSignatures-SenzaMisura.xml



12 ... Clefs

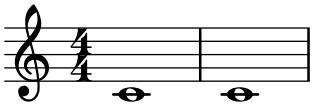
Various clefs: G, C, F, percussion, TAB and none; some are also possible with transposition and on other staff lines than their default (e.g. soprano/alto/tenor/baritone C clefs); Each measure shows a different clef (measure 17 has the " none" clef), only measure 18 has the same treble clef as measure 1.

12a-Clefs.xml



A score without any key or clef defined. The default (4/4 in treble clef) should be used.
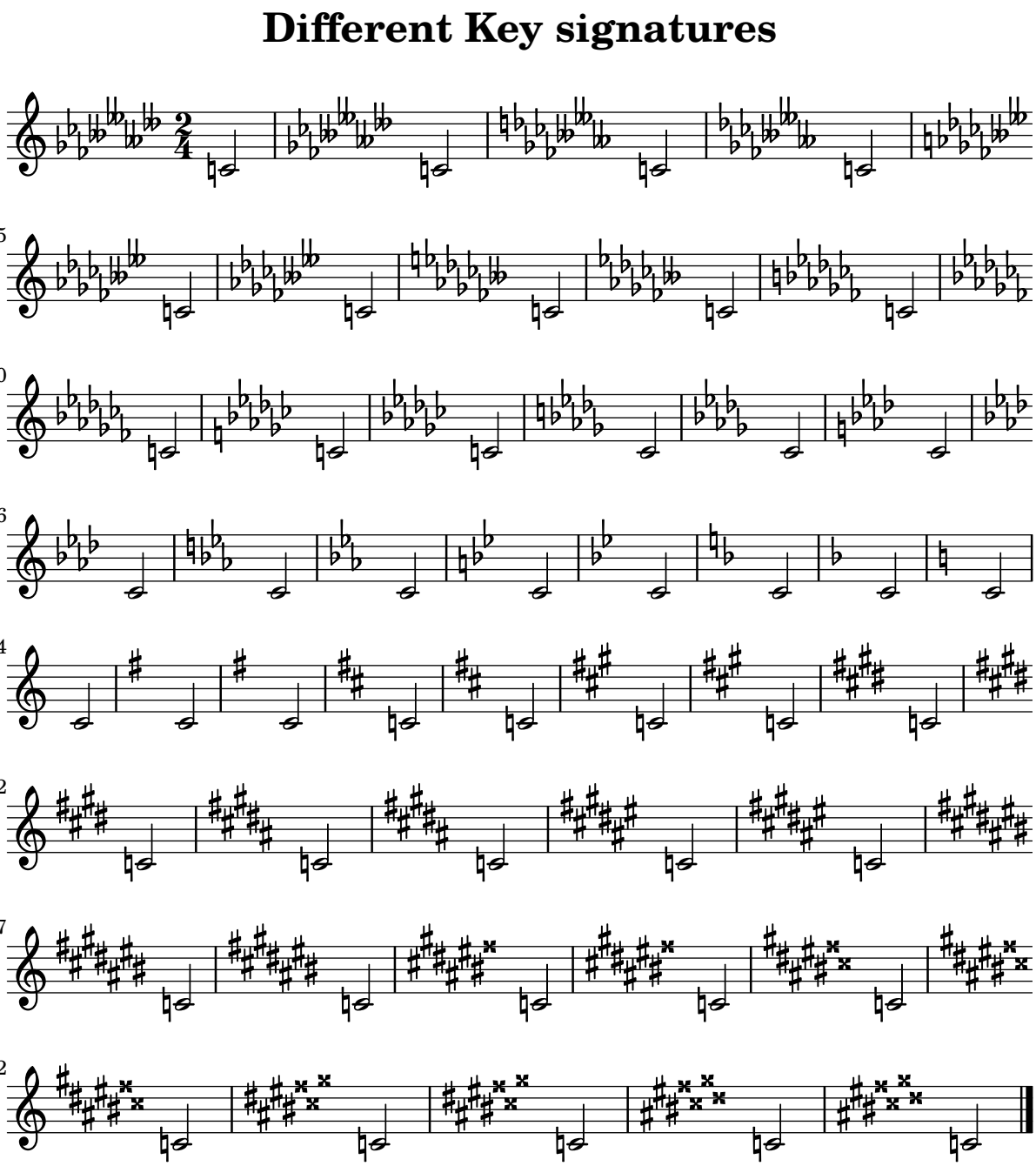
12b-Clefs-NoKeyOrClef.xml



13 ... Key signatures

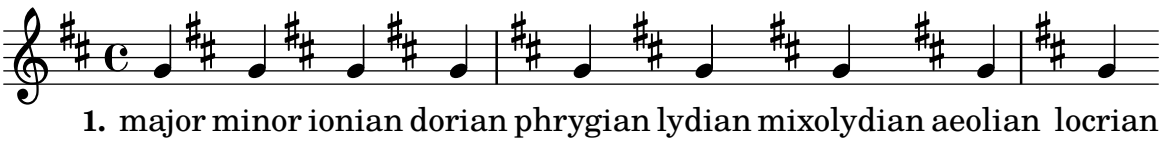
Various key signature: from 11 flats to 11 sharps (each one first one measure in major, then one measure in minor)

`13a-KeySignatures.xml`



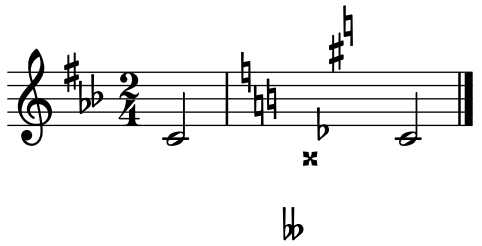
All different modes: major, minor, ionian, dorian, phrygian, lydian, mixolydian, aeolian, and locrian; All modes are given with 2 sharps.

`13b-KeySignatures-ChurchModes.xml`
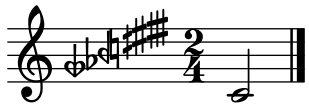


Non-traditional key signatures, where each alteration is separately given. Here we have (f sharp, a flat, b flat) and (c flatflat, g sharp sharp, d flat, b sharp, f natural), where in the second case an explicit octave is given for each alteration.

13c-KeySignatures-NonTraditional.xml



Non-traditional key signatures with microtone alterations: (g flat-and-a-half, a flat, b half-flat, c natural, d half-sharp, e sharp, f sharp-and-a-half).

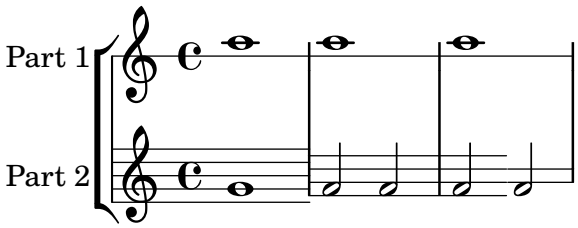
13d-KeySignatures-Microtones.xml



14 ... Staff attributes

The number of staff lines can be modified by using the staff-lines child of the staff-details attribute. This can happen globally (the first staff has one line globally) or during the part at the beginning of a measure and even inside a measure (the second part has 5 lines initially, 4 at the beginning of the second measure, and 3 starting in the middle of the third measure).
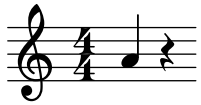
14a-StaffDetails-LineChanges.xml



21 ... Chorded notes
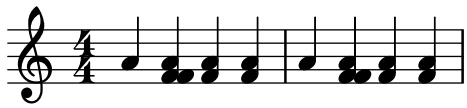
One simple chord consisting of two notes.

21a-Chord-Basic.xml
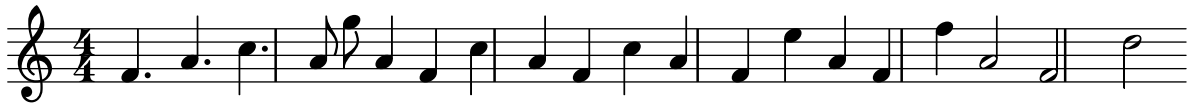


Some subsequent (identical) two-note chords.

21b-Chords-TwoNotes.xml



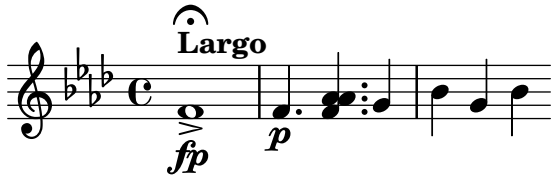
Some three-note chords, with various durations.

21c-Chords-ThreeNotesDuration.xml
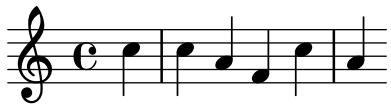


Chords in the second measure, after several ornaments in the first measure and a p at the beginning of the second measure.

21d-Chords-SchubertStabatMater.xml



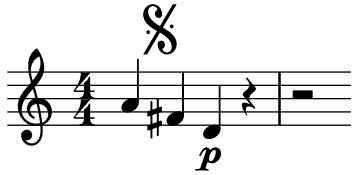
Check for proper chord detection after a pickup measure (i.e. the first beat of the measure is not aligned with multiples of the time signature)!

21e-Chords-PickupMeasures.xml



Between the individual notes of a chord there can be direction or harmony elements, which should be properly assigned to the chord (or the position of the chord).

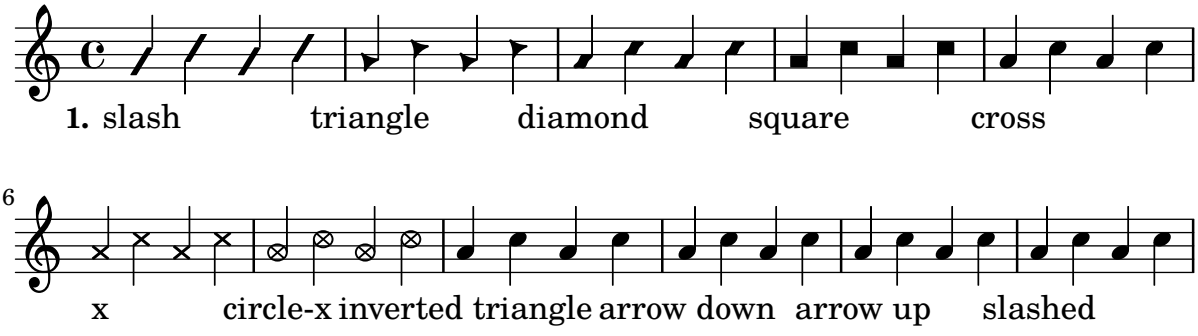
21f-Chord-ElementInBetween.xml

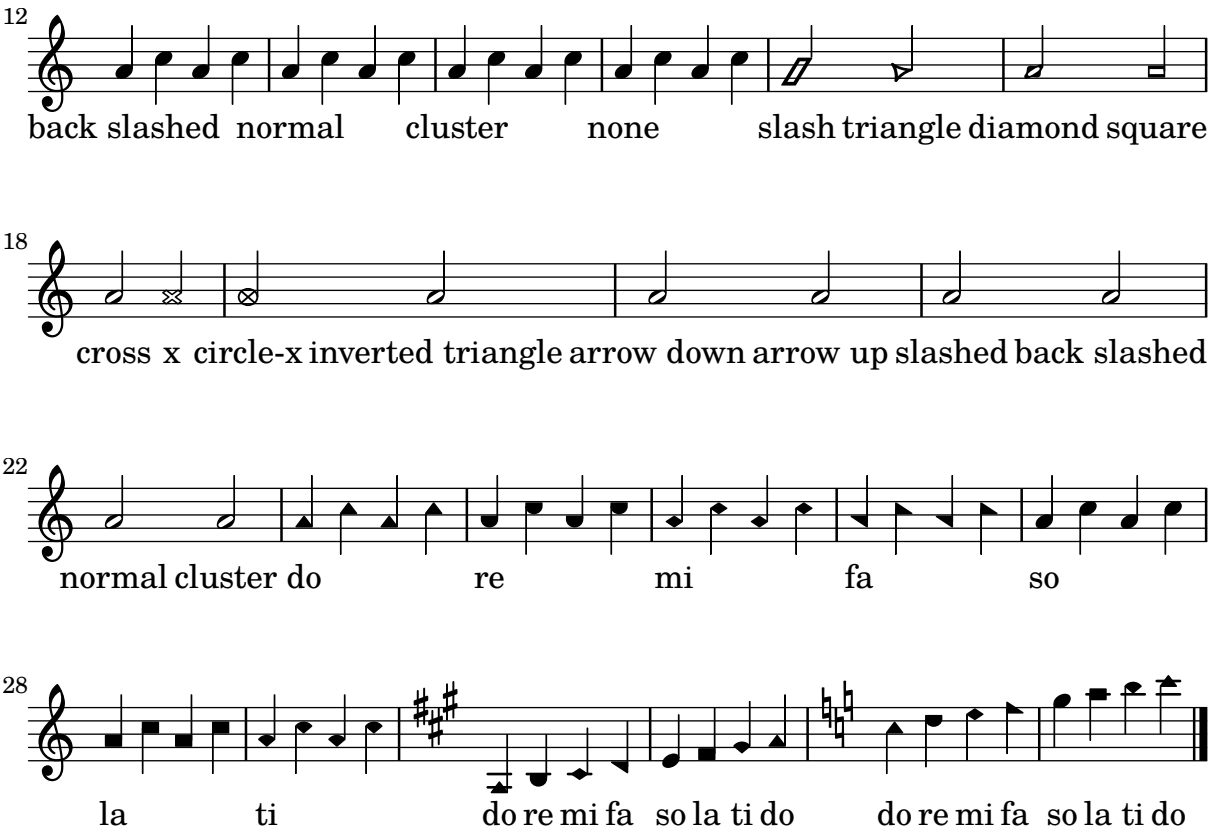


22 ... Note settings, heads, etc.

