

Polymeter Help

Contents

Getting Started

[Introduction](#)

[License](#)

[Requirements](#)

[Installing](#)

[Uninstalling](#)

[Getting help](#)

Concepts

[Polymeter](#)

[Tracks](#)

[Steps](#)

[Modulations](#)

[Dubs](#)

[Songs](#)

[Parts](#)

[Presets](#)

[Mappings](#)

[Convergences](#)

[List controls](#)

[Grid controls](#)

Tracks

[Note tracks](#)

[Controller tracks](#)

[Tempo tracks](#)

[Modulator tracks](#)

[Internal tracks](#)

Modulation

[Mute modulation](#)

[Note modulation](#)

[Velocity modulation](#)

[Duration modulation](#)

[Range modulation](#)

[Position modulation](#)

[Tempo modulation](#)

[Scale modulation](#)

[Chord modulation](#)

[Index modulation](#)

[Voicing modulation](#)

[Offset modulation](#)

[Queue modulation](#)

Views

[Views](#)

[Track View](#)

[Track View](#)

[Track Grid](#)

[Track Mutes](#)

[Track Steps](#)

[Track Selection](#)

[Step Selection](#)

[Velocities Pane](#)

[Song View](#)

[Live View](#)

Docking Windows

[Docking windows](#)

[Arranging docking windows](#)

[Menu bar](#)

[Toolbar](#)

[Properties](#)

[Channels](#)

[Presets](#)

[Parts](#)

[Modulations](#)

[MIDI Input](#)

[MIDI Output](#)

[Piano](#)

[Graph](#)

[Phase](#)

[Step Values](#)

[Mapping](#)

Properties

[Tempo](#)

[Timebase](#)

[Meter](#)

[Key Signature](#)

[Song Length](#)

[Start Position](#)

[Note Overlap](#)

[Record Offset](#)

[Loop From](#)

[Loop To](#)

Options

[MIDI](#)

[Input Device](#)

[Output Device](#)

[Latency](#)

[Buffer Size](#)

[Thru](#)

[Default Velocity](#)
[Input Quantization](#)
[Record Input](#)
[Send MIDI Clock](#)
View
[Update Frequency](#)
[Show Current Position](#)
[Show Note Names](#)
[Show General MIDI Names](#)
[Show Quant as Fraction](#)
[Show Track Colors](#)
[Unique Track Names](#)
[Zoom Step](#)
[Live View Font Height](#)
Graph
[Graphviz Folder](#)
[Graph Handles Unicode](#)
[Use Cairo](#)
General
[Show Property Descriptions](#)
[Automatically Check for Updates](#)
[Always Record](#)

Menus

File
[New](#)
[Open](#)
[Close](#)
[Save](#)
[Save As](#)
[Export](#)
[Import](#)
[Exit](#)
Edit
[Standard editing commands](#)
[Undo](#)
[Redo](#)
[Cut](#)
[Copy](#)
[Paste](#)
[Insert](#)
[Delete](#)
[Select All](#)
[Deselect](#)
[Rename](#)
[Find](#)
[Replace](#)

View

[Docking Windows](#)

[Status Bar](#)

[Application Look](#)

[Zoom In](#)

[Zoom Out](#)

[Zoom Reset](#)

[Properties](#)

[Channels](#)

[Modulations](#)

[Groups](#)

[Velocities](#)

[Track](#)

[Song](#)

[Live](#)

[Customize](#)

Track

[Length](#)

[Shift](#)

[Rotate](#)

[Reverse](#)

[Invert](#)

[Tie Notes](#)

[Transpose](#)

[Velocity](#)

[Fill](#)

[Stretch](#)

[Mute](#)

[Solo](#)

[Group](#)

[Preset](#)

[Modulation](#)

[Sort](#)

[Color](#)

Transport

[Play](#)

[Pause](#)

[Record](#)

[Rewind](#)

[Loop](#)

[Go To Position](#)

[Next Convergence](#)

[Previous Convergence](#)

[Convergence Size](#)

[Record Tracks](#)

Tools

[Statistics](#)
[Time To Repeat](#)
[Prime Factors](#)
[Velocity Range](#)
[Check Modulations](#)
[Devices](#)
[Options](#)
[Import Table](#)
[Export Table](#)
[Convergences](#)
[Select Unmuted](#)
[MIDI Learn](#)

Window

[New Window](#)

[Full Screen](#)

[Reset Layout](#)

[New Horizontal Tab Group](#)

[New Vertical Tab Group](#)

[Windows](#)

Help

[Help Topics](#)

[Polymeter on the Web](#)

[Check for Updates](#)

[About](#)

Loose Ends

[Audio versus events](#)

[Recording MIDI input](#)

[Maximum track length](#)

[Normalized angles](#)

[Period](#)

[Primes](#)

[Modular arithmetic](#)

[Clipping](#)

[Unicode](#)

[Undersized graph text](#)

Reference

[Errors](#)

[Options Summary](#)

[Shortcuts](#)

Getting Started

Introduction

Polymeter is a specialized MIDI sequencer for composing music in [polymeter](#), meaning music that uses multiple meters simultaneously. The application doesn't generate audio directly, instead it outputs a stream of MIDI messages to a [device](#) that converts MIDI to audio, such as a hardware or software synthesizer. By default, the application outputs to the GS Wavetable Synth that ships with Windows. The application also [exports](#) a Standard MIDI file that can be imported into most music software.

A Polymeter document consists of an array of [tracks](#), each of which produces a single MIDI note or control message at a time. Each track has its own [length](#) and loops independently through its own array of equal-sized [steps](#). For [note tracks](#), the steps determine the starting times, durations, and velocities of notes, but steps are interpreted differently for other [track types](#). Tracks of differing lengths can be juxtaposed, and then edited while they're looping. Tracks can also [modulate](#) other tracks, and can be arranged into [songs](#).

The application features multiple [views](#), corresponding to different stages of the composing process, and various [docking windows](#). Most application parameters can be remotely controlled by [mapping](#) them to MIDI messages.

License

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111 USA.

Requirements

Recommended:

Windows 7 or higher, 64-bit
4GB memory or more
Quad-core CPU
Decent graphics card

Installing

Polymeter comes in two flavors (Installer and Portable), each of which is available in both 64-bit and 32-bit versions. If you have a 64-bit CPU, it's strongly recommended to use the 64-bit version of Polymeter. Most users want an installer, because it's easier. For example, installing adds Polymeter to the Windows start menu, and allows Polymeter to register its file types, so that they have appropriate icons and can be opened via double-clicking in Explorer.

Installing for the first time:

Make sure you're logged on as Administrator. The Installer flavor of Polymeter is distributed as a zip file. Unzip the distribution file, using [7-Zip](#) or an equivalent program, and then double-click on Polymeter.msi to launch the installer. The installer is about as simple as an installer can be: for the default setup, just keep hitting "Next." Note that on the "Select Installation Folder" page, you have the option to install for all users, but the default is to only install for the current user.

Upgrading a previous installation:

1. Make sure you're logged on as Administrator.
2. Unzip the distribution file to a folder, if you haven't already.
3. In that folder, double-click Upgrade.bat to launch the installer.
4. In the installer, select the "Repair" option.

Note that double-clicking Polymeter.msi won't work for upgrades: you will get the message "Another version of this product is already installed".

To use the portable app, simply unzip the distribution file to a folder and double-click Polymeter.exe to run the application.

Uninstalling

To uninstall Polymeter, use Add/Remove Programs in the Control Panel, or double-click on Polymeter.msi and select the "Remove" option.

To remove the portable version, delete the folder you unzipped the application files into, and also delete the following registry key:

HKEY_CURRENT_USER\Software\Anal Software\Polymeter

Getting help

Polymeter has context-sensitive help, which means that pressing F1 shows different topics depending on the situation. Context-sensitive help is available for the following types of user-interface objects:

Object type	To get help:
Menus and context menus	Hover the cursor over a menu item and press F1.
Toolbar buttons	Hover the cursor over a button and press F1.
Property bars	Left-click a property name and press F1.
Other toolbars	Left-click a control and press F1.

Modal dialogs	Press F1 while the dialog is displayed. For some dialogs, the topic shown depends on which control has focus.
---------------	---

If none of the above cases apply, or if help is accessed via [Help Topics](#), a default topic is shown.

Concepts

Polymeter

Polymeter is the use of multiple meters simultaneously within a single piece of music. In this application, polymeter means combining [loops](#) of different lengths, so that they slip relative to each other, the way oscillators with different frequencies shift phase relative to each other. Polymeter is more interesting when the loop lengths being combined are [relatively prime](#), meaning they share no common factors except one. Such loop combinations gradually drift apart (diverge) and then drift back together (converge) in a predictable way, producing intricate patterns of interference as they do so. Various tools are provided to work with [convergences](#).

Complex polymeter

Complex polymeter is the simultaneous use of three or more relatively prime meters. For example, a composition that uses the meters 4/4, 5/4, and 7/4 all at the same time is in complex polymeter.

Polymeter versus polyrhythm, odd time, and phasing

- Polyrhythm means combining different rhythms. There's no requirement that the rhythms be different lengths or exhibit phase shift. Polyrhythm is a very general category, of which polymeter is a subset.
- Odd time means using an odd meter (5/4, 7/4, 11/4 etc.), or alternating between several odd meters, one after the other. It's the use of multiple odd meters *simultaneously* that distinguishes polymeter from odd time.
- Phasing is a more general category than polymeter. Any system that combines oscillations of different frequencies exhibits phasing. Loops of different lengths "slip" in and out of phase relative to each other, i.e. they phase. Polymeter is a specialized subset of phasing. Polymeter adds a constraint that the different loop lengths must share a common unit (e.g. a sixteenth note). In other words, polymeter is *quantized phasing*. In phasing, the slippage is continuous, whereas in polymeter, the slippage is discrete: it occurs in [steps](#). Phasing is good for ambient music; polymeter is good for rhythmic music.

Tracks

A track consists of an array of [steps](#) and some [properties](#) that affect the track's behavior. A [polymer](#) composition is a collection of tracks, each of which [cycles](#) through its steps

independently. The tracks are organized similarly to a table; each track corresponds to a row in the [track view](#). There are various [track types](#), and the type determines what the track outputs, and how it interprets its steps. A track can output only one MIDI note or control message at a time. To make chords, create multiple tracks, one for each of the chord's tones.

- [Note tracks](#) output a MIDI note, and have special capabilities.
- [Controller tracks](#) output a MIDI controller, pitch bend, aftertouch, or program change.
- [Tempo tracks](#) output a tempo change.
- [Modulator tracks](#) output nothing, and are used to alter the behavior of other tracks.
- [Internal tracks](#) alter the sequencer's behavior.

A track can be edited via the track view, which is vertically split into two [panes](#). The [left pane](#) lets you edit the track's properties, while the [right pane](#) lets you mute or unmute tracks and edit the track's steps.

The list of tracks can be edited via the [standard editing commands](#). Tracks can be [selected](#), and many commands operate on selected tracks. Selected tracks can also be reordered by dragging, or [sorted](#). Tracks can be [named](#), and giving them unique, meaningful names is highly recommended.

The simplest way to create polymeter is by juxtaposing tracks of different lengths, assuming the lengths are [relatively prime](#). More complex methods include using tracks to alter the behavior of other tracks, in a process called [modulation](#).

Track properties

Name	The track's name
Type	The track's type
Channel	Which MIDI channel the track outputs on
Note	Which MIDI note or controller the track outputs
Length	Length of the track, in steps
Quant	Duration of each of the track's steps, in ticks
Offset	Time offset applied to the track's steps, in ticks; positive = later, negative = earlier
Swing	Time offset applied to the track's even-numbered steps, in ticks; positive = later, negative = earlier
Velocity	Offset added to the track's step velocities
Duration	Offset added to the track's note durations
Range Type	Selects a method for limiting the range of the track's notes
Range Start	Lowest note of the track's range
Mute	True if the track is muted

Track types

Type	Outputs	Step
Note	Note message	Unsigned note velocity; steps can also be tied together to form longer notes
Key Aftertouch	Key aftertouch message	Unsigned key aftertouch pressure
Control	Continuous controller message	Controller value; signed or unsigned depending on context
Patch	Program change message	Unsigned program number
Channel Aftertouch	Channel aftertouch message	Unsigned channel aftertouch pressure
Wheel	Pitch bend message	Signed pitch bend value (most significant byte only)
Tempo	Tempo change message	Signed fractional tempo change
Modulator	Nothing	Depends on the modulation type
Internal	Depends on controller number	Internal controller value

Steps

At the most abstract level, a step is simply a number. Each [track](#) contains an array of one or more steps. A step array is a sequence of integer values, evenly spaced in time. The count of steps in the array is the track's [Length](#), and the duration of each step is the track's [Quant](#). A track's steps are played sequentially in a loop.

A track's step array resembles a time signature or [meter](#), with Length as the numerator and Quant as the denominator, except that the Quant is specified in [ticks](#). For example, assuming the default [timebase](#) of 120 ticks per quarter note, a five-step track with a Quant of 120 is equivalent to five quarter notes, or 5/4 time.

Specifically, a step is a 7-bit integer that can be [unsigned or signed](#) depending on the context. Steps can be interpreted as note velocities, controller values, or modulator values. A step's interpretation depends on what [type](#) of track the step belongs to, except for [modulator tracks](#), in which case the interpretation depends on the [modulation type](#).

Steps can be edited in the following ways:

- The [track steps](#) pane lets you set and clear steps. For [note tracks](#), you can also [tie](#) adjacent steps together to form longer notes.
- The [velocities pane](#) displays the step values of the [selected tracks](#) as vertical bars. The step values can be changed by dragging the bars. Various drawing functions can be accessed by pressing modifier keys while dragging.

- The [Step Values bar](#) displays the step values of the selected tracks on a [grid](#) so that they can be edited numerically.
- The [Fill](#) command lets you fill selected tracks or steps with a series of values. The series can be linear, or a periodic function.
- Steps can be copied, cut, pasted, inserted, and deleted using the [standard editing commands](#).
- Steps can be transformed using Track menu commands such as [Shift](#), [Rotate](#), [Reverse](#), [Invert](#), [Velocity](#), and [Stretch](#).

Step values can also be [imported](#) from a text file containing comma-separated values (CSV format), which is useful if you want to generate step arrays with an external application.

Because a track's steps are evenly spaced in time, undesirably stiff rhythms may result. Rhythmic stiffness can be mitigated by adjusting the track's [swing](#), or by applying [offset modulation](#) to the track.

Unsigned versus signed steps

Steps are stored in seven binary bits, and therefore have 2^7 or 128 possible values. The values range either from 0 to 127, or from -64 to +63, depending on whether the step is interpreted as an unsigned or signed number. This convention derives from the MIDI standard, which commonly uses 7-bit data for message parameters and data. For example the data of a MIDI control change message is considered unsigned for volume controls, but signed for pan controls. To convert a value from unsigned to signed, subtract 64 from it; to do the reverse, add 64. The following table shows some correspondences between unsigned and signed values.

Unsigned	Signed
127	+63
96	+32
64	0
32	-32
0	-64

Whether a step is interpreted as unsigned or signed depends on the context in which the step is used. The context is determined by the [track type](#), except for [modulator tracks](#), in which case it's determined by the [modulation type](#).

Ticks

A tick is the smallest unit of relative time that can be used within a given MIDI composition. Ticks are related to beats (quarter notes) by the [timebase](#), which specifies the number of ticks per beat. Various properties of a [track](#), including the duration of its [steps](#) (also known as its [Quant](#)) are specified in ticks. The length of a tick in absolute time depends on the [tempo](#), as shown below:

`tick_length = 60 / tempo / timebase`

At the default tempo (120 BPM) and timebase (120 TPQN), a tick is approximately four milliseconds long.

Modulations

Besides juxtaposition, another important method of combining [tracks](#) is *modulation*, in which a track alters the behavior of another track. The track that does the modulating is called the modulator or *source*, whereas the track that's being modulated is called the target or *sink*. Modulations are created via the [modulations bar](#) or the Track menu's [Modulation](#) command. There are many types of modulation, and they vary in terms of the targets they support, and how they interpret step values, as shown below.

Modulation types

Type	Targets	Effect	Step
Mute	All	Mutes or unmutes target	Boolean; non-zero value mutes target
Note	Note tracks, scale modulators, chord modulators	Offsets note or tone	Signed offset in semitones or scale tones
Velocity	All track types except Tempo and Modulator	Offsets note velocity or controller value	Signed MIDI data offset
Duration	Note tracks only	Offsets note duration	Signed offset in ticks
Range	Note tracks only	Offsets range start	Signed offset in semitones
Position	All	Rotates step array	Signed rotation in steps
Tempo	Tempo tracks only	Scales tempo	Signed fractional tempo change
Scale	Note tracks only	Adds note to current scale	Signed note offset in semitones
Chord	Note tracks only	Adds scale tone to current chord	Signed zero-based scale tone index
Index	Note tracks only	Picks a scale or chord tone	Signed zero-based scale or chord tone index
Voicing	Note tracks only	Drops or raises a scale or chord tone	Signed one-based voice index
Offset	All track types except Tempo	Offsets playback timing	Offset in ticks; unsigned for Note or Controller, signed for Modulator
Queue	Note tracks only	Assigns a sequence of pitches to notes	Signed offset in semitones, or in scale or chord tones if indexing

Modulation relationships

The relationship between source and target tracks can be any of the following:

Relationship	Meaning
One to one	A single source modulating a single target
One to many	A single source modulating multiple targets
Many to one	Multiple sources modulating a single target
Many to many	Multiple sources modulating multiple targets

In the above relationships, the modulations don't necessarily have to be all of the same type. For example a note track could have its mute state modulated by one source track, and its note modulated by a different source track.

Nested modulation

Nested relationships are also possible, in other words a modulator can modulate a modulator, though there are limits on this, as explained below. The only thing that's definitely not allowed is a track modulating itself, either directly, or indirectly via a loop, as this would lead to infinite recursion. The user interface tries to prevent self-modulation, but if it occurs you'll get the error message "Modulation exceeded the maximum number of recursions." The maximum modulation depth is 32 levels, however only [mute](#), [position](#), and [offset](#) modulation fully support nesting. The modulation types that fully support nesting can form chains, for example:

mute modulator → position modulator → note modulator → note track

Modulation rules

The table below summarizes the supported modulation behavior. For each track type, it tells you what types of modulation the track can be targeted by. Use the [Check Modulations](#) command to verify that your modulations comply with these rules.

Target track type	Supported modulation types
Note	All except Tempo
Key Aftertouch, Control, Patch, Channel Aftertouch, Wheel, Internal	Mute, Velocity, Position, Offset
Tempo	Mute, Position, Tempo
Modulator	Mute, Position, Offset; also Note but only if target is a Scale or Chord modulator

Dubs

A dub is an [event](#) that consists of one or more [tracks](#) being [muted](#) or unmuted at a particular time. Dubs are useful for arranging tracks into a [song](#), which is essentially a sequence of dubs.

You can create dubs by drawing them in [song view](#), but it's often easier to create them by [recording](#) an arrangement that you perform in [live view](#).

Dubs don't support fading. To fade a track in or out, either create a [velocity modulation](#) that targets the track, or create a [controller track](#) that changes the instrument's volume (typically controller number seven). These fader tracks can also be dubbed in and out, like any other track. Another option is to use a MIDI control surface to change instrument volumes live, and capture the control changes by [recording MIDI input](#). If you're [exporting](#) your sequence to another music software, you may prefer to create fades within that software, as it may have better automation support.

Songs

A song is a particular arrangement of your [tracks](#), created by muting and unmuting them over time. In other words, a song is a sequence of [dubs](#), each of which mutes or unmutes one or more tracks at a particular time. The song can be visualized and edited in [song view](#). It's possible to create a song entirely by editing in song view, but it's often more intuitive to [record](#) a rough arrangement in [live view](#) and then use song view to refine the transitions.

Creating a song is optional. The main advantage of creating a song is that you can "paint" with your [polymeter](#) loops on a timeline, in a completely non-destructive way. Editing the song never modifies the underlying tracks, and the tracks always maintain correct phase, no matter how much they're dubbed.

During [playback](#), if the song view is active, the [track mutes](#) are automated, meaning tracks are automatically muted or unmuted as required by the song arrangement. You can start playback at any [position](#), or skip around within the song while it's playing, and the track mutes "chase" to the appropriate state. Note however that a track must have at least one dub in order to be automated.

If the song exists, meaning at least one track has dubs, the MIDI file [export](#) applies the dubs to the tracks, so that the resulting file follows the arrangement.

A song may also include [recorded MIDI input](#) such as note or control change messages.

Parts

A part is a group of [tracks](#) that correspond to a single item in the [live view](#)'s track list. Polymeter compositions tend to involve many tracks, because each track outputs only one note or control event at a time. The proliferation of tracks can become overwhelming and cumbersome to perform with, particularly if the track list gets long enough to require frequent scrolling. Grouping tracks into parts reduces visual clutter, and may shorten the live view's track list enough to avoid scrolling during a performance.

A part's member tracks all share the same [mute](#) state in live view, meaning if one member is muted, they're all muted. Put another way, by grouping tracks into a part, you lose the ability to

mute some of them but not others in live view. For example, if you're using several tracks to form a chord, and the entire chord should always be muted or unmuted as a single entity, you may want to group the tracks into a part. For another example, if you have modulator tracks that can remain unmuted all the time, they can all be grouped into a single part.

Parts are edited using the [Parts bar](#). To create a new part, first [select](#) its member tracks, and then use the Track menu's [Group](#) command. A new part initially has the same name as its first member track, but you can rename it.

A track can belong to at most one part. When you create a new part, if any of the selected tracks already belong to a part, a warning [message](#) is displayed, giving you the option to continue or cancel. If you continue, the offending tracks are removed from their previous part before being reassigned to the new one.

To update an existing part, in other words to add or remove members, do as follows:

1. Select the part's member tracks, by right-clicking the part's list item and then using the Tracks command on the Parts bar's [context menu](#).
2. In the [track view](#), select or deselect tracks as desired.
3. Right-click the part's list item and use the Update command on the Parts bar's context menu. If any of the selected parts belong to a different part than the one you're updating, a warning message is displayed, as explained above.

To avoid confusion, it's recommended to keep all the member tracks of a part contiguous in track view. You may want to use [color](#) to distinguish your parts from each other.

Presets

A preset is a snapshot of the [mute](#) states of all the [tracks](#). Applying a preset mutes or unmutes tracks as needed, so that after it's applied, the same tracks are muted as when the preset was created. Presets are especially useful for creating section changes, because applying a single preset is easier than muting or unmuting many tracks at once.

Presets are edited using the [Presets bar](#). To create a new preset, use the Track menu's [Preset](#) command. A new preset initially has an automatically generated name, but you can rename it.

To update an existing preset, in other words to change its snapshot, do as follows:

1. Apply the preset, either by double-clicking the preset's item within the preset list, or by right-clicking the preset's list item and then using the Apply command on the Preset bar's [context menu](#).
2. In the [track view](#), mute or unmute tracks as desired.
3. Right-click the preset's list item and use the Update command on the Preset bar's context menu.

Mappings

Mapping is the process of assigning input MIDI messages to properties, so that those properties can be remotely controlled, typically via a MIDI control surface. Each mapping consists of a description of the input message, and a description of the target property. The mapping feature also supports translation. An input MIDI message can be translated to any other MIDI message, and/or mapped to any [track property](#) except track name. Other possible mapping targets include [parts](#), [presets](#), and transport commands. Mappings are edited via the [Mapping](#) docking window.

An input message can have multiple mappings, i.e. it can be translated to multiple messages, or mapped to multiple properties, or to the same property in multiple tracks, or any combination of these. The reverse is also true: different input messages can be translated to the same output message, or mapped to the same track property.

Translation facilitates tweaking synthesizer settings with a control surface. It's likely that different songs will use different synthesizers and different control schemes. By setting up appropriate translations in Polymeter, the user can avoid reprogramming the control surface for each song. The control surface can always output the same messages; only the mappings need to change, and the mapping details are conveniently stored in the song's Polymeter document. More importantly, since Polymeter can [record the MIDI input](#) as well as the dubs, a performance can be recorded as [events](#) and then completely reconstructed afterwards.

Mapping is particularly interesting for track properties that affect the rhythmic feel, such as [Offset](#) and [Swing](#). The [Range Start](#) property is another fruitful target.

For some target properties, the mapping's output range may need to be adjusted for optimal results. For example, suppose a knob that outputs 0 to 127 as usual is mapped to a target property that has only three states. The knob's first three values step through the states, while its remaining range of motion does nothing, making the knob awkward to use. This is solved by giving the mapping an output range of 0 to 2, and thereby spreading out the target's states over the knob's entire range of motion.

Convergences

In [polymeter](#), loops of different lengths gradually drift apart (diverge) and back together (converge). Alternatively we can say that the loops drift out of phase, and back into phase. Two loops are exactly in phase when they reach their respective starting points at exactly the same time, and this moment is called a *convergence*.

