Gig Performer 3
User Guide

v. 3.0

January 2019

https://www.gigperformer.com/

Version 3 update by Larry the O, Toys In the Attic Communications

Original manual written by Howard Massey

Manual version f4 19Mar19

INTRODUCTION

Welcome to GigPerformer—the world’s most versatile plugin host application for live performance. GigPerformer lets you take complete control of your entirerig and provides instant access to all of your instrument and effects plugins—no DAW software necessary.

Whether you’re a keyboard player, a guitarist, a vocalist, a drummer, or a live sound engineer, Gig Performer has much to offer:

- Keyboardists can control all theirsoftware synths, samplers, and sequencers—as well as effects—directly from the sliders, knobs and switches on their instruments.

- Using their MIDI pedalboards, guitarists can freely switch between multiple amp/cabinet models and digital effects in real time without any audio glitching.

- Singers can easily add compression, EQ and other desired effects to their vocals.

- Setlists let you effortlessly change sounds and entire plugin setups for every part of every song of every gig.

- Live sound engineers can use Gig Performer to intercept audio coming from the FOH or monitor system, add desired effects processing, and then send the results directly back to the mixing board. Gig Performer can simultaneously record multichannel audio from the mixing board for later processing and mixing in your favorite DAW.

- You can even run multiple Gig Performer instances simultaneously, allowing different members of a band toaccess different plugin setups independently!

Gig Performer’sintuitive interface makes it remarkably fast to start with, and easy to use. Beneath the surface, however, are many advanced features, which we willcover in the pages of this User Guide. The User Guide is basically divided into two sections: a series of Tutorials to familiarize you with basic operation, and a Reference section that documents all of the details.

Gig Performer Features (Only Some of Them)

- Supports VST2 and VST3 (OS X and Windows) and AU (OS X only) plugins
- Runs on both Mac (OS X) and Windows platforms so you can create, share, and modify your files on any computer, even if some plugins are missing
- Minimal CPU usage
- Rock solid performance, extremely stable
- Easily play multiple instruments and controllers, layer sounds, and add effects
- Intuitiveuser interface: simply connect any plugin to any other plugin in any configuration—no confusing busses, sends or inserts
- Build fully customizable rackspaces consisting of one or more front panels
- Instantaneous glitch-free switching between sounds—even while holding notes—for seamless transition from one song to the next or from one part of a song to another
- Use keyboards, MIDI pedals, and control surfaces to alter plugin parameters in real time by connecting them to knobs, sliders, buttons, labels, meters, and other widgets
- Learn mode for easy assignment of MIDI controls to widgets and widgets to plugin parameters
- Build setlists that automatically change to the right sound for each part of each song in the list. Capture on-the-spot tweaks right into song parts. A gig becomes as simple as stepping through its setlist.
- Support for ChordPro. Create and edit ChordPro song documents, and set each to be automatically displayed when its song is selected in the setlist.
- Automatic mixing of signals when multiple outputs are connected to a single input
- Deactivate any plugin at any time to remove a sound or bypass an effect

- Widget scaling allows full motion while restricting the parameter range
- Widget groups for making cross-faders, grouped level controls, and other ganged behaviors.
- Rig Manager simplifies switching a Gig Performer rig to different physical controllers
- Audio Mixer plug-in forcentralized gain controlsand instant rerouting of sources to effects
- LEDs show input/output levels on each channel as well as audio and MIDI activity
- Built-in full screen chromatic tuner
- Audio metronome using built-in or user-supplied sounds
- Create multiple independent running instances for different members of the band or for when you want to control some plugins while leaving others alone
- Audio Input Recorder captures audio from input channels to WAV files for post-production editing, processing, and mixing in a DAW
- MIDI Recorder captures MIDI input from all devices, placing each on its own track of a Type 1 Standard MIDI File that can be imported for post-production in a DAW.
- Support for multiple audio interfaces
- Predictive Loadingreduces RAM and CPU demand in systems with limited resources
- Automatic detection of new MIDI devices—connect and reconnect on the fly
- Full MIDI remapping, transposition, layering, and keyboard splitting
- Send program change and other messages to external MIDI devices automatically
- VST Bus Layout support

- OSC supportfor remote control viatablets and smartphones

- Compatible (via OSC) withMax/MSP and Bidule for the development of complex interactive control algorithms

- GP Script proprietary programming language for complex orcustom needs

Be sure to check http://www.gigperformer.com/features-and-benefits.html for a full list of current features.

Going Above and Beyond: GP Script

Most users will find everything they need in Gig Performer's extensive feature set, but there are always some users that require sophisticated custom functions that go beyond Gig Performer's main feature set. For those users, we have created a proprietary programming language called GP Script. GP Script enables you to perform incredible transformations and perform complex tasks. While any language takes some understanding of programming, GP Script is written to be accessible. If you can program spreadsheet macros or in a language such as VBA, you are likely to be comfortable with GP Script.

Because GP Script will not be needed by most people and requires substantial documentation of its own, we will not cover it in this manual. However, if you are interested,you can find an introduction to the GP Script language in this blog entry:

https://www.gigperformer.com/the-gpscript-programming-language

GP documentation can be found here:

https://gigperformer.com/downloads/GPScript/SystemFunctionList.html

https://gigperformer.com/downloads/GPScript/LanguageManual.pdf

For discussion of GP Script and help getting it to do what you need, check out the GP Script section of the Gig Performer forums:

https://www.gigperformer.com/forums.html

Basic Terminology

Connections View: The Gig Performer screen in which you can connect pluginsto one another.

Gig:A Gig Performer file containing all the data you have created—rackspaces, variations, and all widgets and widget groups.

Instance:Multiple copies of Gig Performer can be launched and run simultaneously on a single computer. Each of these independent copies, running in its own window, is called an "instance" of Gig Performer. This is useful for applications like giving different band members the ability to change the rackspace affecting their particular instrument without changing the rackspaces affecting different instruments. Click here for more information.

MIDI:An acronym for Musical Instrument Digital Interface—a standard protocol for the interchange of musical information between computers, controllers, and hardware-based electronic musical instruments such as synthesizers and samplers.

MIDI controller:A physical device such as a keyboard or pedalboard typically connected to your computer via USB.

OSC:An acronym for Open Sound Control—a communications protocol that is optimized for modern networking technology. Click here for more information.

Panel:In Gig Performer, a virtual screen space (visually similar tohardwarerack-mounted sound modules and signal processors) containing all the widgets needed to control the plugins in a rackspace. A rackspace can contain multiple panels, each containing one or more widgets.

Performer View:The Gig Performer screen containing the panels and widgets used to control your sounds during performance.

Plugin:A package of functionality that can be employed by any compatible host software. There is a tremendous variety of plugins available in the audio/music marketplace, but many popular plugins are software emulations of physical instruments or hardware devices. Originally developed for use within DAW (Digital Audio Workstation) applications, some plugins (called "virtual instruments")generate audio, for example, emulations of musical instruments and synthesizers, while effects plugins alter incoming audio signals, such as from a microphone, electric guitar, or virtual instrument, offering processes like equalization, filtering, compression, limiting, reverb, delay, flanging, chorusing, phasing, and the like.

Rackspace:A collection of interconnected plugins, along with one or more panels containing widgets to control plugin parameters in real time. A rackspace can actually be pretty much anything you like: It could represent a single sound or all the elements you need for an entire song—or even just one part of a song. You can switch from one rackspace to another instantaneously—and with no interruption of sound—either from your laptop, with up/down pedals, or via MIDI program change messages.

Setlist: A collection of songs arranged in a desired order. Setlists are assembled from the library of available songs; you might have one setlist for a set you perform regularly and other setlists for individual performances requiring different songs or a different order. A song in a setlist is made active simply by clicking on it.

Song: A collection of song parts arranged in a desired order. Songs can execute actions when they are made active, such as sending out a MIDI program change message, overriding the current tempo with the song tempo, or opening a ChordPro file showing chords and lyrics. A library of songs can be built up, making them available to be used in setlists.

Song Part: An individual section of a song, such as a verse, chorus, bridge, or interlude. Selecting a song part makes its associated rackspace variation active, and can perform other actions like overriding the current tempo or sending out a custom string of MIDI messages. As the lowest level building block of setlists, song parts can be easily accessed by MIDI program change commands from an external controller, the large song part tiles in the Setlist/Song view, or by song part tiles in the Gig Performer Lemur template (or another custom OSC template).

Variation:These are essentially rackspace presets, all containing the same plugins and interconnections,but with different widget settings. For example, you can create a rackspace to add phasing to a piano sound, and thenadd several variations, each with different degrees of phase intensity. Since variations actually store widget settings, every parameter you want stored in a variation must be assigned to a widget on a rack panel. As with rackspaces, you can switch from one variation to another instantaneously—and with no interruption of sound—either from your laptop, with up/down pedals, or via MIDI program change messages.

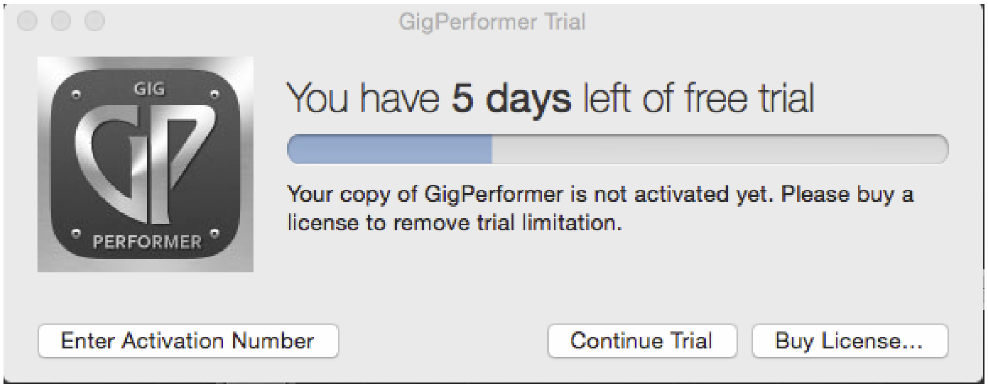
Widget:A virtualobject (such as aknob, slider, button, or meter) used to control and/or display plugin settings. Widgets can be operatedeither withyour laptop's mouse or trackpad, or remotely from a MIDI controller. With the use of OSC, widgets can also be operated from iOS / Android tablets or smartphones.

Widget group:A collection of widgets that operate ganged together, where changing one widget in the group changes all of the others in the group, as well. This is a powerful method for controlling multiple plugin parameters simultaneously. For example, you could create several slider widgets that control the volume of various plugins and then raise or lower the volume of all of them by moving any one of them. In addition, individual widget scaling allows simple implementation of effects such as cross-fading. For more information, see the widget group reference section.

Before You Start

Before launching Gig Performer, **make sure that all audio and MIDI interfaces connected to your systemare powered on**. If no external audio interface is connected, Gig Performer will use your computer's built-in input and output. If you have more than one interface connected, you'll be able to choose among them after the app launches. Note that you can use different interfaces for input and output if desired.

When Gig Performer is first launched, the following dialog will appear:


GigPerformer Trial

You have **5 days** left of free trial

Your copy of GigPerformer is not activated yet. Please buy a license to remove trial limitation.

