

The Harp Gliss Playback Plugin

Bob Zawalich June 26, 2016

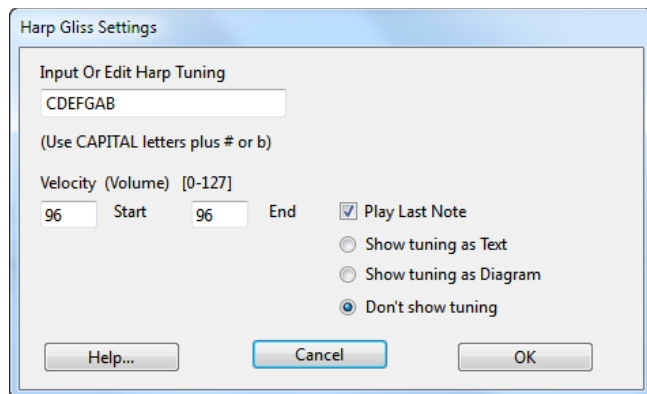
Updated November 18, 2017 to discuss Magnetic Glisses

The *Harp Gliss Playback* plugin is a rewrite and enhancement of the downloadable *Harp Gliss* plugin, which was written by the late Ian Cugley. I originally wanted to make some minor bug fixes and improvements to *Harp Gliss*, but I was unable to get permission to do so from Ian's estate, so instead I wrote a new plugin from scratch that can produce the same output as *Harp Gliss*, and has some additional features.

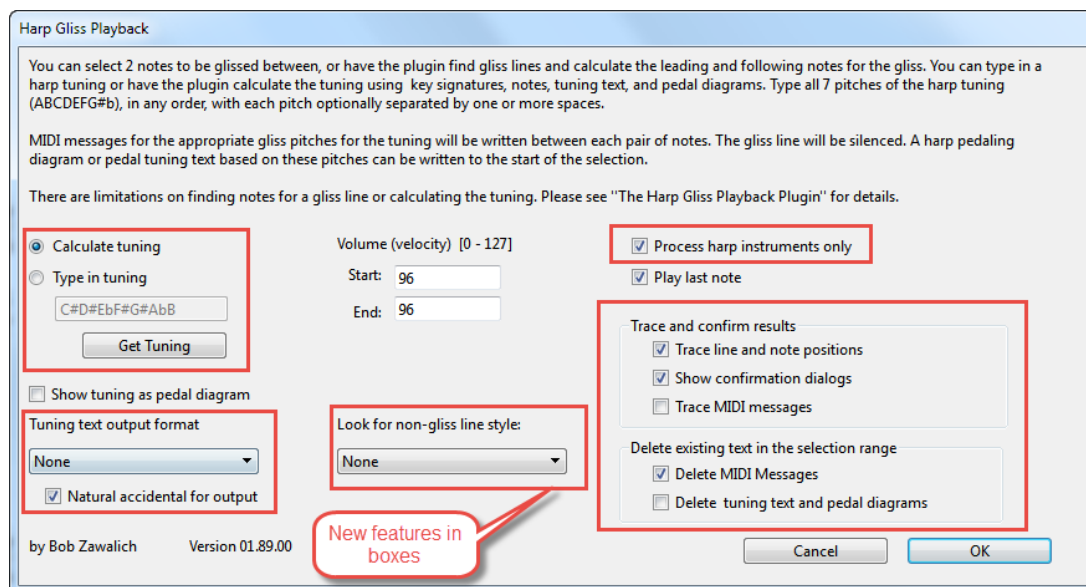
These plugins exist because the only glissando playback for a harp in Sibelius is "white notes", which are the notes for a C major / A minor scale, regardless of the key or any modified tuning of the harp. Any harp glisses that use notes not contained in C major will play back the wrong pitches. Also, gliss lines that cross staves do not play back at all in Sibelius.

The plugins work around this by generating MIDI On and MIDI off messages for the notes that should be played. This could be accomplished by creating notes in a different voice or staff, but such notes have a significant impact on layout that is avoided by using MIDI messages.