Different note styles, using the <notehead> element. First, each note head style is printed with four quarter notes, two with filled heads, two with unfilled heads, where first the stem is up and then the stem is down. After that, each note head style is printed with a half note (should have an unfilled head by default). Finally, the Aiken note head styles are tested, once with stem up and once with stem down.

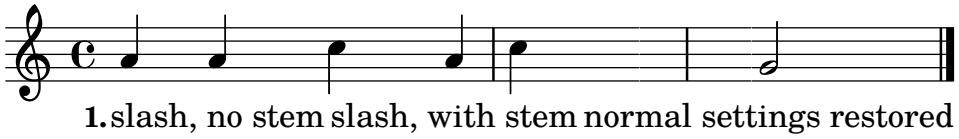
22a-Noteheads.xml





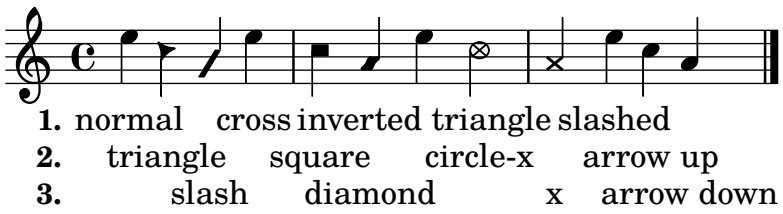
Staff-connected note styles: slash notation, hidden notes (with and without hidden staff lines)

`22b-Staff-Notestyles.xml`
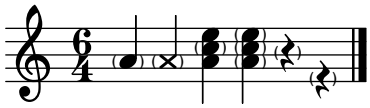


Different note styles for individual notes inside a chord, using the <notehead> element.

`22c-Noteheads-Chords.xml`



Parenthesized note heads. First, a single parenthesized note is tested, once with a normal and then with a non-standard notehead, then two chords with some/all parenthesized noteheads and finally a parenthesized rest.
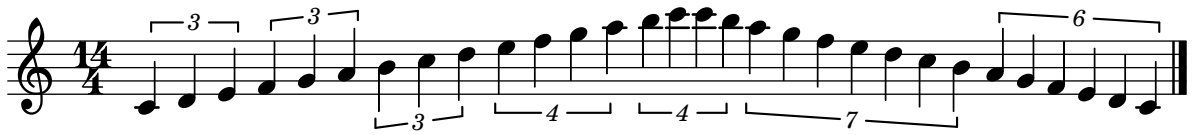
`22d-Parenthesized-Noteheads.xml`



23 ... Triplets, Tuplets

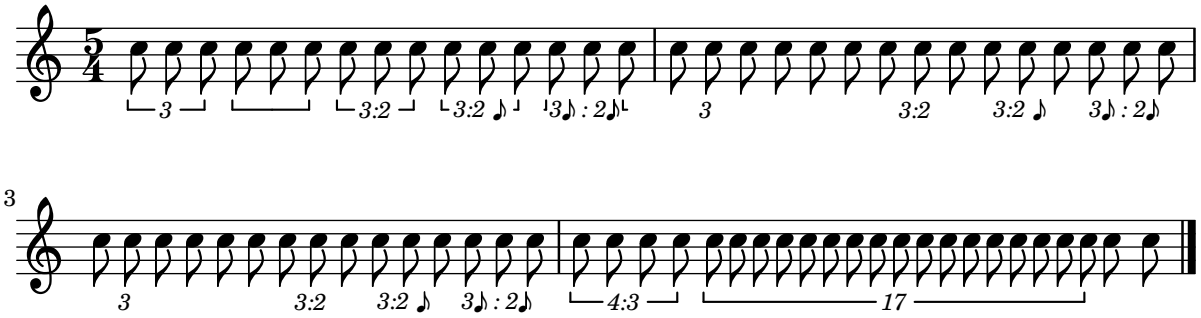
Some tuplets (3:2, 3:2, 3:2, 4:2, 4:1, 7:3, 6:2) with the default tuplet bracket displaying the number of actual notes played. The second tuplet does not have a number attribute set.

23a-Tuplets.xml
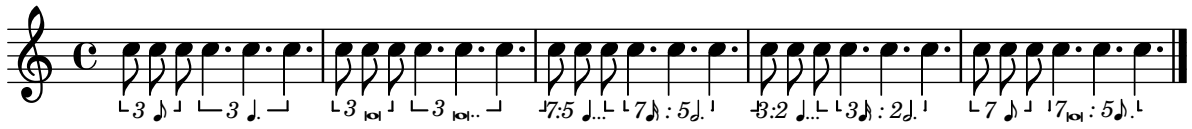


Different tuplet styles: default, none, x:y, x:y-note; Each with bracket, slur and none. Finally, non-standard 4:3 and 17:2 tuplets are given.

23b-Tuplets-Styles.xml
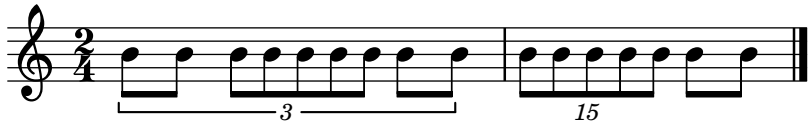


Displaying tuplet note types, that might not coincide with the displayed note. The first two tuplets take the type from the note, the second two from the <time-modification> element, the remaining pair of tuplets from the <tuplet> notation element. The tuplets in measure 3 specify both a number of notes and a type inside the <tuplet-actual> and <tuplet-normal> elements, the ones in measure 4 specify only a note type (but no number), and the ones in measure 5 specify only a number of tuplet-notes (but no type, which is deduced from the note's type). The first tuplet of measures 3-5 uses 'display-type=" actual"', the second one 'display-type="both"'. FIXME: The tuplet-normal should coincide with the real notes!

23c-Tuplet-Display-NonStandard.xml



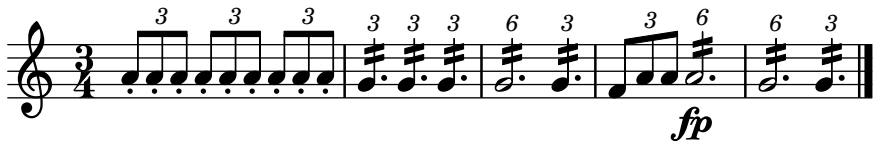
Tuplets can be nested. Here there is a 5:2 tuplet inside a 3:2 tuple (all consisting of written eighth notes).

23d-Tuplets-Nested.xml
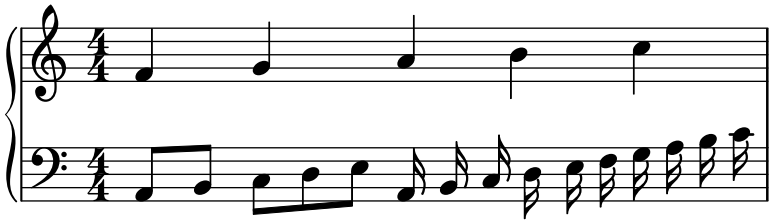


Tremolo tuplets are tuplets on single notes with a tremolo ornament. The application shall correctly import these notes with 2/3 or their time...

23e-Tuplets-Tremolo.xml



Some " triplets" on the end of the first and in the second staff, using only <time-modification>, but not explicit tuplet bracket. Thus, the duration of the notes in the second staff should be scaled properly in comparison to staff 1, but no visual indication about the tuplets is given.
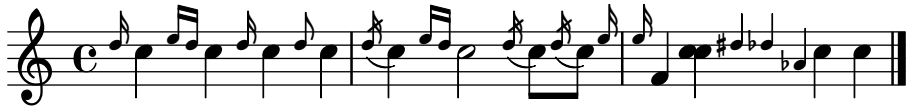
23f-Tuplets-DurationButNoBracket.xml



24 ... Grace notes
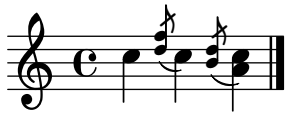
Different kinds of grace notes: acciaccatura, appoggiatura; beamed grace notes; grace notes with accidentals; different durations of the grace notes.

24a-GraceNotes.xml
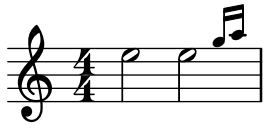


Chords as grace notes.

24b-ChordAsGraceNote.xml
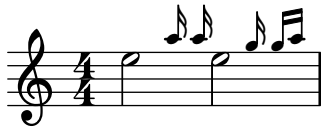


A grace note that appears at the measure end (without any steal-from-* attribute set). Some applications need to convert this into an after-grace.

24c-GraceNote-MeasureEnd.xml
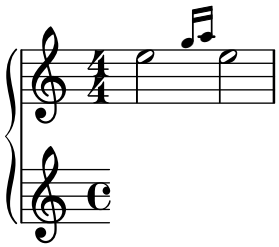


Some grace notes and after-graces (indicated by steal-time-previous and steal-time-following).

24d-AfterGrace.xml
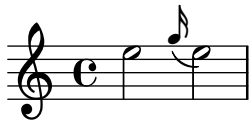


A grace note on a different staff than the actual note.

`24e-GraceNote-StaffChange.xml`



A grace note with a slur to the actual note. This can be interpreted as acciaccatura or appoggiatura, depending on the existence of a slash.