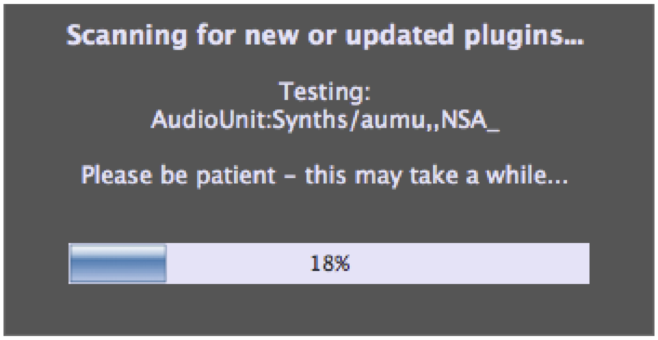
If you have purchased a license, click on Enter Activation Number and type or paste in the activation code emailed to you. (Make sure your anti-spam blocker doesn't prevent the confirmation email with your license from being received).

If you have not yet purchased a license,click on Continue Trial to accessa fully operational trial version of Gig Performerfor a period of 14 days. (The dialog shows you how many days you have remaining.). You can purchase an activation code at any time by clicking on the Buy License... button.

Note: When Gig Performer is in trial mode, there will be momentary dropouts in the audio and the screen will flash red.

Before using Gig Performer, please make sure you agree to the license agreement, a copy of which is included here.

Immediately after launching Gig Performer, it will scan your system looking for plugins and the following dialog will appear:



Scanning for new or updated plugins...

Testing:
AudioUnit:Synths/aumu,,NSA_

Please be patient – this may take a while...

18%

When running Gig Performer on Mac (OS X) systems it will scan for both VST (Virtual Studio Technology) and Audio Unit (AU) plugins; when running on Windows systems it will look for all VST plugins.

Note: If you are a Windows user, you will have to tell Gig Performer what folders to search, since there are no default locations where all plugins are guaranteed to be installed. To do so, go to the Windows menu, then select "Plugin manager." In the resultant window, click on Manage, then choose the "Set VST folders for scanning" option. We suggest you begin by having Gig Performer search in the folder C:\Program Files\VstPlugins. If you have additional plugins stored in other folder locations (as some plugin and DAW manufacturers automatically create), the Plugin manager allows you to easily add other folders to scan. Click here for more information.)

This scanning process can take some time the first time you use Gig Performer but will be nearly instantaneous on subsequent launches unless you have installed or updatedany plugins since the last usage. (If you like, you can later disable this scan completely with an option in the Preferences dialog.)

Gig Performer does not include any synth or effects plugins because there are so many available plugins out there—both free and commercial—we're sure that most users will want to use their own favorites. (Click here for a list of some of our favorite free ones.) However, it does come with a number of specialized plugins to handle basic MIDI and audio operations.

A Word About Cross-Platform Usage

Files created by Gig Performercan be opened byboth Mac (OS X) and Windows computers. If the computer you're using to access the file is missing some of the plugins that were present when the original file was created, Gig Performer will create a "chameleon" plugin that mimics all the properties of the original plugin except the actual processing. This will allow you to modify the file, save it and later re-open it on a computer that has all required plugins installed.

Similarly, if the audio interface connected to the current system does not have enough inputs or outputs to accommodate the setup utilized by the file, Gig Performer will prompt you to create "Virtual channels" that produce no input or output but retain all connections. Doing so will allow you to modify the file, save it and later reopen it on a computer that is connected to the original audio interface.

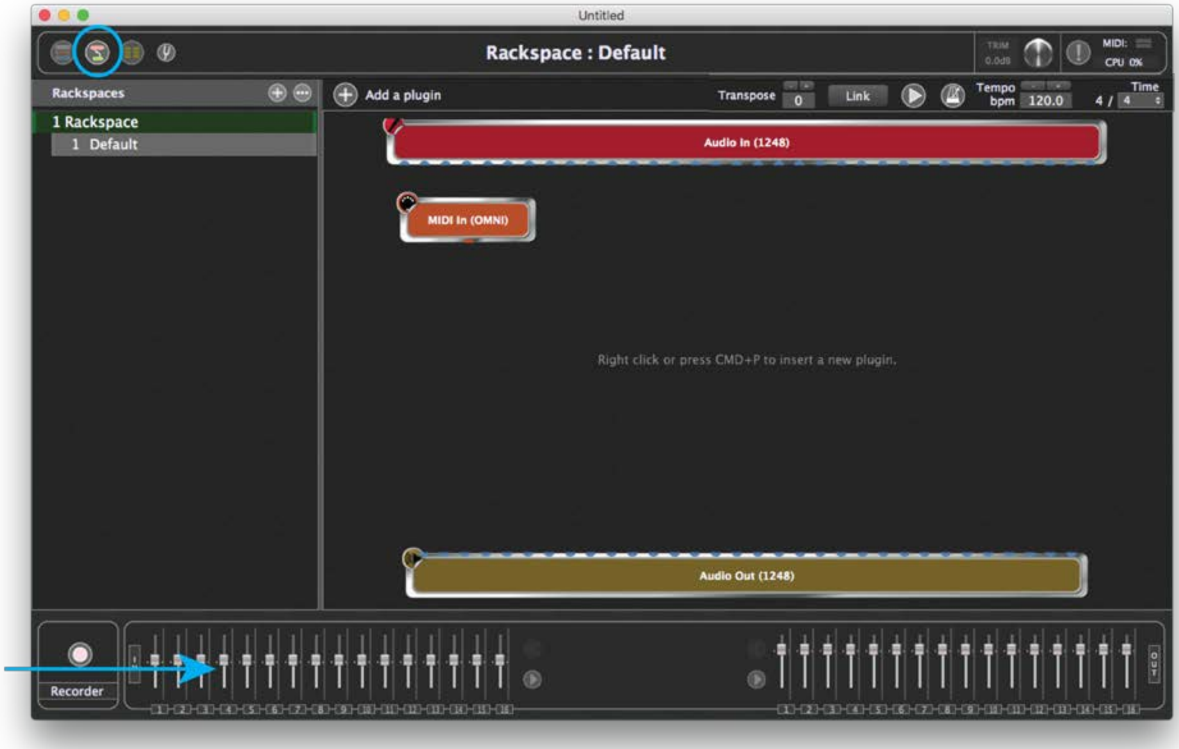
USING GIG PERFORMER

This section presents a series of tutorials to familiarize you with Gig Performer's major features. For more detail, please see the Reference section.

Controlling Plugins From Your Midi Controller

This tutorial will show you how to use Gig Performer to play plugins fromyourMIDI controller.

1. **Launch Gig Performer**, then click on the Connections View icon (circled in blue) to enter Connections View:

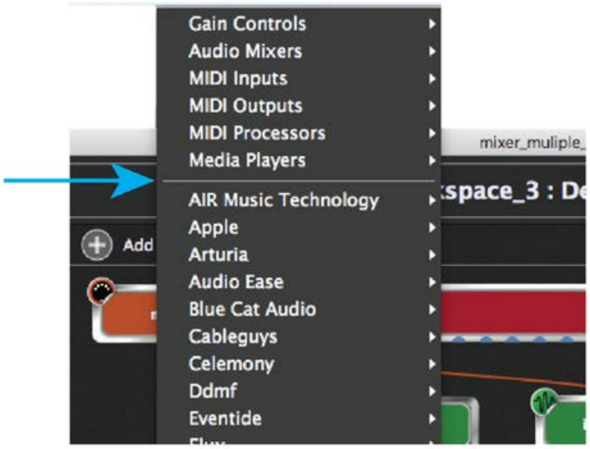


This screen allows you to insert plugins and connect them to other plugins any way you like.

NOTE: Your screen may look slightly different from the one shown above, since the audio input and output blocks will reflect the name of your connected audio interface as well as the number of inputs and outputs it provides.)

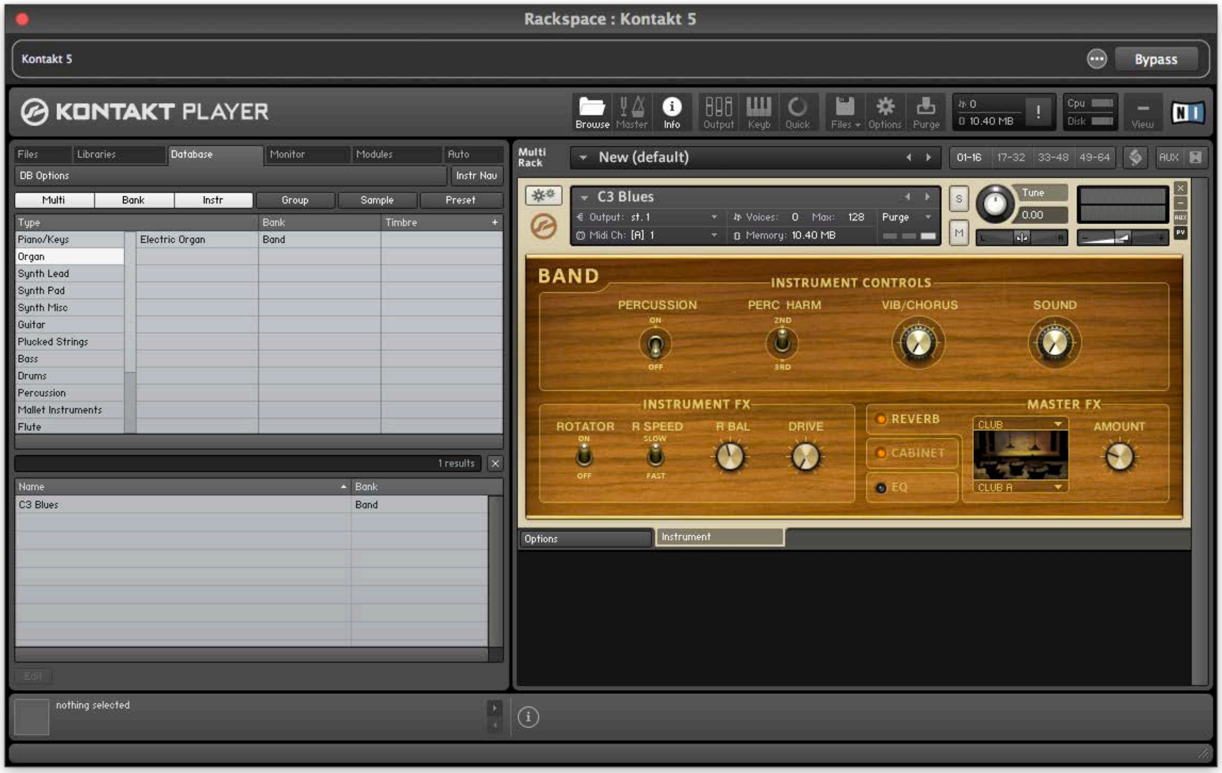
The orange block at top left represents the MIDI input to your system, typically from a MIDI keyboard or other MIDI controller. The red block at the toprepresents the input of your audio interface, typically from microphones, instruments connected via DI (such as electric guitars), and line-level sources. The brown block at the bottom represents the output of your audio interface, and the panel beneath it (indicated by the arrow) provides metering and virtual faders for each of your audio interface's inputs and outputs. (Flickering in the input meters indicates presence of signal.) If no audio interface is connected, Gig Performer will use your computer's built-in input and output. If an audio interface is connected but does not appear, see the "Troubleshooting" section in this User Guide.