Here are the main dialogs for the *Harp Gliss* and *Harp Gliss Playback* plugins:



The **Harp Gliss Settings** dialog box is used to configure the Harp Gliss plugin. It includes a text field for "Input Or Edit Harp Tuning" with the value "CDEFGAB". Below this is a note: "(Use CAPITAL letters plus # or b)". There are two spinners for "Velocity (Volume) [0-127]" with "Start" and "End" values both set to 96. A checkbox "Play Last Note" is checked. Three radio buttons are present: "Show tuning as Text", "Show tuning as Diagram", and "Don't show tuning" (which is selected). At the bottom are buttons for "Help...", "Cancel", and "OK".



The **Harp Gliss Playback** dialog box provides instructions and settings for the playback plugin. It contains a large text area with instructions on how to use the plugin. Below the instructions are several settings sections. On the left, there are radio buttons for "Calculate tuning" (selected) and "Type in tuning", with a text field for the latter containing "C#D#EbF#G#AbB" and a "Get Tuning" button. Below these are checkboxes for "Show tuning as pedal diagram" and "Tuning text output format" (set to "None" with a dropdown arrow). A checkbox "Natural accidental for output" is checked. In the center, there are spinners for "Volume (velocity) [0 - 127]" with "Start" and "End" values both set to 96. A dropdown menu "Look for non-gliss line style:" is set to "None". On the right, there are checkboxes for "Process harp instruments only" and "Play last note". Below these is a section "Trace and confirm results" with checkboxes for "Trace line and note positions", "Show confirmation dialogs", and "Trace MIDI messages". At the bottom right, there is a section "Delete existing text in the selection range" with checkboxes for "Delete MIDI Messages" and "Delete tuning text and pedal diagrams". At the bottom left, it says "by Bob Zawalich Version 01.89.00". A red callout bubble points to the "Look for non-gliss line style:" dropdown with the text "New features in boxes". At the bottom are "Cancel" and "OK" buttons.

Both plugins let you select exactly 2 notes and type in the 7-pitch tuning for a harp. When run this way, both plugins will generate MIDI Note On and Note Off messages that are equivalent to a harp glissando using the notes of the tuning. After running the plugin you will see a “cloud” of messages if you are showing hidden objects. Here is an example, including a tracing of the generated MIDI messages.

The screenshot shows the **Harp Gliss Playback** plugin window and its **Plug-in Trace** window. Red callouts highlight specific features:

- 2 selected notes:** Points to two notes on a musical staff.
- "Cloud" of MIDI messages:** Points to a dense cluster of MIDI messages in the trace window.
- trace of selected notes and tuning:** Points to the first few lines of the MIDI trace, showing note on/off events and the tuning.
- Tuning for F major:** Points to the text input field containing 'CDEFGABb'.
- Trace of MIDI cloud:** Points to the full list of MIDI messages in the trace window.

The **Harp Gliss Playback** window includes the following options:

- Calculate tuning** (radio button)
- Type in tuning** (radio button, selected)
- Tuning text:** CDEFGABb
- Get Tuning** (button)
- Show tuning as pedal diagram** (checkbox)
- Tuning text output format:** None
- Volume (velocity) [0 - 127]:** Start: 96, End: 96
- Play last note** (checkbox, checked)
- Process harp instruments only** (checkbox, checked)
- Trace MIDI messages** (checkbox, unchecked)
- Delete existing text in the selection range:**
 - ☐ Delete MIDI Messages
 - ☐ Delete tuning text and pedal diagrams

The **Plug-in Trace** window shows the following MIDI messages:

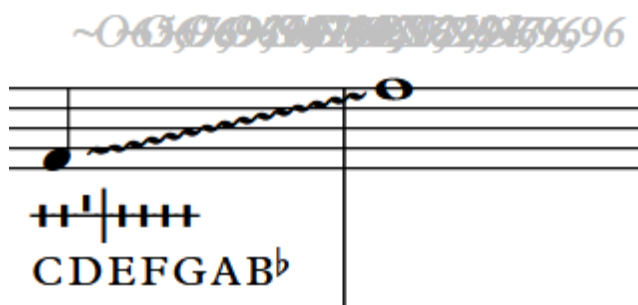
```

First note: F4 1-Harp, 1, 4 (768), 1
Following note: F5 1-Harp, 2, 1 (0), 1
Tuning: CDEFGABb

MIDI text, (staff, bar, position as beat, (position in units), voice)
~065,96 ~N67,96 (1-Harp, 1, 4.1406 (804), 0) (065 = F4 N67 = G4)
~067,96 ~N69,96 (1-Harp, 1, 4.2852 (841), 0) (067 = G4 N69 = A4)
~069,96 ~N70,96 (1-Harp, 1, 4.4258 (877), 0) (069 = A4 N70 = A#/Bb4)
~070,96 ~N72,96 (1-Harp, 1, 4.5703 (914), 0) (070 = A#/Bb4 N72 = C5)
~072,96 ~N74,96 (1-Harp, 1, 4.7109 (950), 0) (072 = C5 N74 = D5)
~074,96 ~N76,96 (1-Harp, 1, 4.8555 (987), 0) (074 = D5 N76 = E5)
~076,1 (1-Harp, 1, 4.9883 (1021), 0) (076 = E5)
~N77,96 (1-Harp, 2, 1 (0), 0) (N77 = F5)
~077,1 (1-Harp, 2, 5 (1024), 0) (077 = F5)
  
```

When played back, the MIDI messages generate the desired pitches in the desired order and timing.

Both plugins can write a harp pedal diagram for the tuning. *Harp Gliss* can write the tuning as text in the order CDEFGAB.



Harp Gliss Playback can also write the text tuning in several other formats. It can optionally use natural signs for natural notes.

Volume (velocity)

This was an unchanged feature of Ian’s original plugin. You can have the volume increase or decrease in the gliss by choosing appropriate start and end values. By default velocity stays at 96, which is roughly equivalent to *f*. Use 84 for *mf*, or 70 for *mp*.

Play Last Note sets the volume (velocity) of the last note to match the final glissando volume. If you don't want this, untick this box.

Here are some additional features added to *Harp Gliss Playback*, in addition to processing 2 selected notes.

1. Processing a selection of more than 2 notes (Large Selection Mode)

The main new feature of *Harp Gliss Playback* is its ability to process multiple glissandos at once. If you make a passage (box) selection, the plugin will look for each Glissando line in the selection, either straight or wavy.

The plugin will process each Glissando line by looking for a leading note in the same voice and staff as the line that starts immediately before the line. If it finds that note it will look for a following note, which must be the next note in the same voice as the first note. There can be no rests between the notes. The following note can be in the same staff of, if the instrument is a grand staff instrument, the following note can be in the other staff of the instrument.

The plugin will only process instruments that are 1 or 2 staves.

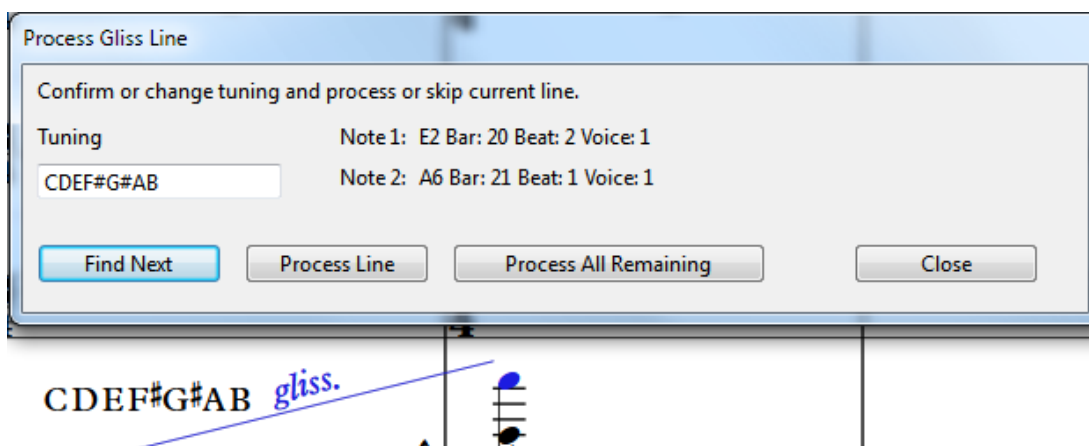
If the plugin finds both notes, it will process the notes and gliss line as if you had just selected the notes. It will silence the gliss line so it will not affect playback. The plugin will trace the locations of each gliss line, its notes, and the tuning used.

