LilyPond

The music typesetter

MIDI test suite

The LilyPond development team

Table of Contents

Introduction 1

Regression test cases 2

ambitus-after-midi.ly 2
articulate-accel-midi.ly 2
articulate-repeat-midi.ly 2
articulate-trill-midi.ly 2
crescendo-gap-compatible-target-midi.ly 3
crescendo-return-crescendo-midi.ly 3
crescendo-return-louder-target-midi.ly 3
crescendo-return-softer-target-midi.ly 3
crescendo-return-unspecified-target-midi.ly 3
crescendo-single-compatible-target-midi.ly 3
crescendo-single-unspecified-target-midi.ly 3
decrescendo-multiple-compatible-target-midi.ly 3
decrescendo-single-contrary-target-midi.ly 4
dynamic-initial-midi.ly 4
dynamic-voices-sequential-midi.ly 4
dynamic-voices-simultaneous-midi.ly 4
key-initial-midi.ly 4
key-option-all-staves-midi.ly 4
key-option-midi.ly 4
lyric-combine-switch-voice-midi.ly 5
lyric-repeat-count-midi.ly 5
lyric-tie-midi.ly 5
lyrics-addlyrics-midi.ly 5
midi-overlapping-notes-midi.ly 5
midi-polymeter-midi.ly 5
music-after-fine-warning-midi.ly 6
pan-position-midi.ly 6
partcombine-midi.ly 6
performance-name-utf8-midi.ly 6
quantize-duration-2-midi.ly 6
quantize-duration-midi.ly 6
quantize-start-midi.ly 7
rehearsal-mark-midi.ly 7
repeat-segno-midi.ly 7
rest-dynamic-midi.ly 7
rest-midi.ly 7
section-label-midi.ly 7
sequence-name-midi.ly 7
sequence-name-scoping-midi.ly 8
spacer-no-notes-midi.ly 8
staff-map-instrument-midi.ly 8
staff-map-voice-midi.ly 8
tempo-midi.ly 9
varcoda-midi.ly 9
voice-2-midi.ly 9

voice-4-midi.ly 9
voice-5-midi.ly 9

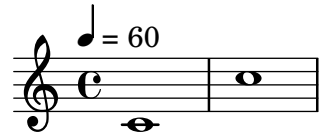
Introduction

This document presents regression tests for LilyPond development that are related to MIDI output and thus do not contain visually significant output in most cases; it mainly exists for completeness and to make the test documentation better available.