When the number of loop lengths being considered is three or more, we can distinguish between full and partial convergence. Full convergence is when *all* of the loops are in phase, whereas partial convergence is when only a subset of them are in phase. No matter how many different lengths are used, they will all fully converge at some point, after which the overall pattern repeats.

For [relatively prime](#) lengths, determining when full convergence will occur is straightforward: it's the product of the lengths. So for 3, 5, and 7, it's $3 \times 5 \times 7 = 105$. Each pair of relatively prime lengths also converges separately. In this case, partial convergences occur at multiples of 15 (3×5), 21 (3×7), and 35 (5×7). For details, see the [example](#) below. As more relatively prime lengths are combined, the number of partial convergences increases, and the time required for the overall pattern to repeat (full convergence) also increases.

The application provides various tools for working with convergences:

- The [Time to Repeat](#) command calculates how long it will take for the [selected tracks](#) to repeat, in beats and time.
- The [Prime Factors](#) command calculates the prime factors of the lengths of the selected tracks.
- The [Convergences Calculator](#) computes all of the convergences between the lengths of the selected tracks, or if no tracks are selected, between a list of comma-separated values.
- The [Next Convergence](#) and [Previous Convergence](#) commands move the current position to the next or previous convergence between the lengths of the selected tracks, or all tracks if none are selected. The minimum convergence size is selectable via the [Convergence Size](#) submenu of the Transport menu.

It's more interesting if something happens at these convergences. In 4/4 music, significant changes often occur at powers of two, e.g. 16, 32, 64, etc. In polymeter, significant changes should naturally occur at the convergences between the different lengths.

Convergence example

This table shows the partial convergences of 3, 5, and 7 up to their first full convergence at 105. Subsequent full convergences would occur at 210, 315, 420, and so on. The table could be extended infinitely far into the future or past, and would continue to repeat this same pattern of convergences.

Convergence	Factors
15	3,5
21	3,7
30	3,5
35	5,7
42	3,7
45	3,5
60	3,5
63	3,7
70	5,7
75	3,5
84	3,7

90	3,5
105	3,5,7

List controls

The application makes frequent use of lists, some of which are editable [grid controls](#). Within a list, rows can be selected either via the mouse, or via the keyboard.

Selecting rows via mouse

- To select a single row, left-click it.
- To select a range of adjacent rows, left-click the range's first row, press and hold down the Shift key, left-click the range's last row, and then release the Shift key. If the range is large, you may need to scroll the list in between selecting its first and last rows.
- To add or remove a row to or from an existing selection, left-click the row while pressing the Ctrl key.
- To deselect all rows, left-click below the last row, or to the right of the last column.

Selecting rows via keyboard

You can also select list rows via the keyboard, using the up/down arrow keys, possibly in combination with other keys.

- To select a single row, repeatedly press (or press and hold) the up or down arrow key until the desired row is selected. Doing so navigates through the list by moving the single selection from row to row, scrolling the list if necessary. You can also use the Page Up, Page Down, Home, and End keys to navigate through the list more rapidly.
- To select a range of adjacent rows, do the same as for a single row, but while pressing the Shift key.
- To add or remove a row to or from an existing selection, first navigate to the desired row as described above, but while pressing the Ctrl key. This moves the list selection mark (usually a dotted rectangular outline) without disrupting the existing selection. When the selection mark is on the desired item, press Ctrl+Alt+Space to toggle the item's selection state. The Alt key is necessary because Ctrl+Space is reserved for the [Play](#) command.

List columns

You can change the width of list columns by using the mouse. To change the width of a column, drag the boundary on the right side of the column heading until the column is the width that you want. To auto-size a column so that it's the same width as its widest visible item, double-click the boundary on the right side of the column heading.

A list may allow its columns to be reordered. To move a column, left-click the column heading and drag it to where you want it. Not all lists support this feature.

A list may support restoring its columns to their original state. To reset the columns to their default width and order, right-click in the column header and select Reset in the context menu. Not all lists support this feature.

Grid controls

A grid control is a type of [list control](#) that allows its items to be edited. The [Track view](#) uses a grid control, as do the [Channels](#), [Mapping](#), [Modulations](#), and [Step Values](#) bars. To edit an item, left-click it. This causes either a pop-up edit box or a drop list to appear, depending on the data type. If a grid row is selected, you can also edit its first item by pressing Tab. While the pop-up control is displayed, you may edit the value or type a new one. To end your edit, left-click outside of the pop-up control, or press one of the following keys:

Enter	Saves the edit.
Escape	Discards the edit.
Tab	Saves the edit and moves to the next column.
Shift+Tab	Saves the edit and moves to the previous column.
Ctrl+Down	Saves the edit and moves to the next row.
Ctrl+Up	Saves the edit and moves to the previous row.

Selection works the same way as in a list control, with one exception: to [select rows via mouse](#), you must left-click within the *primary column*, otherwise you'll edit an item instead of selecting. The primary column is usually the leftmost column, unless you've reordered the columns. For example in the Track view, the primary column contains the track numbers, and has a number sign (#) in the column header.

To change an item in multiple rows at once, first select the desired rows, and then edit the item within any one of the selected rows.

Tracks

Note tracks

Note tracks output MIDI notes, and have special capabilities compared to other [track types](#). One difference is that note tracks allow adjacent steps to be [tied](#) together in order to form notes longer than one step. Another difference is that many [modulation types](#) can only target note tracks. By default, a note track outputs a constant note, selected by its [note](#) property. The simplest way to make a melody is to create multiple note tracks and assign them different notes. However it's also possible to make a melody with only a single note track, by using another track to alter the note, in a process called [note modulation](#). Another more advanced method is to modulate the note track with a combination of [scale modulation](#) and [index modulation](#), and possibly also [chord modulation](#). A note track can only output one note at a time. To make chords, create multiple note tracks, each playing a different note.

The placement of notes, their durations, and their velocities are all determined by a note track's steps. If a note is tied, meaning it occupies multiple adjacent steps, the first of its steps determines its velocity. Note placement and duration are edited using the [track steps](#) pane. To edit note velocities, use the [velocities pane](#) or the [step values bar](#); for other methods, see [steps](#). To offset all of a track's note velocities by a constant, enter a non-zero value for the [track velocity](#) property. Note tracks can be limited to a specified range of notes via the [range](#) properties.

Tied notes

Notes longer than one [step](#) are created by joining two or more adjacent steps. Steps joined in this way are said to be *tied*. Each step has an attribute called a *tie bit*, and if a step's tie bit is set, that step joins itself with the next step. If the next step is occupied, it extends the duration of the existing note by a step, instead of triggering a new note. If the next step isn't occupied, the tie bit has no effect.

- To toggle a step's tie bit from tied to untied or vice versa, press the Shift key, and while holding it down, left-click the step in question.
- To toggle the tie bits of multiple steps at once, first [select](#) the steps by right-clicking and dragging; then press the Shift key, and while holding it down, left-click anywhere within the steps.

[Note tracks](#) display tied and untied steps differently: a tied step fills the entire rectangle allocated for it, whereas an untied step only fills the left half of its rectangle. The following example shows four adjacent steps, all of which are occupied (not empty) and untied. Assuming the track's [quant](#) is a sixteenth note, these steps output four consecutive sixteenth notes.



In this example, the same four steps are shown, except that the first three have their tie bit set. This joins the four steps together so that they output a single quarter note.



A note may be extended past the end of the track, in which case the track's last step must be tied. Assuming the track is four steps long, this example shows a quarter note triggered by the third step.



At least one of a note track's steps must be either empty or untied, otherwise the track will be silent. Tying only applies to note tracks; for all other [track types](#), the tie bit is irrelevant and isn't shown. You can specify whether steps are initially tied or untied, using the [Tie Notes](#) command.

Triggering samples

[Tied notes](#) are more likely to be useful with melodic or chordal parts, because the instruments associated with such parts are typically sensitive to note duration. Tied notes may be irrelevant when triggering drums, because drum sounds commonly play their entire sample, regardless of the triggering note's duration. Many hardware and software drum machines ignore note duration, including the drum channel (channel ten) of a General MIDI synth, such as the GS Wavetable Synth that ships with Windows. If you're writing drum parts for a device that ignores note duration, you may find it convenient to use only untied steps, so that setting consecutive steps always retriggers the sample.

Note off

It's sometimes desirable to cut off an existing instance of a note, for example in order to "choke" a high-hat or cymbal. This can be done by creating a note with a velocity of zero, which sends a *note off* command, as explained below. The note off command instructs the [output device](#) to release the specified note. This technique can be used to create more complex mutual exclusion groups, and can be applied to pitched instruments as well as percussion.

A normal note with a velocity greater than zero starts by sending a note on command, and ends by sending a corresponding note off command, after the note's duration expires. But if a note's velocity is zero, the note off command is sent at the start of the note, and no action is taken at the end of the note, hence the note's duration is irrelevant in this case.

There's a trick to creating a note with velocity of zero. The [velocities pane](#) doesn't let you drag a note's velocity below one, but you can work around this by using a negative [track velocity](#) to offset the note's *net* velocity down to zero. The method with the least impact is to set the track's velocity offset to -1 ; this causes notes with velocities of one to send note off commands, while notes with higher velocities behave normally. If you want a track to *always* send note off commands, regardless of its note velocities, set its velocity offset to -127 . Negative [velocity modulation](#) can similarly offset note velocities down to zero.

Controller tracks

Controller tracks are a group of [track types](#) that output a control value instead of a note. The group includes the following track types:

Type	MIDI message	P1	P2
Key Aftertouch	Polyphonic Key Pressure	track note	step value
Control	Control Change	track note	step value
Patch	Program Change	step value	N/A
Channel Aftertouch	Channel Pressure	step value	N/A
Wheel	Pitch Bend Change	step value	N/A

A controller track's [steps](#) are interpreted as data values for the corresponding MIDI message. Each step causes a MIDI message to be sent to the [output device](#), except when a step has the

same value as the preceding step, in which case no message is sent for the later step, thereby avoiding what would otherwise be a duplicate message. The steps are interpreted as [unsigned or signed](#) data depending on the context. To achieve smooth control changes, it may be necessary to use a relatively small [Quant](#), but don't make it smaller than necessary, otherwise you risk exceeding the bandwidth limitations of the output device.

The [track velocity](#) property has a different meaning for controller tracks than it does for [note tracks](#): instead of offsetting the note velocities, it offsets the control values. In other words, the track's velocity offset (if any) is added to the track's step values to compute the output data values.

Tempo tracks

A tempo track *scales* the base [tempo](#) by a variable amount. The amount of scaling can vary over time, and its effect on the tempo is multiplicative, not additive. To create a tempo [track](#), change a track's [Type](#) to Tempo, and also set the track's [Duration](#) to a non-zero value, e.g. 100. Then set the track's steps to whatever shape you want the tempo change to have. For tempo tracks, Duration is reused as the tempo range, expressed as a percentage. If you set it to 100, the tempo ranges from half-time to double time. If you set Duration to 50 instead, the tempo range is half as wide. Tempo is an exponential function, much like frequency. The actual formula is:

$$\text{scaled_tempo} = \text{base_tempo} * 2^n$$

with the value of the exponent n given by:

$$n = \text{signed_step_value} / 64 * \text{tempo_range} / 100$$

Tempo scaling is mapped to a linear parameter (the range of a step), so that you can manipulate tempo as if it were linear, even though it isn't. The following table lists the tempo percentages that result from a range of step values, assuming a Duration of 100. Notice that +32 doesn't give 150%, because tempo isn't linear.

Step Value	Scaling Factor	Tempo %
+64	$2^1 = 2$	200
+32	$2^{0.5} = 1.414$	141.4
0	$2^0 = 1$	100
-32	$2^{-0.5} = 0.707$	70.7
-64	$2^{-1} = 0.5$	50

In actuality the tempo can't quite reach double time, because the maximum [signed](#) step value is +63, not +64. The table below shows the actual tempo ranges that result from a series of Duration values. Notice that the maximum values are slightly less than ideal (198% instead of 200%, 391% instead of 400%). If this asymmetry is problematic, you can avoid it by only using the lower half of the step value range, from -64 to 0.

Duration	Tempo Range		
	-64	0	+63
25	84%	100%	119%
50	71%	100%	141%
100	50%	100%	198%
200	25%	100%	391%

Only one tempo track can be active (unmuted) at a time. If you have multiple tempo tracks unmuted at once, they will compete, which probably isn't what you want. To get second-order change (tempo changing in multiple ways at once), use [tempo modulation](#).

During [playback](#), the net tempo is shown in the [status bar](#).

Modulator tracks

A modulator track is simply a track that alters the behavior of another track, in a process called [modulation](#). To make a modulator track operational, you must connect it to one or more target tracks, via the [Modulations bar](#) or the Track menu's [Modulation](#) command. In most cases, you'll want to prevent a modulator track from outputting MIDI messages, which you can do by changing its [track type](#) to "Modulator." A modulator track's [steps](#) typically contain values that only make sense in the context of a particular [modulation type](#), in which case also interpreting those same step values as note velocities or controllers would be unhelpful. One possible exception is a [note track](#) gating another note track via [mute modulation](#), in order to prevent collisions between them. But in general, creating separate "Modulator" type tracks is the most flexible method of setting up modulation sources.

It's recommended to give modulator tracks a small negative time [offset](#), so that they're slightly ahead of their targets. This avoids races and track order dependencies.

Internal tracks

An internal track alters the sequencer's behavior. Internal tracks are analogous to [controller tracks](#), but instead of transmitting control change messages to the [output device](#), they're handled within the sequencer.

An internal track accesses an [internal controller](#). Up to 128 internal controllers are possible, though relatively few are currently implemented. Which internal controller is accessed depends on the [control number](#), which is specified via the track's [note](#) property. Some of the internal controllers [emulate](#) existing MIDI controllers, and in such cases the same control number is used. For example, one of the internal controllers implements sustain, and like the MIDI sustain controller, it uses control number 64. The supported internal controllers are shown in the table below.

Internal tracks are *per-channel*, meaning that each internal track operates on a single MIDI channel, selected via the track's [channel](#) property. To operate on multiple channels, create multiple internal tracks, one for each of the desired channels.

Internal controllers

Control Number	Name	Function
62	Duplicate Notes	Sets whether the channel prevents duplicate notes . Track steps are interpreted as unsigned values. A value of zero selects Allow, and any other value selects Prevent. This controller overrides the Duplicates setting in the Channels bar .
63	Note Overlap	Sets the channel's method of correcting note overlaps . Track steps are interpreted as unsigned values. A value of zero selects Split, and any other value selects Merge. This controller overrides the Overlaps setting in the Channels bar. This controller only works if the note overlap master property is set to Prevent, otherwise it has no effect.
64	Sustain	Sets the channel's sustain switch. Track steps are interpreted as unsigned values. A value of zero turns sustain off, and any other value turns it on. While sustain is on, all active notes on the channel are sustained, regardless of their durations. This applies both to notes that are already active when sustain is switched on, and to notes that are triggered while sustain is on. When sustain is switched off, notes with expired durations stop immediately, but notes with duration remaining continue until their durations expire.
66	Sostenuto	Sets the channel's sostenuto switch. Track steps are interpreted as unsigned values. A value of zero turns sostenuto off, and any other value turns it on. Sostenuto is similar to sustain, except that it only sustains notes that are already active when sostenuto is switched on. Notes that are triggered while sostenuto is on are <i>not</i> sustained. Sostenuto is typically used to hold a note or chord, and then play melodies over the held note or chord. When sostenuto is switched off, notes with expired durations stop immediately, but notes with duration remaining continue until their durations expire. Sostenuto can be combined with sustain.
102	Queue Mod Reset	Resets all queue modulators that target the channel. The target track's channel is what matters; the modulator track's channel is irrelevant. The internal controller track's steps are interpreted as unsigned values. When the internal controller transitions from zero to any non-zero value, all queue modulators targeting the channel jump to their first step, regardless of whether they're currently muted or unmuted.
123	All Notes Off	Turns off all notes on the channel. Track steps are interpreted as unsigned values. Transitioning from zero to any non-zero value stops all notes immediately, regardless of their durations. Notes that are being

	held by the sustain or sostenuto internal controllers are also stopped immediately.
--	---

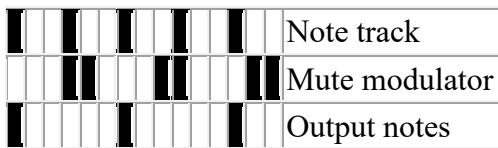
Internal versus external controllers

Why are internal sustain and sostenuto controllers useful? One reason is that the output device may not implement these controllers correctly or at all. Another reason is that the internal controllers modify the actual note durations, both during [playback](#) and in the [MIDI export](#). Because internal sustain and sostenuto are applied to the note durations, the exported MIDI file will play as intended even if the playback device doesn't support these controllers. The [Piano bar](#) also shows the actual note durations resulting from internal sustain and sostenuto, which it doesn't do for the corresponding external controllers.

Modulation

Mute modulation

In mute [modulation](#), one track mutes (or "gates") another. The effect is similar to manually muting and unmuting a track, except that it's automated. As the target track loops, each step is processed sequentially. If that step contains the start of a note, the current step in the modulator track is examined, and if it's zero, the note is played, otherwise the note isn't played. In other words, the modulator track's step is interpreted as a mute switch for the target track. In the example below, a note track with a length of 3 steps is muted by a modulator track with a length of 5 steps. Since the pattern repeats after 15 steps (3×5), only 15 steps are shown.



A modulator can mute multiple targets, all potentially having different lengths. The reverse is also true: a target can be muted by multiple modulators, all potentially having different lengths. In the latter case, the target is muted if any of its mute modulators are non-zero.

Juxtaposed [polymeter](#) tracks can produce overly dense sequences, particularly if the tracks contain little empty space. Mute modulation directly addresses this problem. Periodically muting the juxtaposed tracks adds more space to the composition. Mute modulation is analogous to viewing a pattern through the gaps in a stencil.

Mute modulation creates repeating "windows" of time. The target tracks play only within their time window, leaving periodic unoccupied space or "holes" in the composition. The holes can be left empty, or occupied by something else, e.g. a different group of target tracks, mute modulated by the opposite pattern. This leads to "call and answer" effects. This method isn't limited to

alternating between two groups, for example one could alternate between three or more groups, or partially overlap the groups instead of strictly alternating between them.

Note modulation

In note [modulation](#), a track's note is altered by one or more tracks. A note modulator track is interpreted as an array of [signed](#) offsets (in semitones), which are added to the target track's default note. For example, suppose the target track has a default note of C. If the corresponding step in the note modulator track contains zero, C is output as usual, but if the modulator step contains 3, the target track outputs Eb instead, because $C + 3 \text{ semitones} = Eb$.

If a target track has multiple note modulators, they're summed. This means that a pair of note modulations can potentially cancel each other out, for example if one modulator produces 3 and the other produces -3, the net offset is zero, and the target note remains unchanged.

It's often useful to know what notes a given set of note modulators will produce, and whether those notes form a subset of any common scales. For details, see the [Table of Addends and Their Associated Pitch Sets](#).

When many note modulators are summed, the resulting pitch can potentially vary over a large range. This can be addressed by limiting the output to an octave. For more on this, see [range](#) and [range modulation](#).

Note modulation usually targets [note tracks](#), but it can also target [scale modulators](#) and [chord modulators](#).

Velocity modulation

In velocity [modulation](#), a track's note [velocities](#) are altered by one or more modulator tracks. A velocity modulator track is interpreted as an array of [signed](#) offsets, which are added to the note velocities. If a target track has multiple velocity modulators, they're summed. If the offset becomes too large or too small the velocity [clips](#), but otherwise the target's original dynamics are preserved. If negative velocity modulation causes a note's net velocity to be zero, a [note off](#) is output.

It's interesting to combine velocity modulations that operate on different time scales, for example a longer modulator that applies a gradual crescendo could be combined with a shorter modulator that applies "ripple" to the velocities.

Velocity modulation can also be applied to a [controller track](#), in which case it offsets the controller value instead of the note velocity.

Duration modulation

In duration [modulation](#), a track's note [durations](#) are altered by one or more modulator tracks. A duration modulator track is interpreted as an array of [signed](#) offsets (in MIDI [ticks](#)), which are added to the note durations. If a target track has multiple duration modulators, they're summed. The duration is prevented from being less than one.

Decreasing the duration causes the notes to become more staccato and percussive, whereas increasing it causes the notes to become more legato and potentially [overlapped](#). Duration modulation is more effective if the output instrument has a rapid attack and decay.

Range modulation

In range [modulation](#), a track's notes are transposed by one or more octaves as needed to keep them within a specified range. The size of the range is typically one octave, though other sizes are possible. For range modulation to be effective, the target track's [range type](#) must be set to a value other than None.

The range is movable, and its bottom note is specified by the target track's [Range Start](#) property, which can also be altered by one or more modulator tracks. A range modulator track is interpreted as an array of [signed](#) offsets in semitones, which are added to the target track's range start note. If a target track has multiple range modulators, they're summed.

Because the transposition is by octaves only, the notes remain in the same scale and key; only their octaves are affected. If several tracks forming a chord are range modulated together, the chord cycles through its possible inversions.

Position modulation

In position [modulation](#), a track's step array is temporarily [rotated](#) left or right by one or more steps. Rotation is similar to shifting, except that it's lossless, because any shifted elements that "fall off" one side wrap around to the other side. For example, rotating 1234 right by one step yields 4123, because the 4 wraps around to the left.

A position modulator track is interpreted as an array of [signed](#) offsets (in steps), which determine the amount of rotation. A positive value rotates to the right (later in time), a negative value rotates to the left (earlier in time), and a rotation of zero has no effect. If a target track has multiple position modulators, they're summed.

Position modulation is a powerful technique for generating variations. It has a downside however, which is that it can easily cause multiple overlapping instances of the same note on the same channel. On some instruments this may produce undesired artifacts, such as notes being cut off prematurely. For this reason, you may want to prevent [note overlaps](#).

Tempo modulation

Tempo [modulation](#) is when a [tempo track](#) is modulated by one or more modulator tracks having a modulation type of Tempo. The modulator tracks also must have non-zero Duration, in other words each modulator has its own range of tempo scaling. Initially you can just set them all to 100. Like the tempo track, a tempo modulator can have any function in its steps. The actual tempo at any time is the product of the base tempo, the tempo track, and all of its unmuted tempo modulators. The idea is that tempo can change in [polymeter](#), for example changing slowly in one meter, while changing faster in some other meter.

During [playback](#), the net tempo is shown in the [status bar](#).

Scale modulation

A scale is broadly defined as a set of notes. A scale can contain any notes in any order, with or without duplicates, spanning any number of octaves. Only [note tracks](#) support scale modulation.

When a note track has one or more unmuted scale [modulators](#), it doesn't compute its output note in the usual way. Instead, it builds a temporary scale, consisting of the current step values of each of the scale modulators. Each step value is a [signed](#) offset in semitones from the target track's [base note](#). The order of the scale modulators within the track's modulation list matters, because the step values are appended to the scale in that order. Muted scale modulators are excluded.

After the scale is built, one its tones is picked via [index modulation](#). A track that uses a scale should also have at least one index modulator. The index modulator's current step value is interpreted as a zero-based index into the scale. If the index exceeds the range of the scale in either direction, it's [wrapped around](#) as needed. If a track has multiple index modulators, they're summed.

For example, suppose the target track has seven unmuted scale modulators, and their current steps contain 0, 2, 4, 5, 7, 9, and 11 respectively. Assuming the target track's base note is C, this constitutes a C major scale. If the target track's index modulator is zero, the first scale tone (C in this case) is picked; if the index modulator is one, the second scale tone (D in this case) is picked, and so on.

Scale Modulation							
Scale Tone	0	2	4	5	7	9	11
Index	0	1	2	3	4	5	6
Note	C	D	E	F	G	A	B

The [track range](#) also works differently in this case. The range is applied to the entire scale *before* the indexing operation, i.e. before a scale tone is picked.

The target track can still be [note modulated](#). In this case the note shift is added to all of the scale tones, before range modulation and indexing. This is useful for transposition.

Scale modulators can also have note modulators, which is useful for periodically altering one or more scale tones. Note modulators are signed offsets in semitones, and they're applied first as the current scale is being built. Scales can also be affected by [chords](#) and [voicings](#).

Chord modulation

A chord is a subset of a [scale](#). A chord can contain any number of tones, up to and including all of the tones of the scale. Chords are created by modulating a note track with scale, chord, and index modulators. Voicing modulators may optionally be added also. In the same way that scale modulators build a current scale consisting of notes, chord modulators build a current chord consisting of scale tones. A chord modulator's step values are [signed](#) zero-based indexes into the current scale, [wrapped](#) as needed to avoid range errors. The order of the chord modulators within the track's modulation list matters, because the step values are appended to the chord in that order. Muted chord modulators are excluded.