2. Calculating the harp tuning pitches

There are 2 ways to define the tuning to be used: Type in Tuning and Calculate Tuning. If Calculate Tuning is used, tuning is recalculated for each gliss, but the tuning edit box is not updated. The tuning used will be written to the plugin Trace window.

It is not feasible to find all the possible ways that harp tunings are indicated in a score, so the plugin will bring up a dialog showing each set of selected notes and the tuning, and it will bring the gliss line into view. You can confirm that the tuning is correct with Process Line, change the tuning, or skip this line using Find Next. If you are comfortable that the tunings will be correct, you can use Process All Remaining to avoid checking each one. If there is not enough context, choose Close to look around then make a new selection and restart the plugin.

This process is similar to what many applications do for Find and Replace.



Calculate Tuning

The plugin can calculate the tuning that would be in effect for the notes before and after each gliss line. It can calculate the tuning even if you only select 2 notes.

It calculates the tuning by looking from the start of the score in the current instrument staves up to the leading note. It will record the original key signature and any following key signature changes, any harp pedal diagrams, 1 to 6-pitch pedal changes (which is text that includes only ABCDEFG, spaces, and approved accidental characters), and 7-pitch tuning text strings, and all the notes in the staves, in chronological order. It will include the leading and following notes in the tuning.

For simple music, the tuning should be correct nearly all the time, even when keys change. If you have abrupt tuning changes without enough notes before the gliss to identify the tuning, you may need to add a (possibly hidden) pedal diagram, specify a tuning by typing it in, or process those glisses separately by selecting the leading and following notes only.

Typing in Tuning

Choose the Type in Tuning option and the edit box will be enabled. Type 7 pitches using CDEFGAB (capital letters only) and b for flat and # for sharp, where the accidentals follow the pitch name. Spaces between pitches are optional, and the pitches can be in any order you wish. There can only be 1 use of each pitch letter name in a tuning, and all 7 pitches must be present.

The plugin will calculate a suggested tuning the first time it is run in a session, and put it in the tuning edit box even if you do not use the Type in Tuning option. It will not update the tuning shown in the edit box for the rest of the session unless you use the Get Tuning button to see the current calculated tuning.

3. Limiting processing to Harp instruments

There is an option, on by default, to only process instruments defined as Harp or Lever Harp, which is useful if you have glisses in other instruments that you do not want to process. Turn this off if you have a different instrument you want to treat as a harp. You can select only a single staff in a harp instrument if you want, even if *Process harp instruments only* is true.

In processing instruments, the plugin will only work with instruments that contain 1 or 2 staves. If you have additional staves, or ossia staves on your instrument, that instrument will be ignored. You can still process glisses in such an instrument by selecting each pair of notes you want to gliss between.

4. Deleting existing messages

If you want to run the plugin again on a passage that contains MIDI messages, you can use the option (on by default) to delete existing MIDI messages. This will clear any ~O or ~N messages in the selection. If you do not delete such messages you can end up with 2 or more sets of them.

You can delete the messages manually if you prefer. The downloadable plugin Filter Other (category Filter and Find) has an option to filter MIDI messages, so you can make a selection, run Filter Other, and then delete the filtered messages.

5. Processing consecutive glissandos

If you are in Large Selection Mode and Delete MIDI messages is on, the plugin will delete the existing messages for the entire selection before processing the gliss lines.

Some of the generated MIDI messages will extend past the following note. If you want to process each pair of notes in consecutive glisses separately, turn off Delete MIDI messages, or the second pair of notes processed will delete some of the ending MIDI messages from the previous gliss and adversely affect playback.

It works much better to use Large Selection Mode and select all the lines in such a gliss and process them all at once, which avoids this problem.

