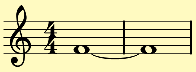
An Extensive MusicXML 2.0 Test Suite

Reinhold Kainhofer, reinhold@kainhofer.com
Vienna University of Technology *and* LilyPond Music notation software project

The MusicXML format

- Industry-standard interchange format for music notation (XML-based)
- Syntax definition via DTD (Document Type Definition) and XSD (XML Schema Document) by Recordare (`http://www.musicxml.org`)

```
<?xml version="1.0"
       encoding="ISO-8859-1"
       standalone="no"?>
<!DOCTYPE score-partwise PUBLIC
       "-//Recordare//DTD MusicXML
       0.6b Partwise//EN" [...] >
<score-partwise>
 <identification>
  <miscellaneous>
   <miscellaneous-field
          name="description">Two
          tied
          notes</miscellaneous-field>
  </miscellaneous>
 </identification>
 <part-list><score-part
        id="P1"/></part-list>
 <part id="P1">
  <measure number="1">
   <attributes>
    <divisions>1</divisions>
    <key><fifths>0</fifths></key>
    <time>
     <beats>4</beats>
     <beat-type>4</beat-type>
    </time>
    <staves>1</staves>
    <clef number="1">
     <sign>G</sign>
     <line>2</line>
    </clef>
   </attributes>
   <note>
    <pitch>
     <step>F</step>
     <octave>4</octave>
    </pitch>
    <duration>4</duration>
    <tie type="start"/>
    <voice>1</voice>
    <type>whole</type>
    <notations><tied
           type="start"/></notations>
   </note>
  </measure>
  <measure number="2">
   <note>
    <pitch>
     <step>F</step>
     <octave>4</octave>
    </pitch>
    <duration>4</duration>
    <tie type="stop"/>
    <voice>1</voice>
    <type>whole</type>
    <notations><tied
           type="stop"/></notations>
   </note>
  </measure>
 </part>
</score-partwise>
```

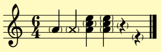


Reasons for a test suite

- No freely available reference implementation, no official sample files, only comments in the DTD
- Sample files needed for applications supporting MusicXML (in our case: MusicXML importer `musicxml2ly` for LilyPond)

Aims of the test suite

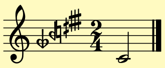
- General MusicXML test suite (>120 test cases)
- Complete coverage not feasible (music notation can be arbitrarily complex)
- Aim is a set of representative test cases to cover as much of the specification as possible: The most common features but also some exotic features
- Each file checks only one aspect of MusicXML, possibly within different contexts. E.g. bracketing note heads and rests:



- Files organized by name: 2-digit category (see right), 1 letter enumerating inside category and a description of the purpose, e.g. `01b-Pitches-Intervals.xml`, `33b-Spanners-Tie.xml`, ...
- Description provided inside the MusicXML file
- Freely available under the MIT license for all applications implementing MusicXML

Example of an exotic test case

```
13d-KeySignatures-Microtones.xml
          ...
          <attributes>
            <key>
              <key-step>4</key-step>
              <key-alter>-1.5</key-alter>
              <key-step>6</key-step>
              <key-alter>-0.5</key-alter>
              <key-step>0</key-step>
              <key-alter>0</key-alter>
              <key-step>1</key-step>
              <key-alter>0.5</key-alter>
              <key-step>3</key-step>
              <key-alter>1.5</key-alter>
            </key>
          </attributes>
          ...
```



Availablility

The MusicXML 2.0 test suite is available at:

`http://www.kainhofer.com/musicxml/`

Sample renderings (using `musicxml2ly` and LilyPond) are provided for all test cases, sample renderings from other applications are planned.

Categories of test cases

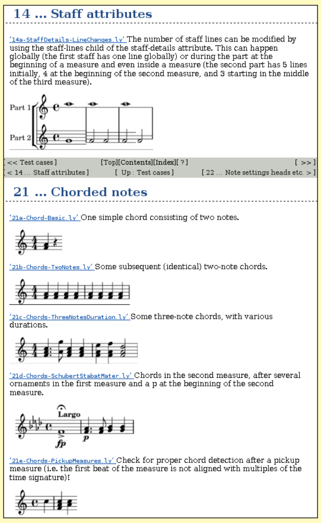
01-09	**... Basics**
01	Pitches
02	Rests
03	Rhythm
10-19	**... Staff attributes**
11	Time signatures
12	Clefs
13	Key signatures
20-29	**... Note-related elements**
21	Chorded notes
22	Note settings, heads, etc.
23	Triplets, Tuplets
24	Grace notes
30-39	**... Dynamics, artic., spanners**
31	Dynamics and other single symbols
32	Notations and Articulations
33	Spanners
40-44	**... Parts**
41	Multiple parts (staves)
42	Multiple voices per staff
43	One part on multiple staves

45-49	**... Measures and repeats**
45	Repeats
46	Barlines, Measures
50-54	**... Page-related issues**
51	Header information
52	Page layout
55-59	**... Exact positioning**
60-69	**... Vocal music**
61	Lyrics
70-75	**... Instrument-specific**
71	Guitar notation
72	Transposing instruments
73	Percussion
74	Figured bass
75	Other instrumental notation
80-89	**... MIDI and sound**
90-99	**... Other aspects**
90	Compressed MusicXML files
99	Compat. with broken MusicXML

Structure of the files, categorized by file name. Not all numbers are assigned to leave room for future extensions of the test suite and the MusicXML specification.