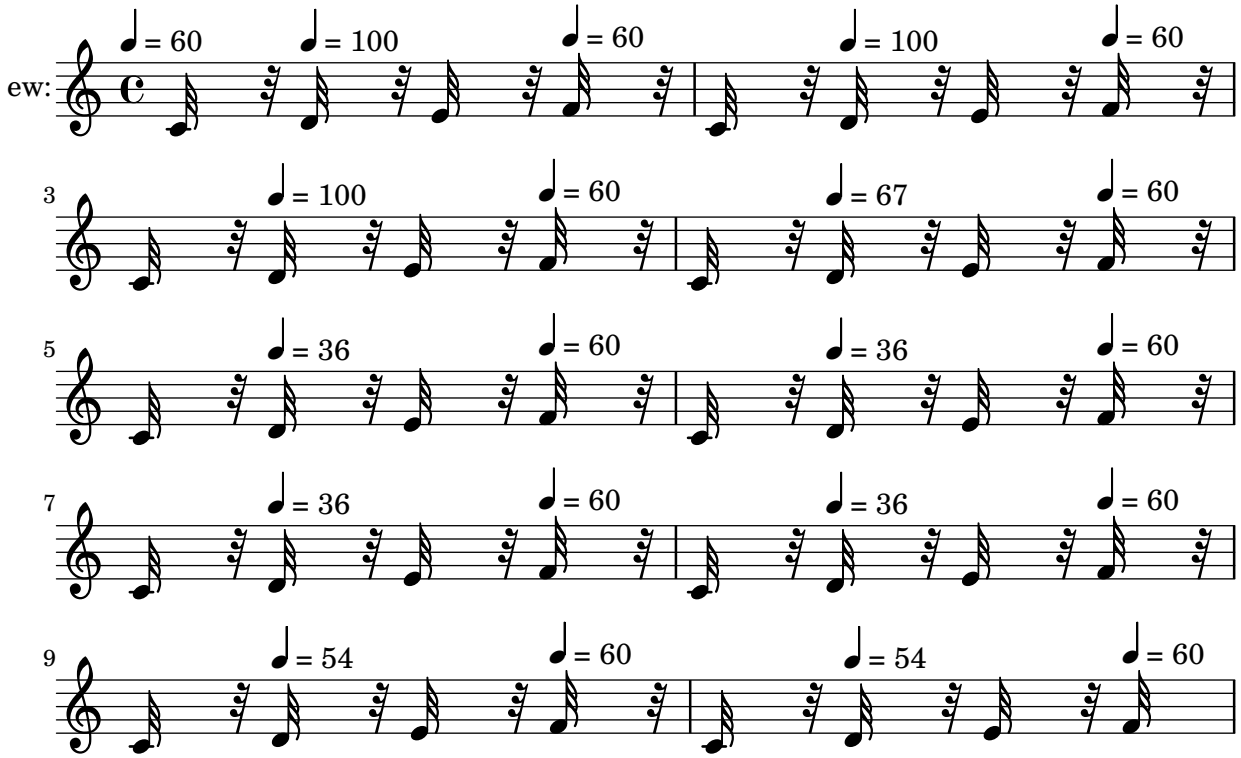
In the web version of this document, you can click on the figure for each example to see the corresponding input file.

Regression test cases

ambitus-after-midi.ly

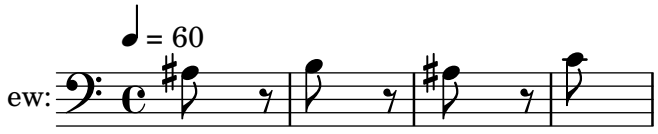


articulate-accel-midi.ly

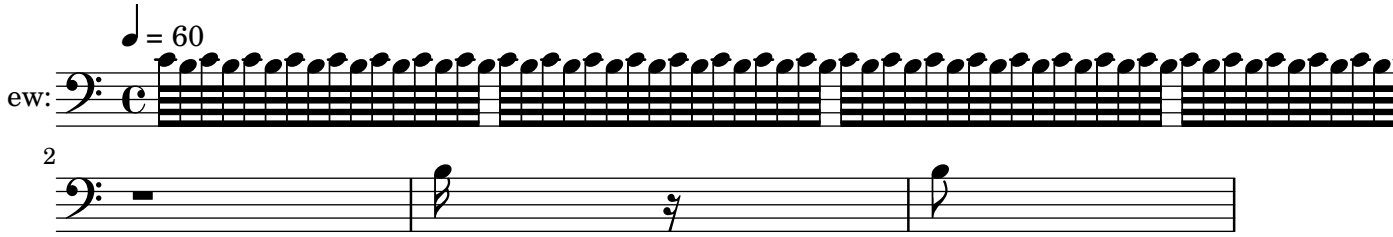


articulate-repeat-midi.ly

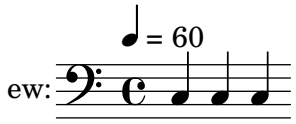
`\articulate` correctly handles repeats with alternatives. They are played in the same order as their visual rendition suggests.