6. Plugin output

The plugin will optionally trace the locations of the gliss line and leading and following note and the tuning used for each gliss line processed.

It will also select each gliss line it finds and the leading and following notes for each gliss line. This can be a quick visual check on whether the operation succeeded.

You can use the *Selected Objects* option of the downloadable *Go To* plugin to browse among the processed lines.

To reduce the amount of output, uncheck *Trace line and note positions*, *Trace MIDI messages*, and *Show confirm dialogs*. Please check the output carefully if you do this.

7. Tracing MIDI messages

The plugin has an option to write out the MIDI message text to the trace window so you can see what pitches are generated. You can instead use the downloadable plugin *Trace MIDI Messages* (category Text) to trace all the messages at once, at any time.

Related tools

You may find these other plugins convenient:

1. *Line Between Notes* for drawing gliss lines between notes (this can be incredibly useful).
2. *Filter Other* for filtering MIDI messages for deletion.
3. *Trace MIDI Messages* to trace the cloud of MIDI messages after running *Harp Gliss Playback*.
4. *Go To* (selected objects) for viewing the processed gliss lines and seeing which operations were successful.
5. *Trace Harp Tuning* will write the calculated tuning at the start of the selection to the plugin Trace window, so you can copy it into other plugins that need to know a tuning. It requires that *Harp Gliss Playback* is installed.

Limitations of *Harp Gliss Playback* in large selection mode

In general, you will obtain the best results if your gliss lines are placed squarely between the notes to be glissed. If the start of the line is before the first note you want to gliss, the plugin will likely find the wrong first note. I recommend using the *Line Between Notes* plugin to place gliss lines.

For best results calculating tunings, be sure that the tuning is clear at the start of a gliss by providing a key signature, harp diagram, 1 to 7-pitch tuning text/pedal change indicator, or enough notes to define the tuning. Other indicators, such as “F major scale” will not be found by the plugin, nor will tuning text with additional characters.

1. Might not find matching notes for gliss lines (following note issues)

Harp Gliss Playback will find matching notes in most cases. It finds the leading note as the closest note to the start of the gliss line so be sure the rhythmic position of the gliss line is later than the note you want to be the leading note. For the following note it will look for the next note in the same voice as the leading note both in the same staff where the line starts and in the other staff of the instrument, if there is one. Sometimes it will not be able to choose between 2 notes at the same location in the 2 staves, and it will skip that gliss line.

2. Can process adjacent notes only

Sibelius itself only plays glisses if the 2 notes are adjacent, and the plugin uses adjacency as one of the conditions defining a following note. If you have a need to gliss between separated notes, select the first and last notes only and run the plugin.