Sample renderings

Sample rendings are possible using `musicxml2ly` and LilyPond on the test files. These cannot be regarded official reference renderings, as `musicxml2ly` does not support every little detail of MusicXML perfectly, and some aspects of the specification are ambiguous in any case.



14 ... Staff attributes

'14a-StaffDetails-LineChanges.ly' 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).

[<< Test cases] [Top][Contents][Index][?] [>>]
[< 14 ... Staff attributes] [Up : Test cases] [22 ... Note settings heads etc. >]

21 ... Chorded notes

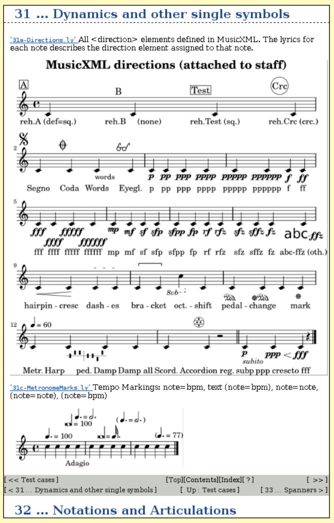
'21a-Chord-Basic.ly' One simple chord consisting of two notes.

'21b-Chords-TwoNotes.ly' Some subsequent (identical) two-note chords.

'21c-Chords-ThreeNotesDuration.ly' Some three-note chords, with various durations.

'21d-Chords-SchubertStabatMater.ly' Chords in the second measure, after several ornaments in the first measure and a p at the beginning of the second measure.

'21e-Chords-PickupMeasures.ly' 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)!



31 ... Dynamics and other single symbols

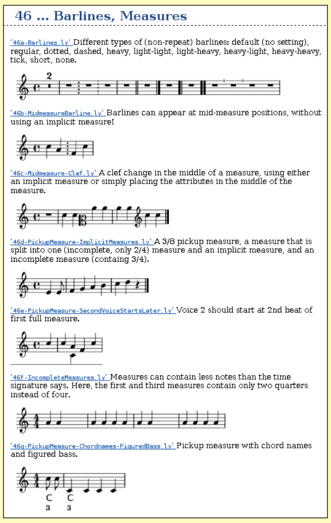
'31a-Directions.ly' All <direction> elements defined in MusicXML. The lyrics for each note describes the direction element assigned to that note.

MusicXML directions (attached to staff)
A
B
Test
Crc
reh.A (def=sq.) reh.B (none) reh.Test (sq.) reh.Crc (crc.)
words
Segno Coda Words Eyegl. p pp ppp pppp ppppp pppppp f ff
fff ffff fffff ffffff mp mf sf sfp sfpp fp rf rfz sfz sffz fz abc-ffz (oth.)
8vb
hairpin - cresc dash - es bra - cket oct. - shift pedal - change mark
♩ = 60
subito
Metr. Harp ped. Damp Damp all Scord. Accordion reg. subp ppp crescto fff

'31c-MetronomeMarks.ly' Tempo Markings: note=bpm, text (note=bpm), note=note, (note=note), (note=bpm)

[<< Test cases] [Top][Contents][Index][?] [>>]
[< 31 ... Dynamics and other single symbols] [Up : Test cases] [33 ... Spanners >]

32 ... Notations and Articulations



46 ... Barlines, Measures

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

'46b-MidmeasureBarline.ly' Barlines can appear at mid-measure positions, without using an implicit measure!

'46c-Midmeasure-Clef.ly' 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.

'46d-PickupMeasure-ImplicitMeasures.ly' 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).

'46e-PickupMeasure-SecondVoiceStartsLater.ly' Voice 2 should start at 2nd beat of first full measure.

'46f-IncompleteMeasures.ly' Measures can contain less notes than the time signature says. Here, the first and third measures contain only two quarters instead of four.

'46g-PickupMeasure-Chordnames-FiguredBass.ly' Pickup measure with chord names and figured bass.

These sample renderings are available on the test suite's web page.

Issues in MusicXML

During the creation of the test suite, we encountered several shortcomings of the MusicXML definition:

- Child-elements of `<note>` (specifying properties of a note) have a fixed, counter-intuitive order!
- MusicXML is a syntax definition, semantic restrictions not covered
- Chords are fragile (e.g. what should happen with a `<forward .../>` or `<backwards.../>` between the notes of a chord?)
- Display of items not always well-defined
- Lyrics identified by stanza number and name... What if they clash? Position of lyrics is most important ⇒ separation of content and display broken
- Assignment of figured bass numbers to notes not clearly defined
- XML element naming often ignores the context ⇒ e.g. `<pitch><step>`, `<root><root-step>`, `<staff-tuning><tuning-step>`
- Metronome markings and non-standard key signatures use bad XML design
- Document-wide headers/footers missing
- Credit elements (text for title, composer, lyricist, arranger, etc.) do not contain their purpose, only the pure text and position on the page

A detailled discussion of the encountered issues was published in:

[1] R. Kainhofer. A MusicXML Test Suite and a Discussion of Issues in MusicXML 2.0. In: *Proceedings of the Linux Audio Conference (LAC) 2010*, Utrecht, May 2010.

Acknowledgements

This work would not have been possible without the help of the LilyPond community, as well as countless helpful clarifications by Michael Good, the creator of the MusicXML specification. This work was also supported by the Christian Doppler Laboratory for Portfolio Risk Management (PRisMa Lab) at the Vienna University of Technology.