2. **Choose a plugin.**Right-click (or two-finger tap if you're using a trackpad) anywhere in the main screen to bring up the plugin selector menu, which allows you to choose the plugin you want to access.The built-in Gig Performer plugins (including MIDI In and MIDI Out blocks for each connected MIDI controller) appear above the line separator (indicated by the arrow), grouped together by category: Gain Controls, Audio Mixers, MIDI Inputs, MIDI Outputs, MIDI Processors, and Media Players. All other plugins appear under the separator, arranged alphabetically by the name of the plugin manufacturer:

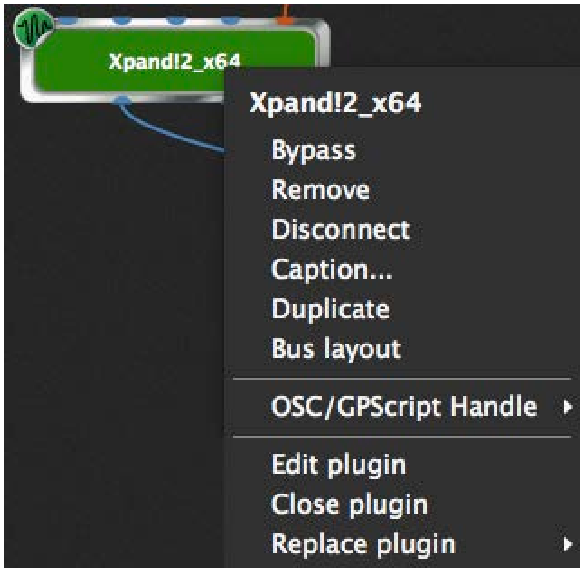


TIP: You can also insert a plug-in by using the Cmd-P keyboard shortcut. This option is actually preferable since it opens a "Quick plugin finder" window that allows you to search for a particular plugin by part or all of its name. Click here for more information.

As soon as you select a plugin block (or double-click on it), Gig Performer opens the editor window for that plugin, allowing you to alter the sound. (You can disable the “open on selection action”with an option in the Preferences dialog.) Let’s start with a Hammond organ plugin. For this example, we have selected“C3 Blues” from Native Instruments’ Kontakt 5 Player (you can choose your favorite). Its editor window is shown below:

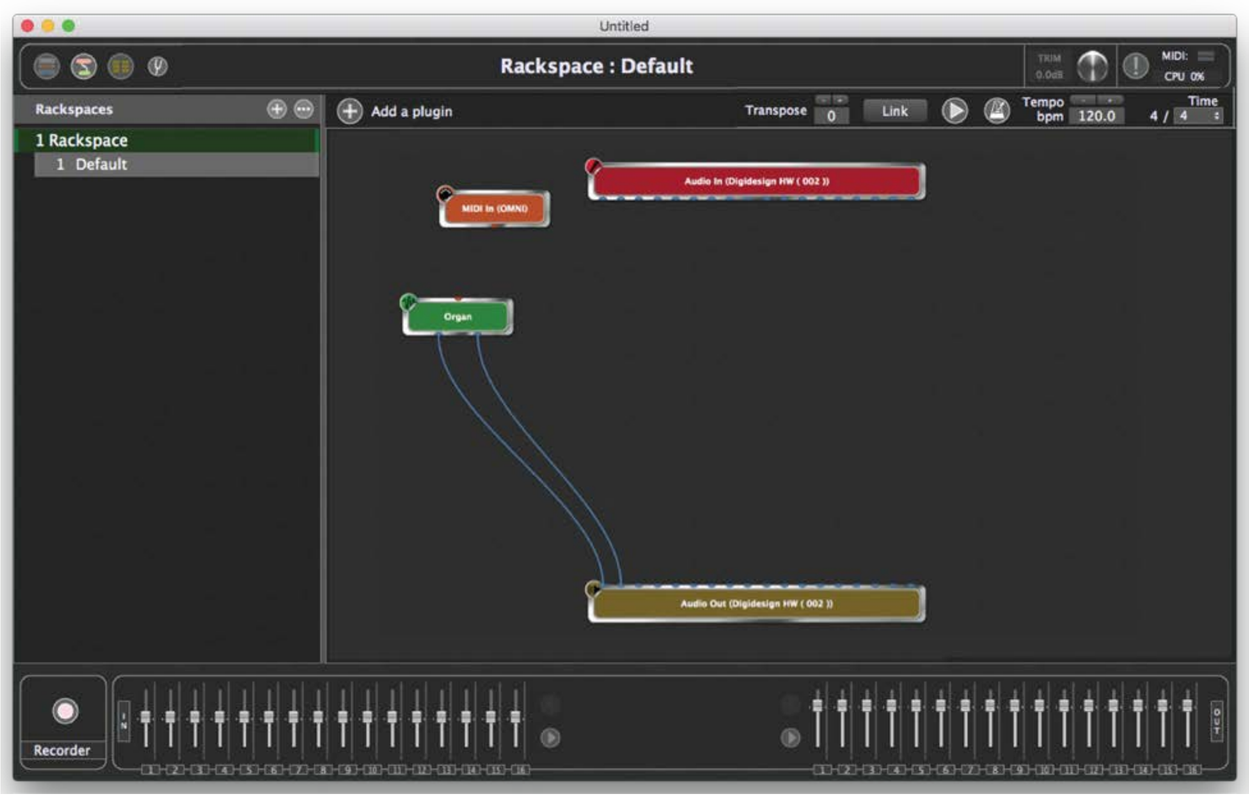


3. **Give your plugin a custom name.**Close the plugin editor window to return to Connections View. Sound-generating plugins such as the one you selected appear as green blocks. (Effects processing plugins, as we will see in the next tutorial, appear as blue blocks.) By default, the name of the plugin is displayed,but you can easily rename it with something that represents the actual sound—very useful for distinguishingamongmultiple copies of the same plugin. Accordingly, right-click on the green block and select the Caption... option, then type in the name “Organ.”



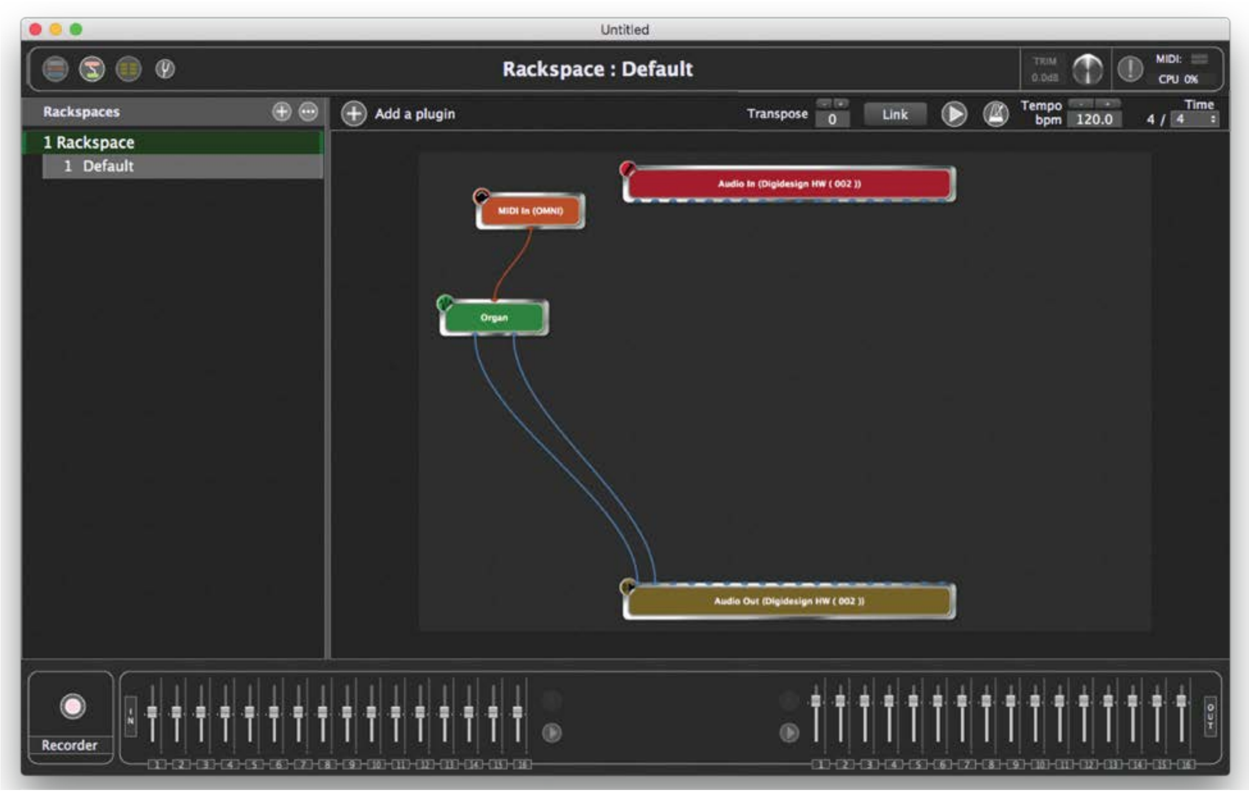
4. **Set the audio output.**Every plugin block has several pins both above and below it. These pins are used to make connections—both MIDI and audio— to otherplugin blocks. Orange pins are for MIDI signals and blue pins are for audio signals. Pins at the top provide input to the plugin, while pins at the bottom send output from the plugin. If you connect the outputs of multiple plugins to the input of another plugin, the signals will be mixed automatically.

Kontakt 5 Player creates a stereo audio signal, so there are two blue pinson the bottom. To be able to hear its sound, we need to connect its output to two inputs of the audio interface. To make these connections, simply click on an output pin and, while holding down the mouse button (or keeping your finger on the trackpad), drag the cursor over the desired input pin and then release the mouse button. (Click here for a shortcut method of making stereo connections.) Audio connections between plugins are represented by blue lines.

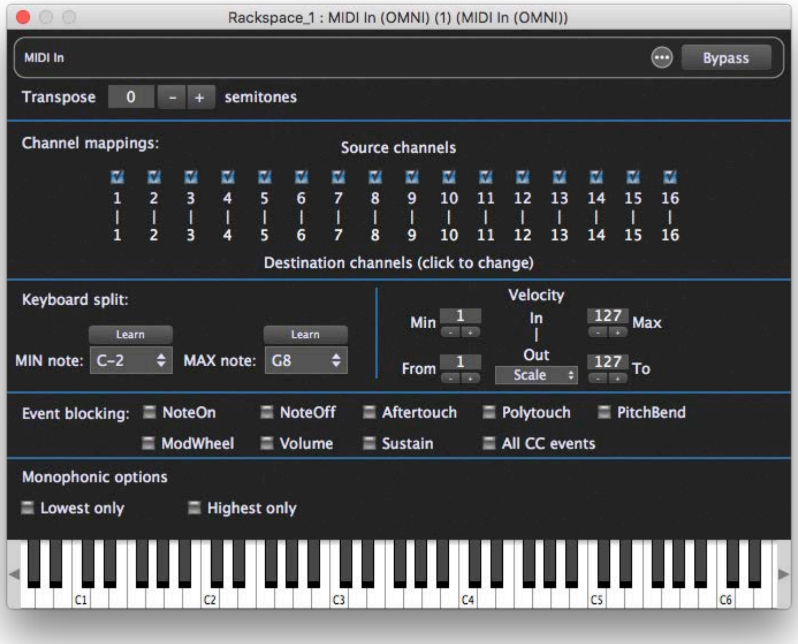


TIP:	Connections and plugin blocks can be moved anywhere in the screen by clicking and dragging on them. To remove (disconnect) a connection, right-click on it and select “Remove this connection” or simply click and drag the connection until it detaches from a plugin and then let go. To remove a plugin block, right-click on it and select “Delete.” To remove all connections from a plugin block, right-click on the block and select “Disconnect.”