Chord modulation occurs when a track with unmuted scale modulators also has unmuted chord modulators. In this case the track's [index modulation](#) accesses the current chord instead of the current scale. This adds an extra level of mapping. The index modulation picks a chord tone, the chord tone maps to a scale tone, and the scale tone maps to a note. Suppose the track has four chord modulators, which contain 0, 2, 4, and 6 respectively. These correspond to the first, third, fifth, and seventh notes of the current scale. Assuming the scale is C major, this chord modulation produces a C major seventh chord.

For a scale, an index value of one accesses the second scale tone, which in this example contains two, which maps to the note D.

Scale modulation						
Scale Tone	0	2	4	5	7	9
Index	0	1	2	3	4	5
Note	C	D	E	F	G	A

For a chord, an index value of one accesses the second chord tone, which in this example contains the value two. Two is then used as an index into the current scale (see above), accessing the third scale tone, which contains four, which maps to the note E.

Chord Modulation				
Scale Tone	0	4	7	11
Chord Tone	0	2	4	6
Index	0	1	2	3
Note	C	E	G	B

Chord modulation changes the meaning of index modulation, such that chord tones are indexed instead of scale tones. Notice that three things are being distinguished: the subset of notes (the scale), the subset of scale tones (the chord), and which chord tone is picked (the index).

If a track uses chords, [voicings](#) are applied to the chord instead of the scale. Chord modulators can have note modulators, and these are applied first as the current chord is being built. Bear in mind that altering a chord modulator alters it in scale tones, not semitones.

Index modulation

Index modulation picks a note from the current [scale](#) or [chord](#). Only [note tracks](#) support index modulation. If a chord exists, the index accesses the chord's tones, otherwise if a scale exists, the index accesses the scale's tones. If neither exists, index modulation has no effect. If a track has multiple unmuted index modulators, they're summed to produce a single index value.

The index is zero-based, meaning zero accesses the first tone, one accesses the second tone, and so on. If the index exceeds the range of the chord or scale, it [wraps around](#). For example, if the scale has seven tones, they correspond to index values 0 through 6. If the index has a value of 7, it wraps around and accesses the first tone. Similarly, if the index has a value of -1, it wraps around in the other direction and accesses the last tone.

The mapping from index values to scale or chord tones may be affected by the order in which the scale or chord modulations are listed for the target track. For this reason, it's advisable to keep scale and chord modulations in ascending sequential order.

Duplicate notes

A duplicate note occurs when an instrument plays the same MIDI note value twice in a row. Duplicate notes may be perceived as flaws, especially in arpeggios. If you're using [index modulation](#) to pick scale or chord tones, you have the option to prevent duplicate notes automatically, by picking the tones in a slightly different order.

Prevention adds a retry to the indexing. The picked tone is compared to the relevant channel's previous note, and if they match, the index value is incremented by one, and the indexing is then retried. This prevents a duplicate note provided 1) the scale or chord contains at least two tones, and 2) all tones in the scale or chord are unique.

Duplicate note prevention is enabled on a per-channel basis, via the Duplicates column in the [channels bar](#). To prevent duplicate notes, change the relevant channel's drop list value from "Allow" to "Prevent". Prevention can also be controlled via [MIDI mapping](#) or an [internal controller](#).

Voicing modulation

A [scale](#) or [chord](#) can be sorted and optionally widened, by applying one or more voicing modulators to the target track. The voicing modulator's steps are interpreted as follows: If the step value is one or more, it selects a tone to be shifted down an octave (dropped). The numbering scheme is one-based and starts from the highest-pitched tone, i.e. a value of 1 drops the highest tone, a value of 2 drops the second-highest tone, and so on, as shown below. Negative values cause the corresponding tone (determined by taking the absolute value) to be *raised* instead of dropped.

N	The Nth-highest tone is shifted down an octave.
3	The third-highest tone is shifted down an octave.
2	The second-highest tone is shifted down an octave.
1	The highest tone is shifted down an octave.
0	The scale or chord is sorted, but no tones are shifted.
-1	The highest tone is shifted up an octave.
-2	The second-highest tone is shifted up an octave.
-3	The third-highest tone is shifted up an octave.
-N	The Nth-highest tone is shifted up an octave.

Multiple voices can be dropped by applying multiple voicing modulators to the track. For example, to create a drop 2/4 voicing of a tetrachord, apply two voicing modulators, one outputting 2 and the other outputting 4. Many other combinations are possible.

To shift a tone two octaves instead of one, specify the same voicing modulation twice in the track's modulation list. If a voicing modulator exceeds the range of the current scale or chord, it's ignored.

After the voicing is applied, the tones are sorted again, and only then is a tone picked via [index modulation](#).

Offset modulation

Offset [modulation](#) changes the time offset of a target track. The supported target types are [note tracks](#), [controller tracks](#) and modulator tracks. For note and controller tracks, an offset modulator's steps are interpreted as [unsigned](#) delays in MIDI [ticks](#). By default, events can only be made later, not earlier, but you can work around this by setting the target track's [offset](#) to a negative value. For example if the target track's offset is -64, that gives its offset modulation an effective range of -64 to +63, in other words from 64 ticks early to 63 ticks late.

Offset modulation changes the rhythmic feel of the target track, periodically speeding it up and slowing it down relative to the beat. It can be used to create rolls on drums, or to emulate the feel changes of a live performer. For best results, use a smooth function such as a triangle or sine wave. Sharp changes in offset modulation may alter the order of events. Offset modulation can cause [notes to overlap](#).

Modulator tracks can also be offset modulated, but in this case the offset modulator's steps are interpreted as signed values, allowing offsets in the range of -64 to $+63$. Positive values delay the target; negative values hasten it. Offset modulation is useful for time-shifting a group of modulators that have different [quants](#). Offset modulation is similar to [position modulation](#), except that the shift is specified in ticks instead of steps.

Queue modulation

Queue [modulation](#) assigns a sequence of pitches to successive notes in order, regardless of note timing. Only [note tracks](#) support queue modulation. Unlike other modulation types, a queue modulator doesn't continuously cycle through its steps during playback. A queue modulator only advances when its target track plays a note. Until the target track plays its next note, the queue modulator remains frozen on its current [step](#) for as long as it takes.

A queue modulator is analogous to a line of customers waiting to pay. The customers stay in order, and the line only moves when the clerk signals for the next customer. The customers are the pitches and the clerk is the note track. But the clerk returns each customer to the back of the line, meaning the pitch sequence loops.

A queue modulator's [quant](#) is irrelevant because the modulator's stepping isn't driven by time. Any quant will do. The [Track Steps](#) view and the [Step Values bar](#) correctly report a queue modulator's current position, but other views may not.

Queue modulation decouples pitch from rhythm. This allows a melody to be constructed by applying a sequence of pitches to an unrelated or unquantized rhythm. Even if the rhythm is erratic due to rests, triplets, or syncopation, pitches won't be skipped.

If the target note track uses [scale](#) or [chord modulation](#), each queue modulator step is a zero-based signed index that selects a tone from the current scale or chord. If [index modulation](#) is also used, the queue and index modulators are summed to produce the final index value. In any case the index value is wrapped as needed to stay within the scale or chord.

If no scale is defined, each queue modulator step is a zero-based signed offset added to the [track's note](#). If [note modulation](#) is also used, the queue and note modulators are summed to produce the final note value.

A note track can have only one active queue modulator. Multiple queue modulators can target a track without conflict provided only one of them is unmuted at a time. If a note track has multiple unmuted queue modulators, only the last one is used and the others have no effect.

A queue modulator can target multiple note tracks at once, but this should be considered carefully. If multiple targets are unmuted, they compete for the modulator's pitches, and this potentially alters the sequence, which may or may not be desirable. If only one of the targets is unmuted at a time, there's no issue and the pitch sequence is preserved.

To repeat a fixed melody, the queue modulator should have one step for every note in the target track. Otherwise, the melody may vary over time due to phasing. In the fixed melody case, a queue modulator can lose synchronization with its target note track. This can happen if the note track is muted during the middle of its cycle, for example. To enforce synchronization, a queue modulator can be periodically reset to its start using [internal controller](#) 102 (Queue Mod Reset).

Queue modulators can be targeted by [mute modulation](#) and [position modulation](#), but they don't support [offset modulation](#).

Position modulation causes a queue modulator to skip forward or backward in its sequence by the specified number of steps, but the effect is temporary and non-destructive. The target track may receive a different queue modulator step than it otherwise would have, but the queue modulator's internal position is unaffected.

Views

Views

The application supports three top-level views, as shown below. Each view is optimized for a different stage of the [polymeter](#) composing process.

View	Purpose
Track	Use for composing tracks . This is the most feature-rich view, with an entire chapter devoted to it.
Song	Use for arranging tracks into a song , or refining an arrangement that was recorded while performing in Live view.
Live	Use for live-arranging or performing with tracks. The performance can be recorded as a sequence of dubs , and the resulting arrangement can be refined in Song view.

To switch between the different views, use the View menu's Track, Song, and Live commands or their keyboard shortcuts. Normally only one view is shown at a time. To show multiple views at once, [split](#) the view area into two or more tab groups.

Song Mode

Which view is shown can affect [playback](#). If song view is shown, the sequencer operates in *song mode*, which means that tracks are automatically muted and unmuted to reflect the [arrangement](#). If [multiple views](#) are shown, which view is *active* determines whether song mode is in effect. When you enter song mode, the [track mutes](#) snap to appropriate states based on your [position](#) within the song.

Track View

Track View

The track [view](#) is the main interface for editing [tracks](#), and contains one row for each track. The view is split vertically into two [panes](#). The left pane contains the [track grid](#), and the right pane contains the [track mutes](#) and [track steps](#). The right pane can also be split horizontally to show the optional [velocities pane](#), which lets you edit the step values graphically by dragging bars. To display the track view, use the View menu's [Track](#) command.

Quick help

- To alter the pattern of a track, left-click in its [steps](#), which are shown as outlined rectangles. The steps area is to the right of the vertical splitter bar and [track mutes](#), and looks similar to graph paper.
- To mute or unmute a track, left-click its mute button, which is the green or red rectangle immediately to the right of the vertical splitter bar. Red means the track is muted.
- To change the track's loop length, edit its [Length](#).
- To change what MIDI channel the track outputs, edit its [Channel](#).
- To change what MIDI note the track outputs, edit its [Note](#).
- To change the track's time unit, edit its [Quant](#). The default is 30 ticks, which is a sixteenth note at the default [timebase](#) of 120 ticks per quarter note.
- The track's [Offset](#) and [Swing](#) are in ticks; they can be positive or negative.
- [Velocity](#) is an offset that's added to the velocities of the track's notes; it can be positive or negative. It's also applied to [controller tracks](#).
- [Duration](#) is also an offset. Negative values make the notes more staccato. Positive values make them more legato, in which case they may [overlap](#).
- To [select](#) a track, left-click on its track number. Ctrl+left-click and Shift+left-click both do what you'd expect (select multiple tracks and select a range of tracks).
- You can copy/cut/paste/delete/insert tracks, using the [standard editing commands](#), or reorder tracks via dragging.
- To edit a field, left-click it. When you're finished editing, press Enter or Tab, or left-click anywhere outside of the field. Pressing Tab or Shift-Tab iterates through the fields. You can also end an edit by pressing Ctrl+Up or Ctrl+Down, in which case the edit focus moves up or down a track. For details, see [grid controls](#).
- To set the same value for multiple tracks, first select the tracks, and then edit the desired field in any one of the tracks; when you end the edit, the new value is applied to all the selected tracks.
- Columns can be resized or reordered via the column header. To reset the columns to their default width and order, right-click in the column header and select Reset in the context menu.

Panes

Certain windows may be horizontally or vertically split into panes, separated by a moveable border called a *splitter bar*. To resize a pane, drag the splitter bar that separates it from its neighbor, in other words left-click on the splitter bar, and while keeping the left mouse button pressed, move the split until the pane is the desired size. While the pointer is positioned over a

splitter bar, the cursor changes to indicate that resizing is possible. The borders that separate [docking windows](#) from each other, and from the document area, behave similarly to splitter bars.

Track Grid

This [grid control](#) constitutes the left pane of the [track view](#), and contains one row for each [track](#). Most [track properties](#) are editable via this grid. The grid's columns are described below.

Number

This column is the track number, which is read-only. The column caption is a number sign (#). Left-click in this column to [select](#) tracks. To move a track, left-click in this column, and while keeping the left mouse button pressed, drag the track to the desired location, and then release the left mouse button. Multiple [selected tracks](#) can be moved in the same manner.

Name

This is the track's name. Appropriately naming tracks makes compositions easier to manage. If you name multiple tracks at once, and the [Unique track names](#) option is enabled, sequential numbers are appended to the track names to make them unique. Polymeter supports [Find](#) and [Replace](#) for track names.

Type

This property selects the track's [type](#), which determines what the track outputs, and how its [steps](#) are interpreted.

Channel

This property selects which MIDI channel the track outputs messages on.

Note

This property has different meanings depending on the [track type](#).

Type	Meaning
Note	Which note is output. The octave may be altered by the track's range , and the note may be altered via note modulation . The note may also be altered via a combination of scale, chord, index, and voicing modulation.
Key Aftertouch	For which note aftertouch is sent.
Control	Which continuous controller number is sent.
Patch	Unused.

Channel Aftertouch	Unused.
Wheel	Unused.
Tempo	Unused.
Modulator	Unused.
Internal	Which internal controller is accessed.

For [note tracks](#), the note property is normally displayed as a MIDI note name instead of a number (for example C4 instead of 60), however this behavior can be disabled by setting the [Show Note Names](#) option to false, and the note value can be entered numerically regardless of this option. For channel 10 only, the note property is normally displayed as a General MIDI drum name, using a drop list instead of an edit box, but this can be disabled by setting [Show General MIDI Names](#) to false.

Length

This is the length of the track, in steps. It can't be less than one, nor can it exceed the [maximum track length](#). Increasing the length appends empty steps to the end of the track. Decreasing the length removes existing steps from the end of the track. To set the length visually instead of numerically, use the Track menu's [Length](#) command.

Quant

This is the duration of each track step, in [ticks](#). If ticks are inconvenient, you can alternatively specify the duration as a fraction of a whole note, for example you can type 1/16 for a sixteenth note and this is automatically converted to the equivalent number of ticks. Quant values can also optionally be [shown as fractions](#). The table below gives common Quant values for the default [timebase](#) (120 ticks per quarter note).

Fraction	Ticks
1/1	480
1/2	240
1/3	160
1/4	120
1/6	80
1/8	60
1/12	40
1/16	30
1/24	20
1/32	15
1/48	10

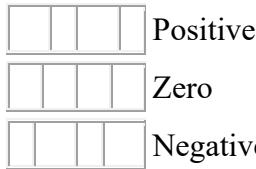
Changing a track's quant causes it to play at a different speed. You can compensate for this by [stretching](#) the track. Multiplying the track's quant by its [length](#) gives the duration of one cycle through the track, also known as its [period](#).

Offset

This is the track's time offset, in [ticks](#). Negative values make the track earlier; positive values make it later. The offset can also be altered via [offset modulation](#).

Swing

This is the track's amount of swing, in [ticks](#). Swing is a time offset that's applied to even-numbered steps. A positive value delays the even-numbered steps, and lengthens the odd-numbered steps to compensate. A negative value hastens the even-numbered steps, and shortens the odd-numbered steps to compensate. Swing is usually positive, and can't exceed the track's [Quant](#). Setting swing to one-third of the track's Quant results in triplet feel. Swing affects all [track types](#) except Modulator.



Velocity

This is the track's velocity offset. It's a signed value that's added to the velocity values obtained from the track steps. The result is clamped so that it can't exceed the valid range of MIDI data, which can potentially result in [clipping](#). Velocity affects all [track types](#) except Tempo and Modulator. The velocity can also be altered via [velocity modulation](#). If a note's net velocity is zero, a [note off](#) command is sent.

Duration

This is the track's duration offset, in [ticks](#). For note tracks, it's a signed value that's added to the durations of all notes played by the track. Positive values make the note longer (more legato), and negative values make it shorter (more staccato). Positive values may cause notes to overlap, in which case collisions may occur; see [note overlap](#). The duration is clamped to prevent it from being less than one. The duration can also be altered via [duration modulation](#).

For tempo tracks and tempo modulators, Duration is repurposed to be the range of the tempo change, as a percentage; see [tempo tracks](#).

Range Type

This is the type of range to be applied to the track. Range only affects [note tracks](#). The types are listed below.

Type	Meaning
None	No range is applied, and any range modulators targeting this track have no effect.
Octave	The output note is forced to stay within a one-octave range, the bottom of which is determined by the Range Start property. The note is shifted by as many octaves as necessary to keep it within the specified range. For chords and arpeggios, this produces inversions.
Dual	Similar to Octave except that it creates a two-octave range, and assigns the note to the upper or lower octave depending on its pitch. The Range Start property determines the start of the upper octave. If the note is below the Range Start value, it's shifted to the lower octave, otherwise it's shifted to the upper octave.

The range's start note can be altered via [range modulation](#), but only if range type isn't None. Range can be used to limit the width of [note modulation](#). Range behaves differently when [scale modulation](#) and/or [chord modulation](#) are used.

Range Start

This is the starting note of the range to apply to the track. It only affects note tracks, and only has an effect if [Range Type](#) isn't None. Range start can be altered via [range modulation](#).

Track Mutes

Each track has a mute switch, which determines whether the track is enabled or disabled. Muting applies to all [track types](#), and can be modulated via [mute modulation](#). Muting a [note track](#) only prevents it from starting *new* notes; any notes that were already in progress when the track was muted continue until they finish their durations.

Muting is done via the track mute buttons, which are located immediately to the right of the vertical divider that separates the two panes of the [track view](#). If a track is muted, its mute button is red, otherwise it's green. To toggle the mute state of a single track (from unmuted to muted or vice versa), simply left-click the track's mute button. To toggle the mute states of multiple tracks, first [select](#) them, and then left-click anywhere within the mute buttons area, or use the [Track/Mute](#) command.

To unconditionally mute the selected tracks, regardless of their current mute state, left-click anywhere within the mute buttons area while pressing the Ctrl key.

Track Steps

The track steps area constitutes most of the [track view](#)'s right pane. For each track, a row of one or more equally sized rectangles is displayed, to the right of the track's [mute button](#). These rectangles correspond to the track's [steps](#), and their sizes represent the step duration, also known as the track's [Quant](#).

Left-clicking on a rectangle sets or clears the corresponding step: the step value toggles between zero and the [default velocity](#). For more nuanced control of step values, use the [velocities pane](#) or the [Step Values bar](#). For [note tracks](#) only, steps can be [tied](#) together to form notes longer than one step.

Right-clicking on a rectangle selects the corresponding step. A contiguous range of steps can be selected by right-clicking the range's first step, and then while keeping the right button pressed, moving the pointer over the range's last step; for details, see [step selection](#).

Empty (zero) steps are white, whereas occupied (non-zero) steps are black, with the following exceptions: selected steps are blueish, and the current position is greenish if the track is unmuted, or reddish if the track is muted. Each track indicates its current position by highlighting one of its steps, unless the [Show Current Position](#) option is false. For note tracks only, a tied step fills its entire rectangle, whereas an untied step only fills the left half of its rectangle.

The steps area can be zoomed horizontally, using the [Zoom In](#), [Zoom Out](#), and [Zoom Reset](#) commands. You can also zoom by spinning the mouse wheel while pressing the Ctrl key, and this centers the zooming at the cursor position. The width of the steps area is determined by the track with the longest [period](#). The ruler at the top of the steps area is marked in beats and ticks; see [meter](#) for details. Left-clicking in the ruler jumps to the corresponding [song position](#). Right-clicking in the ruler clears the current step selection if any.

Quick help

- Left-click a step to create a note or event. Left-click the step again to empty it.
- For [note tracks](#) only, to create a [tied note](#), use Shift+left-click. Multiple adjacent steps can be tied together to create long notes, but the track must contain at least one empty or untied step, otherwise it will be silent.
- You can specify whether tied or untied notes are created by default, using the [Track/Tie Notes](#) command, which is also available from the [toolbar](#).
- To edit step values visually as bars, use the [velocities pane](#); to edit them numerically, use the [Step Values bar](#).
- The right mouse button behaves differently than usual. In the steps area, right-clicking and dragging creates a [step selection](#).
- Rectangular step selections can be cut, copied, pasted, deleted, and inserted, using the [standard editing commands](#).
- Right-clicking and dragging vertically over the mute buttons selects the corresponding tracks. While tracks are selected, left-clicking within the mute area toggles the selected track mutes, and Ctrl+left-click within the mute area unconditionally mutes the selected tracks, regardless of their current state.
- To change the zoom in the time axis, use the [Zoom In](#) and [Zoom Out](#) commands, or spin the mouse wheel while pressing the Ctrl key.
- Left-clicking in the ruler at the top of the track steps area jumps to the corresponding [song position](#).

Track Selection

Many commands operate on selected [tracks](#). You can select tracks by selecting rows in the [track grid](#), or by right-clicking [track mute](#) buttons. You can also use one of the [track selection commands](#).

To use the first method, left-click in the track number column, which has a number sign (#) in the list header. This is usually the grid's leftmost column, unless you've reordered the columns. You can also select multiple rows, and select via the keyboard; for more help with selecting, see [list controls](#).

You can also select tracks via their mute buttons, which are located immediately to the right of the vertical divider that separates the two panes of the [track view](#).

- To toggle the selection of an individual track, right-click its mute button.
- To toggle the selection of a range of adjacent tracks, right-click over the first track's mute button, and while keeping the right mouse button pressed, move the pointer vertically until it's over the last track's mute button.
- To clear the track selection, use the [Deselect](#) command, or right-click within the ruler at the top of the [track steps](#) area.

Some commands can operate on either a track selection or a [step selection](#). For such commands, if a step selection exists, it has priority over the track selection, meaning the track selection is ignored in that case.

Track Selection Commands

Select All	Select all tracks
Deselect	Deselect all tracks
Select Unmuted	Select all unmuted tracks

Step Selection

Many commands operate on a contiguous range of selected [steps](#). Steps are selected via right button dragging within the [track steps](#) area: right-click on the initial step, and while keeping the right mouse button pressed, move the pointer until it's over the final step, and then release the right mouse button. To clear an existing step selection, right-click within the position ruler at the top of the steps pane, or press the Esc key.

A step selection may be entirely within one track, or may span multiple tracks, in other words it can be one-dimensional or two-dimensional. A multi-track step selection is created using a diagonal motion: right-click one of its corners, and then while keeping the right mouse button pressed, move the pointer to the diagonally opposite corner, and then release the right mouse button.

An existing selection can be extended or trimmed. While holding down the Shift key, right-click the step that should be the new corner or edge of the selection. If that step is within the selection,

the selection is trimmed, otherwise it's extended. This feature is useful for precisely selecting a large area: First do a rough cut by drag selecting while zoomed out, and then refine it by zooming into the selection's borders and extending or trimming them as needed.

Some commands only accept a *rectangular* step selection. A multi-track step selection is rectangular if all of its rows contain the same number of steps. This isn't necessarily the case if the tracks involved have different lengths, particularly if you select near the end of such tracks. If the step selection isn't rectangular, and you try to execute a command that requires it to be rectangular, you'll get the error message "Invalid step selection." For example, the [standard editing commands](#) only accept a rectangular step selection.

Some commands can operate on either a [track selection](#) or a step selection. For such commands, if a step selection exists, it has priority over the track selection, meaning the track selection is ignored in that case. This can potentially cause confusion, particularly if the steps area is scrolled so that the step selection is no longer visible.

Velocities Pane

The velocities pane displays the velocities of notes within the [selected tracks](#), and also lets you edit the velocities. The velocities pane isn't restricted only to [note tracks](#); it can also be used to view and edit the [step values](#) of other [track types](#) including [controller](#), [tempo](#), and [modulator](#) tracks. The velocities or step values are represented by vertical bars that can be clicked and dragged. The velocities pane is primarily intended for gestural drawing. If you want to specify exact values, it may be easier to enter them numerically in the [Step Values bar](#).

The velocities pane is optional, and when visible, it appears at the bottom of the [track steps](#) area within the [track view](#). To show or hide the velocities pane, use the [View/Velocities](#) command. To change the height of the pane, drag the [splitter bar](#) that separates the pane from the track steps.

- At least one track must be selected, otherwise the pane is empty. Multiple tracks can be selected, in which case their events are combined.
- To edit a velocity, left-click on its bar at the point where you want it to be, or left-click and drag the bar up or down until it's where you want it.
- To edit multiple velocities, left-click and drag the cursor over the bars you want to change.

The pane supports a variety of drawing functions, which are accessed by pressing modifier keys while dragging. For more precise control, you may prefer to use the [Track/Fill](#) command which offers similar functions but in parameterized form.