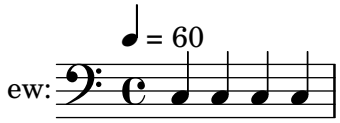
articulate-trill-midi.ly



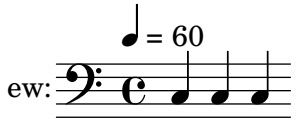
crescendo-gap-compatible-target-midi.ly



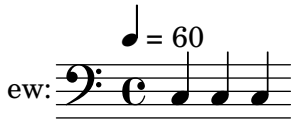
crescendo-return-crescendo-midi.ly



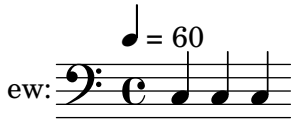
crescendo-return-louder-target-midi.ly



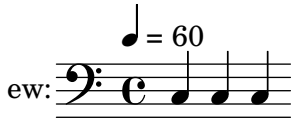
crescendo-return-softer-target-midi.ly



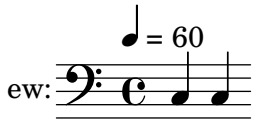
crescendo-return-unspecified-target-midi.ly



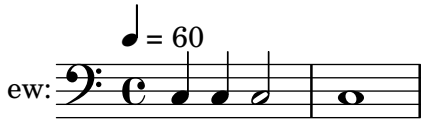
crescendo-single-compatible-target-midi.ly



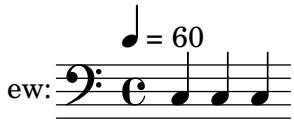
crescendo-single-unspecified-target-midi.ly



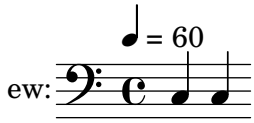
decrescendo-multiple-compatible-target-midi.ly



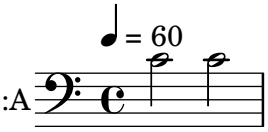
decrescendo-single-contrary-target-midi.ly



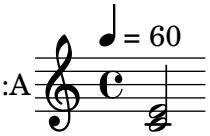
dynamic-initial-midi.ly



dynamic-voices-sequential-midi.ly

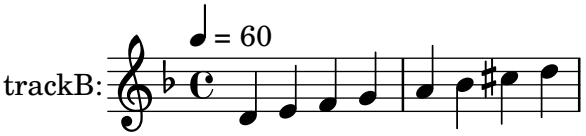


dynamic-voices-simultaneous-midi.ly



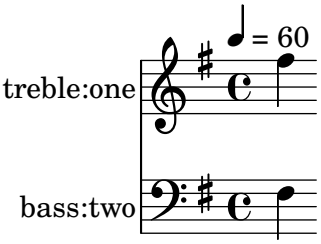
key-initial-midi.ly

keys work in MIDI, this is d-minor



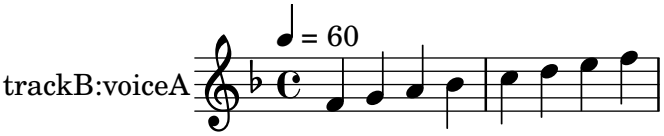
key-option-all-staves-midi.ly

Midi2ly –key works on all staves, this is G major (–key=1)



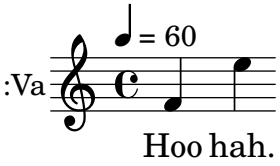
key-option-midi.ly

`midi2ly`'s option `--key` works, this is F major.

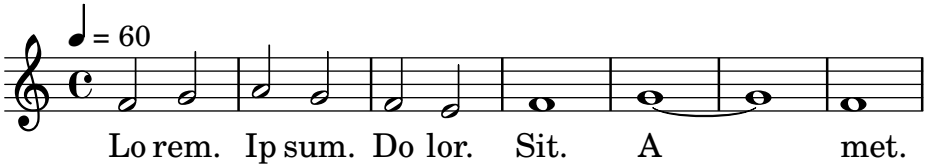


lyric-combine-switch-voice-midi.ly

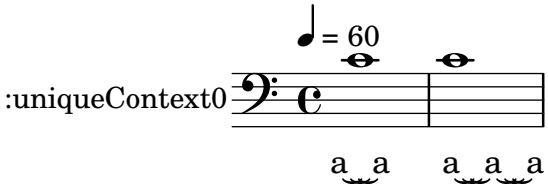
It is possible to switch the voice that `\lyricsto` follows.