5. **Set the MIDI input and play your sound.**Using the same click-and-drag technique, make a connection betweenthe orange output pin on the bottom of the “MIDI In (OMNI)” block and the orange input pin on the top of your plugin. MIDI connections between plugins are represented by orange lines.



The “MIDI In (OMNI)” plugin is included with Gig Performer and is the default MIDI input until you specify a controller-specific one (see below). In addition to passing MIDI messages generated by any connected MIDI controller, it also provides a virtual keyboard, so you can play your plugins even if you are not using an external MIDI controller.



Rackspace_1 : MIDI In (OMNI) (1) (MIDI In (OMNI))

MIDI In

Transpose 0 semitones

Channel mappings:

Source channels

☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Destination channels (click to change)

Keyboard split:

MIN note: C-2 MAX note: G8

Velocity

In: Min 1, Max 127

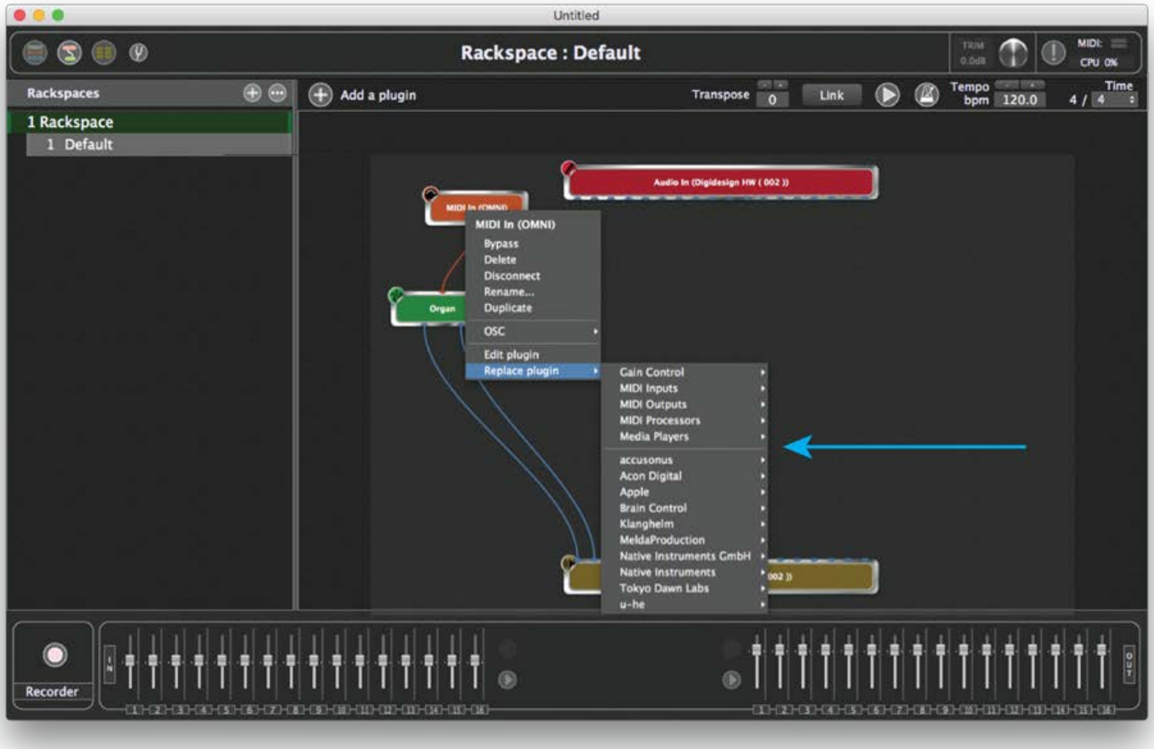
Out: From 1, To 127, Scale

Event blocking: □ NoteOn □ NoteOff □ Aftertouch □ Polytouch □ PitchBend □ ModWheel □ Volume □ Sustain □ All CC events

Monophonic options

□ Lowest only □ Highest only

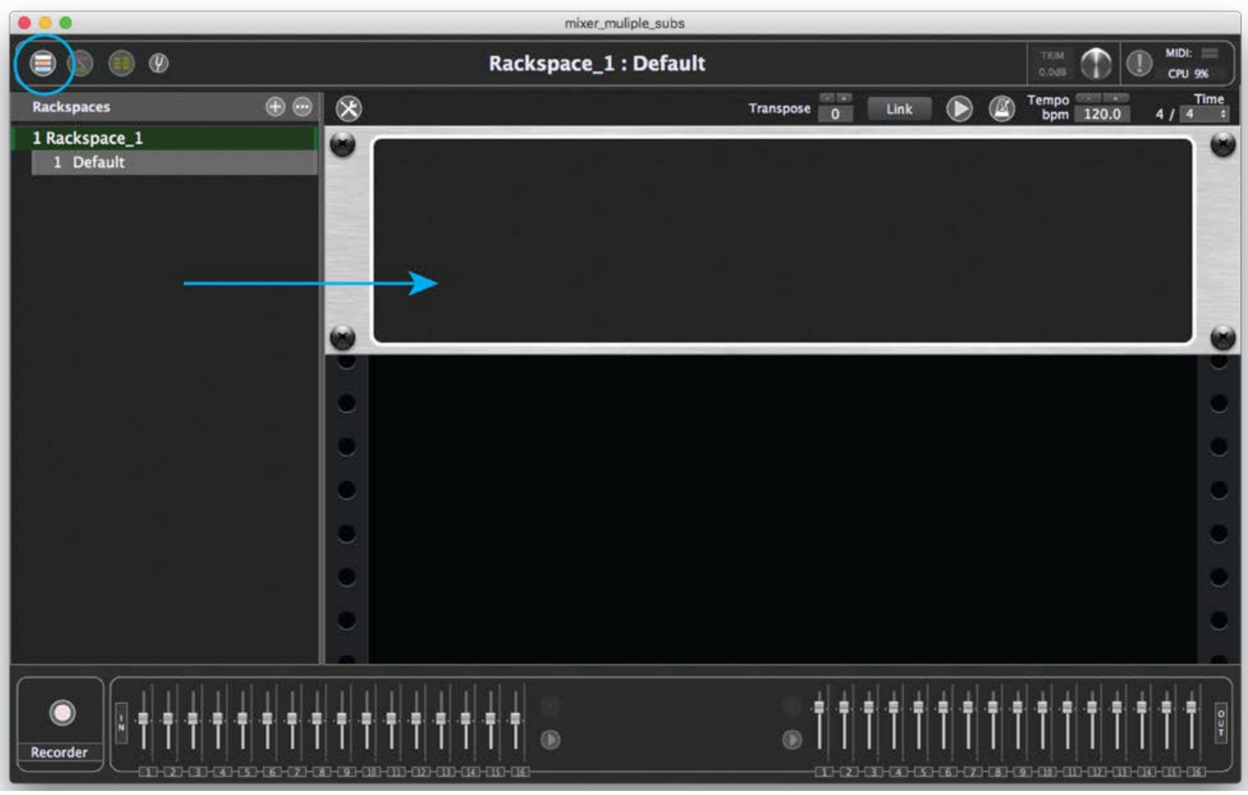
At this point, if you play a few notes on your controller (assuming your audio system is turned up), you should be hearing an organ sound coming from your connected audio interface. Alternatively, you can double-click on the "MIDI In (OMNI)" block to open its editor, then "play" the organ sound by clicking notes on the virtual keyboard. You can also opt to substitute a plugin representing your particular MIDI controller (thus allowing the plugin to be played only by that controller and/or via a specific MIDI port offered by that controller) by right-clicking on the "MIDI In (OMNI)" block and selecting the "Replace Plugin" option.



Again, all built-in Gig Performer plugins (including “MIDI In (OMNI),” as well as MIDI In and MIDI Out blocks for each connected MIDI controller) appear above the line separator (indicated by the arrow), grouped together by category: Gain Control, MIDI Inputs, MIDI Outputs, MIDI Processors, and Media Players.

For more information about the “MIDI In (OMNI)” plugin, see the Supplied Plugins section in this User Guide.

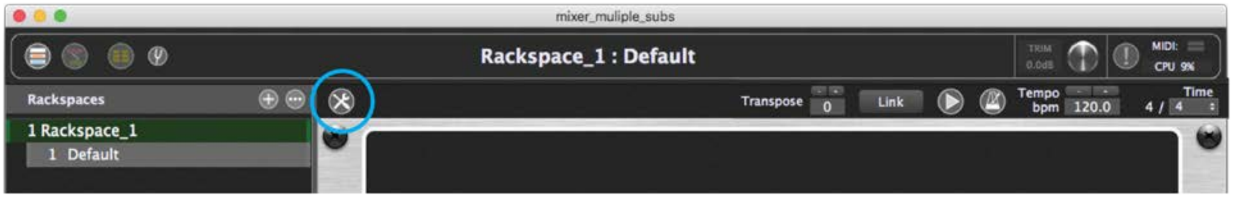
6. **Go to Performer View.**Click on the top left icon (circled) to switch Gig Performer to Performer View.



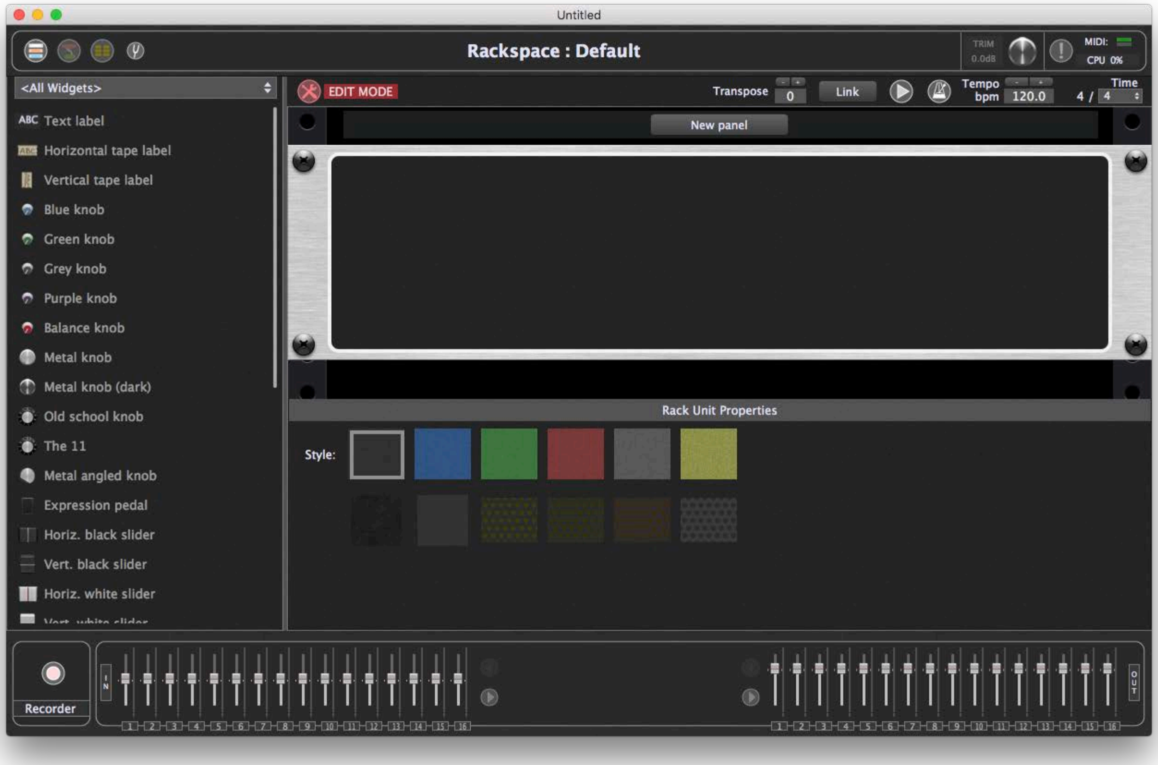
This screen, which is designed to look like a rack of one or more devices, allows you to create widgets (virtual knobs, sliders, buttons, etc.) that can be used to control various aspects of the sound created by your plugins. These widgets can be adjusted manually from your laptop and/or via physical controls (such as real knobs, sliders, and buttons) on your MIDI controller; with the use of an OSC app, they can also be adjusted remotely from iOS / Android tablets or smartphones. (See the OSC tutorial in this user guide for more information.)