- To make all the velocities the same, hold down the Shift key while dragging.
- To make the velocities fit a straight line, hold down the Ctrl key while dragging.
- To make the velocities fit a triangle wave, hold down the Ctrl and Alt keys while dragging diagonally.

- To make the velocities fit a sine wave, hold down the Shift and Alt keys while dragging diagonally.

When using the triangle or sine functions, visualize a rectangle that contains the desired wave, and draw one of the diagonals of that rectangle. The triangle and sine functions are both sensitive to the drawing direction. Drawing from top to bottom inverts the wave vertically. Drawing from right to left shifts the wave's phase by 90°.

The pane also features tool tips that show step values. To learn a bar's exact value, hover the cursor (hold it stationary) over the bar for at least half a second, and its value will appear in a tool tip. The tool tip is also shown continuously while dragging is in progress.

The tool tip can show either [signed or unsigned](#) step values. To switch between the two, left-click the small button on the left border of the velocities pane. The button indicates the step value at the vertical center of the pane. If it shows 64, unsigned tooltips are selected, whereas if it shows 0, signed tooltips are selected.

The minimum height of a velocity bar is one for [note tracks](#), and zero for all other [track types](#). Even though note velocity bars can't be dragged below one, you can still create notes with velocities of zero, by using a negative [track velocity](#) offset to shift the note velocities down. For example, within a note track, you could create a [note off](#) by setting a note's velocity to one, and also setting the track's velocity offset to -1.

Song View

The song [view](#) is for arranging your [tracks](#). The arrangement is also known as a [song](#), and it consists of a sequence of [dubs](#), each of which mutes or unmutes one more tracks. To display the song view, use the View menu's [Song](#) command.

Song view contains one row for each track, and is vertically split into two [panes](#). The left pane contains the track names, and the right pane contains the [song grid](#), explained below. To change the proportions of the panes, drag the splitter bar that separates them.

Instead of creating an arrangement entirely via editing in song view, you may find it more intuitive to first [record](#) a rough arrangement in [live view](#), and then refine it in song view. For complex arrangements, you can repeat the record/edit cycle for each section, instead of trying to get the entire arrangement right in one take.

Song Grid

The song grid is a timeline editor for dubbing. It lets you "paint" with your tracks, by drawing the regions where each track should be audible. Each grid row corresponds to a track, and wherever the row is highlighted in green, the corresponding track is unmuted (audible). Tracks are automatically repeated infinitely, and always maintain correct phase regardless of dubbing.

The vertical axis is tracks, and the horizontal axis is time. The grid's width is determined by the [Song Length](#); to access the time beyond the maximum horizontal scroll position, increase the song length. The grid can be zoomed in the horizontal axis only, via the [Zoom In](#), [Zoom Out](#), and [Zoom Reset](#) commands. You can also zoom by spinning the mouse wheel while pressing the Ctrl key, and this centers the zooming at the cursor position.

A grid row is divided into cells, each of which represent one or more [steps](#) of the corresponding track. The zoom specifies the duration of a grid cell, hence for any given row, zoom determines the relationship between that row's cells and the corresponding track's steps. As you zoom in, more detail becomes visible, because each cell represents fewer steps. Cells that represent at least one non-zero step are marked with a black square. For [tied notes](#), only the note's first step is considered.

The song grid has features in common with the track view's [steps pane](#). For example, left-clicking toggles cells, and a rectangular range of cells can be [drag-selected](#) via the right mouse button, similarly to [step selection](#). The ruler at the top of the grid is marked in beats and ticks, and possibly also in measures, depending on the [meter](#). Left-clicking in the ruler jumps to the corresponding [song position](#). Right-clicking in the ruler clears the current cell selection.

Be careful when editing while zoomed out, because the further out you're zoomed, the more details are hidden. This is most likely to be a problem when extending an existing dub. In such cases you should zoom in and make sure you haven't left unintended gaps. It's also recommended to step through your dubs, using the left and right arrow keys in combination with the Alt key, as this can help you find mistakes.

Quick help

- Each grid cell represents a timespan. A grid cell is highlighted in green if the corresponding track is unmuted (audible) during that cell's timespan.
- To toggle a single cell from muted to unmuted or vice versa, left-click it.
- To mute a cell regardless of its current state, left-click it while pressing the Ctrl key.
- To select a rectangular range of cells, right-click one corner of the range, and while keeping the right mouse button pressed, drag the pointer to the diagonally opposite corner, and then release the right mouse button.
- To toggle multiple adjacent cells, first select them as explained above, and then toggle them by left-clicking anywhere within the grid.
- To mute the selected cells regardless of their current state, keep the Ctrl key pressed while left-clicking.
- To clear a cell selection, right-click in the ruler at the top of the view, or press the Esc key.
- If a cell selection exists, the [standard editing commands](#) apply to cells rather than tracks, provided the grid has the input focus; see below.
- To [select](#) a track, left-click on its name. Ctrl+left-click and Shift+left-click both do what you'd expect (select multiple tracks and select a range of tracks). Note that this transfers the input focus to the track name area, hence subsequent standard editing commands

affect tracks rather than cells, unless you restore the input focus to the grid by clicking within it.

- Zooming changes the duration of grid cells. To see more detail, zoom in; to see the big picture, zoom out.

Cell selection

It's often useful to select a contiguous range of cells in one or more tracks, also known as a cell selection. A cell selection must be rectangular. To make a cell selection, right-click any corner of the desired range, and while keeping the right mouse pressed, drag the pointer to the diagonally opposite corner of the range, and then release the right mouse button. If the pointer reaches an edge of the view while you're dragging, the view is scrolled automatically, and this lets you make a cell selection larger than what would fit on the screen. A selected cell is colored either cyan or yellow, depending on whether the cell is unmuted, as shown below:

Unmuted	Selected	Color
No	No	White
Yes	No	Green
No	Yes	Cyan
Yes	Yes	Yellow

The selected cells can be toggled by left-clicking anywhere in the view, or muted by left-clicking with the Ctrl key down. A cell selection can also be copied, cut, pasted at, inserted at, or deleted via the [standard editing commands](#). When deleting an entire section from an arrangement, or inserting empty space into the middle of an arrangement, make sure your cell selection includes all the tracks.

Empty space can be inserted in two ways. One way is to make a cell selection that's one cell wide and then repeat the insert command as many times as needed. The other way is to find an empty area of the view, select an area of the desired width, copy it, select the upper left corner of the area where the insertion should go, and then paste.

Live View

The live [view](#) makes it easier to perform a live arrangement of your [tracks](#). While it's also possible to perform in [track view](#), live view has advantages, as explained below. To display the live view, use the View menu's [Live](#) command.

- You can [record](#) your live arrangement while you're performing it. Your arrangement decisions are recorded as a sequence of [dubs](#), also known as a [song](#). Recording a live arrangement can be more intuitive than creating an arrangement via editing in [song view](#).
- You can reduce the number of tracks shown in live view by [grouping](#) related tracks together into [parts](#). It's easier to arrange fewer elements, and shortening the track list can

also avoid time-consuming scrolling. You may also be able to avoid scrolling by using a [smaller font](#).

- Live view has a much simpler interface than track view, with less distractions, which helps you concentrate on performing.
- Live view supports a two-stage ready/go process, which lets you prepare a multi-track transition and then execute it, using only one hand, leaving your other hand free to operate a control surface, for example; see below.
- Tracks can't be edited in live view, so you needn't worry about inadvertently modifying your tracks while performing.

Live view features two lists. The left list contains your [presets](#), and the right list contains your tracks and parts. In live view, tracks that belong to a part are hidden; the part's name is listed as a substitute for its member tracks. A part's position in live view is determined by the position of its first member track in track view.

In the track list, muted items are orange, and unmuted items are green. To toggle an item from muted to unmuted or vice versa, left-click it. To mute an item regardless of its current state, left-click it while pressing the Ctrl key.

Items can also be selected via the right mouse button, and then toggled or muted all at once via the left mouse button, in a two-stage process, similarly to the [track mutes](#). This lets you prepare a multi-track transition and then execute it, using only one hand. Selected items are highlighted using dark text on a light background. To toggle the selection of an item, right-click it. To toggle the selection of a group of adjacent items, right-click the group's first item, and then while keeping the right mouse button pressed, drag the pointer to the group's last item. To deselect all items, right-click in the area to the right of the track list. If a selection exists, left-clicking anywhere within the view toggles or mutes the selected items, depending on whether the Ctrl key is down. The selected items can also be soloed, using the solo button at the top of the view.

The presets list works somewhat differently. To apply a preset, left-click the corresponding item. To select a preset, right-click the corresponding item. If a preset is selected, left-clicking anywhere in the view applies the preset. Only one preset can be applied or selected at a time.

Live view shows a progress bar for each track list item. Each progress bar indicates the current position within the corresponding track or part. If a part's member tracks have different [periods](#), its progress bar corresponds to the longest period.

If your arrangement is complex, you may prefer to record each section separately, instead of trying to get all the sections right in one take. To append a new section, move the [song position](#) to the end of the last completed section before you start recording. Your live arrangement can be approximate, because you can refine its transitions later in song view.

It's a good idea to record your rehearsals, in case there's a happy accident. You may want to use the [Always Record](#) option, so that you can't accidentally fail to record a perfect take. In order to further reduce visual clutter, it's recommended to use live view in [full screen](#) mode.

Docking Windows

Docking windows

Docking windows are special windows that can be [arranged](#) in various ways, in order to make the application more convenient to use. Docking windows are shown or hidden using the View/Docking Windows submenu. Many of Polymeter's features are accessed via docking windows.

Standard	The toolbar, containing buttons for frequently used commands
Channels	Lets you edit the properties of each MIDI channel
Graph	Displays graphs of modulations
MIDI Input	Displays input MIDI events in real time
MIDI Output	Displays output MIDI events in real time
Mapping	Lets you edit translations and mappings of input MIDI events
Modulations	Lets you edit modulations for the selected tracks
Parts	Lets you edit the list of parts
Phase	Displays phase relationships for the selected tracks
Piano	Displays output notes in real time on a virtual piano keyboard
Presets	Lets you edit the list of presets
Properties	Lets you edit the master properties
Step Values	Lets you edit track step values numerically, using a grid view

Arranging docking windows

[Docking windows](#) can be arranged in various ways to suit your taste. The arrangement determines how much space remains for viewing and editing the document. The following options are just some of the ways that docking windows can be arranged:

- Dock windows to the edge of a frame.
- Tab-dock windows together with other windows.
- Float windows over or outside the application.
- Hide windows along the edge of the application.
- Display windows on different monitors.
- Reset window placement to the default layout.

Docking windows can be arranged by dragging, by using commands on the Window menu, and by right-clicking the title bar of the window to be arranged.

Floating windows

You can undock a window from its default location in the application and move that window anywhere on the desktop or to another monitor. You can dock multiple windows together in a "raft" and then move them as a unit anywhere on the desktop or to another monitor.

To position a window outside the application:

- Drag the window to the location you want.

To return a window to its most recent docked location:

- Double-click the title bar of the window.

Docking windows can be fastened to one side of a frame in the application or within the editing frame. A guide diamond appears when you drag a window to another location to help you to easily re-dock the window.

To dock windows:

- Click the title bar of the window you want to dock.
- Drag the window toward the middle of the application. Tip: To move a dockable window without snapping it into place, press the Ctrl key while you drag the window. A guide diamond appears. The four arrows of the diamond point toward the four sides of the target pane. An additional four arrows point to the four edges of the application.
- When the window you are dragging reaches the location where you want to dock it, move the pointer over the corresponding portion of the guide diamond. The designated area is shaded.
- To dock the window in the position indicated, release the mouse button.

For example, if the properties bar is docked on the right edge of the application and you want to dock it on the left edge, drag it toward the middle of the application, move the pointer over the far left arrow of the guide diamond, and then release the mouse button.

A floating window can be maximized to cover the entire screen. To maximize a floating window, select the Maximize item from its Window menu. To restore a maximized window to its previous size and location, select the Restore item from its Window menu.

Auto hide

Docking windows support a feature called Auto Hide. Auto Hide causes a window to slide out of the way when you use a different window. When a window is auto-hidden, its name appears on a tab at the edge of the application. To use the window again, point to the tab so that the window slides back into view.

To use Auto Hide:

- On the menu bar, choose Window, Auto Hide to turn Auto Hide on or off. As an alternative, you can click the pushpin icon on the title bar of the window.

Docking windows that have Auto Hide enabled may temporarily slide into view when the window has focus. To hide the window again, select an item outside of the current window. When the window loses focus, it slides back out of view.

Specifying a Monitor

If you have a second [monitor](#) and your operating system supports it, you can choose which monitor displays a window. You can even group multiple windows together in "rafts" on other monitors.

To move windows to different monitors:

- Use the Display settings in the Control Panel to set up a multiple monitor configuration.
- In the application, drag the window to the other monitor.

Resetting window layouts

You can return the application to the original window layout by using the [Window/Reset Layout](#) command.

Menu bar

The menu bar contains the application's main menus. Unlike other [docking windows](#), the menu bar can't be hidden, and can only be docked along the top, bottom, left, or right edge of the application.

Toolbar

The toolbar provides buttons for the most frequently used commands. To show or hide the toolbar, use View/Docking Windows/Standard. Unlike other [docking windows](#), the toolbar can only be docked along the top, bottom, left, or right edge of the application.

Properties

This [docking window](#) lets you edit the [master properties](#). These properties are saved in the document. To edit a property, left-click it and change its value. To save your edit, press Enter or Tab, or left-click outside the property being edited. To cancel your edit, press Escape. While you're editing a property, the arrow keys may have different meanings. For example, for numeric edits and drop lists, the up/down arrows scroll through the possible values. If you're not editing a property, the up/down arrows navigate through the list of properties. To edit the currently selected property, press Tab.

Master properties

Tempo	Tempo, in beats per minute
Timebase	Time division, in ticks per quarter note
Meter	Number of beats in a measure
Key Signature	Key in which notes are displayed
Song Length	Song duration in hh:mm:ss format
Start Position	Starting position of song in beats and ticks
Note Overlap	How to handle overlapping instances of the same note on a channel
Record Offset	Offset added to recorded MIDI event times during playback, in ticks
Loop From	Starting position of loop, in beats and ticks
Loop To	Ending position of loop, in beats and ticks

Channels

This [docking window](#) lets you edit the setup for each MIDI channel. The messages specified here are sent when playback starts. This is useful if you want to initialize your instruments, by sending them patch change messages, for example. The properties are edited via a [grid control](#).

Columns

Column	Description
#	The channel number; use this column to select channels
Patch	The channel's patch number, or -1 for none
Volume	The channel's volume, or -1 for none
Pan	The channel's panning, or -1 for none
Bank LSB	The Least Significant Byte of the channel's bank, or -1 for none
Bank MSB	The Most Significant Byte of the channel's bank, or -1 for none
Overlaps	Selects a method of preventing note overlaps for this channel. This option is only meaningful if the Note Overlap master property is set to Prevent.
Duplicates	To prevent duplicate notes on the channel, set this to "Prevent". The default is "Allow". Only tracks that use index modulation are affected.

Presets

This [docking window](#) lets you edit your [presets](#). When you create a preset, it's given a temporary numeric name which you're free to change. For item selection methods, see [list controls](#).

Context menu

Item	Description	Selection
Create	Creates a new preset from the current state of the track mutes , and appends it to the presets list.	N/A
Apply	This command applies the selected preset, by muting and unmuting tracks as needed.	Single
Delete	Deletes the selected presets.	Multiple
Rename	Renames the selected preset.	Single
Update	Updates the selected preset from the current state of the track mutes.	Single

Parts

This [docking window](#) lets you edit your [parts](#). When you create a part, it has the same name as its first member [track](#), but you're free to rename it. For item selection methods, see [list controls](#).

Context menu

Item	Description	Selection
Create	Creates a new part from the selected tracks, and appends it to the parts list.	N/A
Delete	Deletes the selected parts.	Multiple
Rename	Renames the selected part.	Single
Update	Updates the selected part from the selected tracks.	Single
Tracks	Selects the tracks that belong to the selected parts.	Multiple
Sort	Sorts the parts, by name or by track.	N/A

Modulations

This [docking window](#) lets you edit your [modulations](#). The modulations are edited using a [grid control](#). Only modulators for the [selected tracks](#) are shown. If no tracks are selected, or if the selected tracks don't have any modulators, the window is empty. In addition to showing modulators, the window can optionally also [show targets](#).

Creating a modulation

To create a modulation, do the following:

1. [Select](#) one or more tracks; each of these tracks will be a target of the new modulation.
2. Ensure the Modulations bar is visible, and also ensure it has focus, by left-clicking or right-clicking within it.

3. Insert a new modulation, using either [Edit/Insert](#) or its keyboard [shortcut](#), or Insert on the [context menu](#).
4. Select the desired [modulation type](#) if needed. The type defaults to [mute modulation](#).
5. Select the desired modulation source, in other words the track that will do the modulating. If the source is "<None>" the modulation has no effect.

The new modulation is inserted at the current position in the modulation list. To append, first left-click or right-click below the last list item, so that no item is selected. If your modulation gets inserted in the wrong place, you can move it afterwards by left-clicking and dragging it.

Creating a modulation group

Sometimes it's desirable to create a group of modulations, all targeting the same tracks and having the same [modulation type](#), but with different source tracks. Instead of [creating](#) the modulations one by one, it's easier to create them all at once, as follows:

1. [Select](#) one or more tracks; each of these tracks will be a target of the new modulations.
2. Ensure the Modulations bar is visible.
3. Insert a new modulation group, using either [Track/Modulation](#) or its keyboard [shortcut](#), or Insert Group on the [context menu](#). The Modulation dialog is displayed.
4. In the Modulation dialog, select one or more source tracks and a modulation type, and then press OK to create the modulations.

Show Differences

The Modulations bar normally shows all modulations that the [selected tracks](#) have in common. If the selected tracks don't all have exactly the same modulations, the message "<Differences>" is displayed at the bottom of the list. This message indicates that at least one of the selected tracks has a modulation that's hidden, because it's not shared by the other tracks.

In some cases you may prefer to show all modulations that target any of the selected tracks, regardless of whether all the selected tracks share them. To do this, enable Show Differences mode via the [context menu](#). In Show Differences mode, the modulations list is a summary report. Each row of the report corresponds to a unique combination of modulation type and source track, and the number of instances of that combination is displayed in the index column. The report is sorted into descending order by number of instances. Reordering the modulations via drag and drop isn't supported in Show Differences mode.

Context menu

Item	Description
Insert	Inserts a new modulator at the current position.
Insert Group	Inserts a group of modulators at the current position.
Delete	Deletes the selected modulators.

Cut	Deletes the selected modulators after copying them to the clipboard .
Copy	Copies the selected modulators to the clipboard.
Paste	Inserts the contents of the modulations clipboard at the current position.
Select All	Selects all items in the modulations list.
Show Differences	Toggles Show Differences mode.
Show Targets	Shows or hides the Targets pane.
Sort	Sorts the modulators by modulation type or by source track index.

Show Targets

In addition to sources, the Modulations bar can also show [targets](#), i.e. tracks that are modulated by the selected tracks. The targets are shown in a separate pane. To show or hide the target pane, select Show Targets from the bar's context menu.

The source and target panes are separated by a splitter bar. To change the relative size of the panes, left-click and drag the splitter bar. To change the split type, right-click the splitter bar to display its context menu, and then select Horizontal or Vertical. To move the splitter bar to the middle of the window, select Center.

Only the direct targets of the selected tracks are shown; to show indirect targets, use the [graph bar](#). The target pane respects [Show Differences](#) mode. The target pane only partially supports editing, as shown in the table below. To change the type or target of selected modulations, use the target pane's [grid control](#). Commands supported in both panes are applied to whichever pane has focus.

Edit	Supported
Change type or target	Yes
Insert	No
Insert Group	Yes
Delete	Yes
Cut/Copy/Paste	No
Sort	No
Drag Reorder	No

MIDI Input

This [docking window](#) displays input MIDI events in real time, which can be useful for debugging your MIDI setup. To display output events, use the [MIDI Output bar](#), which works similarly.

By default, the window shows all input received from the MIDI [input device](#). The data can be filtered, so that only a subset of the events are shown, using optional filters which are accessible via [context menu](#). The filtering is non-destructive, i.e. if you remove a filter, any events that were rejected by that filter reappear.

For example, to only see events on channel 10, select 10 in the Channel Filter submenu. To only see Note On messages, select "Note On" in the Message Filter submenu. To remove all filters, use the context menu's Reset Filters command.

To pause the display, select Pause in the window's context menu. Events continue to be captured while you're paused, so you won't miss any. To resume the display, reselect Pause in the context menu. The context menu also allows you to clear the event history, via Clear History.

Context menu

Channel Filter	To show only a particular channel, select the corresponding submenu item, or select "All" to show messages from all channels.
Message Filter	To show only a particular message type, select the corresponding submenu item, or select "All" to show all message types.
System Messages	Check this item to enable the display of system messages.
Reset Filters	Resets all filters to their default settings (show all channels and all messages, including system messages).
Pause	Pauses and unpauses the display.
Clear History	Removes all stored messages from the history.
Export	Exports the current message history to a CSV file. All messages are exported, regardless of filtering.
Note Names	Check this item to display note names instead of numbers.
Controller Names	Check this item to display controller names instead of numbers.
Reset Columns	This command resets the list column header to its default state.

MIDI Output

This [docking window](#) displays output MIDI events in real time, which can be useful for debugging your MIDI setup. By default, the window shows all output sent to the MIDI [output device](#). For event filtering and other details, see the [MIDI Input bar](#).

Piano

This [docking window](#) dynamically displays the output notes on a virtual piano keyboard. The piano keyboard's orientation can be horizontal or vertical, depending on whether the window is docked horizontally or vertically. To monitor MIDI output messages other than notes, e.g. controllers, use the [MIDI Output bar](#) instead.

By default, output notes are displayed for all MIDI channels. To show only a particular channel, use the Channel Filter submenu of the window's [context menu](#). This submenu also lets you choose [multiple channels](#), in which case each channel is shown on its own keyboard. The context menu also lets you change the size of the piano keyboard, show or hide the key labels, and rotate the key labels. The window can optionally send MIDI notes when the corresponding keys are left-clicked. To enable this feature, use the context menu's Output Channel submenu to select the desired MIDI channel.

Context menu

Channel Filter	To only show notes for a particular MIDI channel, select the corresponding submenu item, or select "All" to show notes for all channels. You can also select "Multiple" to choose multiple channels .
Output Channel	In this submenu, select the MIDI channel on which left-clicked piano keys should send the corresponding notes, or select "None" to disable this feature.
Piano Size	To resize the keyboard, select a standard piano size from the submenu.
Key Labels	Check this item to show labels on the piano keys that identify the corresponding MIDI notes.
Rotate Labels	Check this item to rotate the key labels 90° counterclockwise, so that they're vertical instead of horizontal. This may allow the labels to use a larger font. This option is only supported when the window is docked horizontally.
Color Velocities	Check this item to color the piano keys according to the velocities of the output notes. The colors range spectrally from blue (softest) to red (loudest).
Insert Track	This command inserts a track for the selected note. Position the cursor over the desired note on the piano keyboard before using this command. The new track's note property will be set appropriately.

Multiple keyboards

The piano bar can display up to sixteen keyboards, each of which only shows notes for a specific MIDI channel. This is useful when the same notes are being played on multiple instruments; with a single keyboard, the instruments mingle, making it hard to distinguish them. The keyboards have labels indicating which channels they correspond to. To configure multiple keyboards, select "Multiple" in the [context menu](#)'s Channel Filter submenu. This displays the Select Channels dialog, which has a checkbox for each channel. Check the desired channels, and then press OK to proceed. The dialog provides additional selection features, explained below.

Button	Action
--------	--------

All	Checks all channels.
None	Unchecks all channels.
Used	Checks all channels that are used by the current document.

Graph

This [docking window](#) graphs your [modulations](#), in order to make them easier to visualize. To use this feature, you must first download and install [Graphviz](#), an excellent free graphing software that's used by many notable projects. If you're running 64-bit Windows, it's recommended to use a 64-bit build of Graphviz. More recent versions of Graphviz only support Windows 10.

Polymeter looks for Graphviz in two system folders: Program Files, and Program Files (x86). If it's not found in either place, you'll be prompted for the [location](#) of Graphviz.

The graphing options are selected via [context menu](#). The scope of the graph can be one of these:

Scope	Description
All	Show all modulations regardless of track selection
Sources	Show modulators of selected tracks
Targets	Show targets of selected tracks
Bidirectional	Show modulators and targets of selected tracks

Various graph layouts are supported, including dot (for hierarchical graphs) and neato (for non-hierarchical graphs). The edges are [colored](#) to indicate modulation type, and edge labels can optionally be displayed. The graph is generated as vectors (SVG format) so it's zoomable, and can be exported for printing. Hovering over a node's label shows a tooltip containing the track number. The graph is displayed using a web browser control (IWebBrowser2). Note that the recommended Internet Explorer version is 11 or later (Edge); earlier versions of IE may display the graph incorrectly or not at all.