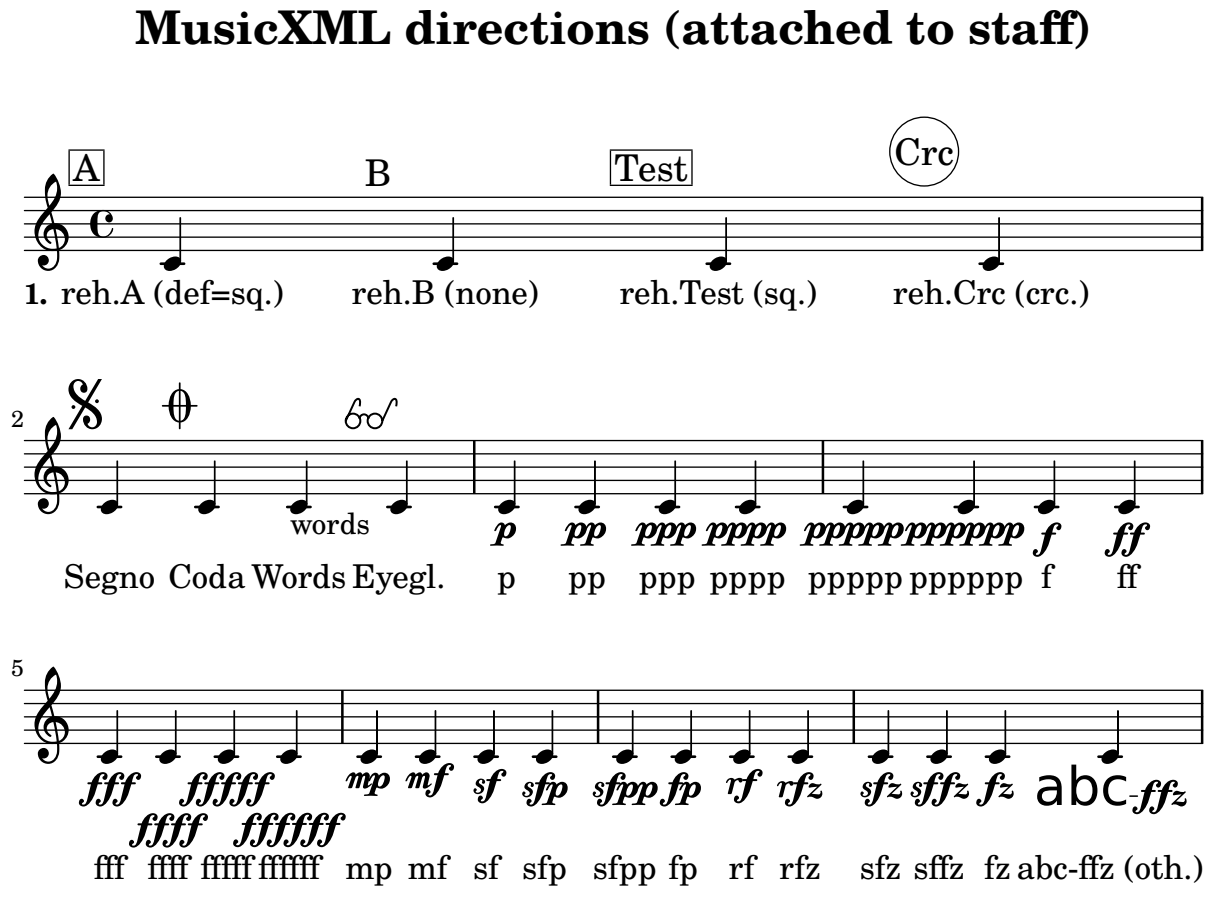
`24f-GraceNote-Slur.xml`
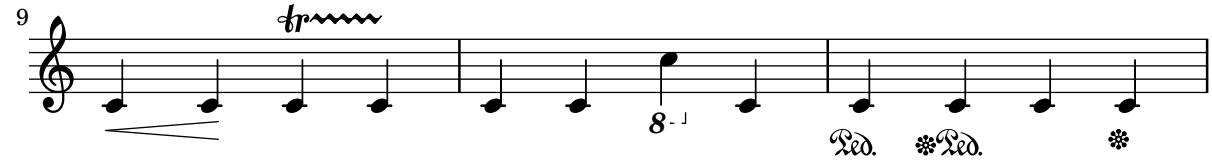


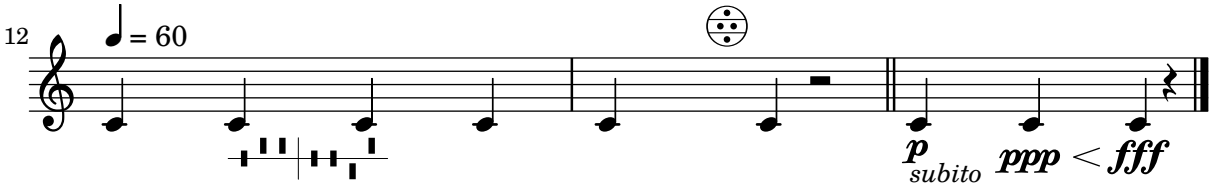
31 ... Dynamics and other single symbols

All <direction> elements defined in MusicXML. The lyrics for each note describes the direction element assigned to that note.

`31a-Directions.xml`
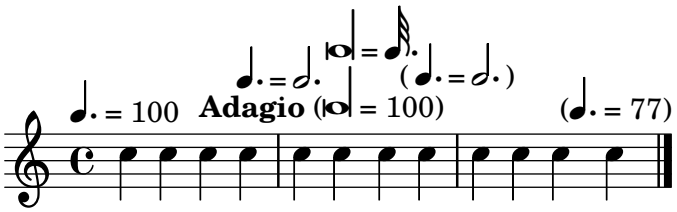






Tempo Markings: note=bpm, text (note=bpm), note=note, (note=note), (note=bpm)

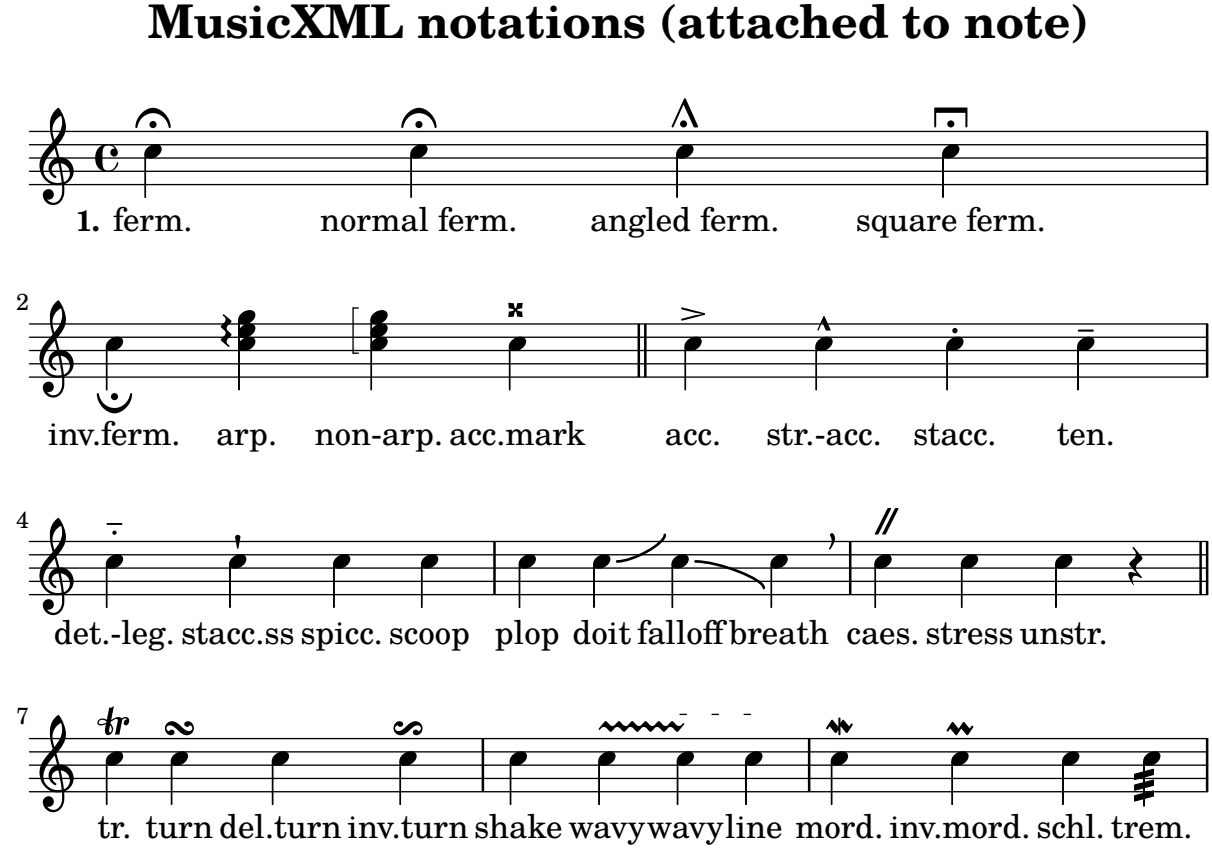
`31c-MetronomeMarks.xml`

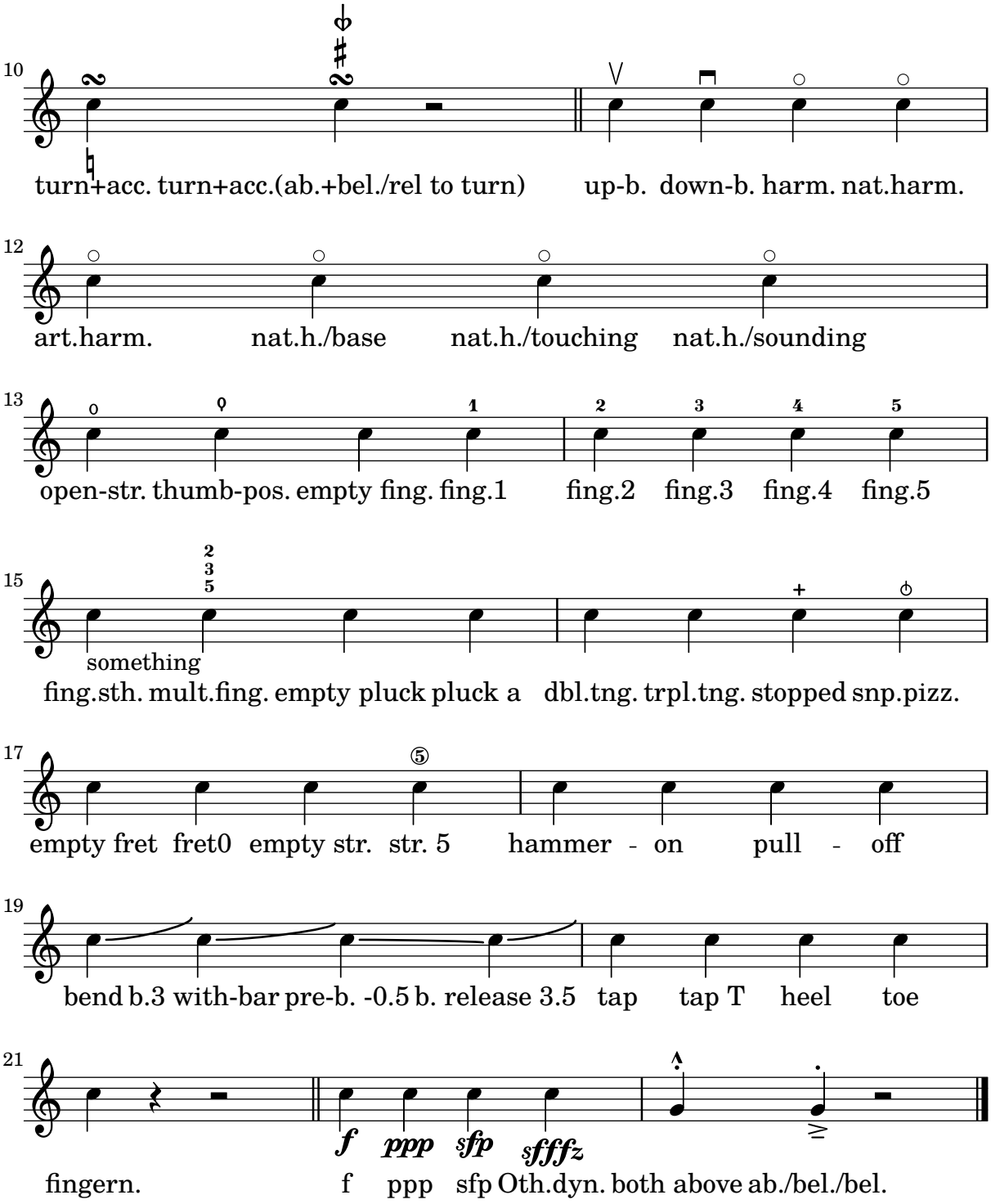


32 ... Notations and Articulations

All <notation> elements defined in MusicXML. The lyrics show the notation assigned to each note.