When you switch to Performer View for the first time, you will see a rackspace containing a single empty panel (indicated by the arrow) into which you can insert widgets. (Extra panels can be added to group your widgets as desired; more about this below). You can create multiple rackspaces—each representing a collection of interconnected plugins—and you can switch from one rackspace to another manually from your laptop or via MIDI program change messages (with optional bank switching messages), or by using UP/DOWN commands from a MIDI pedalboard controller. You can also switch rackspaces with OSC commands (see Appendix A in this User Guide for more information).

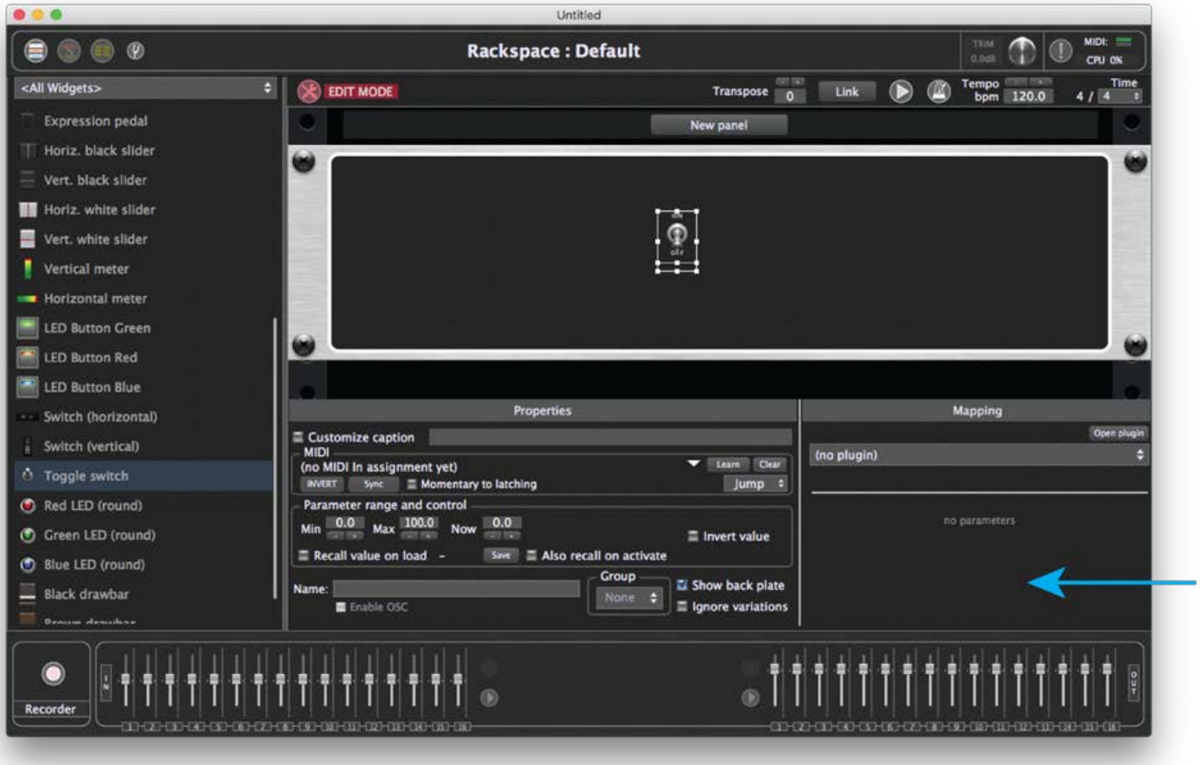
7. **Create a widget.**Click on the Widget Layout Designer icon (circled) to enter Edit mode.



In the screen shown below, the list of rackspaces on the left is replaced by a list of Gig Performer widgets, including virtual knobs and sliders of different shapes and colors, along with text labels, meters, buttons, and toggle switches.



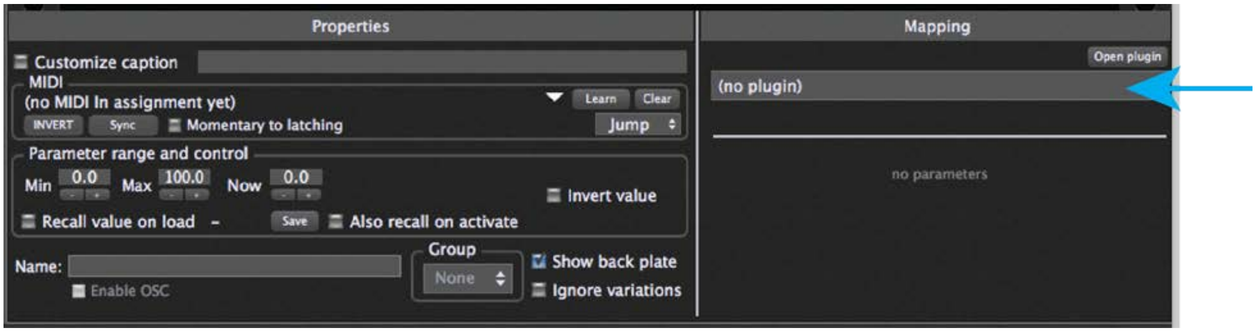
Widgets can be placed anywhere in a panel by simply dragging them to the desired location. To start, scroll down the list of widgets and drag a toggle switch from the left column somewhere into the panel. It will appear highlighted within a white rectangle and you can resize it by dragging any of the corner points. (The smaller section directly beneath the white rectangle displays acaption for the widget.)



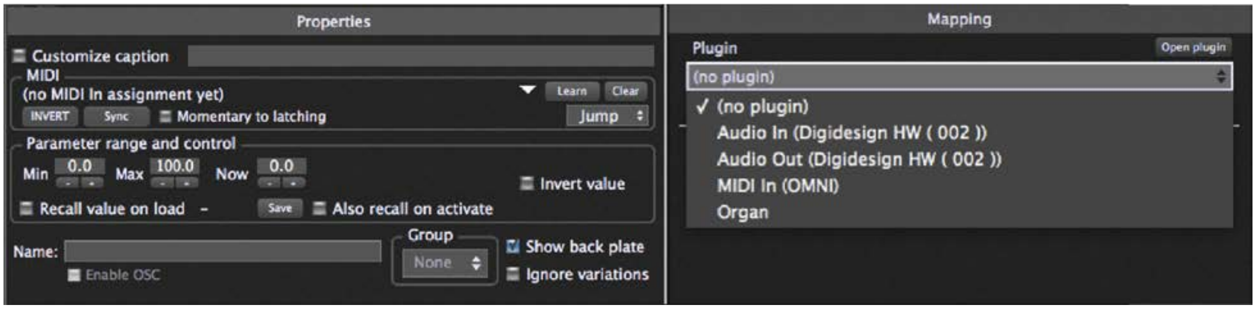
When a widget is highlighted, a Widget Properties “inspector” area appears below(indicated by the arrow). The “New Panel” button above itallows you to create additional panels within the rackspace if desired.The inspector is divided into two sections: Properties and Mapping. The Properties section on the left allows you to associate the widget with a physical button, knob or slider on a connected MIDI controller; it also allows you to optionally assign it a name that can be used with GP Script and OSC. The Mapping section on the right allows you to assign the widget to control a particular plugin parameter. Together, this enables you to use the buttons, pads, sliders and knobs on your controller to change parameters in real time.

Note: There are no technical differences between the various knob and slider widgets—only cosmetic differences. They all do the same thing: they generate values when you move them. But their different appearances allow you to easily group related items or have certain kinds of parameters (for example, all volume parameters) stand out clearly.

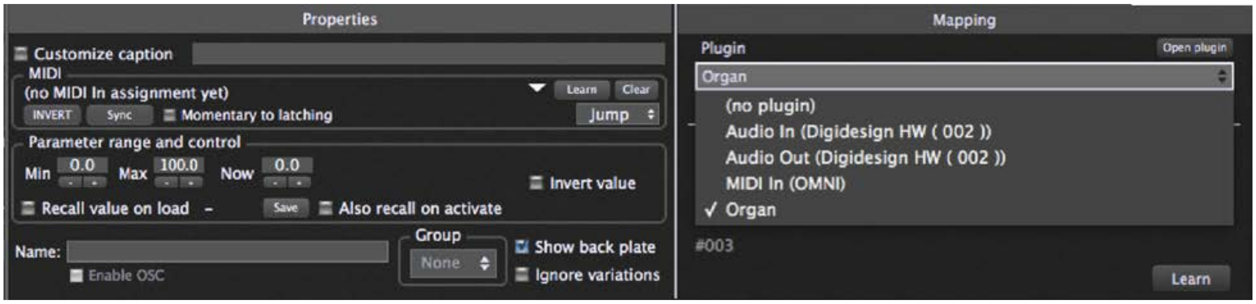
8. **Assign the widget to a plugin.**We're going to use this toggle switch to select fast or slow Leslie speeds for our Hammond organ sound. (Pretty much every Hammond organ plugin offers such a parameter). Start by telling Gig Performer to which plugin you want to apply the widget. This is accomplished by clicking on the plugin list dropdown menubeneath the word "Plugin"in the mapping section of the inspector (indicated by thearrow).



The default selection when you first create a widget is always "(no plugin)," as indicated by the checkmark to the left.