Context menu

Scope	Use this submenu to select the scope of the graph. Scopes are explained above. For scopes other than All, the modulation network is crawled by starting from the selected tracks and traversing their modulators, targets, or both. The traversal can be limited to a specific distance via the Depth option.
Depth	Use this submenu to limit the recursion depth of the graph to a specific number of levels. To allow unlimited recursion depth, select Maximum. This option lets you simplify the graph by only including modulations that are closely connected to the selected tracks. It has no effect if the Scope is All.
Layout	Use this submenu to select a Graphviz layout engine. The dot layout engine draws a hierarchical graph, which is often the most useful way to visualize a modulation

	network. The other layout engines draw various types of non-hierarchical graphs. For more information on layouts, see the Graphviz documentation .
Filter	Use this submenu to filter the graph by modulation type . To show all types, select All. To show only a single type, select its name. To show multiple types, select Multiple, check the desired list items in the resulting dialog, and press OK. By default, all types are shown. The filter is applied after the modulation network has been crawled.
Channel	Use this submenu to select which MIDI channels are shown. To show all channels, select All. To show only a single channel, select its number. To show multiple channels, select Multiple, check the desired list items in the Select Channels dialog , and press OK. By default, all channels are shown. The channel selection is applied after the modulation network has been crawled. Tracks with a track type of Modulator are only included if they directly or indirectly modulate a non-Modulator track that matches the channel selection.
Zoom	Use this submenu to zoom the graph in or out, or reset zoom to its default value. The shortcuts are Ctrl++ to zoom in, Ctrl+- to zoom out, and Ctrl+0 to reset the zoom. To zoom in/out to/from the cursor position, spin the mouse wheel while pressing the Ctrl key.
Highlight	This item enables or disables highlighting of the selected tracks in the graph. When this option is enabled, the selected tracks are indicated by drawing the corresponding graph nodes with a different background color.
Labels	This item enables or disables edge labels. When this option is enabled, a label appears next to every connecting line in the graph, identifying the corresponding modulation type. This is only recommended for simple graphs, because otherwise the graph is hard to read.
Legend	This item enables or disables showing a legend that maps the edge colors in use to their corresponding modulation types.
Muted	This item determines whether the graph includes muted tracks . If this item is checked (the default), muted tracks are included, and the graph is unaffected by muting or unmuting tracks. Uncheck this item to exclude muted tracks. Excluding muted tracks can simplify the graph, but the graph also changes more often, because it updates whenever track mutes change. To match the Modulations bar , check this item; to match what you're hearing, uncheck it.
Save As	This item saves the current graph to a file. The default output format is SVG, which is vector-based and scalable. Other supported formats include PNG, BMP, and PDF. The command displays a file dialog, which lets you specify the name and location of the output file. To change the output format, use the "Save as type" drop list near the bottom of the file dialog. The command also displays an Image Export dialog, which lets you specify the output image size. If "Same size as window" is selected, the output image will have the same size as the graph window. To specify a different size, select "Custom size" and enter the desired width and height in pixels.
Reload	This item redraws the graph.

Graph edge colors

Color	Modulation Type
Black	Mute
Red	Note
Green	Velocity
Blue	Duration
Dark Cyan	Range
Magenta	Position
Brown	Tempo
Purple	Scale
Dark Green	Chord
Olive	Index
Orange	Voicing
Turquoise	Offset

Full screen mode

The Graph bar can be displayed in full screen mode, in which case the graph is shown without borders and fills the entire monitor. To make the bar fill a secondary monitor, drag the bar onto that monitor before making it full-screen. The shortcut key to enter or exit full screen mode is Ctrl+F11. You can also exit full screen mode by pressing the Escape key. To use either key, you may need to give the Graph bar the input focus first, by left-clicking within the graph.

Phase

This [docking window](#) shows an animated diagram that helps you visualize the phase relationships between your [tracks](#). The diagram consists of a series of concentric circles, each of which corresponds to a unique loop [period](#) in [ticks](#). Loop period is the product of a track's [Length](#) and [Quant](#). The periods are sorted so that the innermost circle corresponds to the shortest period, and the outermost circle to the longest period. An animated dot moves along each circular path, representing the phase, in other words the current position within the corresponding period. The representation is analogous to a planetary system, with the dots as planets and the circles indicating their orbits.

Multiple tracks having the same period are coalesced into a single orbit because they all share the same phase. If all of the tracks have the same period, only one orbit is shown. The diagram makes it easier to visualize [convergences](#). When two or more planets are shown at zero degrees (noon), the corresponding periods converge. All periods converge at the start of the first beat of the first measure. Subsequent convergences may be located using [Time to Repeat](#), the [Convergences calculator](#), or the [Next Convergence](#) and [Previous Convergence](#) commands.

The diagram indicates track [muting](#). A planet is drawn in red if *all* tracks having the corresponding length are muted. It's also possible to [exclude muted](#) tracks from the diagram, but doing so has [side effects](#).

The diagram also indicates [track selection](#) if any. Since each orbit can represent multiple tracks, the orbit is highlighted if any of the tracks it represents is selected. It's also possible to select tracks by left-clicking within the diagram, as explained below:

- To select a single orbit, left-click it. This selects only loops having the corresponding period.
- To select a range of adjacent orbits, left-click the first orbit, and then left-click the last orbit while pressing the Shift key.
- To toggle the selection of an orbit, left-click it while pressing the Ctrl key.
- To deselect all orbits, left-click on the diagram's background, outside the outermost orbit.

If you hover the pointer over an orbit, after a short delay, a tooltip is shown containing the corresponding Length and Quant.

Exporting video

The phase visualization can be exported as a series of frames in sequentially numbered PNG files. The resulting image sequence can then be converted to a video using [ffmpeg](#) or an equivalent program. To start exporting, select Export Video from the phase bar's [context menu](#). This command is only available while playback of the document is [stopped](#).

The command first displays a folder dialog, prompting you to select a destination folder for the image sequence. The export typically creates many files, so it's advisable to store them in their own folder. The dialog permits you to create a new folder if needed. Press OK to continue.

The command then displays a record dialog that lets you specify the duration of the recording, the frame rate and the frame size. The duration can be specified in either hh:mm:ss time or in frames. Press OK to start exporting frames. During the export, a progress bar is displayed.

This command can potentially consume a large amount of disk space. You're responsible for ensuring that sufficient space exists. If the destination folder already contains an image sequence from a previous run of this export, you'll be warned. If you proceed anyway, the previous image sequence is overwritten.

Context menu

Item	Description
Export Video	Exports video of the animated diagram as an image sequence; see above .
Circular Orbit	If checked, the orbits are circular, otherwise the orbits are elliptical with the same aspect ratio as the window.
3D Planets	If checked, the planets are drawn in 3D instead of 2D.

Night Sky	If checked, the background is dark instead of light.
Crosshairs	If checked, the diagram includes crosshairs.
Periods	If checked, the diagram includes labels that specify each orbit's period . The periods are displayed in ticks by default, but you can optionally convert them to another unit via division; see Period Unit below. Showing labels increases overhead.
Convergences	If checked, the diagram indicates convergences slightly before and after they occur. The participating orbits are increasingly highlighted as the convergence approaches, and gradually return to normal as the convergence recedes. To specify the minimum number of orbits that must participate in a convergence, use the Convergence Size options on the Transport menu. To skip lesser convergences, increase the size. If orbits are selected, only those orbits participate in convergences.
Exclude Muted	If checked, the diagram excludes muted tracks. This potentially simplifies the diagram. An orbit is only omitted if <i>all</i> tracks having the corresponding length are muted. Excluding muted tracks may conflict with showing convergences, and also affects the convergence commands , as explained below .
Full Screen	Makes the Phase bar enter full screen mode, or if it's already full screen, restores it to its previous size and position. In full screen mode, the diagram is shown without borders and fills the entire monitor. To make the bar fill a secondary monitor, drag the bar onto that monitor before making it full-screen. The shortcut key to enter or exit full screen mode is Ctrl+F11. You can also exit full screen mode by pressing the Escape key. To use either key, you may need to give the Phase bar the input focus first, by left-clicking the diagram's background.
Period Unit	This command lets you convert the period labels to a unit other than ticks, via division. It's only meaningful if you include period labels on the graph, by checking the Periods option as explained above. The command displays the Period Unit dialog; enter the desired unit in ticks, and click OK. All periods are divided by the specified unit before being displayed. For example, if the timebase is 120, specifying 30 will cause periods to be displayed in sixteenth notes. To revert to displaying periods in ticks, repeat this command and set the unit to zero.

Excluding muted tracks

When many different loop lengths are used, the phase diagram can have an excessive number of orbits, making it hard to read. If a significant portion of the tracks are [muted](#), the diagram can be simplified by excluding muted tracks. To do this, check [Exclude Muted](#) in the phase bar's [context menu](#). Normally, if *all* tracks having a given length are muted, the corresponding planet is drawn in red, but when Exclude Muted is checked, that planet and its orbit are omitted from the diagram. Excluding muted tracks has the following side effects:

- The [convergence commands](#) only consider unmuted tracks, and if all tracks are muted, these commands do nothing.
- The convergence commands can behave unexpectedly in [Song View](#), because in that view, which tracks are muted depends on the current [song position](#). The nearest convergence is calculated for whichever tracks are unmuted when the command occurs. Depending on the [dubs](#), moving to the convergence's position may mute tracks that participated in the convergence, in which case the corresponding orbits may not be [highlighted](#), or may disappear from the phase diagram entirely.
- When iterating through convergences in Song View, if you reach a position at which all tracks are muted, the iteration halts. This is particularly likely at the end of the song. To get unstuck, [navigate](#) to a position where at least one track is unmuted.

Step Values

This [docking window](#) lets you edit [step](#) values numerically, using a [grid control](#). In some cases this may be more convenient than dragging velocity bars in the [Velocities pane](#). It's particularly useful for editing [note modulator](#) tracks and similar cases.

Step values are shown for the [selected tracks](#) only. Note that selecting a large number of tracks at once may cause the Step Values window to perform poorly. It's recommended to hide the Step Values window when it's not needed.

This bar supports the [standard editing commands](#), allowing selected rows to be copied, cut, pasted, deleted and inserted. Selected rows can also be moved by left-clicking and dragging them. Note however that these commands are only supported if the selection is [rectangular](#), meaning its rows are contiguous and all contain the same number of steps. This may not be the case if the selected tracks have different lengths. Although the [track steps](#) area also supports the standard editing commands, editing may be easier in the Step Values bar because it shows the contents of the steps.

For each selected track, the step value corresponding to the current [song position](#) is highlighted in red or green depending on whether the track is [muted](#). To move the song position to a particular step within a selected track, left-click the step value while pressing the Ctrl key.

Display formats

Various display formats can be selected using the window's [context menu](#). If Signed is enabled, the steps are interpreted as [signed](#) values from -64 to +64, otherwise they're considered unsigned values. If Notes is enabled, the steps are displayed as notes, with or without octave information, depending on whether Octaves is enabled. If Ties is enabled, the [tied note](#) bit (the MSB) is included, otherwise it's masked off; this is an advanced feature, as is Hexadecimal display.

Context menu

Item	Description
------	-------------

Format	This submenu lets you select a display format .
Cut	Deletes the selected steps after copying them to the clipboard .
Copy	Copies the selected steps to the clipboard.
Paste	Inserts the contents of the steps clipboard at the current position.
Insert	Inserts a new step.
Delete	Deletes the selected steps.
Select All	Selects all steps.
Export	Exports the steps as text to the system clipboard.
Options	Configures the export. You can choose Columns = Steps versus Columns = Tracks, and Comma Delimited (CSV) versus Tab Delimited.

Mapping

This [docking window](#) lets you edit your [mappings](#). Mappings can be inserted, revised, deleted, copied, cut, pasted, and reordered via dragging. Mappings can also be sorted by left-clicking the appropriate item within the column header. To sort in descending order, press the Shift key while left-clicking.

The mappings are displayed in a [grid control](#). Each mapping consists of input and output event properties, an output range, and a target track. The input properties include the input event type (note on, control, etc.), the input MIDI channel, and the note or control number if appropriate. Input properties can be entered manually, or [learned](#) from the MIDI input.

The output properties start with the [output event type](#), which may be either a MIDI message (note on, control, etc.) or a target property. If the output event is a MIDI message, the mapping is a translation, in which case the output MIDI channel and the note/control number are also relevant. Otherwise, the mapping targets a property, in which case the output MIDI channel is irrelevant, the output control may not matter either, and a target track may be required.

To map a MIDI message to the same property in multiple tracks, create multiple mappings, one for each target track. It's easiest to do this via copy and paste: create a mapping for one of the tracks first, and when it's set up correctly, clone it as many times as needed, and finally edit the copies to target different tracks.

The output range (start value and end value) allow the input MIDI data to be shifted or scaled to any integer range. If the start is bigger than the end, the control direction is reversed, in which case turning the knob up makes the value smaller.

Mapping columns

Input Event	The type of input MIDI event being mapped; these are all MIDI channel messages.
-------------	---

Input Event	The MIDI channel on which to listen for the input event.
Input Control	The parameter that further specifies the input event. For Note On and Key Aftertouch events, it's the note number, for Control events it's the controller number, and for other input event types it's ignored.
Output Event	The type of output event that's being mapped to. For translations, these are MIDI channel messages, otherwise they're properties.
Output Channel	The MIDI channel of the output event. This is only applicable for translations.
Output Control	The parameter associated with the output event. This is only applicable for certain event types; see below.
Range Start	The starting value of the output range.
Range End	The ending value of the output range.
Track	The target track. This is only applicable when mapping to track properties.

Output event types

Translations			
Output Event	Output Control	Track	Comment
Note On	Note number	N/A	Output channel matters
Key Aftertouch	Note number	N/A	Output channel matters
Control	Controller number	N/A	Output channel matters
Patch	N/A	N/A	Output channel matters
Channel Aftertouch	N/A	N/A	Output channel matters
Wheel	N/A	N/A	Output channel matters
Mappings			
Type	N/A	Required	Enumeration
Channel	N/A	Required	Zero-based, from 0 to 15
Note	N/A	Required	In semitones
Length	N/A	Required	In steps
Quant	N/A	Required	In ticks
Offset	N/A	Required	Signed offset in ticks
Swing	N/A	Required	Signed offset in ticks
Velocity	N/A	Required	Signed offset in velocity
Duration	N/A	Required	Signed offset in ticks

Range Type	N/A	Required	Enumeration
Range Start	N/A	Required	In semitones
Mute	N/A	Required	Non-zero to mute track
Step	Step Index	Required	Event data is step value; see step mapping
Tie	Step Index	Required	Non-zero to tie step; see tie mapping
Presets	N/A	N/A	Event data is preset index
Parts	Part Index	N/A	Non-zero to activate part
Play	N/A	N/A	Non-zero to play
Pause	N/A	N/A	Non-zero to pause
Record	N/A	N/A	Non-zero to record
Loop	N/A	N/A	Non-zero to loop
Tempo	N/A	N/A	Fractional multiplier; see tempo mapping
Overlaps	N/A	N/A	Sets output channel's note overlap method, to Merge if non-zero, otherwise to Split
Duplicates	N/A	N/A	Sets output channel's duplicate notes method, to Prevent if non-zero, otherwise to Allow

Learning mappings

The application also supports [MIDI learn](#) mode, i.e. the input messages can be learned from the MIDI input device. To do this, select a mapping in the Mapping window, and then enable learn mode via Tools/MIDI Learn, or via the keyboard shortcut Ctrl+Shift+L, or via the Mapping window's [context menu](#). The background color of the selected mapping changes to green, indicating that this mapping will be learned. Now move the associated knob on the MIDI input device, and the mapping's input properties will snap to the appropriate values for that knob. To learn other mappings, select each mapping one at a time and move its associated knob. To learn multiple mappings at once, select them all and move the knob that you want to assign to all of them. After all mappings have been learned, disable MIDI learn mode.

Context menu

Insert	Inserts a new mapping at the current position
Delete	Deletes the selected mappings
Cut	Deletes the selected mappings after copying them to the clipboard
Copy	Copies the selected mappings to the clipboard
Paste	Inserts the contents of the mapping clipboard at the current position
Select All	Selects all mappings
Learn	Toggles MIDI Learn mode
Map Tracks	Creates mappings for the selected tracks

Tempo mapping

Controlling [tempo](#) is a special case, because unlike other [mapping](#) targets, tempo is exponential rather than linear. For example, suppose the normal tempo is 100, in which case half-time is 50 and double-time is 200. The values 50, 100, and 200 are clearly not linearly spaced. Tempo mapping handles this complication for you, so that your control behaves as if tempo were linear, even though it isn't. A pitch bend wheel works similarly.

To specify the range of tempo change your control covers, adjust the mapping's Range Start and Range End properties. Some useful tempo mapping setups are given below.

Range Start	Range End	Tempo Range	Comment
0	127	100% to 200%	Normal to double-time, the default setup
-127	0	50% to 100%	Half-time to normal
-127	127	50% to 200%	Half-time to double-time
-63	63	71% to 141%	Approximately half the previous range, from the square root of $\frac{1}{2}$ to the square root of 2

Tempo mapping produces a *multiplier*, which the base tempo is multiplied by. The tempo may also be multiplied by a [tempo track](#) if one exists, along with any [tempo modulators](#) the tempo track may have. The product of these multiplications is the *net tempo*. During [playback](#), the net tempo is shown in the [status bar](#).

Position mapping

Position mapping lets you control the [rotation](#) of one or more target tracks. Tracks don't have a rotation property, so it's necessary to use a two-stage workaround that consists of [position modulation](#) and [step mapping](#). You want to create a one-step position modulator track that targets the track(s) to be rotated, and then create a mapping that targets your position modulator track's one and only step. Do as follows:

1. Create a new track, change its [type](#) to Modulator, and change its [length](#) to one.
2. Create a [modulation](#) targeting the track(s) you want to rotate. Change the modulation's [type](#) to Position, and its source to the modulator track you created above.
3. Create a new mapping, and set its input properties appropriately for your control, or [learn](#) them.
4. Change the mapping's [Output Event](#) to Step, its Output Control to zero, and its Track to the modulator track you created above.
5. You may also want to adjust the mapping's range so that the possible rotations are spread out evenly across your control's range. To do this, set the mapping's Range Start to 64,

and its Range End to 64 plus the length of the track(s) being rotated. The offset of 64 is necessary because position modulation interprets steps as [signed](#) values.

Rotating multiple tracks with a single control makes more sense if the target tracks are all the same length, but that's not a requirement. If the above procedure isn't working, make sure the mapping's Output Control is zero; it's specifying which step to control, and step indices are zero-based. If you show your position modulator track in the [velocities pane](#) or the [step values bar](#), you should see its step value changing as you move your control. When in doubt, use the [MIDI input bar](#) to verify that you're receiving the expected MIDI message, and if not, check your [input device](#).

Step mapping

Step mapping means assigning a MIDI message to control the value of a step within a track. The mapping's Output Event must be Step, and its Track property must specify the target track. In a step mapping, the Output Control property is a zero-based index that specifies which step to access. The valid range of a step index is from zero to the target track's length minus one. For example, if the target track is four steps long, to access its last step, set Output Control to three. If the step index is out of range, the mapping has no effect. Because [steps](#) serve many purposes, there are many possible step mapping scenarios, only some of which are presented below.

Position mapping	This is already explained above.
Mapping a single control to the same track property in many tracks	Normally you must create one mapping for each track you want to control, but this is potentially cumbersome when controlling many tracks. If the track property you want to control has a modulation type associated with it—as is the case for note , offset , velocity , duration , and range start —it may be easier to modulate the target tracks with a one-step source track, and then create a single mapping that targets your source track's step. This is similar to the position mapping procedure explained above, except you'll select a different modulation type.
Independently controlling each of the step values in a track	By mapping a different control to each step of a note track, you can use a control surface to create rhythmic patterns. Your controls determine not only the pattern, but also the note velocities. To emulate the step buttons on a drum machine, use switches instead of knobs or sliders, toggling between 0 and 100 for example. To create melodies, map controls to the steps of a note or index modulator. You can also alter the contents of scales or chords by mapping controls to their steps. Controller tracks can also be interesting step mapping targets.

Tie mapping

Within [note tracks](#), a step can be [tied](#), so that it extends the preceding note instead of triggering a new note. You can assign a MIDI message to control whether a track step is tied or not. Set the mapping's Output Event to Tie, its Track property to the target track, and its Output Control to

the zero-based step index. The input control should ideally be a button that toggles between zero and a non-zero value. [Step mapping](#) and tie mapping work together to give you complete control of a note track: use step mapping to change the note velocities, and tie mapping to merge adjacent steps into longer notes. Tie mapping is only supported for note tracks and has no effect on other track types.

Properties

Tempo

This is the playback tempo, in beats per minute. The document is [played](#) at this tempo, unless this base tempo is modified by a [tempo track](#). The tempo can be changed at any time, including during playback. Changing the tempo may change the current time shown in the [status bar](#). To change the tempo, left-click it, edit the value or type a new one, and press Enter or Tab, or left-click elsewhere. To increment or decrement the tempo, use the arrow buttons that appear when you left-click the tempo. For more information on editing, see the [Properties bar](#). Fractional tempos are permitted.

Tempo is normally expressed as a frequency, but it can also be expressed as a [period](#), in other words how long each beat takes in seconds. To compute the period, divide 60 by the tempo. For example, at 120 beats per minute, each beat takes half a second, because $60 / 120 = \frac{1}{2}$.

Tempo can be remotely controlled; see [tempo mapping](#). During [playback](#), the net tempo is shown in the [status bar](#).

Timebase

This is the time division, also known as the timebase, in [ticks](#) per quarter note (TPQN). It relates *ticks*, which are the smallest unit of time in MIDI, to *beats*, which are quarter notes. In other words, it specifies the number of ticks in a beat, and thereby determines the precision with which time can be represented. A range of predefined [timebase values](#) are supported. At the default timebase of 120, a beat has 120 ticks. For a list of common note durations in ticks when using the default timebase, see [Quant](#).

The default timebase is recommended. Larger timebase values improve timing precision, but they also reduce the range of [duration](#) and [offset](#) modulation. That's because these modulation types interpret track [steps](#) as time offsets in ticks, and since the range of a step is finite, the larger the timebase, the less impact such time offsets have.

The timebase can't be changed during [playback](#), and only one timebase can be used within any given document. Changing the timebase of an existing document causes all settings in ticks to be scaled, so that their values remain proportionally similar in the new timebase. Specifically track Quant, [Offset](#), [Swing](#), and [Duration](#) are all scaled, along with the times of any [dubs](#). The scaling

is only guaranteed to be exact if the new timebase is an integer multiple of the current one, otherwise it's approximate and may cause loss of precision.

Timebase values

This table lists the supported timebase values, all of which are multiples of 24 to ensure compatibility with [MIDI clock](#) synchronization. The table also shows the Greatest Prime Factor (GPF) for each timebase, in other words the largest [prime](#) number each timebase is evenly divisible by, which is useful information if you want to use odd subdivisions of a beat.

Ticks	GPF
24	3
48	3
72	3
96	3
120	5
144	3
168	7
192	3
216	3
240	5
288	3
336	7
360	5
384	3
480	5
504	7
672	7
768	3
840	7
960	5

Meter

This is the number of beats in a measure. It determines how [song positions](#) are displayed throughout the application. If meter is one, song positions are displayed as beats and [ticks](#), whereas if meter is greater than one, song positions are displayed as measures, beats and ticks (also known as MBT). Meter is one by default, and can't be less than one. Meter is purely cosmetic, in the sense that changing it has no effect on the application's MIDI output. Meter

affects how the current position is displayed in the [status bar](#), and also affects the display of [Start Position](#), [Loop From](#), and [Loop To](#).

Since [polymeter](#) is the use of multiple meters at once, it might seem pointless to specify a meter, but it can still be useful to change how song positions are displayed, particularly if your composition emphasizes one meter more than others.

Song position

A song position is a location within the output MIDI sequence. Song positions are maintained internally in [ticks](#), but they're displayed either as beats and ticks, or as measures, beats and ticks (also known as MBT), depending on the [Meter](#) property. Ticks are zero-based, whereas both beats and measures are one-based, hence a song position of zero ticks is displayed as 1:000 if meter is one, or 1:1:000 if meter is greater than one.

Song positions are unaffected by the [tempo](#). But if a song position is converted to a time, the resulting time does depend on the tempo. The [status bar](#) shows the current position in both measures, beats, and ticks on the left, and time on the right.