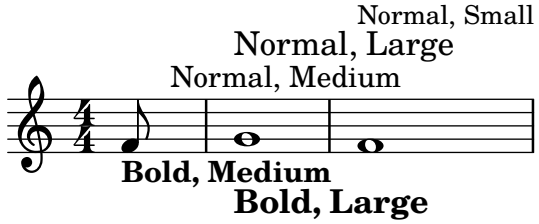
`32a-Notations.xml`





Text markup: different font sizes, weights and colors.
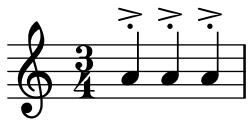
`32b-Articulations-Texts.xml`



It should not make any difference whether two articulations are given inside two different notation elements, inside two different articulations children of the same notation element or
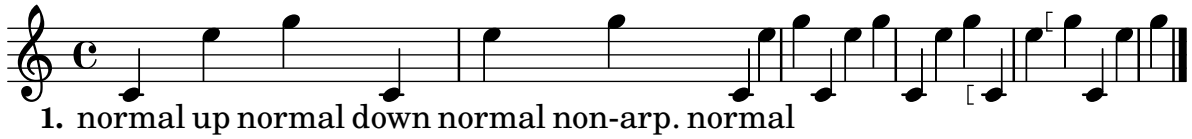
inside the same articulations element. Thus, all three notes should have a staccato and an accent.

32c-MultipleNotationChildren.xml



Different Arpeggio directions (normal, up, down, non-arpeggiate)
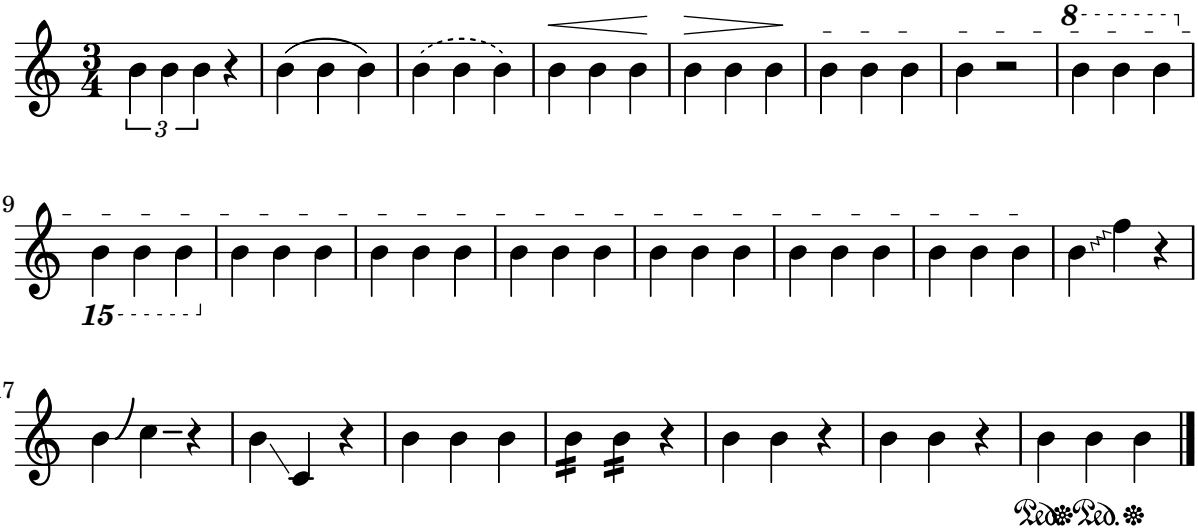
32d-Arpeggio.xml



33 ... Spanners
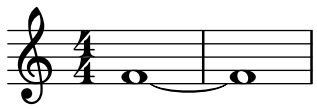
Several spanners defined in MusicXML: tuplet, slur (solid, dashed), tie, wedge (cresc, dim), tr + wavy-line, single-note trill spanner, octave-shift (8va,15mb), bracket (solid down/down, dashed down/down, solid none/down, dashed none/up, solid none/none), dashes, glissando (wavy), bend-alter, slide (solid), grouping, two-note tremolo, hammer-on, pull-off, pedal (down, change, up).

33a-Spanners.xml
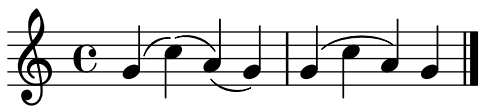


Two simple tied whole notes

33b-Spanners-Tie.xml
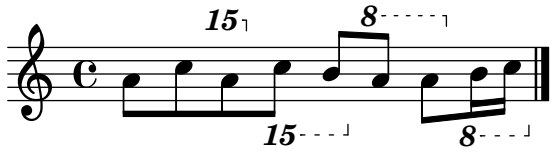


A note can be the end of one slur and the start of a new slur. Also, in MusicXML, nested slurs are possible like in the second measure where one slur goes over all four notes, and another slur goes from the second to the third note.

33c-Spanners-Slurs.xml



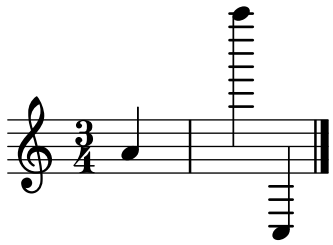
All types of octave shifts (15ma, 15mb, 8va, 8vb)

33d-Spanners-OctaveShifts.xml
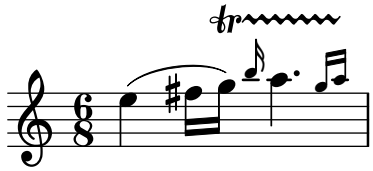


Invalid octave-shifts: 27 down, 11 up.

33e-Spanners-OctaveShifts-InvalidSize.xml
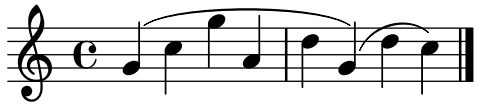


A trill spanner that spans a grace note and ends on an after-grace note at the end of the measure.

33f-Trill-EndingOnGraceNote.xml
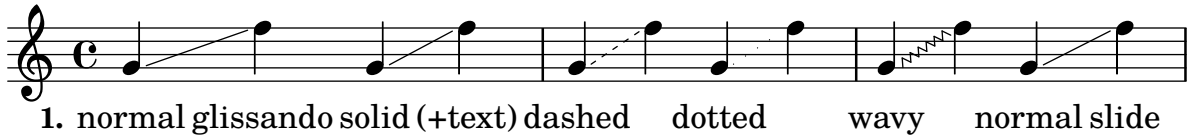


Slurs on chorded notes: Only the first note of the chord should get the slur notation. Some applications print out the slur for all notes – these should be ignored.
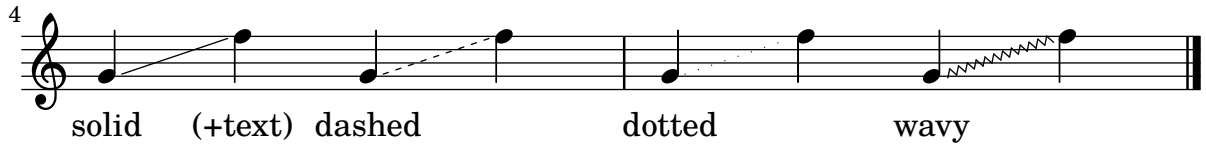
33g-Slur-ChordedNotes.xml



All different types of glissando defined in MusicXML

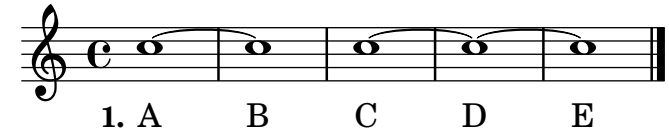
33h-Spanners-Glissando.xml





Several ties that have their end tag missing.

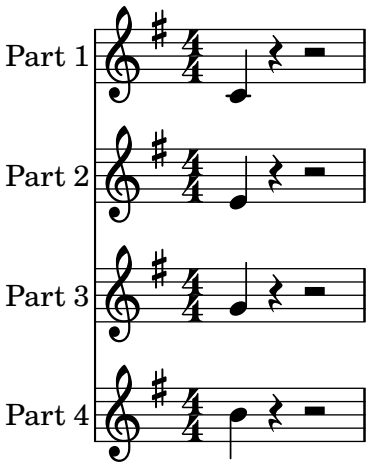
`33i-Ties-NotEnded.xml`



41 ... Multiple parts (staves)

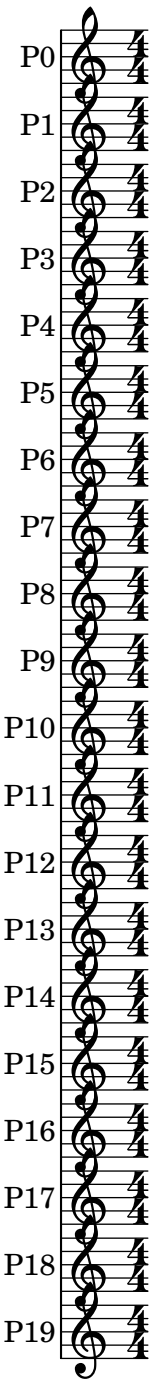
A piece with four parts (P0, P1, P2, P3; different from what Finale creates!). Are they converted in the correct order?

`41a-MultiParts-Partorder.xml`



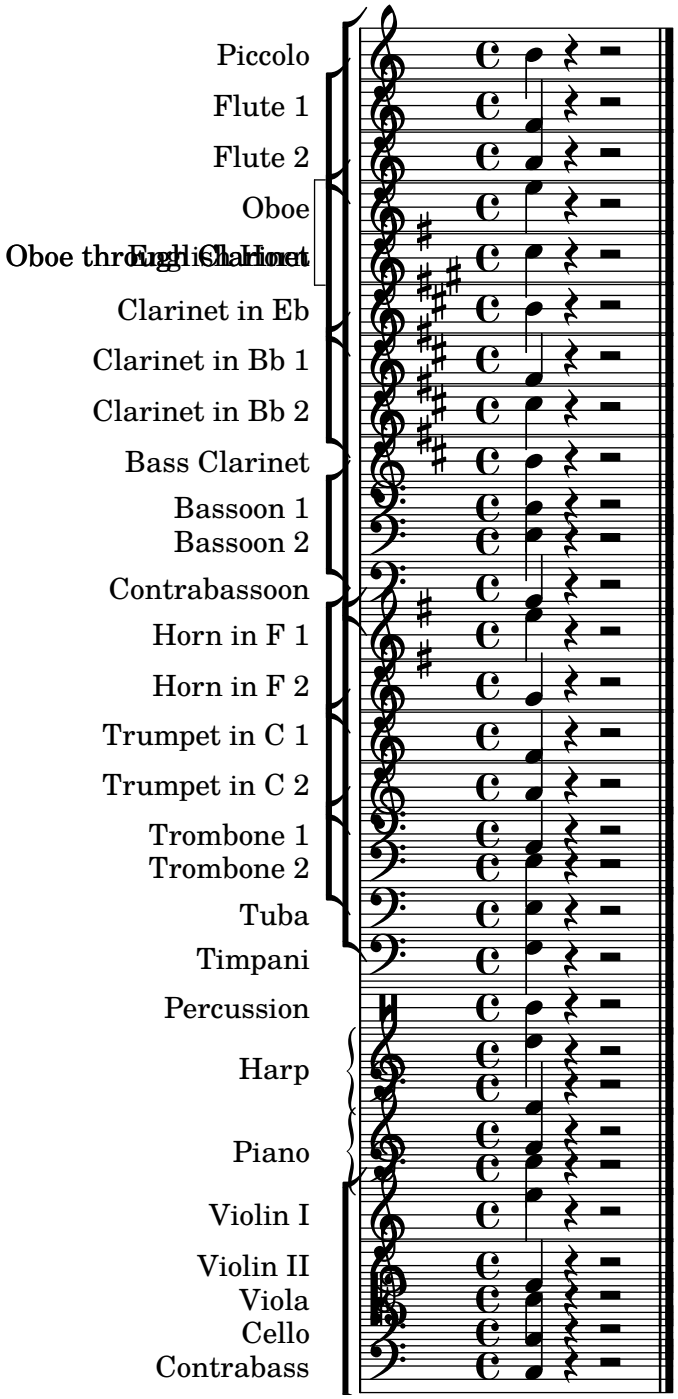
A piece with 20 parts to check whether an application supports that many parts and whether they are correctly sorted.