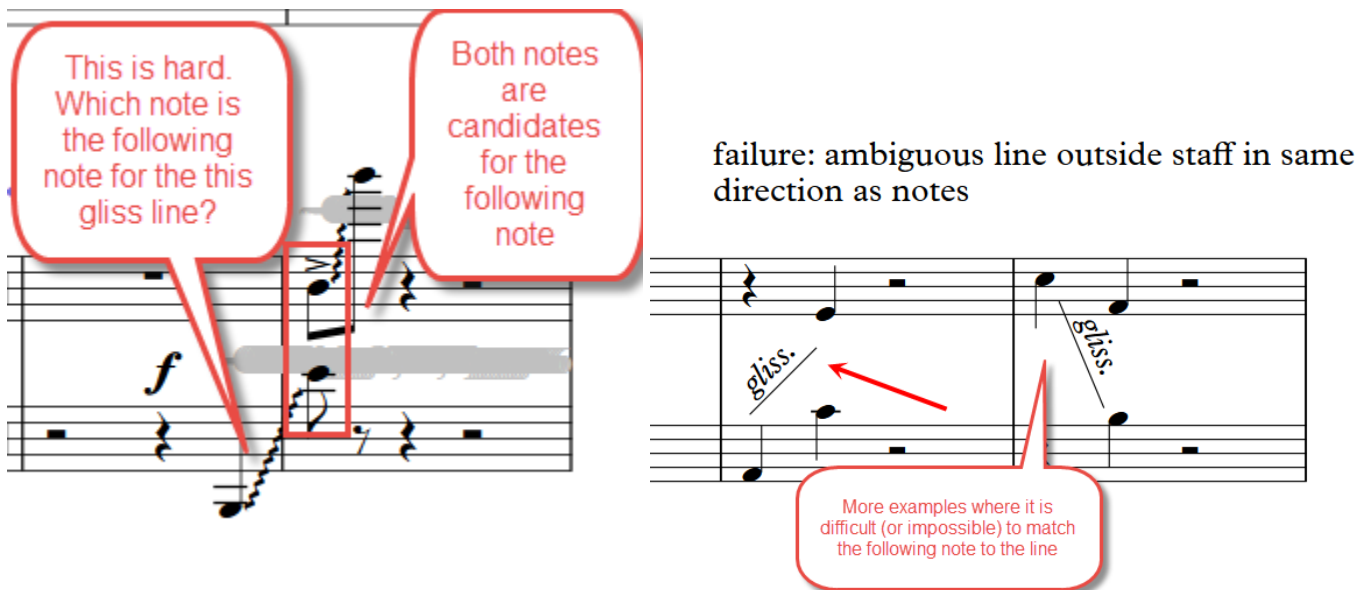
Troubleshooting and avoiding trouble

1. Matching notes to the gliss lines.

If at all possible, add the gliss lines with the *Line Between Notes* plugin, which puts the line ends as close as it can to the noteheads. If you add them manually, note that the plugin only compares the rhythmic position of the line to that of the notes. If you shift the X offset of either the line or notes with the Property Window/Inspector, the line may visually be between the notes, but the plugin cannot detect that. If you can run Reset Position on the lines, especially for horizontal position, you will be more likely to avoid the plugin matching an incorrect note.

If the gliss line crosses staves, the plugin can only guess where it ends, because its positions are all relative to the staff the line begins in, and the plugin has no way to know how much space is between the staves. If there is a valid following note on only one staff, there is no problem, but there will be situations where there is a following note on both staves and the plugin will not be able to tell which note actually matches the line. (*You can see that the line crosses staves or not, but the plugin cannot*).

It will use a number of heuristics to try to make a good guess, but it could be wrong, so check the results carefully if there are multiple notes at the following note position.



2. Calculating the tuning.

If your tuning always follows the key signature, this is easy. If not, there has to be enough information before the gliss to calculate the tuning. The plugin will look at all the notes in the instrument before the leading note, and will also take pitches from harp pedal diagrams, and tuning text and pedal changes.

If the plugin chooses the wrong tuning, add a pedal diagram (hidden is fine) before the line, or type in the tuning explicitly and select just the lines that use that tuning.

3. Deleting existing messages

If you are in Large Selection Mode, the plugin will delete all the MIDI messages in the selection first, and with then add new messages. This works fine. If you are selecting pairs of notes and running the plugin, it will delete all messages between the 2 notes. Some messages are added past the following note, and if you have consecutive glisses, the second time you run it the plugin will delete the overhanging notes from the first note pair.

Either use Large Selection Mode for consecutive glisses or manually delete the MIDI messages before you process the note pairs, and turn off the Delete MIDI messages option.

4. 8va and 8vb lines

The plugin will figure out octave lines that affect the pitch of a glissando, and it will process such glisses correctly so the extra octave is played. (The *Harp Gliss* plugin does not deal with these lines). When the plugin displays the note names that are involved, it shows the written name (as Sibelius does in the status bar), which does not include the 8va line.