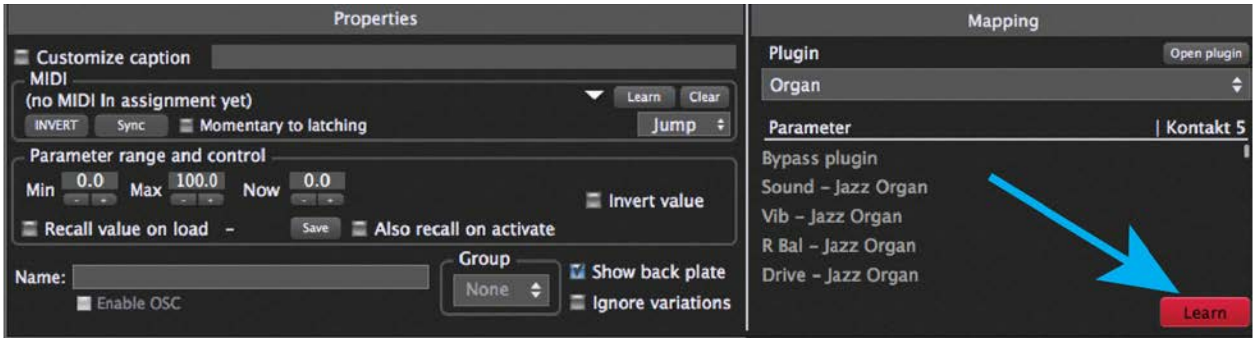


Scroll down the list of plugins—currently, there are justfour of them: your audio interface input and output, the built-in MIDI In (Omni) plugin, and your Organ plugin—and select Organ.

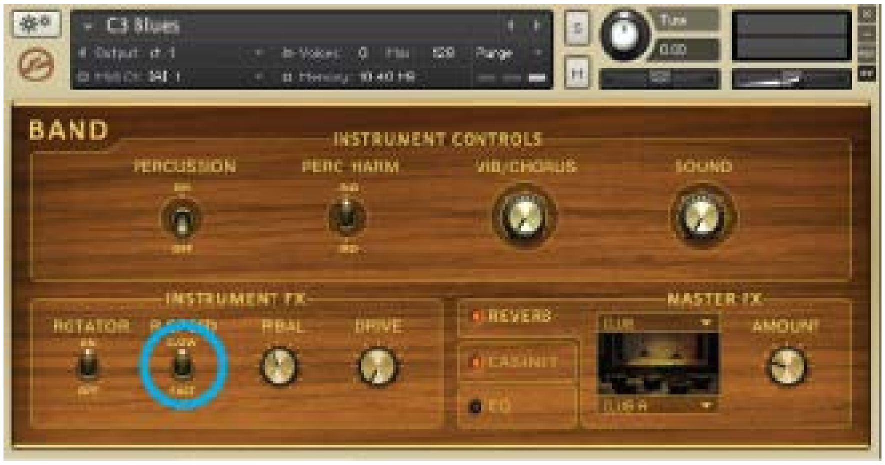


9. **Assign the widget to a plugin parameter.**Next, we have to tell Gig Performer which parameter we want to have the widget control. This is done

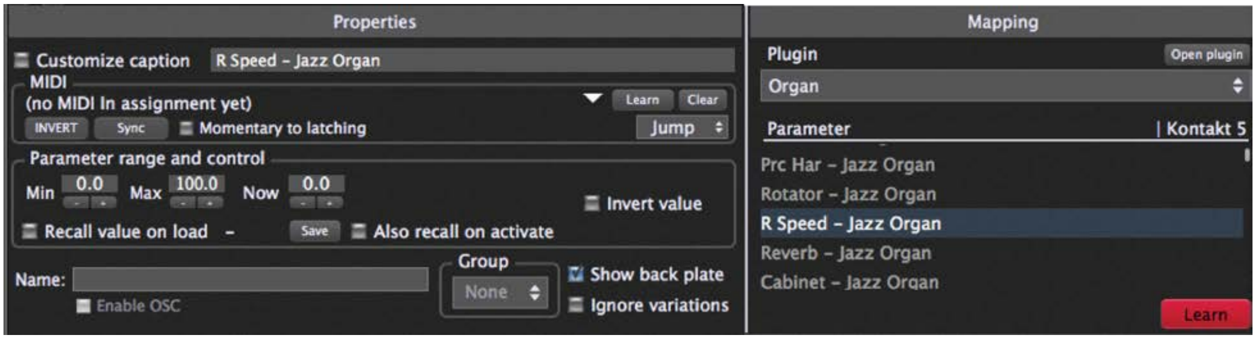
in the Parameter area of the mapping section of the Widget Properties inspector, immediately below the plugin list menu.As soon as you select a plugin, this area is populated with a list of available parameters that can be controlled by widgets viahost automation. This list always begins with the “Bypass plugin” option, enabling you to use a widget to simply bring the selected plugin in or out of the signal chain. The remainder of the controllable parameters are plug-in specific. Many plugins provide a list that is easily readable (i.e., “volume,”“overdrive,”“reverb level,”etc.), in which case all you need to do is to select the parameter you want the widget to affect, and you’re done with this part of the process. However, other plugins simply offer a list of parameter numbers, leaving you to decipher which number affects what aspect of the sound. Even if that’s the case, no problem, thanks to Gig Performer’s handy Learn mode,whichenables it to instantly recognize and learn the appropriate parameter. To use Learn mode, simply click on the “Learn” button in the right-hand Plugin section(as indicated by the arrow) so that it turns red.



Doing so immediately opens the selected plugin’s editor window. Now all you have to do is use your mouse to move the control you want the widget to affect, and Gig Performer instantly learns the parameter. In the case of the “C3 Blues” instrument, the parameter is called “R Speed – Jazz Organ” (circled below).



10. **Use the widget to control the sound.** If you now close your plugin's editor screen, the mapping section of the Widget Propertiesinspector will display the parameter assigned to your widget (in this case, "R Speed - Jazz Organ").



Properties

□ Customize caption R Speed – Jazz Organ

MIDI

(no MIDI In assignment yet) Learn Clear

INVERT Sync □ Momentary to latching Jump

Parameter range and control

Min 0.0 Max 100.0 Now 0.0 □ Invert value

□ Recall value on load – Save □ Also recall on activate

Name: Group None ☑ Show back plate

□ Enable OSC □ Ignore variations

Mapping

Plugin Open plugin

Organ

Parameter | Kontakt 5

Prc Har – Jazz Organ

Rotator – Jazz Organ

R Speed – Jazz Organ

Reverb – Jazz Organ

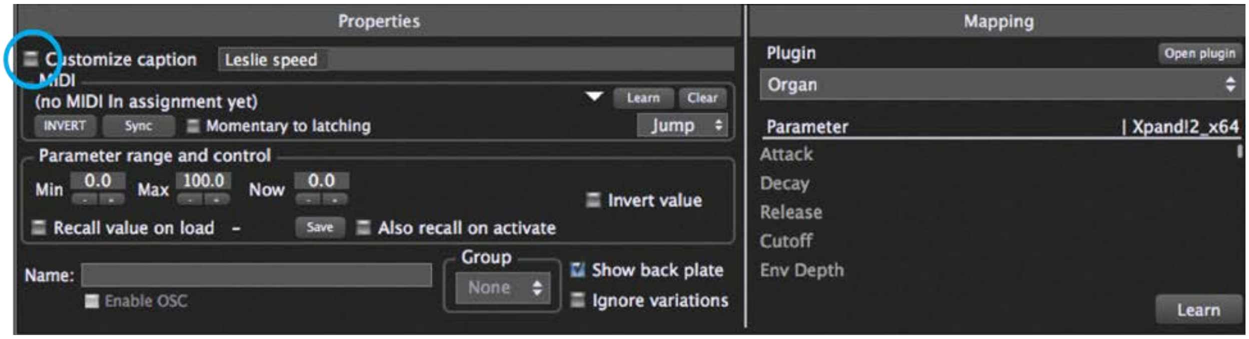
Cabinet – Jazz Organ

Learn

At this point, if you exit Edit mode by clicking on the Widget Layout Designer icon in red (above your rackspace), you will see a panel with your toggle switch, in this case also named "R Speed – Jazz Organ." (We'll show you how to change the name in a moment.) More importantly, if you click on the toggle switch with your mouse or trackpad, it will flip between the "ON" and "OFF" positions, and your organ sound will switch between a slow Leslie and a fast Leslie speed. (Try it!) When you're done, disable Learn mode by clicking on the "Learn" button again so that it no longer lights red.

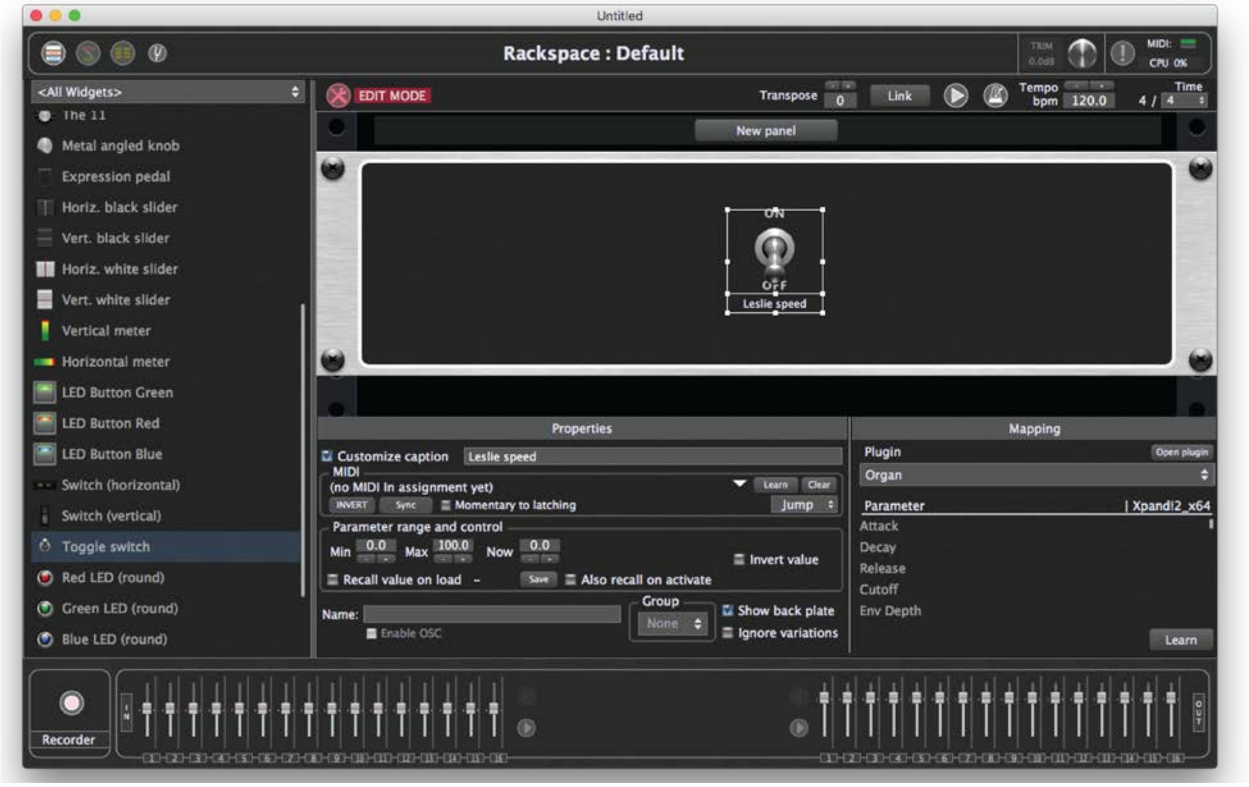
If for some reason a widget works the opposite way you'd like it to (for example, if the "ON" position of a toggle switch gives you a slow Leslie speed instead of the fast speed), you can easily change that in Edit mode. Just click on the widget to open the Inspector, then click on the "Invert value" button in the Properties section on the left; alternatively, you can swap the MIN and MAX values. Click here for more information.