lyric-repeat-count-midi.ly

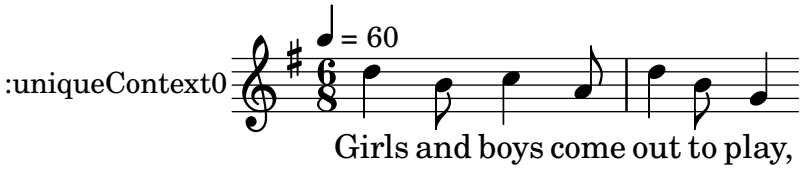


lyric-tie-midi.ly

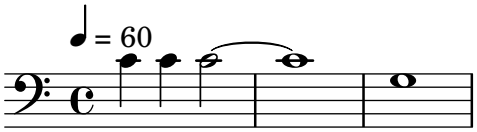


lyrics-addlyrics-midi.ly

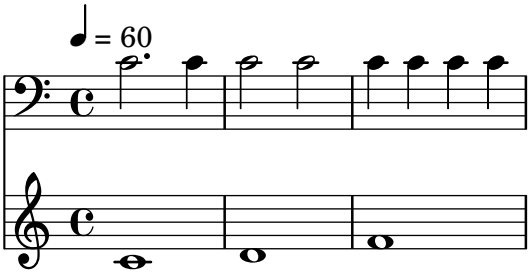
Lyrics are preserved



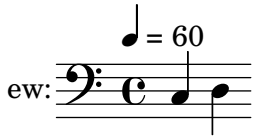
midi-overlapping-notes-midi.ly



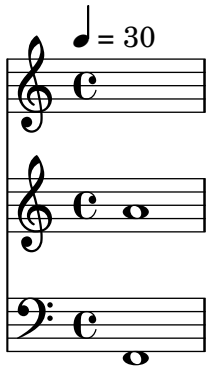
midi-polymeter-midi.ly



music-after-fine-warning-midi.ly

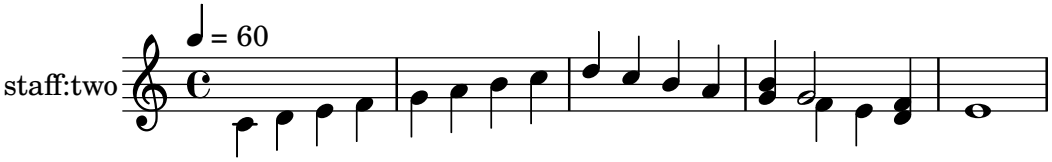


pan-position-midi.ly

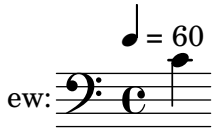


partcombine-midi.ly

PartCombined music is preserved

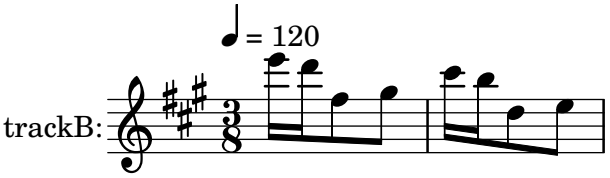


performance-name-utf8-midi.ly



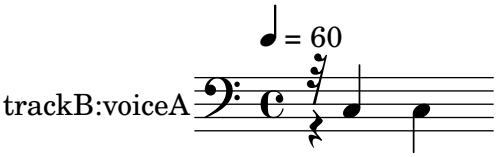
quantize-duration-2-midi.ly

`midi2ly`'s option `--duration-quant` preserves first note length (16).



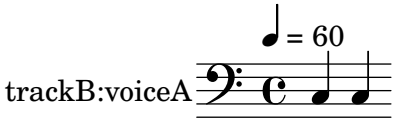
quantize-duration-midi.ly

`midi2ly`'s option `--duration-quant` quantizes durations of notes.