The image shows a music score with a glissando and a corresponding plugin trace window. The score has a key signature of one sharp (F#) and a common time signature (C). The glissando is marked with a blue line and the word "gliss." in blue. A red box highlights the "8va" line above the glissando, with a callout stating "8va line recognized and gliss is extended". The plugin trace window shows the following information:

Glissando (straight): duration: 717 duration in beats: 2.8008 bar: 3 at beat: 1.1992 staff: 2 -
First note: G#3 Staff: 2-Lever Harp Bar: 3 Beat: 1 Voice: 1
Following note: F#6 Staff: 1-Lever Harp Bar: 4 Beat: 1 Voice: 1

The trace window lists a series of notes and their corresponding MIDI messages. The notes are listed in a table format:

Note	Staff	Lever	Harp	Bar	Beat	Voice
~056,96	2	2-Lever	Harp	3	1.1094 (28)	0
~056,96	2	2-Lever	Harp	3	1.2188 (56)	0
~059,96	2	2-Lever	Harp	3	1.332 (85)	0
~061,96	2	2-Lever	Harp	3	1.4414 (113)	0
~063,96	2	2-Lever	Harp	3	1.5547 (142)	0
~063,96	2	2-Lever	Harp	3	1.6641 (170)	0
~066,96	2	2-Lever	Harp	3	1.7773 (199)	0
~068,96	2	2-Lever	Harp	3	1.8867 (227)	0
~068,96	2	2-Lever	Harp	3	2 (256)	0
~071,96	2	2-Lever	Harp	3	2.1094 (284)	0
~073,96	2	2-Lever	Harp	3	2.2188 (312)	0
~075,96	2	2-Lever	Harp	3	2.332 (341)	0
~075,96	2	2-Lever	Harp	3	2.4414 (369)	0
~078,96	2	2-Lever	Harp	3	2.5547 (398)	0
~080,96	2	2-Lever	Harp	3	2.6641 (426)	0
~080,96	2	2-Lever	Harp	3	2.7773 (455)	0
~083,96	2	2-Lever	Harp	3	2.8867 (483)	0
~085,96	2	2-Lever	Harp	3	3 (512)	0
~087,96	2	2-Lever	Harp	3	3.1094 (540)	0
~087,96	2	2-Lever	Harp	3	3.2188 (568)	0
~090,96	2	2-Lever	Harp	3	3.332 (597)	0
~092,96	2	2-Lever	Harp	3	3.4414 (625)	0
~092,96	2	2-Lever	Harp	3	3.5547 (654)	0
~095,96	2	2-Lever	Harp	3	3.6641 (682)	0
~097,96	2	2-Lever	Harp	3	3.7773 (711)	0
~099,96	2	2-Lever	Harp	3	3.8867 (739)	0
~099,1	2	2-Lever	Harp	3	3.9922 (766)	0
~N102,96	1	1-Lever	Harp	4	1 (0)	0
~O102,1	1	1-Lever	Harp	4	4 (768)	0

The trace window also shows the following MIDI messages:

- (056 = G#/Ab3 N56 = G#/Ab3)
- (056 = G#/Ab3 N59 = B3)
- (059 = B3 N61 = C#/Db4)
- (061 = C#/Db4 N63 = D#/Eb4)
- (063 = D#/Eb4 N63 = D#/Eb4)
- (063 = D#/Eb4 N66 = F#/Gb4)
- (066 = F#/Gb4 N68 = G#/Ab4)
- (068 = G#/Ab4 N68 = G#/Ab4)
- (068 = G#/Ab4 N71 = B4)
- (071 = B4 N73 = C#/Db5)
- (073 = C#/Db5 N75 = D#/Eb5)
- (075 = D#/Eb5 N75 = D#/Eb5)
- (075 = D#/Eb5 N78 = F#/Gb5)
- (078 = F#/Gb5 N80 = G#/Ab5)
- (080 = G#/Ab5 N80 = G#/Ab5)
- (080 = G#/Ab5 N83 = B5)
- (083 = B5 N85 = C#/Db6)
- (085 = C#/Db6 N87 = D#/Eb6)
- (087 = D#/Eb6 N87 = D#/Eb6)
- (087 = D#/Eb6 N90 = F#/Gb6)
- (090 = F#/Gb6 N92 = G#/Ab6)
- (092 = G#/Ab6 N92 = G#/Ab6)
- (092 = G#/Ab6 N95 = B6)
- (095 = B6 N97 = C#/Db7)
- (097 = C#/Db7 N99 = D#/Eb7)
- (099 = D#/Eb7 N99 = D#/Eb7)
- (099 = D#/Eb7)
- (N102 = F#/Gb7)
- (O102 = F#/Gb7)

The trace window also shows the following tuning information:

Tuning: C#D#EbF#G#AbB

Callouts from the image:

- "8va line recognized and gliss is extended" points to the "8va" line in the score.
- "top note named F#6" points to the "Following note: F#6" line in the trace window.
- "Top played MIDI message is for F#7" points to the "(O102 = F#/Gb7)" line in the trace window.