41b-MultiParts-MoreThan10.xml



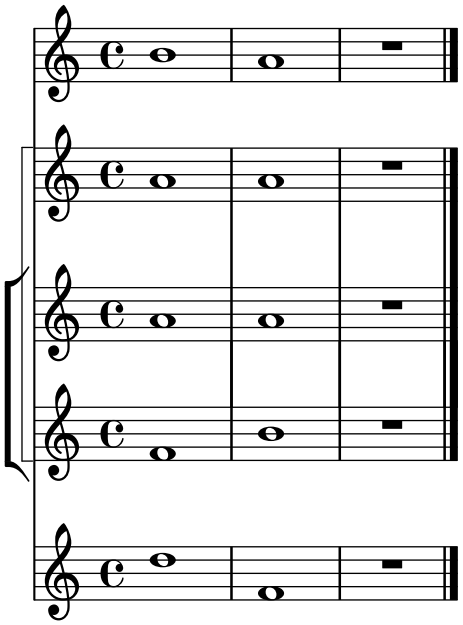
A huge orchestra score with 28 parts and different kinds of nested bracketed groups. Each part/group is assigned a name and an abbreviation to be shown before the staff. Also, most of the groups show unbroken barlines, while the barlines are broken between the groups.

41c-StaffGroups.xml



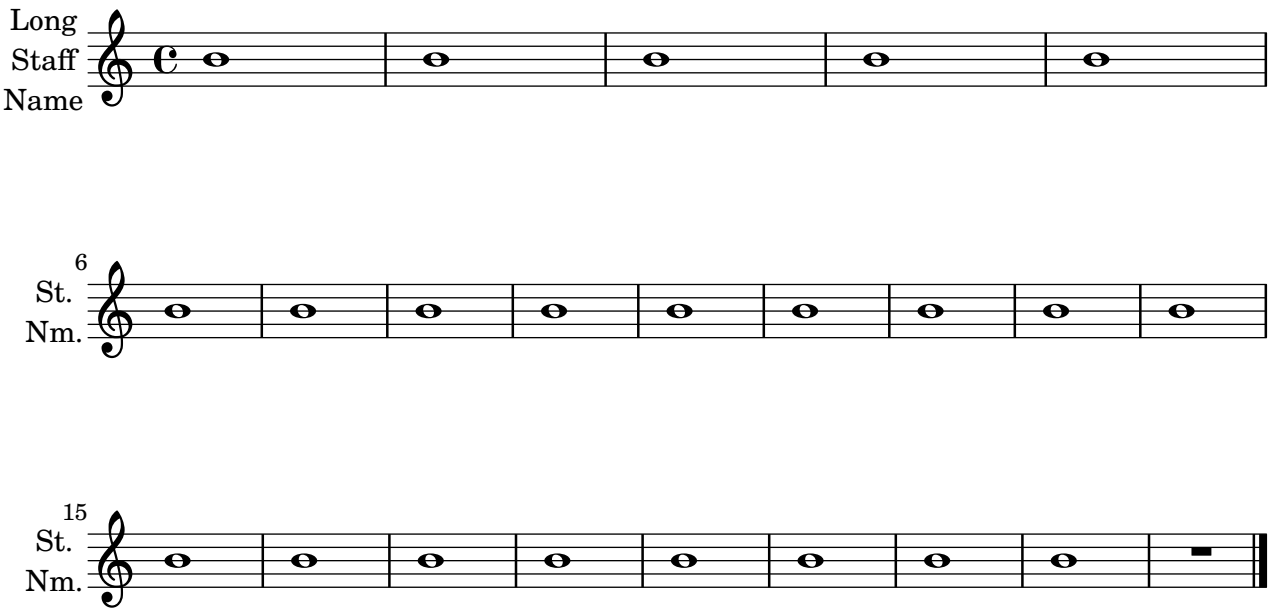
Two properly nested part groups: One group (with a square bracket) goes from staff 2 to 4) and another group (with a curly bracket) goes from staff 3 to 4.

41d-StaffGroups-Nested.xml



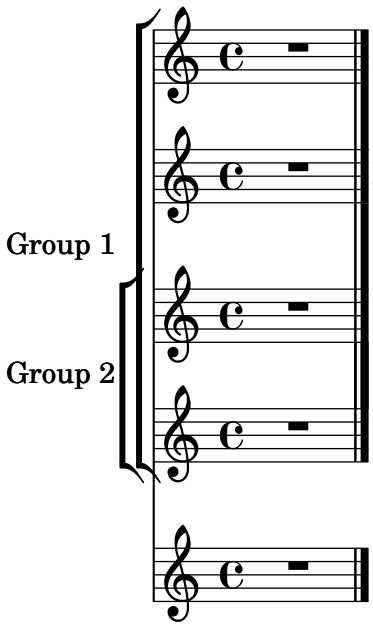
Part names and abbreviations can contain line breaks.

41e-StaffGroups-InstrumentNames-Linebroken.xml



MusicXML allows for overlapping part-groups, while many applications do not allow overlapping groups, but require them to be properly nested. In this case, one group (within parenthesis) goes from staff 1 to 4 and another group (also within parenthesis) goes from staff 3 to 5.

41f-StaffGroups-Overlapping.xml

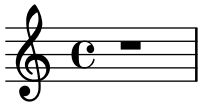


A part with no id attribute. Since this piece has only one part, it is clear which part is described by the one part element.

41g-PartNoId.xml

This piece has more part elements than the part-list section gives. One can either convert all the parts present, but not listed in the part-list, or simply not import / ignore them.

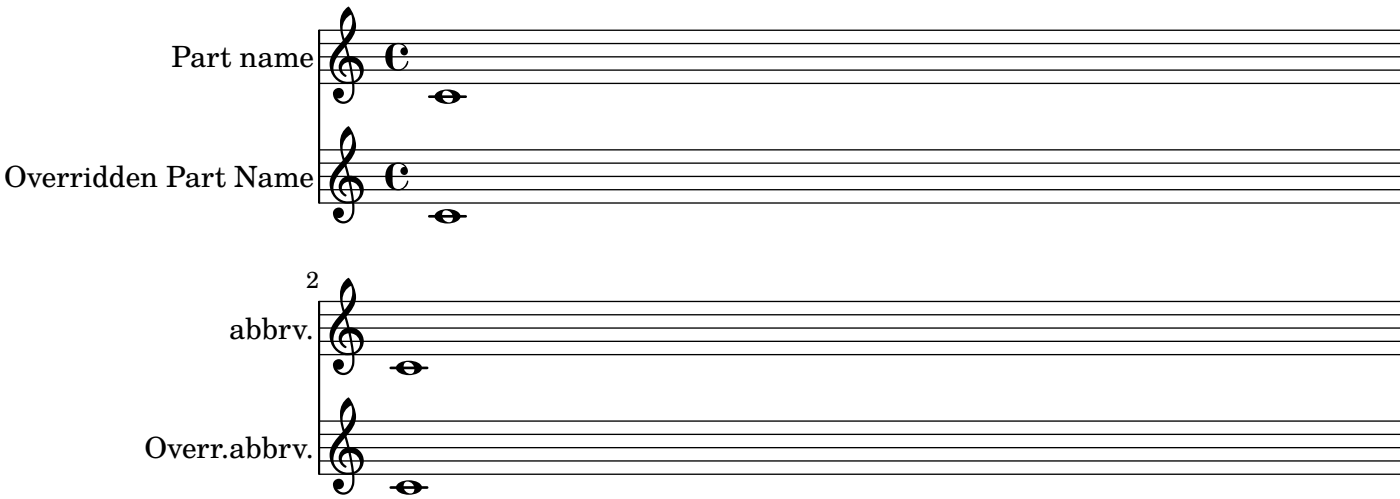
41h-TooManyParts.xml



MusicXML allows part-name and part-name-display in the score-part element. If part-name-display is given, it overrides the part-name for display.

The first staff uses only part-name, while the second one (same part-name) overrides it with a custom text. Similar for the part-abbreviation used in subsequent staves.

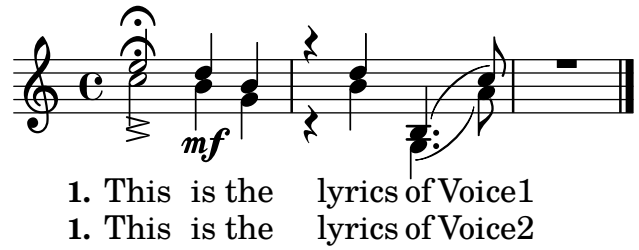
41i-PartNameDisplay-Override.xml



42 ... Multiple voices per staff

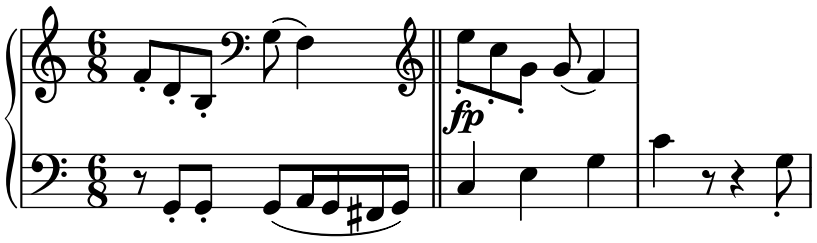
Two voices share one staff. Each voice is assigned some lyrics.

`42a-MultiVoice-TwoVoicesOnStaff-Lyrics.xml`



A multi-voice / multi-staff part with a clef change in the middle of a measure and a <backward> for voice 2 jumping back beyond that clef change.

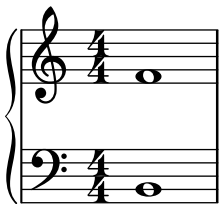
`42b-MultiVoice-MidMeasureClefChange.xml`



43 ... One part on multiple staves
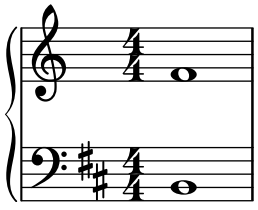
A simple piano staff

`43a-PianoStaff.xml`
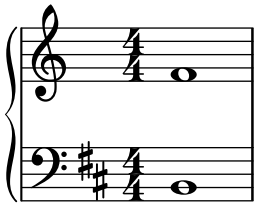


A piano staff with different keys and clefs for each of its staves. The keys and clefs for both staves are given at the very beginning of the measure.

`43b-MultiStaff-DifferentKeys.xml`



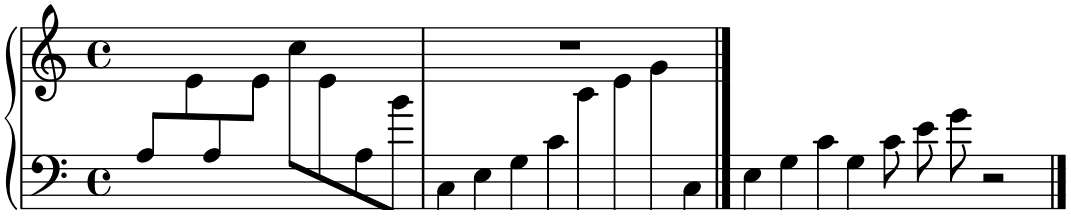
A piano staff with different keys and clefs for each of its staves. The key and clef for the second staff is given only after a backward, just before the first note of the second staff is given, but after the whole measure for staff 1 has been given.

43c-MultiStaff-DifferentKeysAfterBackup.xml



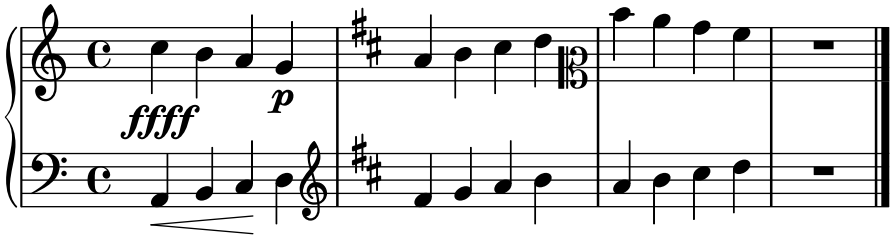
Staff changes in a piano staff. The voice from the second staff has some notes/chords on the first staff. The final two chords have some notes on the first, some on the second staff.

43d-MultiStaff-StaffChange.xml



A piano staff with dynamics and clef changes, where each element (ffff, wedge and clef changes) applies only to one voice or one staff, respectively.