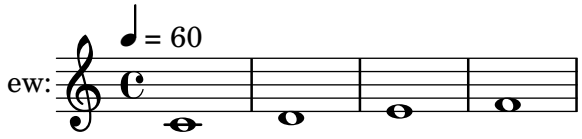


quantize-start-midi.ly

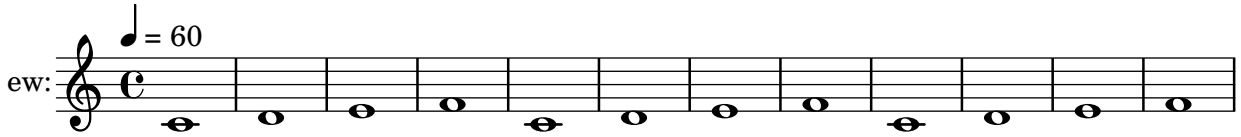
`midi2ly`'s option `--start-quant` quantizes start of notes.



rehearsal-mark-midi.ly

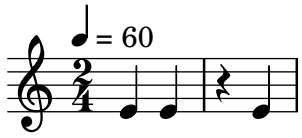


repeat-segno-midi.ly



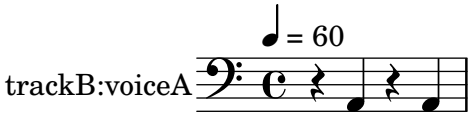
rest-dynamic-midi.ly

LilyPond respects rests, also when there are dynamics

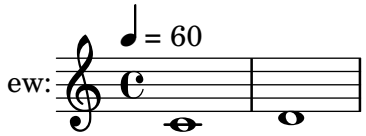


rest-midi.ly

midi2ly identifies rests



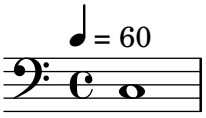
section-label-midi.ly



sequence-name-midi.ly

The MIDI sequence generated from a score should get its name from the title defined in the score's `\header` block (if any). The title used for layout can be overridden for MIDI output by specifying a separate `miditítle` in the `\header` block. If the score does not define a title of its own, and has no enclosing `\bookpart`, `\book`, or top-level scope with a `\header` block that defines a title, either, the MIDI sequence should get the default name.

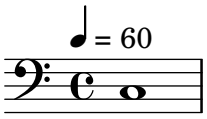
Title shared between layout and MIDI



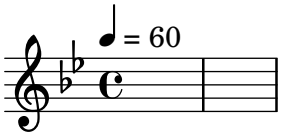
sequence-name-scoping-midi.ly

If a score has a `\header` block which defines a title, this title should override any title defined in a `\header` block of the score's enclosing `\bookpart` or `\book` (or a title defined in a top-level `\header` block) when naming the MIDI sequence generated from the score. Otherwise, if the score has no title defined, the MIDI sequence generated from the score should get named using the title defined in the `\header` block of the nearest enclosing `\bookpart`, `\book`, or top-level scope that contains a title definition.