Magnetic Glisses in Sibelius 8.6 and later (as of November 18 2017)

Magnetic Glisses were introduced in Sibelius 8.6 to replace Gliss Lines. Magnetic Glisses are properties of a Note object, rather than separate lines, and glisses added in Sibelius 8.6 and later will be Magnetic Glisses by default.

This plugin does not recognize Magnetic Glisses, and while you can still select 2 notes and have the plugin add MIDI messages between them, the plugin will not find any note pairs if you make a passage selection.

This plugin was already making guesses to find the notes associated with a gliss line, and with Magnetic Glisses it has less to work with to correctly find the following note for a Magnetic Gliss. It is also trickier to disable playback for a Magnetic Gliss, since you must silence its note to silence it. I may upgrade the plugin in the future to do what it can with Magnetic Glisses, but for now, if you want to have the plugin process a passage selection you should convert Magnetic Glisses to non-magnetic gliss lines.

The easiest way to do that is to export your score to a previous version prior to 8.6. Sibelius will convert gliss-type Magnetic Glisses to gliss Lines (it will leave slides alone) and will do a better job of it than a plugin could do. Open the converted score and then run the plugin. If you want to add more glisses to the score, use the Line Between Notes plugin, which will create non-magnetic glisses.

Appendix: Interpreting the MIDI cloud

Here is the trace of the MIDI messages we used earlier as an example. The MIDI message itself is only the part of these lines up to the first “(“. The rest of the data is location information and a translation of the MIDI note numbers into pitch names

MIDI text, (staff, bar, position as beat, (position in units), voice)
~O65,96 ~N67,96 (1-Harp, 1, 4.1406 (804), 0) (O65 = F4 N67 = G4)
~O67,96 ~N69,96 (1-Harp, 1, 4.2852 (841), 0) (O67 = G4 N69 = A4)
~O69,96 ~N70,96 (1-Harp, 1, 4.4258 (877), 0) (O69 = A4 N70 = A#/Bb4)
~O70,96 ~N72,96 (1-Harp, 1, 4.5703 (914), 0) (O70 = A#/Bb4 N72 = C5)
~O72,96 ~N74,96 (1-Harp, 1, 4.7109 (950), 0) (O72 = C5 N74 = D5)
~O76, 1 (1-Harp, 1, 4.8477 (985), 0) (O76 = E5)
~O74,96 ~N76,96 (1-Harp, 1, 4.8555 (987), 0) (O74 = D5 N76 = E5)
~N77,96 (1-Harp, 2, 1 (0), 0) (N77 = F5)
~O77,1 (1-Harp, 2, 5 (1024), 0) (O77 = F5)

As an example, if we have

~O65,96 ~N67,96 (1-Harp, 1, 4.1406 (804), 0) (O65 = F4 N67 = G4)

There are 2 MIDI messages, each preceded by a tilde (~)

~O65,96 turns **off** MIDI pitch 65, and sets its velocity to 96
~N67,96 turns **on** MIDI pitch 67, and sets its velocity to 96

The next part of the trace shows the location of the start of the MIDI message, with comma-separated fields

(1-Harp, 1, 4.1406 (804), 0)

- *1-Harp* is the staff number and name
- *1* is the bar number, in this case bar 1, so we are in the first bar of the first staff
- *(804)* is the rhythmic position in the bar, first in beats (4.1406), and then in internal units (804), where there are 256 units in a quarter note.
- *0* is the voice number. Voice 0 means “all voices”

Finally, if there are MIDI pitch numbers in the message, the last section translates these to pitch names

(O65 = F4 N67 = G4)

So the message *O65* is turning off the pitch F4, and *N67* plays pitch G4.