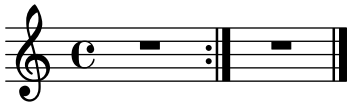
43e-Multistaff-ClefDynamics.xml



45 ... Repeats
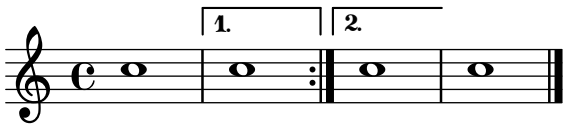
A simple, repeated measure (repeated 5 times)

45a-SimpleRepeat.xml



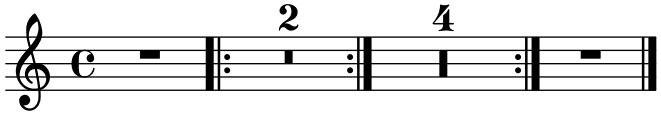
A simple repeat with two alternative endings (volta brackets).

45b-RepeatWithAlternatives.xml



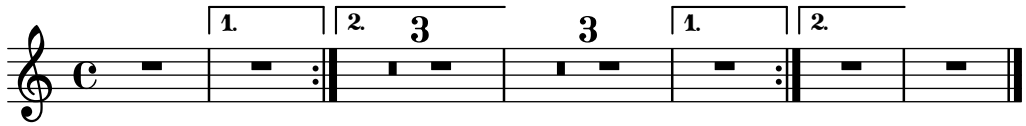
Repeats can also be nested.

45c-RepeatMultipleTimes.xml



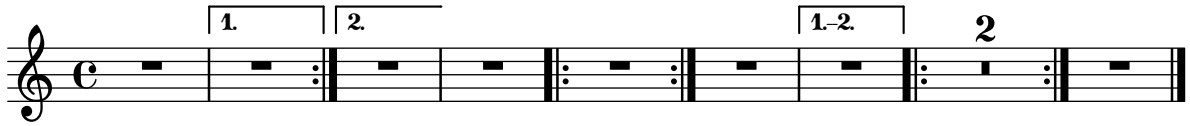
Nested repeats, each with alternative endings.

`45d-Repeats-Nested-Alternatives.xml`



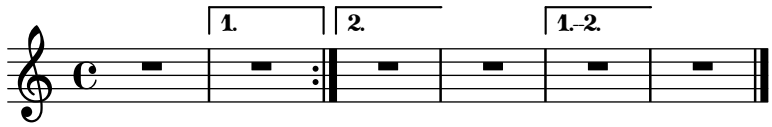
Some more nested repeats with alternatives. The barline between measure 7 and 8 will probably be messed up! (Should be a repeat on both sides!)

`45e-Repeats-Nested-Alternatives.xml`
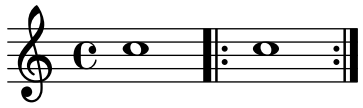


Some more nested repeats with alternatives, where the MusicXML file does not make sense in the first place. How well are applications able to cope with improper repeats and alternatives?

`45f-Repeats-InvalidEndings.xml`



A forward-repeating bar line without an ending repeat bar.

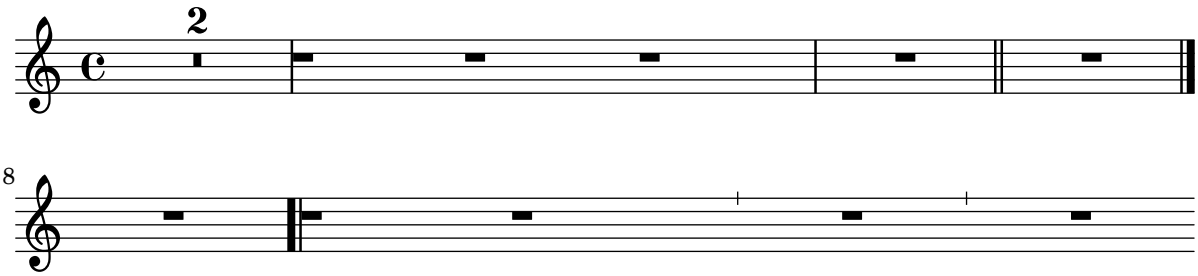
`45g-Repeats-NotEnded.xml`



46 ... Barlines, Measures
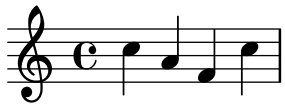
Different types of (non-repeat) barlines: default (no setting), regular, dotted, dashed, heavy, light-light, light-heavy, heavy-light, heavy-heavy, tick, short, none.

`46a-Barlines.xml`
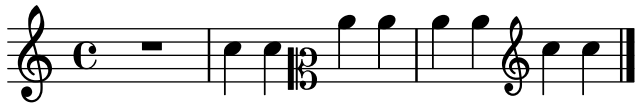


Barlines can appear at mid-measure positions, without using an implicit measure!

`46b-MidmeasureBarline.xml`
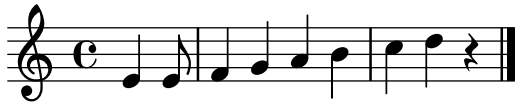


A clef change in the middle of a measure, using either an implicit measure or simply placing the attributes in the middle of the measure.

`46c-Midmeasure-Clef.xml`
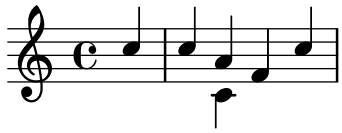


A 3/8 pickup measure, a measure that is split into one (incomplete, only 2/4) measure and an implicit measure, and an incomplete measure (containg 3/4).

`46d-PickupMeasure-ImplicitMeasures.xml`
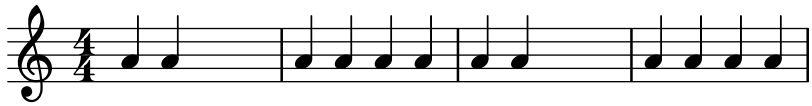


Voice 2 should start at 2nd beat of first full measure.

`46e-PickupMeasure-SecondVoiceStartsLater.xml`



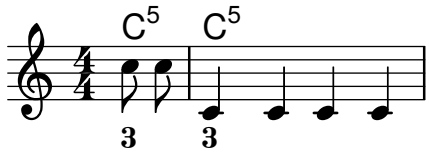
Measures can contain less notes than the time signature says. Here, the first and third measures contain only two quarters instead of four.

`46f-IncompleteMeasures.xml`



Pickup measure with chord names and figured bass.

`46g-PickupMeasure-Chordnames-FiguredBass.xml`
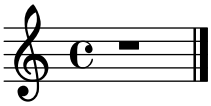


51 ... Header information

Several header fields and part names can contain quotes ("). This test checks whether they are converted/imported without problems (i.e. whether they are correctly escaped when converting).

`51b-Header-Quotes.xml`
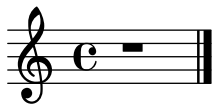
" Quotes" in header fields

Some " Tester" Name

Staff "Test" 

There can be multiple <rights> tags in the identification element of the score. The conversion shall still work, ideally using both of them.

51c-MultipleRights.xml

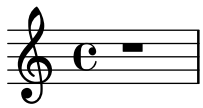


A piece with an empty (but existing) work-title, but a non-empty movement-title. In this case the movement-title should be chosen, even though the work-title exists.

51d-EmptyTitle.xml

Empty work-title, non-empty movement-title

Empty work-title, non-empty movement-title

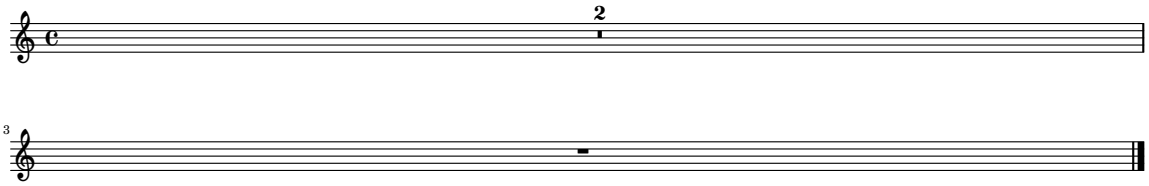


52 ... Page layout

Several page layout settings: paper size, margins, system margins and distances, different fonts, etc.
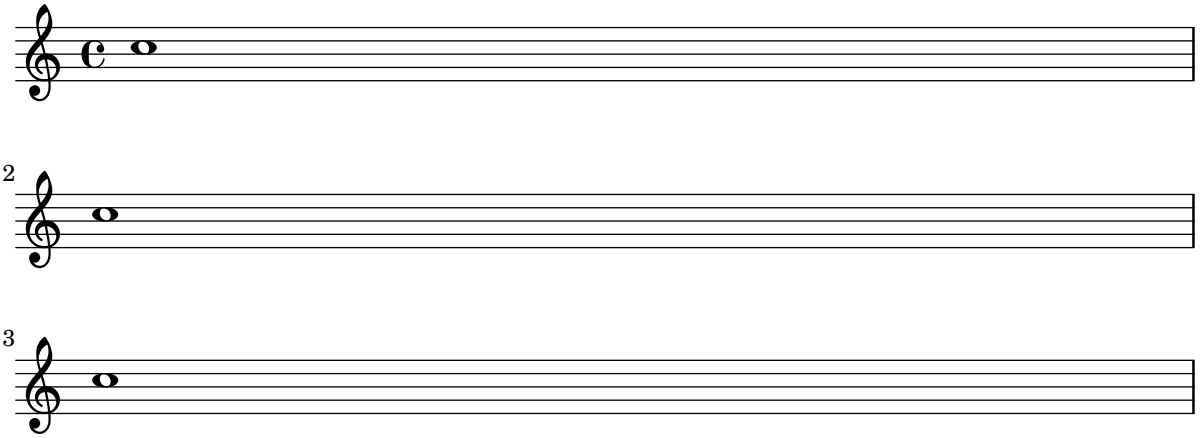
52a-PageLayout.xml

Layout options



System and page breaks, given in a <print> element
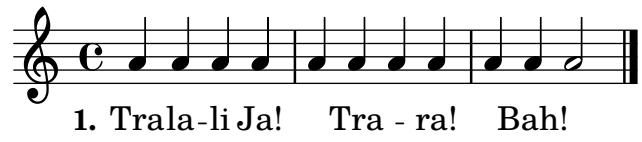
52b-Breaks.xml



61 ... Lyrics
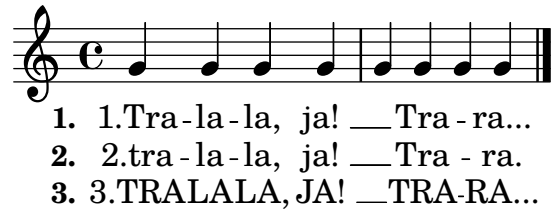
Some notes with simple lyrics: Syllables, notes without a syllable, syllable spanners.

61a-Lyrics.xml



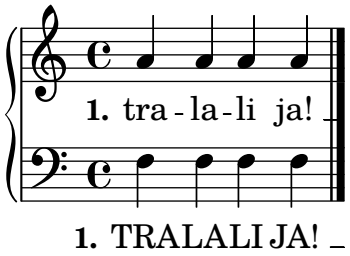
Multiple (simple) lyrics. The order of the exported stanzas is relevant (identified by the number attribute in this test case)

61b-MultipleLyrics.xml
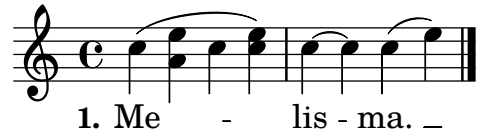


Lyrics assigned to the voices of a piano staff containing two simple staves. Each staff is assigned exactly one lyrics line.

61c-Lyrics-Pianostaff.xml
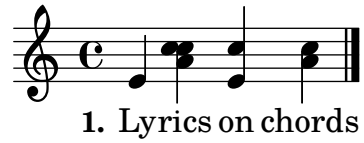


How to treat lyrics and slurred notes. Normally, a slurred group of notes is assigned only one lyrics syllable.

61d-Lyrics-Melisma.xml
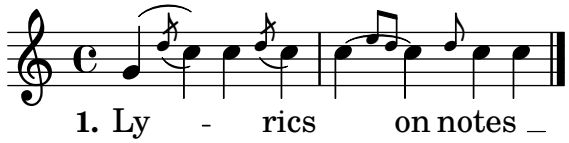


Assigning lyrics to chorded notes.