Book

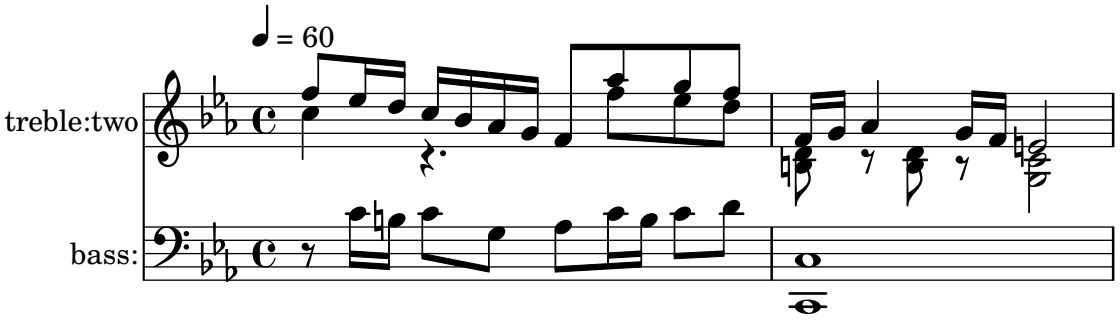


spacer-no-notes-midi.ly



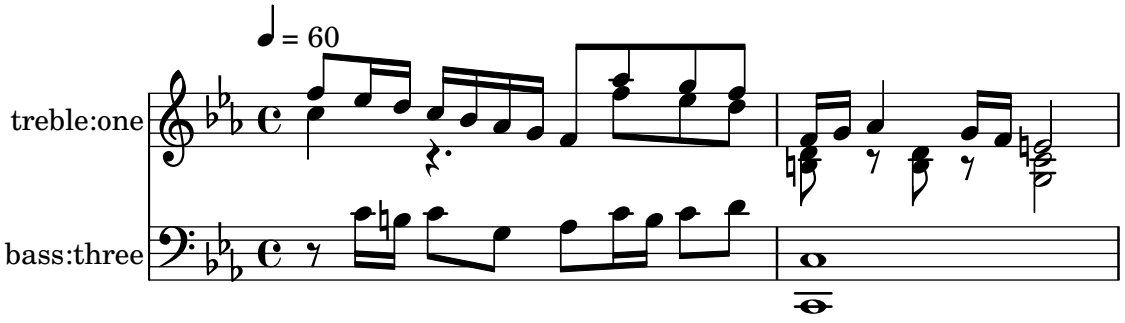
staff-map-instrument-midi.ly

Midi2ly remaps voices correctly to staves in MIDI-files that use instrument<->channel mapping when combined with voice<->track mapping. TODO: pianostaff

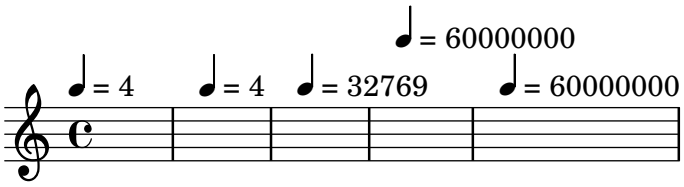


staff-map-voice-midi.ly

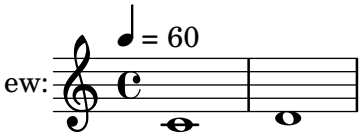
Midi2ly remaps voices correctly to staves in MIDI-files that use voice<->channel mapping when combined with staff<->track mapping. TODO: pianostaff



tempo-midi.ly

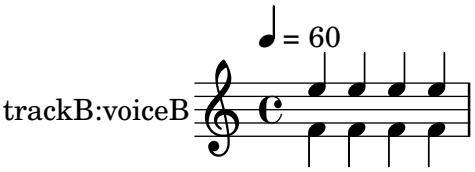


varcoda-midi.ly



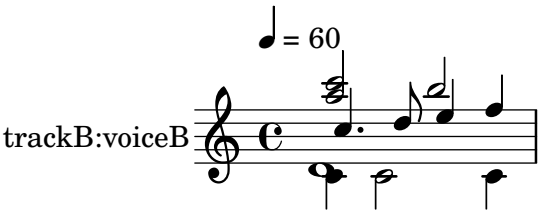
voice-2-midi.ly

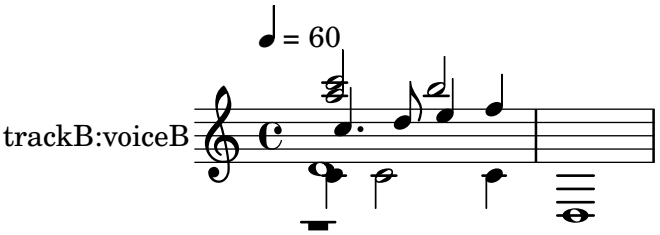
midi2ly maps two voices nicely on one staff as \voiceOne, \voiceTwo



voice-4-midi.ly

midi2ly maps four voices nicely on one staff as \voiceOne, \voiceTwo, \voiceThree, \voiceFour



voice-5-midi.ly

midi2ly still produces output for a staff with five voices. However, in such cases, most probably the the correct \voiceOne, \voiceX... mapping is lost.