The following table shows the effect of some common meters on various positions in ticks, assuming the default [timebase](#) of 120 ticks per beat. The asymmetry in how negative positions are displayed is due to measures and beats being one-based.

Ticks	Meter		
	1	3	4
480	5:000	2:2:000	2:1:000
360	4:000	2:1:000	1:4:000
240	3:000	1:3:000	1:3:000
120	2:000	1:2:000	1:2:000
0	1:000	1:1:000	1:1:000
-120	0:000	0:3:000	0:4:000
-240	-1:000	0:2:000	0:3:000
-360	-2:000	0:1:000	0:2:000
-480	-3:000	-1:3:000	0:1:000

Navigating within a song

To move the [song position](#) to a specific location, use one of these methods:

- In the [track grid](#) or [song grid](#), left-click the desired location in the ruler at the top of the view.
- To move to the [start position](#), use the Transport menu's [Rewind](#) command.

- To specify the desired location numerically, use the Transport menu's [Go To Position](#) command.
- To move to the nearest [convergence](#), use the [Next Convergence](#) and [Previous Convergence](#) commands on the Transport menu, or their keyboard shortcuts: Alt+Right for next convergence, or Alt+Left for previous convergence.
- In the [Step Values bar](#), left-click a step value to move to the corresponding location.
- In [Song view](#) only, use Shift+Right to move to the next [dub](#), or Shift+Left to move to the previous dub.

Key Signature

This is the key in which notes are displayed throughout the application. It determines which notes are displayed as sharps rather than flats. The key signature is purely cosmetic, in the sense that changing it has no effect on the application's MIDI output.

Song Length

This is the song duration, in hh:mm:ss format. It determines the maximum horizontal scroll position in [song view](#). Stopping a [recording](#) sets the song length automatically, to the time when recording was stopped.

Start Position

This is the [position](#) at which the song starts. It determines the [Song view](#)'s left margin, and the [Rewind](#) command's destination. It's displayed either in beats and [ticks](#), or in measures, beats and ticks, depending on the [Meter](#) property.

Negative start positions are allowed, and this may be useful because a full [convergence](#) is guaranteed at position zero (1:000). By starting at a negative time, you effectively move this initial convergence to later in your song.

Note Overlap

This property specifies how to handle overlapping instances of the same note on the same MIDI channel. If overlapping instances are allowed, the output device may release the later note prematurely, or the overlapped portion may exhibit acoustic artifacts such as flanging. To prevent overlapping instances, set this property to "Prevent". Preventing note overlaps is strongly recommended if [position modulation](#) or [offset modulation](#) are used.

For example, suppose two overlapping instances of the same note on the same channel occur as follows:

Time	Event
1:000	first note on

2:000	second note on
3:000	first note off
4:000	second note off

In this case, the note will likely be cut off at 3:000, in other words the second instance is aborted by the first instance's note off command. Preventing note overlaps results in the following output:

Time	Event
1:000	first note on
2:000	first note off, second note on
4:000	second note off

Instead of the default method shown above, you also have the option to merge the overlapping instances into a single note. In this case the result would be as follows:

Time	Event
1:000	first note on
4:000	first note off

The merge option is selectable on a per-channel basis, using the Overlaps column in the [Channels bar](#). The following diagram shows different types of note overlap and possible solutions.

Partial Overlap



Full Overlap



Uncorrected



Split



Merge



Note Overlap Methods	
Split	Preserves the overlapping notes. Note messages are added or removed as necessary to ensure that the overlapping notes don't cut each other off.
Merge	Eliminates all but the first note. The overlapping notes are merged into a single note that spans all of them, and gets its velocity from the first note.

Selecting a note overlap method is only meaningful if overlap correction is enabled, by setting the Note Overlap [master property](#) to **Prevent**. If Note Overlap is set to **Allow**, overlap correction is disabled and Split versus Merge is irrelevant. The Note Overlap master property enables overlap correction *as a whole*, whereas the Overlaps column in the Channels bar determines which correction method (Split or Merge) is used for each channel. The correction method can also be controlled by an [internal track](#), or remotely via MIDI [mapping](#).

Merge has uses beyond overlap correction. It can also be used to concatenate adjacent instances of the same note, so that they become one long note. To do this, you must force the adjacent notes to overlap, typically by slightly increasing their durations, using the [track duration](#) property or via [duration modulation](#).

Record Offset

This offset is added to the times of recorded MIDI events during playback. It's specified in [ticks](#). It allows you compensate for latency during recording, by playing recorded MIDI events later (positive offset) or earlier (negative offset), relative to the playback of [tracks](#). For details, see [Recording MIDI Input](#).

Loop From

If playback is [looped](#), this is the starting position of the loop. It's displayed either in beats and [ticks](#), or in measures, beats and ticks, depending on the [Meter](#) property. The ending position is given by the [Loop To](#) property. The starting position must be less than the ending position, otherwise no looping occurs.

Instead of entering the loop positions numerically, you can alternatively create a [step selection](#) before using the Loop command. In that case the step selection's first and last steps determine the loop's starting and ending positions respectively. In the [song grid](#), a loop can similarly be specified by selecting a range of cells and then using the Loop command.

If a valid loop exists, it's indicated by a highlighted area in the rulers at the top of the [track steps](#) and song grid. The loop's range can be modified by horizontally dragging the left or right edge of this highlighted area.

Loop To

If playback is [looped](#), this is the ending position of the loop. For more information, see [Loop From](#).

Options

MIDI

Input Device

This is the currently selected MIDI input device. Only one input device can be used at a time. The application doesn't require an input device, except for a few supplemental features such as [MIDI mapping](#) and [Record Tracks](#).

Output Device

This is the currently selected MIDI output device. Only one output device can be used at a time. The application requires an output device in order to play the sequence, and if one isn't selected, [playback](#) is disabled. The default output device is the GS Wavetable Synth that ships with Windows.

The output device can be hardware, such as a MIDI-capable synthesizer or drum machine. Some hardware instruments connect directly to your computer via USB, whereas others require a MIDI interface. In either case you may need to install driver software, depending on the device. Multiple devices can be controlled, either by daisy-chaining them together, or via specialized MIDI hardware.

Alternatively the output device can be software, such as a software synthesizer or Digital Audio Workstation (DAW). The audio software may create one or more MIDI ports that appear in Polymeter's list of output devices, but if not, you'll need to install a virtual MIDI port, such as Tobias Erichsen's loopMIDI.

Latency

This is the desired MIDI latency during [playback](#), in milliseconds. The shorter the latency is, the less the user interface lags behind playback, and the closer the application is to being in real time. On the other hand, the shorter the latency is, the more often the application runs, and the less time it has to do its work. For example if the latency is 10, the application has at most 10 milliseconds to prepare each chunk of the sequence for output. If the latency is too short, [underruns](#) may occur. Underruns can't occur in exported MIDI files because the [export](#) can take as long as it needs.

Issue	Symptoms
Latency too short	Underrun errors; erratic timing of output MIDI messages; audible gaps in sequence

Latency too long	User interface lags behind playback; delay between your actions and their effects
------------------	---

Underruns

During [playback](#), the MIDI sequence is prepared ahead of time in chunks. The length of the chunks is determined by the [Latency](#) setting. Ideally the application prepares the next chunk while the previous chunk is being played, and thus stays ahead of demand. But if a chunk takes too long to prepare, the previous chunk may run out before the next chunk can be delivered. This situation is called an *underrun*.

An underrun means that the sequencer took too long and failed to keep up. It can occur because the sequence is too dense, or the latency is too short, or CPU power is lacking, or the output device is too slow, or any combination of these. An underrun can affect the timing of the output MIDI messages, and in extreme cases may cause an audible gap in the sequence. If it's the first underrun since playback started, an error message is displayed. Subsequent underruns are [counted](#), but the error message is suppressed. To solve underrun errors, do one or more of the following:

- Make your sequence less dense. For example you may be sending continuous controller messages more often than necessary.
- Increase the latency. This has the downside of making the user interface lag further behind the playback.
- Eliminate competing applications so that more CPU power is available, or use a more powerful computer.
- Use faster output devices, or faster connections to them.

If you're using MIDI hardware, be advised that a serial MIDI interface is limited to roughly a thousand MIDI messages per second, and approaching this limit may cause underrun errors. This is less likely to be a problem if the hardware is connected to the computer via USB, or if the output device is a software synth or virtual MIDI port.

Buffer Size

This is the size of the output MIDI event buffer, in bytes. If the error "MIDI event buffer size exceeded" occurs during [playback](#), try increasing this size. Decreasing it is not recommended.

Thru

If true, MIDI input received from the [input device](#) is passed through to the [output device](#). This lets you play along with your sequence. Any MIDI messages that are [mapped](#) aren't passed through.

Default Velocity

This is the default note velocity or controller value for new [steps](#).

Input Quantization

This is the quantization for converting MIDI input to steps. It's used when [importing](#) a MIDI file, and also by the [Record Tracks](#) command. It can be set to none, in which case the input is converted to the finest resolution available in the current [timebase](#), which is a [Quant](#) of one tick.

Record Input

This determines what types of data are captured by the [Record](#) command. The options are as follows:

Dubs only	Only mute/unmute events, also known as dubs , are recorded.
Dubs + MIDI	Both dubs and MIDI data received from the input device are recorded.
MIDI only	Only MIDI data received from the input device is recorded.

For more information, see [Recording MIDI Input](#).

Send MIDI Clock

If true, MIDI synchronization messages are transmitted to the [output device](#). If you want the output device to synchronize itself with the application, set this option to true. The output device must support MIDI clock sync for this to work.

View

Update Frequency

This is the frequency at which the view is updated during playback, in Hertz (updates per second). It specifies the desired frame rate, but the actual frame rate may differ, depending on your hardware's capabilities. Windows that track [song position](#) may animate more smoothly if you increase the frame rate, but CPU and/or GPU usage may also increase, so proceed cautiously.

Show Current Position

If true, the current position within each track is indicated visually in the [Track Steps](#) area.

Show Note Names

If true, notes are displayed as names instead of numbers, for example C4 instead of 60.

Show General MIDI Names

If true, General MIDI patch and drum names are displayed instead of numbers. Patch names are shown in the [Channels bar](#). Drum names are shown in the [note](#) column of the [Track View](#), but only for tracks that output on channel 10.

Show Quant as Fraction

If true, [Quants](#) are shown as fractions of a whole note; otherwise, quants are shown in [ticks](#). For example using the default [timebase](#) of 120, a quant of 120 is shown as 1/4 (a quarter note), 60 is shown as 1/8, and so on. If you're using common quants, fractions are easier to work with than ticks, because the numbers are smaller and the application does the tick arithmetic for you. You can always *enter* a quant as a fraction, regardless of this option; this option only controls how quants are *shown*.

Some quants may still be shown in ticks even when this option is enabled. If a quant can be converted to a fraction having a denominator that's either a power of two, or three times a power of two, and the denominator is reasonably small, the quant is shown as a fraction, otherwise it's shown in ticks. For example you may enter the fraction 1/5, but it will be shown in ticks, because the denominator is non-conforming.

If this option is enabled when you edit a quant in the [track grid](#), a dropdown list containing common quant fractions is displayed next to the edit box. To select a fraction from the list, either left-click it, or use the up and down arrow keys to scroll to the desired fraction, and then [end the edit](#). If the desired fraction isn't in the list, type the fraction in the edit box, as two integers separated by a slash. You can alternatively type a single integer, in which case it's interpreted as a value in ticks. Fractions are reduced automatically, for example if you type 6/8 it will be shown as 3/4. If a given fraction can't be exactly converted to ticks, the closest value is substituted.

Show Track Colors

If true, tracks can be assigned custom colors, using the [Track Color](#) command on the [toolbar](#).

Unique Track Names

If true, when renaming multiple tracks in the [track grid](#), sequential numbers are appended to the track names as necessary in order to keep them unique.

Zoom Step

This is the zoom step size, as a percentage. It only applies to zooming the [Track View](#).

Live View Font Height

This is the height of the [Live View](#) font, in pixels. Making the font smaller fits more rows in the live view's lists of presets and tracks, which may let you avoid scrolling the lists, at the cost of making them harder to read. Another way to avoid scrolling is by grouping related tracks into [parts](#).

Graph

Graphviz Folder

This is the location of the folder containing the Graphviz executables. Graphviz is an excellent free graphing software that's used by many notable projects. Polymeter uses Graphviz to make graphs of your [modulations](#). If you wish to display modulation graphs, the Graphviz executables are required. This option is set automatically by the Graphviz wizard when you show the [Graph bar](#).

You may set this option manually, either by typing a path, or via the browse button. The browse button is normally hidden, but it appears on the right when the option is selected. Click the browse button to display a Browse for Folder dialog. Note that the path must be fully-qualified, and the folder it identifies must contain the Graphviz executables, which are normally found in the bin subfolder of the Graphviz distribution.

Graph Handles Unicode

If true, the [Graph bar](#) correctly handles [non-ASCII](#) characters in [track names](#). This option is only meaningful if Graphviz is installed. Enabling this option adds overhead due to UTF-8 encoding, hence it's disabled by default. With this option disabled, non-ASCII characters may display as question marks in the graph, and can also cause incomplete or erroneous graphs. If you regularly use non-ASCII characters, enable this option.

Use Cairo

If true, the [graph](#) is rendered with the Cairo graphics library, which is included in Graphviz. If you're having problems with the graph's text, enabling this option may help. Graphviz normally outputs high-level instructions in SVG format, which are then rendered by a web browser control. This is efficient, but it can cause problems with fonts. Enabling this option instructs Graphviz to convert fonts to lines and curves instead of relying on the web browser's fonts. This brute force solution greatly increases the size of the SVG files and may reduce graphing performance. Before enabling this option, consider that the most likely cause of *undersized* graph text is the system display scaling, and a specific fix is available for that, as explained [here](#).

General

Show Property Descriptions

If true, descriptions are shown in the [Properties bar](#).

Automatically Check for Updates

This option determines whether Polymeter automatically checks for updates when it launches. If a newer version is available, you will be prompted with the message "A newer version of Polymeter is available. Do you want to download and install it?" The options are as follows:

Yes	Download and install	The newer version is downloaded and installed.
No	Skip this version	The newer version is NOT downloaded, and no further notifications are given <i>for this version</i> , though you'll continue to be notified of subsequent versions.
Later	Defer the update	The newer version is NOT downloaded, but you'll be notified about it again the next time you launch Polymeter.

Always Record

If true, [dubs](#) are always recorded during playback, and the resultant recordings are saved automatically. Enabling this option prevents data loss due to forgetting to press [Record](#). The recordings are stored in the current user's profile, and named using a combination of the document title and the current date and time. For a user named foo, the destination folder would typically be:

C:\Users\foo\AppData\Roaming\Polymeter

Menus

File

New

Use this command to create a new Polymeter document. To open an existing document, use [Open](#) instead.

Open

Use this command to open an existing Polymeter document. A document can also be opened by drag/drop from Windows Explorer, or by specifying the document's path on the command line.

Close

Use this command to close the active document. If the document has been modified, you'll be prompted to save your changes.

Save

Use this command to save the active document to its current name and directory. If you want to change the name or directory of an existing document before you save it, use [Save As](#) instead.

Save As

Use this command to save and name the active document.

Export

This command exports the current document as a type 1 MIDI file, containing at most 17 tracks. The first track is reserved for the tempo map, and it's followed by one track for each MIDI channel that's utilized in the document. If the document contains song [dubs](#), they're applied, otherwise the unmuted tracks are looped until the specified duration is reached.

For each MIDI channel, all Polymeter [tracks](#) that output to that channel are merged into a single MIDI file track. The resulting MIDI file track gets its name from the first Polymeter track that contributed to it. If no Polymeter tracks output to the channel, the corresponding track is omitted from the MIDI file to save space.

The command displays a dialog that prompts you to choose the name and location of the destination file. After you press OK to continue, another dialog prompts you for the export duration in hours, minutes and seconds. For [songs](#), the export duration defaults to the current [song length](#), otherwise it defaults to its previous value.

The document can also be [exported as a table](#), which can be useful for debugging.

Import

This command imports a MIDI file into the current document. Only channel messages—notes, continuous controllers, pitch bends, etc.—are imported. Message times are quantized according to the current [Input Quantization](#) setting. Each unique combination of MIDI message type, channel, and note or controller number (if applicable) produces a separate [track](#).

Exit

Use this command to end your Polymeter session. You can also use the Close command on the application Control menu. Polymeter prompts you to [save](#) documents with unsaved changes.

Edit

Standard editing commands

The commands for inserting, deleting, and moving data via the [clipboard](#) are common to most applications. They're available from the Edit menu and certain context menus, and in some cases also from the [toolbar](#).

Command	Shortcut	Effect
Cut	Ctrl+X or Shift+Delete	Copies the selection to the clipboard and then deletes it
Copy	Ctrl+C or Ctrl+Insert	Copies the selection to the clipboard
Paste	Ctrl+V or Shift+Insert	Inserts the contents of the clipboard
Insert	Insert	Inserts a new item
Delete	Delete	Deletes the selection
Select All	Ctrl+A	Selects all items

These commands are supported by the [Track view](#), [Song view](#), [Modulations bar](#), [Mapping bar](#), and [Step Values bar](#). They're also supported within any text edit control, excepting the Insert command. If more than one of these targets is visible, the standard editing commands affect whichever one currently has the input focus.

Within the track view, if a [step selection](#) exists, it takes priority over the [track selection](#). In other words, if both tracks and steps are selected, the editing commands operate on the selected steps. The song view works similarly.

Clipboards

The Polymeter application doesn't use the Windows clipboard, except for text. Instead, Polymeter has internal clipboards for each of the following types of data: [tracks](#), [steps](#), [dubs](#), [modulations](#), and [mappings](#). These clipboards are only accessible within the instance of Polymeter that created them, hence they don't support copying data to other applications or other instances of Polymeter, nor do they persist after Polymeter exits. This shouldn't be a problem since Polymeter is an MDI (Multiple Document Interface) application, meaning it allows multiple documents to be opened at once. To move data between two Polymeter documents via clipboard, open both documents within a single instance of Polymeter.

The behavior of Cut/Copy/Paste is slightly non-standard due to each supported data type having its own clipboard. For example, if you copy some tracks, and then copy some steps, the steps don't replace the tracks, as would normally be the case if the Windows clipboard were used. Both the copied tracks and the copied steps are available for pasting in their respective views, because they're stored separately.

Certain types of data can also be moved between documents using tables, using the [Export Table](#) and [Import Table](#) commands, and such tables persist after Polymeter exits, unlike its clipboards.

Undo

This command undoes the last action. Polymeter supports unlimited undo and [redo](#).

Redo

This command redoes the previously undone action. Polymeter supports unlimited [undo](#) and [redo](#).

Cut

This command deletes the selection after copying it to the [clipboard](#).

Copy

This command copies the selection to the [clipboard](#).

Paste

This command inserts the contents of the [clipboard](#) at the current position.

Insert

This command inserts a new item at the current position.

Delete

This command deletes the selection.

Select All

This command selects all items.

Deselect

This command deselects all tracks.

Rename

This command renames the selected items.

Find

This command finds a string in the [track names](#).

Replace

This command finds a string in the [track names](#), and replaces it with a different string.

View

Docking Windows

This submenu lets you show or hide any of the [docking windows](#).

Status Bar

The status bar displays hints for the menus, and other status information, including the current [position](#) and time.

During [playback](#), the status bar also shows the *net tempo*, which accounts for [tempo tracks](#), [tempo modulation](#), and [tempo mapping](#).

Application Look

This submenu allows you to customize Polymeter's appearance. The different looks correspond to various versions of Windows.

Zoom In

This command zooms into the center of the current view. It's supported by the [Track View](#) and the [Song View](#). In both cases only the horizontal axis is zoomed. To zoom at a location other than the center, move the pointer to the desired location and spin the mouse wheel up while pressing the Ctrl key. The track view's [zoom step](#) is adjustable.

Zoom Out

This command zooms out of the center of the current view. It's supported by the [Track View](#) and the [Song View](#). In both cases only the horizontal axis is zoomed. To zoom at a location other than the center, move the pointer to the desired location and spin the mouse wheel down while pressing the Ctrl key. The track view's [zoom step](#) is adjustable.

Zoom Reset

This command resets the current view's zoom to its default.

Properties

This command shows or hides the [Properties bar](#).

Channels

This command shows or hides the [Channels bar](#).

Modulations

This command shows or hides the [Modulations bar](#).

Groups

This submenu contains commands to show or hide windows associated with groups, such as [presets](#) and [parts](#).

Presets

This command shows or hides the [Presets bar](#).

Parts

This command shows or hides the [Parts bar](#).

Velocities

This command shows or hides the [Velocities pane](#) within the [Track View](#).

Track

This command shows the [Track View](#).

Song

This command shows the [Song View](#).

Live

This command shows the [Live View](#).

Customize

This command lets you customize your toolbars, menus, and keyboard shortcuts. It displays a tabbed dialog. The various tabs are explained below.

Commands tab

This tab contains commands that you can drag onto your menus and toolbars. It allows you to add commands and modify existing commands on the Menu bar or any toolbar. For detailed instructions, see the [Menu tab](#).

The commands are categorized by which menu they belong to. To see the commands for a particular menu, select that menu in the **Categories** list box. To see all the available commands, select **All Commands**.

You can also select **New Menu**, which allows you to drag a new submenu onto the menu you're customizing. After you've created the submenu, you can drag commands onto it. To rename the submenu, right-click its name, and in the resulting context menu, select **Button Appearance**; in the resulting dialog, type the new name in the **Button Text** edit box. To give the name a mnemonic, insert an ampersand before the character that should be the mnemonic.

Toolbars tab

This tab allows you to create, rename, delete, and reset your toolbars. The following table describes the tab options.

Option	Description
Toolbars listbox	Displays the toolbars built into the application and any toolbars you create. When you show a toolbar, a check mark appears to the left of it. NOTE: The Menu Bar cannot be hidden. It can only be reset.
Reset	Removes any changes to the built-in toolbars and resets them to their original state. Only available if you select a built-in toolbar.
Reset All	Resets all built-in toolbars and menus to their original state.
New	Opens the New Toolbar dialog box where you type the name for your new toolbar in the Toolbar name box.
Rename	Opens the Rename Toolbar dialog box where you type the new name for your toolbar. Only available if you select a user-defined toolbar.
Delete	Deletes a user-defined toolbar from your project. Only available if you select a user-defined toolbar.

Keyboard tab

This tab allows you to customize keyboard shortcuts (also known as shortcut keys or keyboard accelerators) by assigning them to a command. You can also remove keyboard shortcuts. To assign or remove a keyboard shortcut:

1. In the **Categories** box, select the category that contains the command that you want to assign a keyboard shortcut to or remove a keyboard shortcut from.
2. In the **Commands** box, select the name of the command that you want to assign a keyboard shortcut to or remove a keyboard shortcut from. Any keyboard shortcuts that are currently assigned to that command appear in the **Current keys** box, or below the box with the label **Assigned to**.
3. To assign a keyboard shortcut do the following:
 - a. Begin keyboard shortcuts with CTRL or a function key. In the **Press new shortcut key** box, press the combination of keys that you want to assign. For example, press CTRL plus the key that you want to use.
 - b. Look at **Current keys** (or **Currently assigned to**) to see whether the combination of keys is already assigned to a command. If the combination is already assigned, type a different combination.

- c. Select **Assign**.
- 4. To remove a keyboard shortcut:
 - a. In the **Current keys** box, select the keyboard shortcut that you want to remove.
 - b. Select **Remove**.

Menu tab

This tab allows you to change the appearance of the application frame menus and context menus. First select the menu you wish to customize.

- For frame menus, use the **Show Menus for** drop list; in this case, you must also show the selected menu, by left-clicking it.
- For context menus, use the **Select Context menu** drop list; in this case, the selected context menu is shown automatically.

Once you've selected and shown the menu, you can customize it as follows:

- To move a command up or down within the menu, drag the command to the desired position.
- To remove a command from the menu, drag the command off of the menu.
- To add a command to the menu, first switch to the **Commands** tab. Then drag the desired command from the **Commands** list box onto the menu, and drop it at the desired position.
- To insert a separator, right click on the menu at the position where the separator should go, and in the resulting context menu, select **Start Group**.
- To change the text or appearance of a command, right-click it, and in the resulting context menu, select **Button Appearance**. This same context menu also lets you delete the command, hide the command's image if any, or reset the command to its default state.
- To reset the entire menu to its original state, press the appropriate **Reset** button in the **Menu** tab.

The tab also lists animations you can apply to your menu bar, and lets you enable or disable menu shadows.

Options tab

This tab allows you to change the appearance of your menu bar and toolbars; for example, you can change the size of your toolbar buttons, and show ToolTips and shortcut keys. The following table describes the options.

Option	Description
Show ScreenTips on toolbars	Turns ScreenTips off and on.
Show shortcut keys in ScreenTips	Displays the shortcut keys on the menus.
Large icons	Changes the toolbar icons to a larger size.