61e-Lyrics-Chords.xml
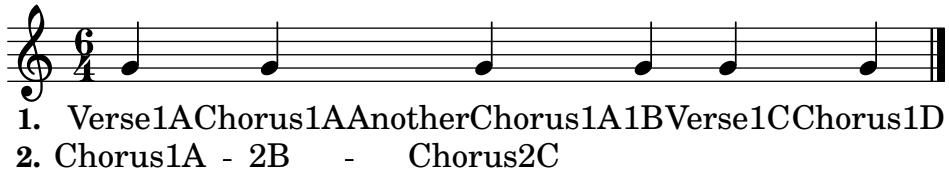


Grace notes shall not mess up the lyrics, and they shall not be assigned a syllable.

61f-Lyrics-GracedNotes.xml
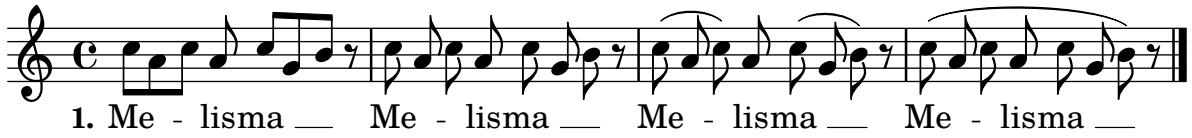


A lyrics syllable can have both a number and a name attribute. The question is: What should be used to put syllables of the same voice together. This example uses different number/name combinations to check how different applications handle this unspecified case (The advice on the MusicXML mailing list was " there is no correct way, each application can do what it thinks is best").

61g-Lyrics-NameNumber.xml
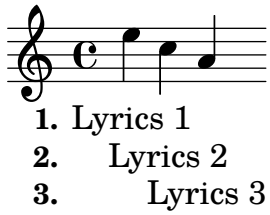


Beaming or slurs can indicate melismata for lyrics. Also make sure that notes without an explicit syllable are treated as if they were part of a melisma.

61h-Lyrics-BeamsMelismata.xml
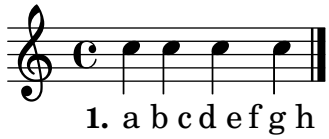


Each note of a chord can have some lyrics attached. In this case, each note of the chord has lyrics of the form " Lyrics [123]" attached, where each lyrics has a different number attribute to distinguish them. These syllables should be imported into three different stanzas and the timing should be correct.

61i-Lyrics-Chords.xml
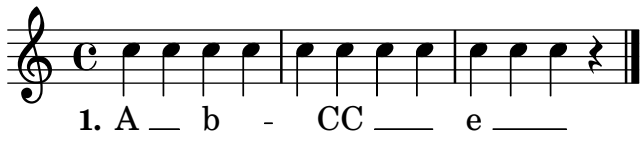


Multiple lyrics syllables assigned to a single note are implemented either using a space in the lyrics or by using the <elision> lyrics element. This testcase checks both of them. First, a note with on syllable is given, then a note with two syllables separated by a spcae and finally a note with two and one with three syllables implemented using <elision> is given.

61j-Lyrics-Elisions.xml



Lyrics spanners: continued syllables and extenders, possibly spanning multiple notes. The intermediate notes do not have any <lyric> element.

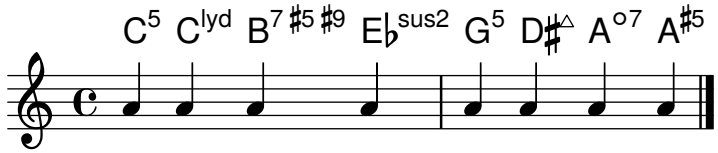
61k-Lyrics-SpannersExtenders.xml



71 ... Guitar notation

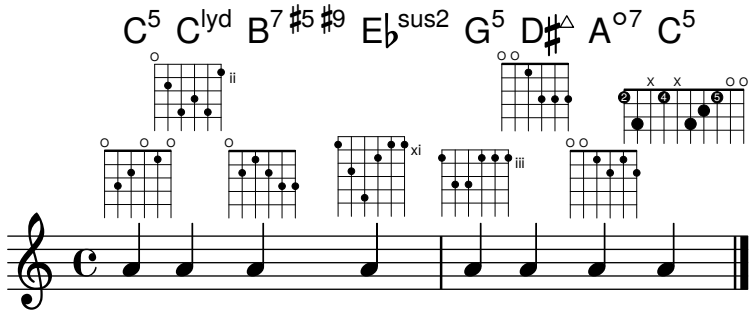
A normal staff with several (complex) chord names displayed.

71a-Chordnames.xml



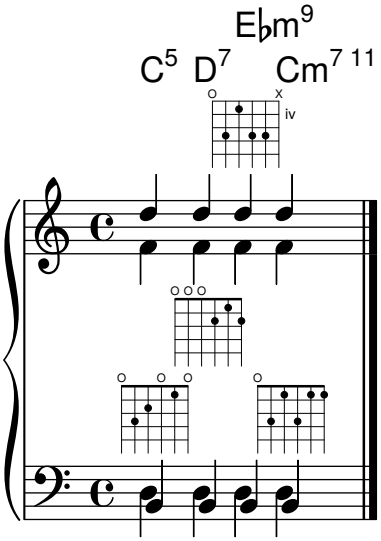
A staff with chord names and some fretboards shown. The fretboards can have an arbitrary number of frets/strings, can start at an arbitrary fret and can even contain fingering information.

71c-ChordsFrets.xml



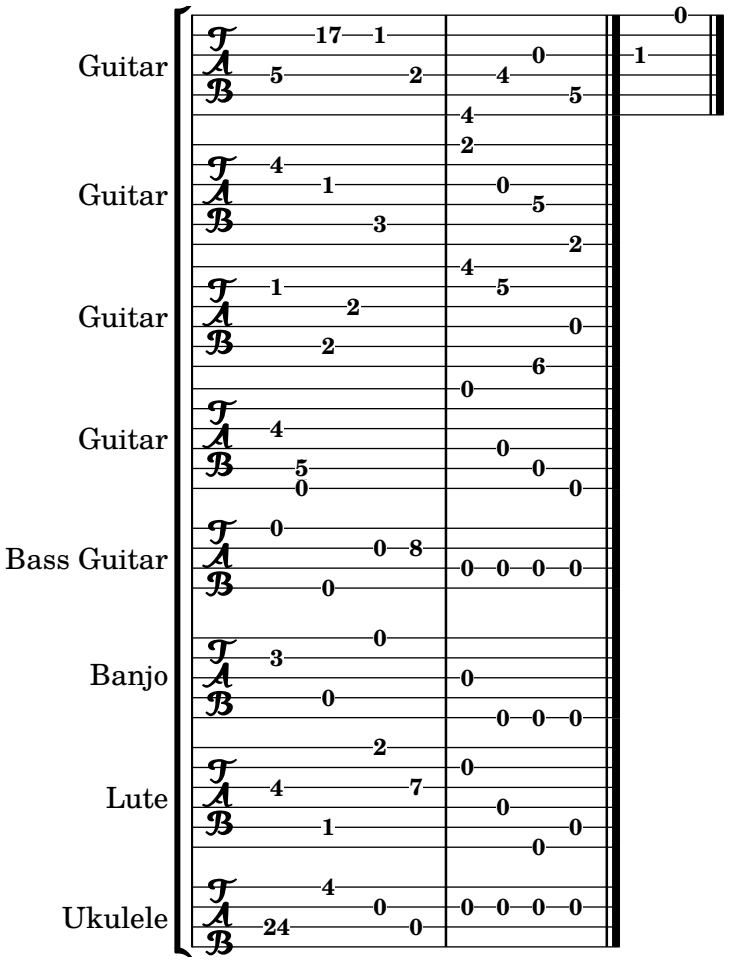
Chords and fretboards assigned to the voices in a multi-voice, multi-staff part. There should be fret diagrams above each of the two staves.

71d-ChordsFrets-Multistaff.xml



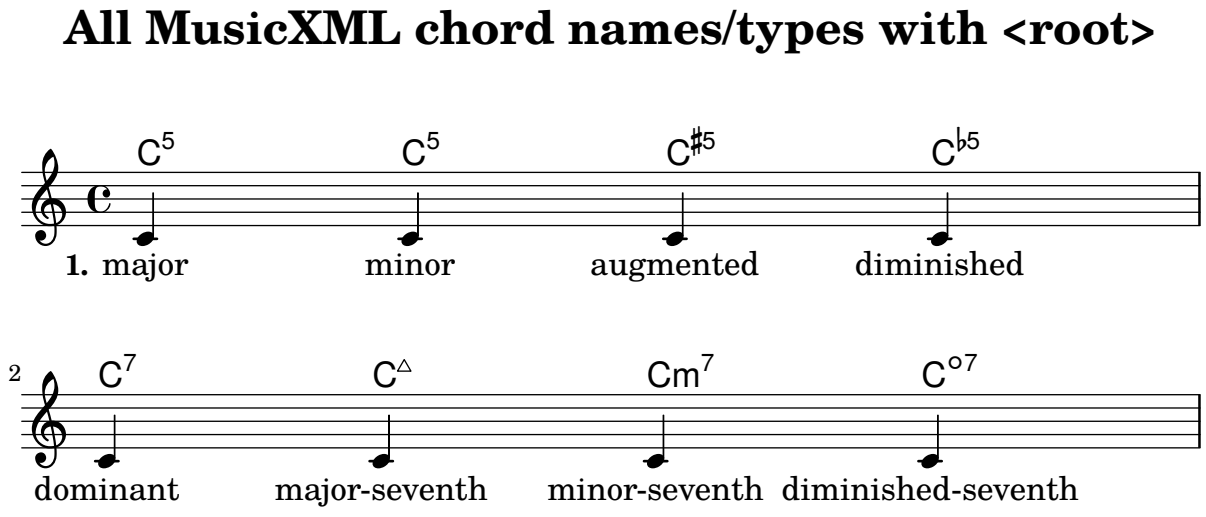
Some tablature staves, with explicit fingering information and different string tunings given in the MusicXML file.
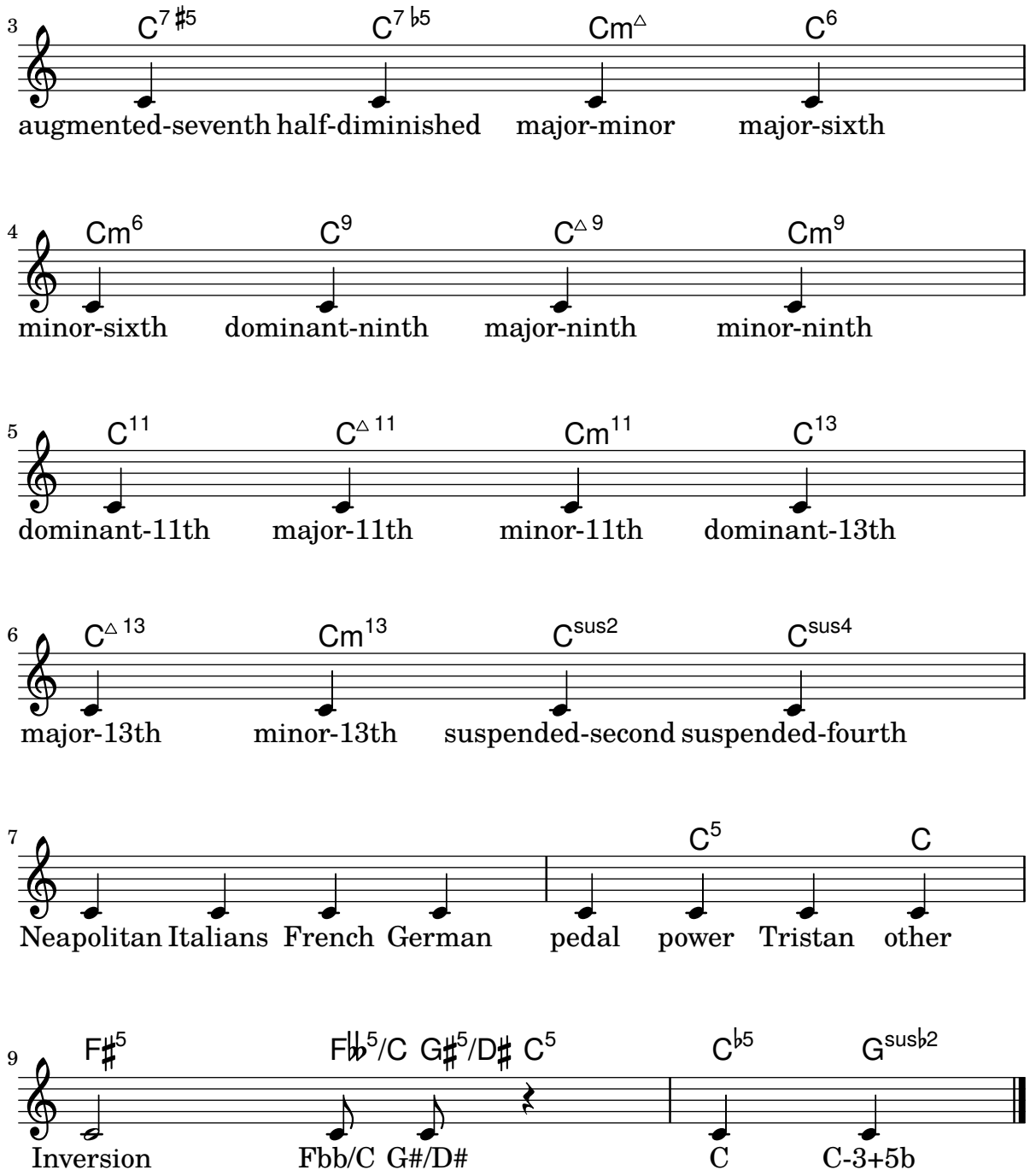
`71e-TabStaves.xml`



All chord types defined in MusicXML. The staff will only contain one c' note (NO chord) for all of them, but the chord names should be properly printed.
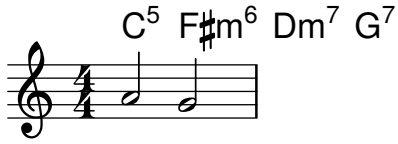
`71f-AllChordTypes.xml`





There can be multiple subsequent harmony elements, indicating a harmony change during a note
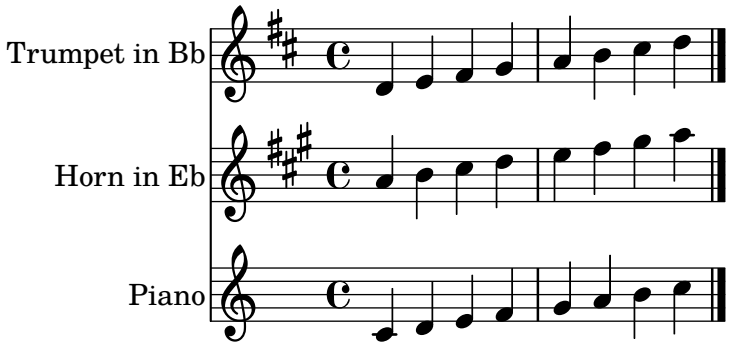
`71g-MultipleChordnames.xml`



72 ... Transposing instruments

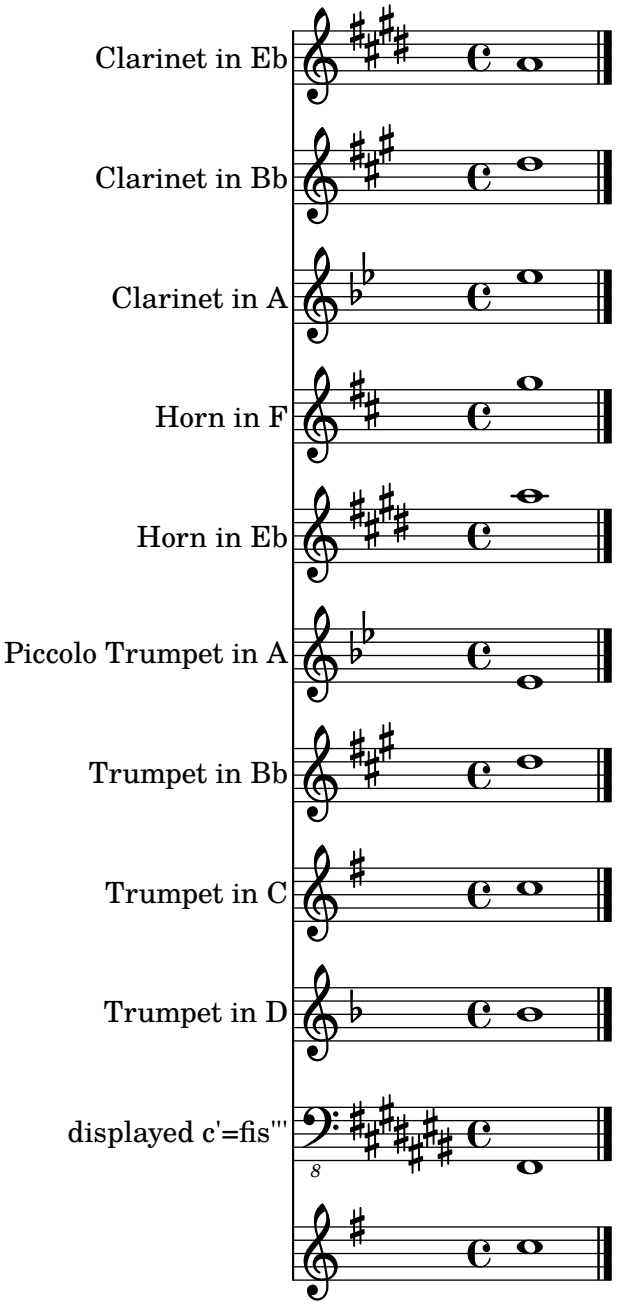
Transposing instruments: Trumpet in Bb, Horn in Eb, Piano; All of them show the C major scale (the trumpet with 2 sharp, the horn with 3 sharp).

72a-TransposingInstruments.xml



Various transposition. Each part plays a c”, just displayed in different display pitches. The second-to-last staff uses a transposition where the displayed c’ is an actual f”’ concert pitch. The final staff is an untransposed instrument.

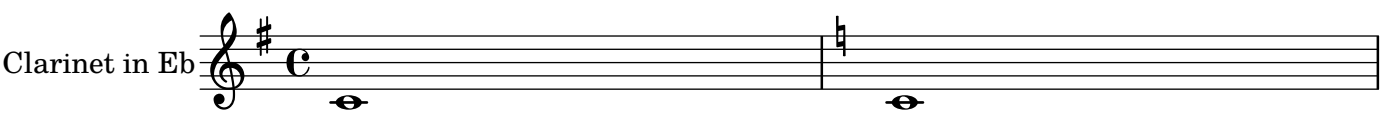
72b-TransposingInstruments-Full.xml

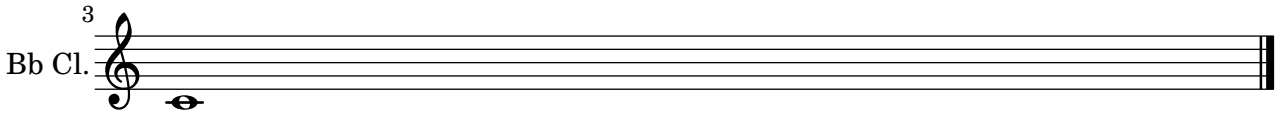


An instrument change from one transposition (Clarinet in Eb) to another transposing instrument (Clarinet in Bb). The displayed instrument name should also be updated.

The whole piece is in Bb major (sounding), so first the key signature should be one flat, after the change it should have no accidentals.

72c-TransposingInstruments-Change.xml
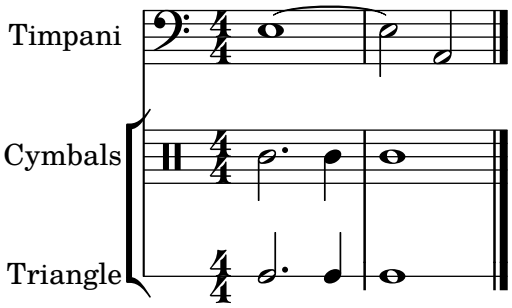




73 ... Percussion

Three types of percussion staves: A five-line staff with bass clef for Timpani, a five-line staff with percussion clef, and a one-line percussion staff with only unpitched notes.

`73a-Percussion.xml`

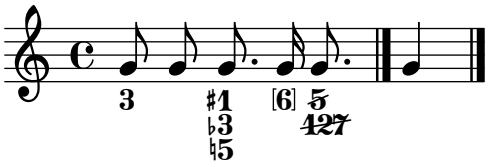


74 ... Figured bass

Some figured bass containing alterated figures, bracketed figures and slashed figures. The last note contains an empty <figured-bass> element, which is invalid MusicXML, to check how well applications cope with malformed files.

Note that this file does not contain any extenders!

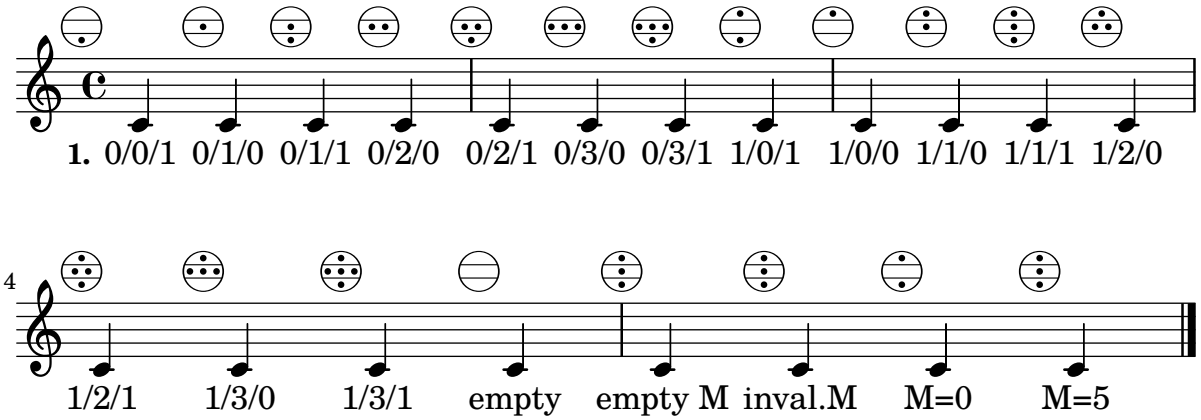
`74a-FiguredBass.xml`



75 ... Other instrumental notation

All possible accordion registrations.

`75a-AccordionRegistrations.xml`
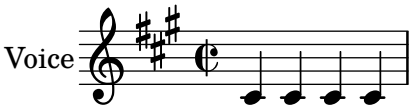


90 ... Compressed MusicXML files

A compressed MusicXML file, containing a simple MusicXML score and the corresponding .pdf output for reference.

`90a-Compressed-MusicXML.mxl`

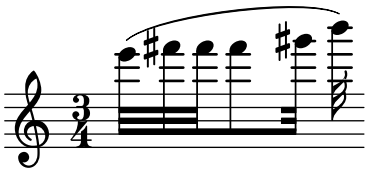
Compressed MusicXML file



99 ... Compatibility with broken MusicXML

Dolet 3 for Sibelius (5.1) did not print out any closing beam tags, only starting and continuing beam tags. For such files, one either needs to ignore all beaming information or close all beams

`99a-Sibelius5-IgnoreBeaming.xml`



If we properly ignore all beaming information from the Dolet 3 for Sibelius export file, make sure that the lyrics syllables are still assigned to the correct notes.
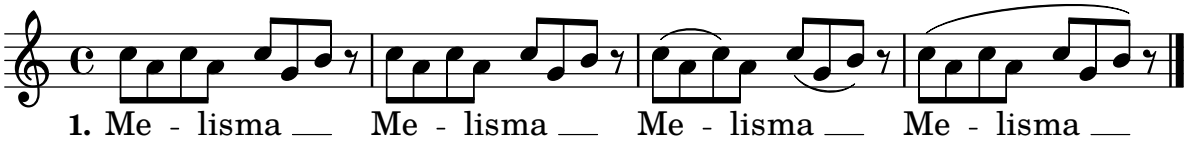
`99b-Lyrics-BeamsMelismata-IgnoreBeams.xml`