Track

Length

This command lets you set the [length](#) of a track visually instead of numerically. To set the length of a track:

1. Indicate the desired length by hovering the cursor over the location where the last step should be.
2. Set the new length via the Track/Length command or its shortcut Ctrl+L.

Tracks can be shortened or lengthened using this method. If a [track selection](#) exists, all of the selected tracks are set to the indicated length.

If the cursor isn't over the step view, the error message "Cursor must be over track step to indicate desired length" is displayed.

If a [step selection](#) exists, it takes priority. In this case the track is shortened so the leftmost step of the step selection becomes the track's last step. If the step selection includes multiple tracks, they're all shortened.

1. Select the step that should be last. The step selection can span multiple tracks.
2. Set the new length via the Track/Length command or its shortcut Ctrl+L.

Shift

These commands shift tracks left or right by some number of steps. Empty steps are shifted in as needed. Shifting is a lossy, destructive operation. Lossless shifting, also known as rotation, is also supported, via the [rotate](#) commands.

If a [step selection](#) exists, it takes priority. In this case only the selected steps are shifted. Otherwise the command operates on the [selected tracks](#) in their entirety.

Shift Left

This command shifts tracks one step to the left. The leftmost step is lost, and an empty step is shifted in on the right to replace it.

Shift Right

This command shifts tracks one step to the right. The rightmost step is lost, and an empty step is shifted in on the left to replace it.

Shift Steps

This command shifts tracks by any amount. A dialog is displayed, prompting you for the desired shift in steps. The shift can be positive (right) or negative (left). After you press OK, the tracks are then shifted by the specified number of steps. Empty steps are shifted in as needed.

Rotate

These commands rotate tracks left or right by some number of steps. Rotation is similar to shifting, except that steps wrap around instead of falling off one end. For example, rotating 1234 right by one step yields 4123, because the 4 wraps around to the left. Rotation is a lossless, non-destructive operation. Shifting is also supported, via the [shift](#) commands.

If a [step selection](#) exists, it takes priority. In this case only the selected steps are rotated. Otherwise the command operates on the [selected tracks](#) in their entirety.

Rotate Left

This command rotates tracks one step to the left.

Rotate Right

This command rotates tracks one step to the right.

Rotate Steps

This command rotates tracks by any amount. A dialog is displayed, prompting you for the desired rotation in steps. The rotation can be positive (right) or negative (left). After you press OK, the tracks are then rotated by the specified number of steps.

Offset Tracks

This command adds a specified amount to the [track offset](#) for the [selected tracks](#). A dialog is displayed, prompting you for the desired offset change, in [ticks](#). The change can be positive (later) or negative (earlier).

Reverse

This command reverses tracks. The order of the steps is reversed, for example 1234 becomes 4321.

If a [step selection](#) exists, it takes priority. In this case only the selected steps are reversed. Otherwise the command operates on the [selected tracks](#) in their entirety.

Invert

This command inverts tracks. Steps containing non-zero values are set to zero, and steps containing zero are set to the [default velocity](#).

If a [step selection](#) exists, it takes priority. In this case only the selected steps are inverted. Otherwise the command operates on the [selected tracks](#) in their entirety.

Tie Notes

This command toggles a variable that determines whether the [steps](#) of a [note track](#) are initially [tied](#) or untied. When you left-click on an empty step, you may prefer the newly created note to be tied or untied, depending on the situation. If you're writing for an instrument that ignores note durations, as many drum machines do, you'll probably prefer notes to be untied by default. But if you're writing for a melodic instrument that respects note durations, you'll probably prefer notes to be tied by default. In either case you can achieve the opposite result by holding down the Shift key while left-clicking, as the following table shows:

	Tie Notes OFF	Tie Notes ON
left-click	Untied note	Tied note
Shift+left-click	Tied note	Untied note

The command is available both from the Track menu and the [Toolbar](#). Its icon resembles a flattened 'U', and if it's highlighted, Tie Notes is on, otherwise it's off. Only note tracks are affected by the Tie Notes state.

Transpose

This command transposes the [selected tracks](#). A dialog is displayed, prompting you for the desired transposition in semitones. It can be positive or negative. After you press OK, this amount is added to the [note](#) for each track. For [controller tracks](#), this command offsets the controller number. Tracks can also be transposed via [note modulation](#).

Velocity

This command changes the velocities of the selected [tracks](#) or [steps](#). It has three functions: Offset, Scale and Replace. If a [step selection](#) exists, it takes priority, and in this case only the selected steps are affected. Otherwise the command operates on the [selected tracks](#). The command supports all [track types](#), even though they interpret step values differently. For [note tracks](#), only notes that start within the step selection are affected. For all functions, the resulting values are [clipped](#) so that they can't exceed the permissible range. If clipping is likely, a "Clipping may occur" warning [message](#) is displayed, giving you the opportunity to cancel or continue. The clipping check accounts for all velocity offsets including [velocity modulation](#).

Offset

Use this function to apply an offset, in other words to add or subtract a value from the velocities. The offset is a positive or negative integer. If a step selection exists, the specified offset is added to the selected steps. If a step selection doesn't exist, the "Apply to" setting determines how the

offset affects the selected tracks: if it's "Tracks," the offset is added to their [velocity offsets](#), whereas if it's "Steps," the offset is added to their step values or note velocities.

Scale

Use this function to apply scaling, in other words to multiply the velocities by a value. The scale may be positive or negative, and may also be fractional. If a step selection exists, each selected step is multiplied by the specified scale, otherwise for each selected track, all of its step values are multiplied by the specified scale. If "Signed" is selected, the steps are interpreted as signed values ranging from -64 to +63, otherwise they're interpreted as unsigned values ranging from 0 to 127.

Replace

Use this function to replace a particular step value with a different one. If a step selection exists, the replacement only occurs with the selected steps, otherwise it occurs within all steps of the selected tracks. The find and replace operation can operate on [signed or unsigned steps](#). You can also replace a range of velocity values; to do this, first check the "Range" checkbox, in order to reveal the "Find start" and "Find end" edit controls, and then use these controls to specify the desired range.

Fill

This command fills a range of steps with a series of values. The series can be linear, or one of the supported periodic functions. Filling may be more convenient than editing the velocity bars in the [velocities pane](#), particularly for periodic functions.

If a [step selection](#) exists, it takes priority. In this case only the selected steps are filled. Otherwise the command fills all of the steps in the [selected tracks](#).

Value Range

This is the range of values to be filled. For linear fills, these are the starting and ending values. For periodic functions, the difference between these two values is the amplitude of the resulting waveform. The range may be [signed or unsigned](#). To fill steps with a constant, set Start and End to the same value.

Step Range

This is the range of steps to be filled. It's taken from the step selection if one exists, otherwise it's the maximum of the lengths of the selected tracks.

Function

This is the type of function to fill with. The options are Linear, Sine, Triangle, Ramp Up, Ramp Down, and Square. All but the first are periodic functions.

Frequency

This is the frequency of the periodic waveform, in Hertz. Frequency applies to all functions except Linear. To fill with one complete cycle of the waveform, set the frequency to one. Frequency is a real number and may be fractional.

Phase

This is the phase of the periodic waveform, in [normalized degrees](#), ranging from zero to one. Phase applies to all functions except Linear. To start the waveform in its usual position, set the phase to one.

Curviness

This makes a periodic waveform curvier. If it's zero, curviness is disabled. Otherwise, the curvature becomes more extreme as curviness diverges from one. This lets you make a wider variety of shapes by distorting the underlying waveform. Curviness applies to all functions except Linear. For a typical exponential curve, select the following: Step Range = 0 to 127 (or -64 to +63), Function = Ramp Up, Frequency = 1, Phase = 0, Curviness = 100. For standard waveforms, leave Curviness zero.

Value	Description
< 0	Not allowed.
0	Linear (disabled).
0 < x < 1	Starts out curving more steeply and then levels off; the smaller the value, the more extreme the curvature.
1	Linear. Use zero instead to avoid needless computations.
> 1	Starts out curving more gradually and then gets steeper; the larger the value, the more extreme the curvature.

Stretch

This command either stretches or shrinks the [selected tracks](#). Unlike simply changing the [track length](#), this command resizes the contents of the [track steps](#) as needed to fit the new length. A dialog is displayed, prompting you to enter the desired scaling as a percentage. For example, entering 200 doubles the size of the track, and entering 50 halves it. For integer multiples of 100, the stretching is lossless, but for all other scaling factors, the process is approximate. The percentage can be fractional, for example entering 66.666 shrinks the track to two-thirds of its size.

To estimate the new steps via linear interpolation, check the Interpolate option. Otherwise step values are repeated or removed as needed, but never changed. Disabling interpolation is useful when stretching or shrinking [modulator tracks](#) that contain notes, to avoid altering the tonality.

For [note tracks](#), the note durations are increased or decreased as needed, and the interpolation option is ignored. The Stretch command can be useful when changing the [quant](#) of a note track. For example, suppose you have a track in eighth notes, and you want to convert it to sixteenth notes, without changing how it sounds. You divide its quant in half, but this causes the track to play twice as fast. Stretching the track by 200% fixes this.

Mute

This command toggles the [mute](#) state of the [selected tracks](#). Tracks that are currently muted become unmuted, and vice versa.

Solo

This command solos the [selected tracks](#). The selected tracks are unmuted, and all other tracks are muted.

Group

This command groups the [selected tracks](#) into a new [part](#). The new part is named after its first member track, and appears in the [Parts bar](#).

If any of the selected tracks already belong to a part, a warning [message](#) is displayed, giving you the option to continue or cancel. If you continue, the offending tracks are removed from their previous part before being reassigned to the new one.

Preset

This command creates a new [preset](#) from the current state of the [track mutes](#). The new preset is given an automatically generated name, and appears in the [Presets bar](#).

Modulation

This command creates one or more [modulations](#) targeting the [selected tracks](#). It's particularly useful for creating a group of modulators all having the same [modulation type](#), as in [scale modulation](#) for example. Instead of creating the modulators one by one in the [modulations bar](#), it's easier to create them all at once using this command.

At least one track must be selected in the [track view](#) for this command to be enabled, because the track selection determines which tracks will be modulated. The command displays a dialog, allowing you to select one or more source tracks and pick a modulation type. Press OK to create the modulations. For help with selecting multiple items, see [list controls](#). It's recommended to keep the modulations bar visible while using this command so that you can see its effects. This command is also accessible via the Insert Group item on the modulations bar's [context menu](#).

You also have the option to sequentially distribute one source to each selected track. To do this, check the Modulation dialog's Distribute option. This is particularly useful when creating [index](#)

[modulation](#) for a chord. Instead of assigning a different index modulator to each chord voice one by one, you can assign them all at once. For example if tracks 1, 2, 3 are selected, and sources 4, 5, 6 are distributed to them, after the command, 4 modulates 1, 5 modulates 2, and 6 modulates 3. Note that if Distribute is checked, the number of source tracks must match the number of selected tracks, otherwise an error message is displayed.

Sort

This command sorts the [selected tracks](#), or all tracks if none are selected. The command displays a dialog which allows you to specify up to three sort levels. Each sort level consists of a [track property](#) and a direction (ascending or descending). To disable a sort level, set its property to none.

It's also possible to sort by ID. This puts the tracks in the order they were in when the document was opened. Any tracks added since then appear in the order they were added.

Sorting tracks may have unintended effects, for example [scale](#) and [chord](#) modulators are potentially sensitive to track order.

Color

This command lets you customize the background color of tracks in the [Track view](#). It's only accessible from the [toolbar](#), and only if the [Show Track Colors](#) option is enabled. To assign a custom color to the [selected tracks](#), left-click the paint bucket icon in the toolbar. A pop-up menu is displayed. Now select a color swatch by left-clicking it, or if the desired color isn't available, left-click "More Colors" for an extended palette. To restore the default color to the selected tracks, select "Automatic."

If the paint bucket icon is disabled, it's because either no tracks are selected, or the Show Track Colors option is false.

Transport

Play

This command starts or stops playback.

Pause

This command pauses or continues playback.

Record

This command starts or stops recording. By default, only [dubs](#) are recorded, but you can change this using the [Record Input](#) option. You can also configure the application to start recording automatically, using the [Always Record](#) option.

Rewind

This command rewinds back to the [start position](#).

Loop

This command enables or disables looped playback. While looping is enabled, playback is restricted to the range of time defined by [loop from](#) and [loop to](#).

Go To Position

This command prompts you to enter a numeric [song position](#) and then navigates to that position. The position can be specified in measures, beats, and [ticks](#), or in time.

Next Convergence

This command goes to the next [convergence](#). If a [track selection](#) exists, the convergence is calculated for the selected tracks only, otherwise it's calculated for all tracks. To specify a threshold of significance below which convergences are skipped, use the [Convergence Size](#) submenu. To go backwards, use the [Previous Convergence](#) command.

Note that this command is affected by [excluding muted tracks](#) from the Phase bar.

Previous Convergence

This command goes to the previous [convergence](#). For more information, see [Next Convergence](#).

Note that this command is affected by [excluding muted tracks](#) from the Phase bar.

Convergence Size

This submenu lets you choose a convergence size for the [Next Convergence](#) and [Previous Convergence](#) commands. Convergence size is the minimum number of unique loop lengths that must reach their start point simultaneously for it to count as a [convergence](#). The size lets you filter the convergences by significance, finding only those where at least the specified number of different lengths converge, while skipping over the lesser ones. At the default size of two, the commands find every convergence. To find only three-way or higher convergences, set the convergence size to three. To find only full convergences, set the convergence size to All, but note that doing so may slow down the search if many large, [relatively prime](#) lengths are in use.

If the desired size isn't listed in the submenu, select Custom to display a dialog that lets you enter other sizes. The size can be set as low as one, in which case the next/previous convergence commands find every time when a loop starts, regardless of whether other loops also start at that time. To display a list of convergences, use the [Convergences](#) command.

The convergence size also affects the [Phase bar](#)'s optional Convergences feature, which indicates convergences slightly before and after they occur.

Record Tracks

This command records input MIDI messages and converts them to [tracks](#). Only MIDI channel messages (notes, control changes, pitch bends, and so on) are recorded; other message types are ignored. The timing of the input messages is quantized in accordance with the [Input Quantization](#) option. Each unique note number or control number used during the recording results in one track. Before they're inserted, the recorded tracks are sorted into descending order by note or control number within message type.

This command is only available if an [Input Device](#) is selected. If you start recording tracks while [playback](#) is stopped, playback is started automatically. You can also start recording tracks during playback, but in this case the recorded tracks will include a chunk of empty steps at the beginning, which corresponds to the period between when playback started and when you started recording. Depending on the [latency](#) setting and [devices](#) in use, the recording can be significantly delayed, though this may be correctable afterwards.

Tools

Statistics

This command displays performance statistics. The statistics are reset every time [playback](#) is started.

Total Callbacks	The total number of MIDI callbacks that have occurred.
CB Underruns	The number of underrun errors that have occurred.
CB Latency	The desired callback latency in milliseconds and ticks.
Min CB Time	The minimum callback duration, in time and as a percentage of the latency.
Max CB Time	The maximum callback duration, in time and as a percentage of the latency.
Avg CB Time	The average callback duration, in time and as a percentage of the latency.

If underrun occurs, the maximum callback length will exceed the callback latency. To avoid underruns, increase the latency so that it's well above the maximum callback length.

Time To Repeat

This command reports how long it takes for the [selected tracks](#) to repeat. The result is displayed in beats, and in time. The result in time depends on the [tempo](#), whereas the result in beats is tempo-independent.

Prime Factors

This command reports the [prime factors](#) of the lengths of the [selected tracks](#). The lengths are computed in steps if possible, otherwise they're computed in [ticks](#); the details are explained [here](#).

Velocity Range

This command reports the minimum and maximum step values in the selected [tracks](#) or [steps](#). The command also checks for [clipping](#), which can be caused by excessive [track velocity](#) or [velocity modulation](#). If a [step selection](#) exists, it takes priority, and only the selected steps are included. Otherwise all steps of the [selected tracks](#) are included. The step values are interpreted as unsigned integers from 0 to 127, regardless of [track type](#). When applied to [note tracks](#), the command reports velocities, but for other track types, the meaning depends on the context.

Check Modulations

This command checks your [modulations](#) for errors. All modulations are checked, regardless of [track selection](#). If errors are found, they're reported in a table that contains one row for each erroneous modulation. The columns consist of the target track's number, the source track's number, the modulation type, and the error message. The errors are explained below.

Error	Severity	Description
Unsupported modulation type	Warning	The modulation has an inappropriate type for the context in which it's used.
Infinite loop	Fatal	The modulation directly or indirectly modulates itself, causing an infinite loop.

An unsupported modulation type indicates a violation of the [modulation rules](#). For example, the setup shown below reports an error for the velocity modulator, because velocity modulators can't be [nested](#). The erroneous modulation has no effect on playback, which is why the error is only a warning.

velocity modulator → mute modulator → note track

The infinite loop error is fatal because such loops can potentially crash or freeze the application. The playback engine protects itself against infinite recursion, but more complex modulation setups can loop in other ways that are harder to detect. Unlike the playback engine, the Check Modulations command can detect all feedback loops, even if they're currently inactive due to tracks being [muted](#).

Devices

This command displays a list of the currently available MIDI devices. To select MIDI devices, use the [Input Device](#) and [Output Device](#) settings in the [Options](#) dialog.

Options

This command displays a dialog that lets you edit Polymeter's options, which are global user preferences that apply to all documents. The options are summarized [here](#). Press OK to close and save your changes, or Cancel to close without saving. To reset all options to their default values, press the Reset All button.

Import Table

Each of these commands imports data from a Comma-Separated Values (CSV) file. In all cases, a file dialog is displayed, prompting you to select the input CSV file. The imports have different targets, and expect differently formatted data, as explained below. Each import has a corresponding [export](#), and in order to better understand a format, it may be helpful to examine the results of its export.

Import Tracks

This command imports [track](#) data from a CSV file. The first row must be a column header specifying which track properties are to be imported. Each column header item must be a valid track property name. It's not necessary to specify all the track properties, as any omitted properties will be initialized to default values. Track properties may be specified in any order, so long as the subsequent data rows use that same order. For a list of the track property names, examine the file resulting from the corresponding [export tracks](#) command.

Each subsequent row corresponds to a track to be inserted, and must contain properly formatted data values for the properties specified by the column header. Values containing commas must be enclosed in double quotes.

Import Steps

This command imports [step](#) data from a CSV file, containing one row per track, with no initial header row. Each row consists of a list of integers ranging from 0 to 255, corresponding to the step values for that track. The tracks needn't all be the same length. Each track's length is inferred from the number of step values found on its row. For note tracks, step values may exceed 127, due to the most significant bit being used for [tying](#) steps together into longer notes. You can also [export steps](#).

Import Modulations

This command imports [modulation](#) data from a CSV file containing one row per modulation, with no initial header row. Each row consists of three integers, corresponding to the modulation's enumerated type, source track index, and sink track index. The [modulation types](#) are zero-based.

The track indices are also zero-based and must be within the range of the current tracks. You can also [export modulations](#).

Export Table

Each of these commands exports data to a Comma-Separated Values (CSV) file. In all cases, a file dialog is displayed, prompting you to specify the name and location of the output CSV file. The exports have different targets, and produce differently formatted data, as explained below.

Export Tracks

This command exports [track](#) data to a CSV file. If a [track selection](#) exists, only the selected tracks are exported, otherwise all tracks are exported. The first output row is a column header specifying the names of the track properties. Each subsequent row corresponds to a track, and contains the data for each of the track's properties. You can also [import tracks](#), using this same file format. If one or more track names contain [non-ASCII](#) characters, the CSV file is encoded as UTF-8 with a BOM (Byte Order Mark).

Export Steps

This command exports [step](#) data to a CSV file, for the selected tracks, or for all tracks if none are selected. Each output row corresponds to a track, and contains one integer for each of the track's steps. For note tracks, step values may exceed 127, due to the most significant bit being used for [tying](#) steps together into longer notes. You can also [import steps](#), using this same file format.

Export Modulations

This command exports [modulation](#) data to a CSV file. All modulations in the current document are exported. Each output row corresponds to a modulation, and contains three integers, corresponding to the modulation's enumerated type, source track index, and sink track index. You can also [import modulations](#), using this same file format.

Export Song

This command is similar to the [MIDI file export](#), except that it exports the MIDI messages as text. The resulting CSV file contains the same data as the MIDI file export, but in a human-readable format. This can be useful for debugging purposes, for example text exports can be meaningfully compared using a file comparison utility. The command displays a dialog prompting you for the desired duration. If any [dubs](#) exist, they're applied during the export, otherwise the unmuted tracks are looped for the specified duration.

Convergences

This calculator shows the [convergences](#) between a given set of numbers. If a [track selection](#) exists, the calculator uses the loop lengths of the selected tracks as its input, otherwise you can enter a set numerically in the dialog.

If the input is from selected tracks, the results are shown in either [steps](#) or [ticks](#), depending on the [Quants](#) of the selected tracks. There are three distinct cases:

1. The selected tracks all have the same Quant. In this case, the results are shown in steps.
2. The selected tracks have different Quants, but all the Quants are integer multiples of the smallest Quant. In this case, the results are shown in steps of the smallest Quant. For example, if the selected tracks have Quants of 30, 60, and 120, the results are shown in steps of 30 ticks.
3. Neither of the preceding cases apply. In this case, the results are shown in ticks. This can make the numbers much bigger, and may cause the calculation to take noticeably longer.

If you compare the results to the [Time to Repeat](#) function, bear in mind that Time to Repeat is shown in beats, which must be converted to either steps or ticks depending on the cases explained above.

If you prefer to enter a set numerically, make sure no tracks are selected when you show the calculator dialog. The set must consist of at least two comma-separated positive integers. Note that zero is not considered a positive integer. The set can be in any order, and duplicate values are ignored. A valid input set could look like this:

2,3,5,7

Simply type the set in the input edit control and press Enter or Tab to calculate the convergences. To export the results, use the Windows clipboard: copy the output edit control's text, using Ctrl+A and Ctrl+C or the context menu, and paste it into the desired application.

The calculator only shows the *first* convergence for each unique subset of the input numbers. The subsequent convergences can be determined via multiplication. To step through convergences one by one, use the [Next Convergence](#) and [Previous Convergence](#) commands.

Select Unmuted

This command [selects](#) all [unmuted](#) tracks. This can be useful in preparation for various commands that operate on selected tracks, such as [Time to Repeat](#), [Prime Factors](#) and [Convergences](#).

MIDI Learn

This command enables or disables MIDI learn mode. Learn mode makes it easier to set up [mappings](#) so that properties can be remotely controlled, typically via a MIDI control surface. Learn mode is only effective if the [Mapping bar](#) is visible and at least one mapping is selected. While learn mode is enabled, any mappings selected in the Mapping bar are highlighted in green.

In learn mode, if a MIDI channel message is received from the input MIDI device, any selected mappings automatically snap their input properties to match the received message. This is often

easier than setting the input properties manually, and makes it unnecessary to know what MIDI messages the control surface sends for each of its controls.

Be sure to disable learn mode when you're done learning input MIDI messages, otherwise you may unintentionally change your mappings the next time you touch the control surface. See also [learning mappings](#).

Window

New Window

Use this command to open a new window with the same contents as the active window. You can open multiple document windows to display different parts or views of a document at the same time. If you change the contents in one window, all other windows containing the same document reflect those changes. When you open a new window, it becomes the active window and is displayed on top of all other open windows.

Splits

You can split the view area horizontally or vertically into two or more tab groups. This lets you show multiple documents at once, which can be useful for comparing them. Splitting also lets you work with multiple views of a single document at the same time. For example, instead of switching back and forth between [track view](#) and [song view](#), you may prefer to show your document in both views at once. To create another view of the active document, use the [New Window](#) command.

For splitting to work, you must have at least two tabs in the same tab group. The tabs can correspond to different documents, or to different views of the same document. Activate whichever tab you wish to move, and then use either the [New Horizontal Tab Group](#) or [New Vertical Tab Group](#) command to create an alternate tab group. The active view is automatically moved into the newly created tab group.

You can also left-click a tab, drag it, and drop it within the view area of a tab group. If you drop it within the tab group it already belongs to, a new split is created, and the active view is moved to the new tab group. If you drop it within a different tab group, the active view is moved to that tab group. Removing the last tab from a tab group also removes the corresponding split if any.

You can create as many tab groups as you want provided they all have the same orientation, in other words you can't mix horizontal and vertical tab groups. To change the proportions of two tab groups, left-click and drag the [splitter bar](#) that separates them.

If you create the initial split by dragging a tab, the drop location determines whether the split is vertical or horizontal. The split will be vertical unless you drop near the bottom of the view area, in which case the split will be horizontal; in the latter case, the bottom half of the view area is temporarily outlined to indicate that a horizontal split will occur.

Multiple monitors

Showing multiple views of the same document is more useful in a multiple monitor setup where each view has enough space to be fully utilized. The optimal setup is two identical monitors placed side by side and set to the same resolution, with the Windows desktop extended onto both monitors. In that case, you can resize the application's main window so that it fills both monitors. This is ideal for creating vertical [splits](#).

Multiple monitors of different sizes and resolutions can also be useful, for example you could place the application's main window on the larger monitor, and use the smaller monitor to show one or more [docking windows](#).

Full Screen

Use this command to toggle full screen mode. In full screen mode, the current view is enlarged to fill the entire monitor, and all other application windows are hidden, including the caption bar and all [docking windows](#). The [shortcut keys](#) still work however.

To enter full screen mode, use Window/Full Screen or press F11. To exit full screen mode, press F11 or the Escape key. If your Windows desktop spans multiple monitors, drag the application onto the desired monitor before going full screen.

Reset Layout

This command restores every [docking window](#) to its default size, position, and visibility. Most application settings are also restored to default values, except for the [options](#), which can be reset separately via the Reset All button in the [Options dialog](#). All [customizations](#) are reset, restoring toolbars, menus, and keyboard shortcuts to their original state. This command can't be undone and causes the application to restart.

New Horizontal Tab Group

This command creates a new horizontal tab group, and moves the active view into the newly created group. For details, see [splits](#).

New Vertical Tab Group

This command creates a new vertical tab group, and moves the active view into the newly created group. For details, see [splits](#).

Windows

This command displays a dialog that lists all views of the currently open documents. The dialog also lets you activate, save, or close views. The list box supports multiple selection.

Help

Help Topics

Use this command to display the help's table of contents and [default topic](#). The application also supports [context-sensitive help](#).

Polymeter on the Web

Use this command to launch your web browser to Polymeter's home page.

Check for Updates

Use this command to determine if a more recent version of Polymeter is available. If a newer version is available, you'll be asked whether you want to download and install it. For more information, see [Automatically check for updates](#). You must be logged on as Administrator to install an update.

About

Use this command to display Polymeter's copyleft notice, version number, and license.

Loose Ends

Audio versus events

Digital music is typically created and edited within two distinct domains: audio space, and event space. Audio space represents music as samples (the musical equivalent of pixels), whereas event space represents music as notes and control changes (the musical equivalent of vectors). Many musically useful operations that are difficult or impossible in audio space are trivial in event space. For example, muting a track without cutting off notes that are already playing is easy in event space, but impossible in audio space. The Polymeter application operates entirely in event space, and generates no audio by itself. Polymeter outputs a stream of MIDI messages which must be routed to an [output device](#) capable of converting MIDI to audio.

Recording MIDI input

The MIDI input recording feature is primarily intended for the case where a MIDI controller device is being routed through Polymeter to the output device. For example you might be using a MIDI controller to tweak the settings of a software synth. If you record a live performance in Polymeter, it's not enough to record only your [dubs](#). In order to entirely capture the performance, the controller moves must also be recorded.

Recording of MIDI input is optional, and it's off by default, for backwards compatibility. To enable it, open the Options dialog and set the [Record Input](#) option to "Dubs + MIDI." Now when you do the [Record](#) command (Transport/Record), in addition to your dubs, any MIDI events received from the input device will also be recorded. Note that playback of the recorded events occurs ONLY in Song view. There's currently no way to directly view or edit the recorded events, though this may change in future versions.

The timing resolution is pretty good, but there can be problems with recording latency. If the recorded events playback out of sync with the sequencer tracks, you can offset their timing via a Master property, [Record Offset](#). This offset is in [ticks](#). To make the recorded events earlier relative to the sequencer tracks (the usual case), enter a negative value.

It's unusual for a record offset to be specified in ticks as opposed to milliseconds, but it has one big advantage: once the record offset is correctly set for a given recording, you can subsequently change the song's tempo and the record offset remains correct.

There's no exact method for determining the amount of recording latency. The most direct approach is to record yourself playing along with a click track and then adjust the record offset by ear until it sounds in sync. You can also inspect the timing differences in the MIDI output view. On a typical system, the record latency is about 40ms, but your results will depend on your hardware configuration. Once you know your recording latency, you can compute the appropriate record offset for a given song via the following function:

$$\text{offset} = (\text{tempo} / 60) \times \text{timebase} \times (\text{latency} / -1000)$$

This assumes the latency is specified in milliseconds. For example if latency is 40ms, tempo is 90 BPM, and timebase is 120, the function is:

$$\begin{aligned}\text{offset} &= (90 / 60) \times 120 \times (40 / -1000) \\ \text{offset} &= 1.5 \times 120 \times -0.040 \\ \text{offset} &= -7.2\end{aligned}$$

So in this case the record offset should be -7 ticks.

Maximum track length

A track can have up to millions of steps, depending on its [Quant](#). Track length is limited by several factors. The hard limit is that the product of the track's [Length](#) (in steps) and Quant (in ticks) can't exceed 2 billion. Attempting to exceed this limit causes the error message "Track too long." But well before reaching this upper limit, you'll experience problems with the track's steps displaying incorrectly or not at all. This occurs because the width of the Steps view is similarly limited, to around 100 million pixels.

For example, a Length of 1 million and a Quant of 30 gives a product of 30 million ticks, which seems doable. But how wide are the steps in pixels? At the default zoom, they're 20 pixels wide, necessitating a view width of 20 million pixels, which is already quite large. Zooming in a few

times makes the track too wide, causing incorrect display. This may be addressed in a future release, but meanwhile it's recommended to keep tracks under a million steps, or even less if unusually large Quant values are used.

Normalized angles

Phase-related properties may be expressed as *normalized angles*. A normalized angle maps the full range of angles (360 degrees) to the range zero to one. The following table shows the relationship between degrees and normalized angles:

Degrees	Normalized
0	0
90	0.25
180	0.5
270	0.75
360	1

Values outside the range [0..1] wrap around, e.g. 1.25 wraps to 90° and -0.25 wraps to 270°.

Period

An oscillation can be described in two ways. One can specify how many cycles it makes per second, also known as its *frequency*, or one can specify how long each of its cycles takes in seconds, also known as its *period*. Frequency and period are inverses of each other: frequency equals one divided by period, just as period equals one divided by frequency. For example, a frequency of two cycles per second is equivalent to a period of half a second.

In this application, a [track](#) is analogous to an oscillator, in that it cycles sequentially through its [steps](#). A track's period, meaning how long it takes to complete one cycle, is the product of its [Length](#) in steps, and the duration of each step, also known as the track's [Quant](#). Because step duration is measured in [ticks](#), a track's period is also measured in ticks. For example, a track that's five steps long and has a quant of 30 has a period of $5 \times 30 = 150$ ticks.

Primes

A prime number is an integer that has no factors other than itself and one, in other words an integer that's evenly divisible only by itself and one. The first ten prime numbers are 1, 2, 3, 5, 7, 11, 13, 17, 19, and 23, after which the series continues infinitely.

Each prime has an extended "family" of integer multiples, for example the 5 family contains 5, 10, 15, 20, 25, and so on. An integer can be the product of two or more primes, e.g. 35 is a multiple of both 7 and 5. In such cases, by convention, the largest factor wins, so 35 belongs to the 7 family.

Two integers are *relatively prime* if they have no common prime factors. Consider 6 and 35: neither is prime, but they have no common factors other than one ($6 = 2 \times 3$, $35 = 5 \times 7$), thus they are relatively prime.

Juxtaposing prime or relatively prime loop lengths results in a "slipping" effect that's characteristic of [polymeter](#). The loops shift phase relative to each other, hence the effect is known as *phasing*. Shown below is the juxtaposition of two loops, having lengths of 3 and 5 respectively. Notice that each time the 3 loop restarts, the 5 loop is in a different position, and vice versa; this behavior is typical of phasing.

123123123123123
123451234512345123451

After 15 steps, the two loops restart at the same time, and their combined cycle begins again. The point at which the loops are back in phase is called their [convergence](#). If both lengths are prime, as in this case, the distance to the convergence is easy to calculate: it's simply the product of the lengths ($3 \times 5 = 15$). For harder cases, use the [Convergences Calculator](#).

Modular arithmetic

In modular arithmetic, numbers "wrap around" when they reach a certain value, called the *modulus*. Modular arithmetic is common in music, particularly modulo 12, due to the chromatic scale having twelve notes. This application uses modular arithmetic for many purposes, for example [index modulation](#).

A familiar example of modular arithmetic is a digital clock: at midnight, the clock shows 00:00, and an hour later it changes from 00:59 to 01:00, because the seconds wrap around back to zero. In other words, seconds are modulo 60. For integers, the operation is equivalent to the remainder of division by the modulo, except that negative results are handled specially. The figure below shows modulo 3 for a range of input values.

x	x mod 3
5	2
4	1
3	0
2	2
1	1
0	0
-1	2
-2	1
-3	0
-4	2

-5	1
-6	0

Clipping

Clipping is when a value that would otherwise exceed its permissible range has to be truncated. Clipping can occur at either or both extremes of the range. [Steps](#), note velocities and MIDI control values are all seven-bit numbers which allow 128 unique values, ranging from 0 to 127 for [unsigned](#) values, or -64 to +63 for signed values. Applying an excessive offset to step values via [track velocity](#) or [velocity modulation](#) can cause the values to exceed their range, in which case they must be truncated or *clipped*.

For example, suppose a track's step values range from 0 to 127. Offsetting the track's velocity by 100 causes the steps to range from 100 to 227, but since values exceeding 127 are clipped to 127, the result has a much smaller range, from 100 to 127. Furthermore, any details that were present in the clipped range (128 to 227) are lost. Squashing of range and loss of detail are both characteristic of clipping.

Clipping is potentially an issue for all [track types](#), but it's a particular concern for [note tracks](#) because the range of note velocities determines the dynamic range of the resulting music. Unintentional clipping of note velocities can reduce the subtlety of your composition. One way to detect it is by using the Color Velocities option in the [Piano bar](#). If you see a lot of bright red or dark blue notes, you're probably clipping. Another method is to use the [Velocity Range](#) command on your note tracks: it reports the range of velocities in use, taking all offsets into account, and also reports clipping.

Unicode

The application generally supports Unicode, which means non-ASCII characters are allowed in [track names](#), as well as [part](#) and [preset](#) names. All valid UTF-16 code points are allowed. Note however that in [exported](#) MIDI files, non-ASCII characters will be replaced by question marks, because the MIDI file format only supports ASCII track names. This limitation also applies to the [Export Song](#) command, because it uses the MIDI export internally.

ASCII here means *printable* ASCII, excluding control characters, in other words the 7-bit characters ranging from 32 to 126 decimal; characters outside this range are considered non-ASCII.

The [Graph bar](#) optionally supports Unicode, but the option is disabled by default because it increases overhead. To enable it, set the [Graph Handles Unicode](#) option to true.

Undersized graph text

If the [graph](#) text is unexpectedly small, the most likely cause is that the system display scaling is greater than 100%, as is often the case when using a high-resolution monitor. To determine whether display scaling is in fact the cause, check its value in Settings / Display / Scale and layout. If display scaling is currently greater than 100%, change it to 100%, and then exit and restart Polymeter. If the graph text is now appropriately sized, display scaling is the cause. Restore the previous display scaling and apply one of the fixes below.

The root cause is that the web browser control is not DPI-aware. An automated fix for that is available from the Polymeter download page. It's a zip archive named Polymeter-Graphviz-small-text-fix.zip, containing a REG file and instructions. The fix makes the web browser control DPI-aware for the Polymeter application only. If you prefer to apply the fix manually using the Registry Editor, follow the instructions below. Note that elevation to administrator privileges may be required.

To fix the problem for the current user *only*, create the following key:

HKEY_CURRENT_USER\Software\Microsoft\Internet

Explorer\Main\FeatureControl\FEATURE_96DPI_PIXEL

and in that key, create a DWORD value named Polymeter.exe with a value of 1.

To fix the problem for ALL users, create the following key:

HKEY_LOCAL_MACHINE\Software\Microsoft\Internet

Explorer\Main\FeatureControl\FEATURE_96DPI_PIXEL

and in that key, again create a DWORD value named Polymeter.exe with a value of 1.

The fix for all users shown above only affects the 64-bit version of Polymeter. For the 32-bit version, the key is instead:

HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Internet

Explorer\Main\FeatureControl\FEATURE_96DPI_PIXEL

If the above fixes don't help, you can also try enabling the [Use Cairo](#) option.

Reference

Errors

Only application-specific errors are listed here. For generic errors, see the Windows documentation.

Sequencer errors

Error	Message	Severity	Comment
Buffer Overrun	MIDI event buffer size exceeded.	Fatal	Too many MIDI events were scheduled during a callback, exceeding the space allocated for them. This

			error could conceivably occur with extremely dense sequences. Try increasing the buffer size .
Bad Time Format	Unsupported MIDI time format.	Fatal	The application couldn't obtain a current position from the MIDI subsystem in the requested format. This error is unlikely, but if it occurs it's possibly a driver issue. Try using a different output device .
Too Many Recursions	Modulation exceeded the maximum number of recursions.	Fatal	The sequencer exceeded the maximum permissible depth while traversing a nested modulation . This error is usually caused by a modulation loop, in which a track directly or indirectly modulates itself. Try running the check modulations command. If it's not a loop and you actually need more depth, fork the code, edit Sequencer.h and increase MOD_MAX_RECursions.
Callback Too Long	Sequencer callback took too long.	Warning	The sequencer took too long generating the next chunk of MIDI data, and delivered it too late. This condition is called an underrun , and it may produce an audible gap in the output. It could be caused by an extremely complex sequence, an unusually slow output device, or setting the latency too short. If it occurs, try increasing the latency. This error is only reported once during a given playback of the document. You can determine how many times the error actually occurred by inspecting CB Underruns in the sequencer statistics .

User interface errors

Error	Message	Comment
Bad Format	'[path]' has an invalid format.	The most likely causes are opening a file that isn't a Polymeter document, or opening a damaged document. This error also occurs if you inadvertently convert a Polymeter document to Unicode, for example by editing it with a text editor. Try comparing the problematic document to one that opens correctly, using a file comparison utility.
Bad Step Selection	Invalid step selection.	This occurs if the step selection isn't <i>rectangular</i> , and the attempted edit requires a rectangular selection. A rectangular selection includes the same number of steps in each of its rows. Non-rectangular selections can occur when selecting steps within multiple tracks that have different lengths , and especially when selecting near the end of a track.
Can't Distribute	Can't distribute unless source count matches target count.	This can occur when Distribute is checked in the Modulation dialog. Make sure you've selected one source track for each selected target track.

Can't Truncate Recording	Can't set song length shorter than the recording.	This can occur when attempting to shorten the song via the Song Length property. It occurs because the song contains one or more dubs after the specified length. Delete all such dubs and then try setting the song length again.
Clip Warning	Clipping may occur.	This occurs when a Velocity command would cause one or more steps to exceed the valid range of a step. It also occurs if a Transpose command would cause one or more tracks to exceed the range of a MIDI note. If you proceed despite the warning, the offending values are clipped .
Convergence Out of Range	Convergence is out of range.	This occurs with the Next Convergence and Previous Convergence commands, if the next or previous convergence would exceed the range of a position in ticks , which is \pm 2.147 billion, the range of a 32-bit integer. It's likely to occur if the convergence is calculated for many loops with different relatively prime lengths, and the Convergence Size is set to a large number or All. Try lowering the convergence size.
Group Overlap	One or more of the selected tracks already belong to a group, and will be reassigned.	This occurs when using the Group command or the Parts bar Update command. It occurs because a track can only belong to one part at a time. If you proceed despite the warning, the offending tracks are removed from the parts they currently belong to, before being made members of the new or updated part.
Newer Version	'[path]' is from a newer document version. Some of its features may not be supported.	This occurs if you open a document that was created by a more recent version of Polymeter than the one you're using. The result depends on which features were added since your version, and whether the document you're opening uses them. The document probably won't work as intended. The solution is to upgrade to the latest version.
Search Text Not Found	Search text not found.	The Find and Replace commands give this warning if no track has the specified name.
Track Length Cursor	Cursor must be over track step to indicate desired length.	This occurs when the Track menu's Length command is unable to determine the intended length. Specify the track length either via a step selection or by positioning the cursor over the desired last step.
Track Too Long	Track too long.	This occurs if one or more tracks would exceed the maximum track length . It occurs when editing the track Length or Quant .

Value Not Found	Value not found.	The Replace Velocity command gives this warning if none of the selected steps match the specified value or range of values.
Export Bad Duration	Invalid duration. Time format is HH:MM:SS and duration must be greater than zero.	This occurs with the Export or Export Song commands if the specified duration is improperly formatted or otherwise unacceptable. The expected format is hours, minutes and seconds separated by colons.
Export Error	Error during export.	The Export command failed.
Grid Error Bad Time Format	Enter time in hh:mm:ss format.	This occurs when entering a time in the Properties bar , for example in the Song Length field. The expected format is hours, minutes and seconds separated by colons.
Hyperlink Can't Launch	Can't launch browser.	The application was unable to display a web page. It could happen with the Polymeter on the Web command. The most likely causes are no Internet connection, or the web site is unavailable. To rule out the former, try your browser.
Import Error Format	Format error.	This occurs with Import Tracks if the input CSV file contains data that doesn't conform to the expected format of the corresponding track property. The message includes the one-based row and column at which the error was found.
Import Error Property	Undefined property name.	This occurs with Import Tracks if the input CSV file's mandatory header row specifies an unrecognized property name. To obtain a list of valid property names, use the Export Tracks command and examine the resulting CSV file.
Import Error Range	Range error.	This occurs with Import Tracks if the input CSV file contains data that exceeds the range of the corresponding track property. The message includes the one-based row and column at which the error was found.
MIDI Devices Missing	MIDI devices are missing.	This means a selected MIDI device is no longer available. Both input and output devices can give this error. It typically occurs when a USB MIDI device is unplugged or powered off. You can recover by making the device available again (for example by plugging it back in, or powering it back on) and then clicking Retry. If you click Cancel, the device selection is reset.
MIDI Incompatible Song Position	Song position is incompatible with MIDI clock sync. Playback	This warning can occur when starting playback with Send MIDI Clock enabled. The sequencer's starting song position is either misaligned or exceeds the

	will start from the nearest sixteenth note.	range of MIDI clock sync. The MIDI Song Position Pointer message can only cue to sixteenth notes, and is limited to 16384 sixteenth notes. If you click OK, the song position is quantized to the nearest acceptable value. To avoid the warning, start playback from a compatible position.
MIDI No Output Device	No MIDI output device is selected.	This occurs if you attempt to start playback with no output device selected. It's most likely to occur after abandoning a missing device by selecting Cancel in the MIDI Devices Missing dialog (see above). The solution is to select an output device.

Options Summary

MIDI	
Input Device	Selected MIDI input device
Output Device	Selected MIDI output device
Latency	Desired MIDI latency, in milliseconds
Buffer Size	Size of MIDI event buffer, in bytes
Thru	Pass MIDI input through to output device
Default Velocity	Default note velocity or controller value for new steps
Input Quantization	Quantization for converting MIDI input to steps
Record Input	Type of data captured by the Record command
Send MIDI Clock	Transmit MIDI synchronization messages to slave device
View	
Update Frequency	Frequency at which view is updated, in Hertz
Show Current Position	Indicate current position within each track
Show Note Names	Display notes as names instead of numbers
Show General MIDI Names	Display General MIDI patch and drum names instead of numbers
Show Quant as Fraction	Show quant as a fraction instead of in ticks
Show Track Colors	Allow tracks to be assigned custom colors
Unique Track Names	When renaming multiple tracks, append number to keep names unique
Zoom Step	Zoom step size, as a percentage
Live View Font Height	Height of live view font, in pixels
Graph	
Graphviz Folder	Location of Graphviz executable files
Graph Handles Unicode	If true, graph handles Unicode characters in track names

Use Cairo	Render graph with Cairo graphics library
General	
Show Property Descriptions	If true, show descriptions area in properties bar
Automatically Check for Updates	Automatically check for newer version of app during startup
Always Record	Always record dubs during playback and automatically save the recordings

Shortcuts

Alt+Backspace	Undo	Undo the last action
Ctrl+Space	Play	Start or stop playback
Ctrl+Shift+Space	Pause	Pause or continue playback
Alt+Left	Previous Convergence	Go to previous convergence
Alt+Right	Next Convergence	Go to next convergence
Insert	Insert	Insert a new item
Ctrl+Insert	Copy	Copy the selection and put it on the Clipboard
Shift+Insert	Paste	Insert Clipboard contents
Delete	Delete	Delete the selection
Shift+Delete	Cut	Cut the selection and put it on the Clipboard
Ctrl+0	Zoom Reset	Reset zoom
Ctrl+1	Rotate Left	Rotate the selected tracks or steps left
Ctrl+Shift+1	Shift Left	Shift the selected tracks or steps left
Ctrl+2	Rotate Right	Rotate the selected tracks or steps right
Ctrl+Shift+2	Shift Right	Shift the selected tracks or steps right
Ctrl+3	Reverse	Reverse the selected tracks or steps
Ctrl+A	Select All	Select the entire document
Ctrl+Alt+A	Show Parts	Show or hide the parts bar
Ctrl+C	Copy	Copy the selection and put it on the Clipboard
Ctrl+Alt+C	Show Channels	Show or hide the channels bar
Ctrl+D	Deselect	Unselect the entire document
Ctrl+E	Export	Export to a MIDI file
Ctrl+Shift+E	Stretch	Stretch or shrink the selected tracks
Ctrl+Alt+E	Show Presets	Show or hide the presets bar
Ctrl+F	Find	Find the specified text
Ctrl+Shift+F	Fill	Fill selected tracks or steps with a range of values

Ctrl+G	Create Group	Group the selected tracks into a part
Ctrl+H	Replace	Replace specific text with different text
Ctrl+I	Import	Import tracks from a MIDI file
Ctrl+Shift+I	Invert	Invert the selected tracks or steps
Ctrl+J	Rotate Steps	Rotate the selection left or right a given number of steps
Ctrl+Shift+J	Shift Steps	Shift the selection left or right a given number of steps
Ctrl+L	Length	Set the length of one or more tracks
Ctrl+Shift+L	MIDI Learn	Learn mappings from MIDI input
Ctrl+Alt+L	Loop	Loop playback
Ctrl+M	Mute	Mute or unmute the selected tracks
Ctrl+Shift+M	Create Modulation	Create a modulation that targets the selected tracks
Ctrl+Alt+M	Show Modulations	Show or hide the modulations bar
Ctrl+N	New	Create a new document
Ctrl+O	Open	Open an existing document
Ctrl+Shift+O	Options	Edit the options
Ctrl+Alt+O	Offset Tracks	Offset the selected tracks by a given number of ticks
Ctrl+Shift+P	Create Preset	Create a new preset from the track mutes
Ctrl+Alt+P	Show Properties	Show or hide the properties bar
Ctrl+R	Record	Start or stop recording
Ctrl+Shift+R	Record Tracks	Start or stop recording MIDI input to tracks
Ctrl+S	Save	Save the active document
Ctrl+Shift+S	Save As	Save the active document with a new name
Ctrl+Alt+S	Sort	Sort the selected tracks
Ctrl+T	Tie Notes	Toggle tie notes mode; if on, new notes are tied
Ctrl+Shift+T	Transpose	Transpose the selected tracks
Ctrl+U	Select Unmuted	Select all unmuted tracks
Ctrl+V	Paste	Insert Clipboard contents
Ctrl+Shift+V	Velocity	Transform the velocities of the selected tracks or steps
Ctrl+W	Rewind	Rewind to start of song
Ctrl+X	Cut	Cut the selection and put it on the Clipboard
Ctrl+Y	Redo	Redo the previously undone action
Ctrl+Z	Undo	Undo the last action
F1	Help Topics	List help topics
F2	Rename	Rename the selected item
Ctrl+F4	Close	Close the active MDI child window

Alt+F4	Exit	Exit the application
F6	Next Pane	Switch to the next window pane
Shift+F6	Previous Pane	Switch back to the previous window pane
Ctrl+F6	Next Document	Activate the next MDI child window
Ctrl+Shift+F6	Previous Document	Activate the previous MDI child window
F7	Show Velocities	Show or hide the track view's velocities pane
F8	Track View	Show the track view
F9	Song View	Show the song view
Ctrl+F9	Live View	Show the live view
F11	Full Screen	Enter or exit full screen mode
Ctrl+=	Zoom In	Zoom in
Ctrl+-	Zoom Out	Zoom out
Ctrl+/-	Go To Position	Go to specified position within song
Ctrl+\	Solo	Solo the selected tracks
Ctrl+Tab	Next Document	Activate the next MDI child window
Ctrl+Shift+Tab	Previous Document	Activate the previous MDI child window