Edit mode also gives you the ability to rename the caption below a widget. To do so, simply click on the "Customize caption" button (circled below) in the Properties section and type in a new name. Let's enter "Leslie speed" as the caption:



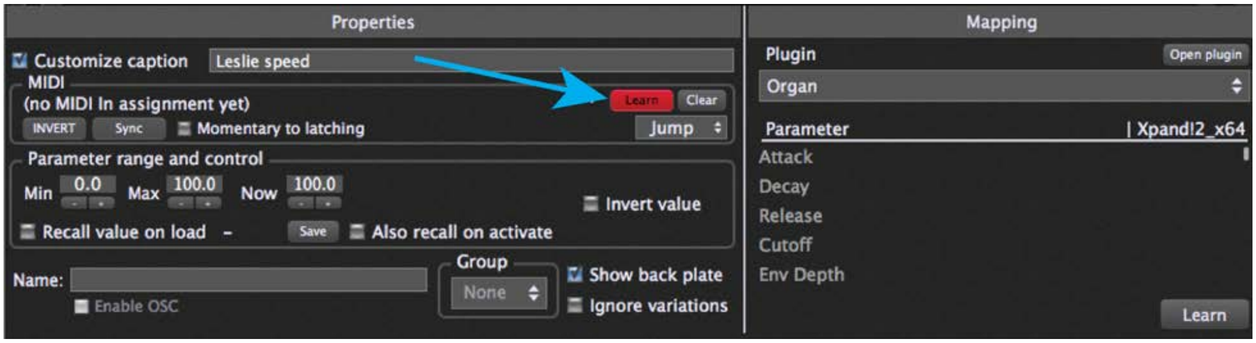
Customize caption: Leslie speed
Min 0.0 Max 100.0 Now 0.0
Plugin: Organ
Parameter | XpandI2_x64

As you may notice, a name of more than a few characters may not display properly within the default caption space. No problem! As mentioned earlier, the white rectangle that appears around the selected widget (and caption below) in Edit mode allows you resize it (andthe accompanying caption) by dragging any of the corner points. Here we've made it considerably larger so as to accommodate the new "Leslie speed" caption:



11. **Associate a widget with a physical device.**The real power of Gig Performer is that it gives you the abilitytoassociate widgets with physical deviceslike the knobs, sliders, pads, buttons, and switcheson your MIDI controller, as well as any pedals connected to it. You can even control widgets by playing specific notes, or by pressing down on keys harder.

Once again, this process is made easy by a Learn mode, which enables Gig Performer to instantly recognize which physical device you want to use to control the selected widget. To enter this mode, simply click on the "Learn" button in the left-hand Properties section of the inspector (indicated by the arrow) so that it turns red.



Now all you have to do is touch or move any switch, button, knob, slider, or pad on your MIDIcontroller, play any note on your keyboard, or step on any connected pedal. Gig Performer instantly learns the MIDI message transmitted by that device, and the toggle switch widget movesat the same time. Try it!

> If your keyboard controller has pads, they likely transmit MIDI note-on/note-off messages (unlike switches and buttons, which typically transmit MIDI controller messages). For that reason, when using notes or a pad to control an on/off widget such as the toggle switch, you may need to click on the "Treat momentary as latching" button in the MIDI section of the Widget Properties inspector for them to work correctly. Click here for more information.

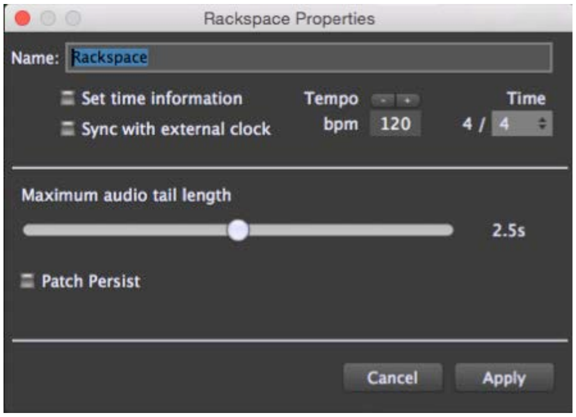
When you're satisfied with the physical controller you've selected, click the "Learn" buttonagain so that it no longer lights red.

Tip: While in "Learn" mode, you can click on different widgets and keep associating them with plugin parameters or physical controls—useful if you want to create all your widgets first and then associate them with parameters and physical controls later.

Important note: While you can use continuous physical controllers like knobs or sliders to control on-off widgets such as buttons or toggle switches (and vice versa: physical on-off controllers such as switches, buttons, or pads can be used to control continuous widgets such as knobs or sliders), this can get awkward and is not recommended practice!

Important note: Using keys to control widgets requires some extra care. When you first press a key while Gig Performer is in Learn mode, it will recognize and learn a Note On event. However, if you release the key while Learn is still enabled, Gig Performer will recognize and learn a Note Off message. In addition, if your keyboard controller generates aftertouch (channel pressure) messages, they will occur as you push the key harder in-between the Note On and Note Off messages. If Learn mode is still enabled while a note is being held, Gig Performer may well recognize and learn one of those values instead of the desired Note On or Note Off message.

12. **Name your rackspace.**Double-clicking on the rackspace name (currently the default name "Rackspace") in the upper left-hand corner of the screen brings up the Rackspace Properties dialog:



Name: Rackspace

□ Set time information

□ Sync with external clock

Tempo bpm 120

Time 4 / 4

Maximum audio tail length 2.5s

□ Patch Persist

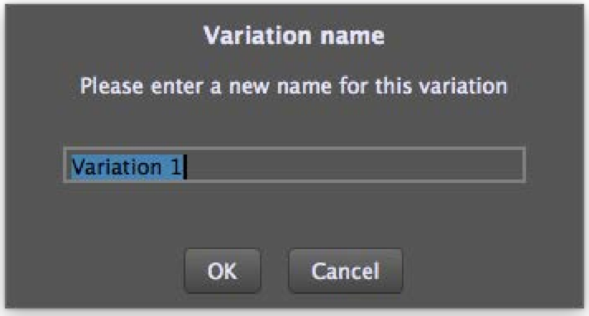
There are a number of options provided here, all of which are discussed in the Reference section of this User Guide. For now, simply type in a new name—let's use "Organ with Leslie speed control"—and click on Apply, which will close the dialog.

13. **Create a variation.**As discussed in the Basic Terminology section of this User Guide, a variation is essentially a rackspace preset which contains the same plugins and interconnections but with different widget settings. Variations allow you to leveragethe plugins in a single rackspace to create many different sounds without having to duplicate the rackspacefor each sound. It's also a great way to save on memory! For example, you could have multiple virtual instrument plugins feeding a mixer plugin,and then mutedifferent plugins, or have a different mix of them,in each variation. Or you could connect multiple effects (i.e., phasing, flanging, tremolo, reverb, echo, etc.) and have different ones enabled or disabled in each variation. Rather than creating a rackspace for an instrument with phasing, and another for the same instrument with echo, the same thing is accomplished by two variations of a single rackspace containing both phasing and echo.

As an example, let's create one variation of our "Organ with Leslie speed control" rackspace that is set to a slow Leslie speed, and another that is set to a fast Leslie speed. You can freely switch between variations instantaneously—and with no interruption of sound—either manually, with up/down pedals, or via MIDI program change messages.

> Variations contain only the value of widgets, which are connected to plugin parameters. Parameters not attached to widgets are not stored in variations.

To create a new variation, either go to **Edit >New variation** ortypeCmd-N. A dialog will open asking you to type in a new name. Let's call this variation "Leslie Fast."

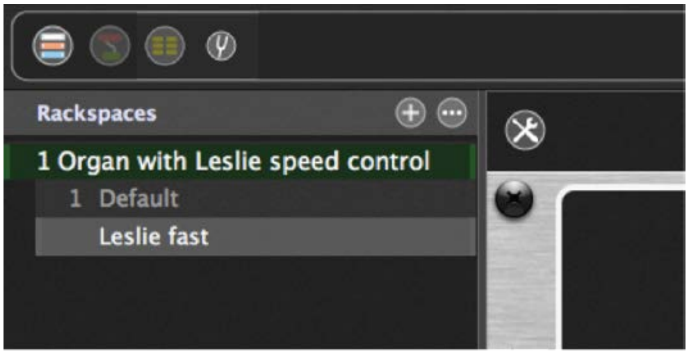


Variation name

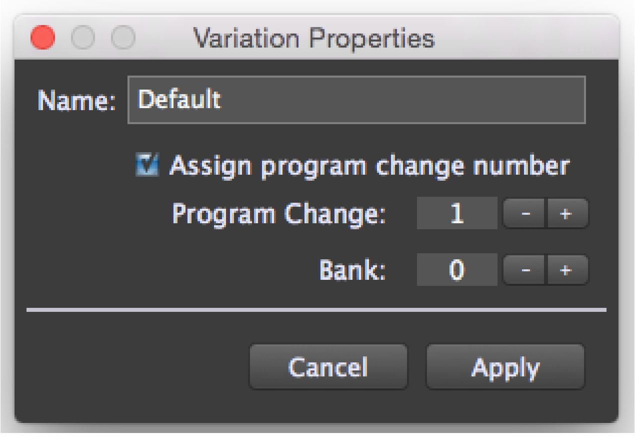
Please enter a new name for this variation

Variation 1

You’ve probably noticed that there’s already a variation for the “Organ with Leslie speed control” rackspace you created, with the name “Default”:



Every rackspace automatically comes with at least one variation, initially named “Default.”(The reason isthat, technically, only variations—not the rackspaces themselves—can be selected with MIDI program change messages.) Now that we’ve created a second variation, let’s go ahead and rename the first one. To do so, double-click on the name, which will bring up the Variation Properties window:



Name: Default

☑ Assign program change number

Program Change: 1 - +

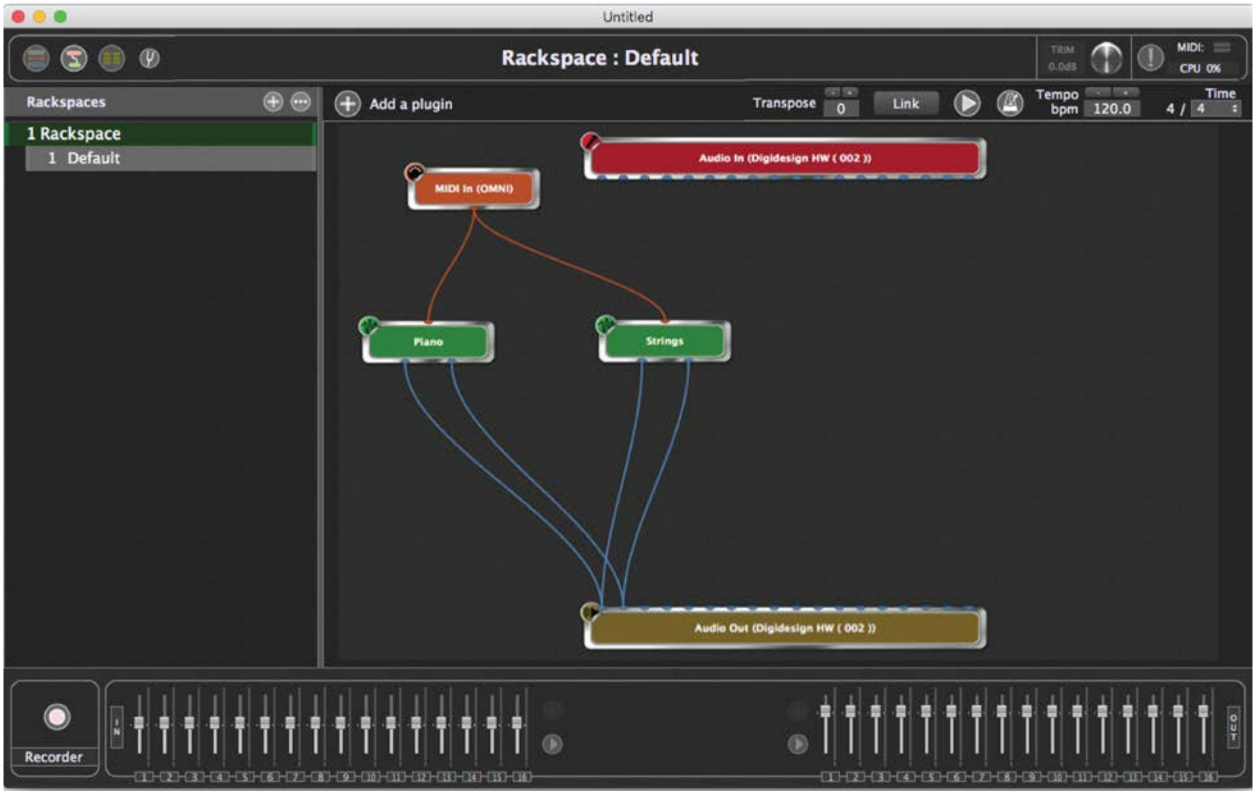
Bank: 0 - +

Enter the name “Leslie slow” in the Name field. If you want to switch to this variation by sending Gig Performer a MIDI program change message, be sure the Assign program change number box is checked, and set the program change number and, if desired, the bank number, using the “+” and “-“ keys next to the fields, or by simply clicking in the fields directly and entering the values. When you’re done, click on Apply, which will save the settings and close the dialog.

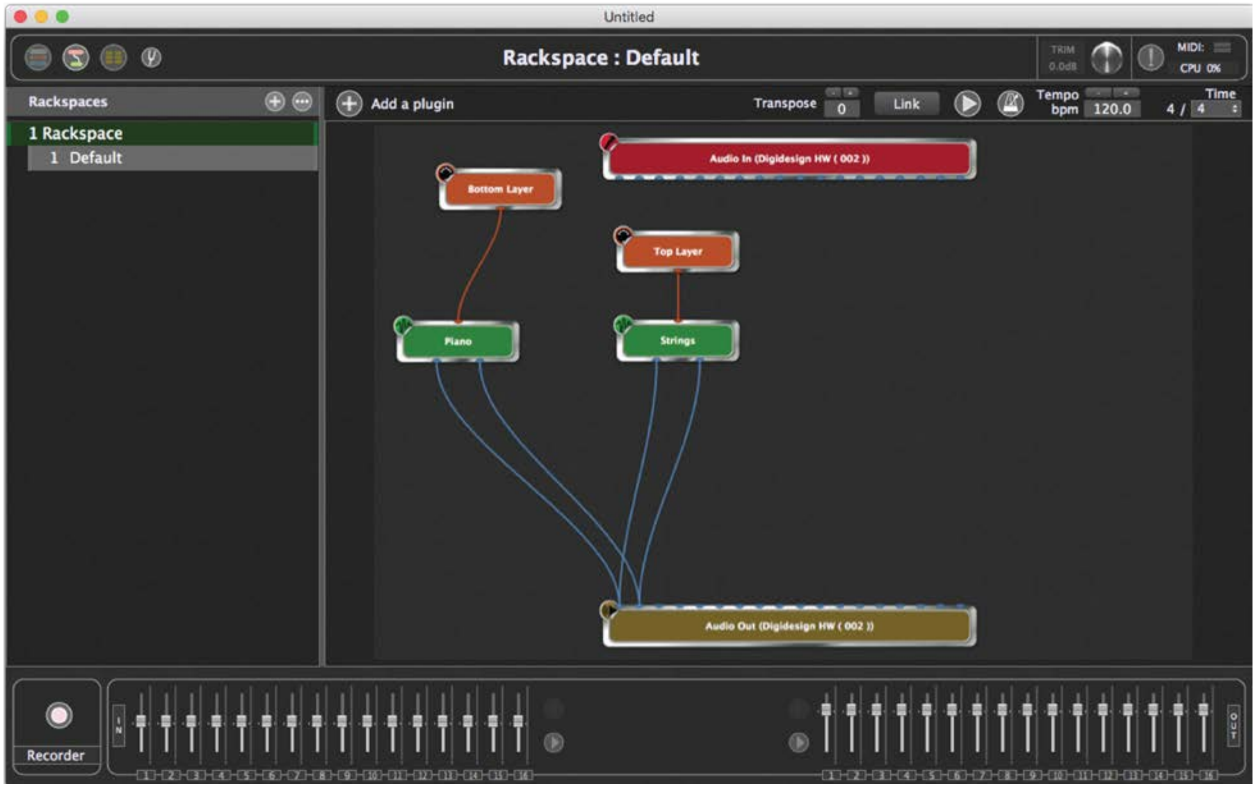
Now all that remains is to make sure the toggle switch widget is set to the “Off” position—something you can do either manually from your laptop or by moving the associated physical device on your MIDI controller (as set in step 11 above) and that it is set to the "On" position when you click on the "Leslie fast" variation. For more information about switching between variations with MIDI pedals or a MIDI controller, click here.

14. **Create a new rackspace with multiple variations.**As discussed in the Basic Terminology section of this User Guide, a rackspace can actually be pretty much anything you like: It can represent a single sound or all the elements needed for an entire song—or even just one part of a song. You can switch from one rackspace to another instantaneously, either manually, with up/down pedals, or via MIDI program change messages.

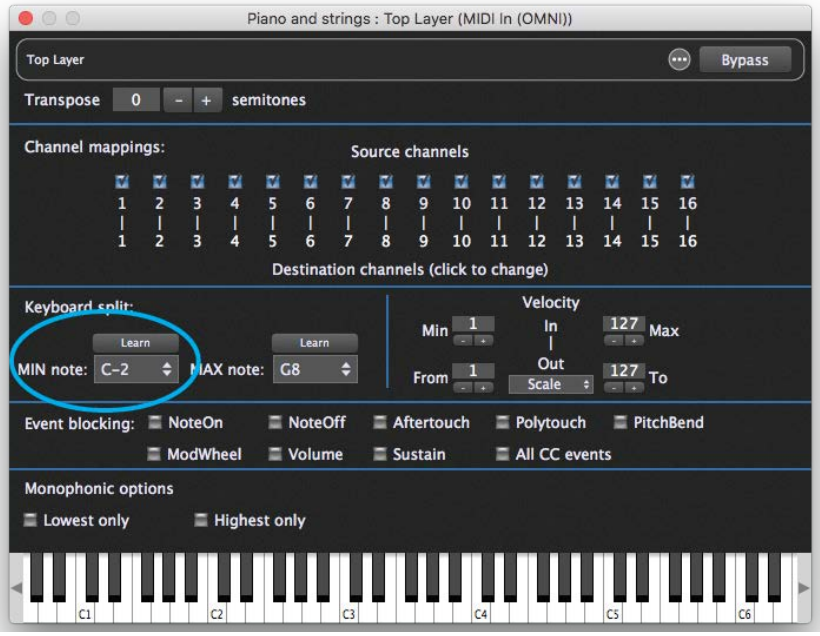
With that in mind, let's create a new rackspace by going to **Edit>New rackspace**, by using the Cmd-Shift-N (Mac)/Ctrl-Shift-N (Windows) keystroke commands, or by clicking on the **"+"** icon above the list of rackspace names. In the Rackspace Properties window that opens, rename it "Piano and strings," then, following the procedures outlined earlier in this tutorial, insert two plugins: one generating a piano sound and the other generating a string sound. Rename them "Piano" and "Strings" respectively, and make the appropriate MIDI and audio connections. Your Connections View screen should now look something like this:



Gig Performer allows a single MIDI input block to control two or more plugins simultaneously, so if you now play your keyboard controller, you should hear both sounds, layered together. However, we're going to create a keyboard split so that the strings only layer with the piano when notes above Middle C are played. Doing this requires separate MIDI input blocks, so right-click in the Connections View screen and add a second "MIDI In (OMNI)" block and rename it "Top Layer," then move the orange connection from the first "MIDI In (OMNI)" block to the "Top Layer" block. Finally, rename the first "MIDI In (OMNI)" block as "Bottom Layer." Your screen should now look something like this:


Untitled
Rackspace : Default
TRIM
0.0dB
MIDI:
CPU 0%
Rackspaces
1 Rackspace
1 Default
Add a plugin
Transpose
0
Link
Tempo
bpm
120.0
Time
4 / 4
Audio In (Digidesign HW (002))
Bottom Layer
Top Layer
Piano
Strings
Audio Out (Digidesign HW (002))
Recorder
IN
OUT
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Next, we'll create the keyboard split. Double-click on the "Top Layer" MIDI Input block to open its plugin editor:



Transpose 0 - + semitones

Source	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
Destination	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

MIN note: C-2

MAX note: G8

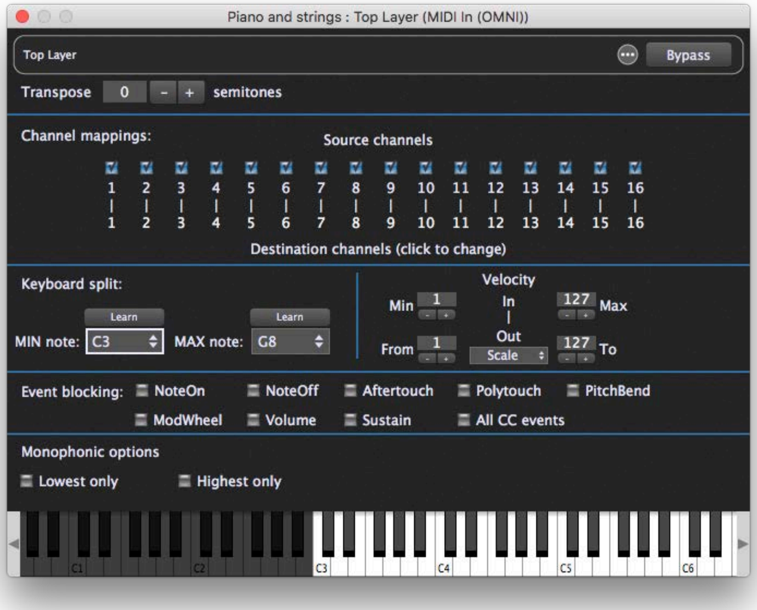
Velocity In Min 1, Max 127

Velocity Out From 1, To 127, Scale

Event blocking: □ NoteOn □ NoteOff □ Aftertouch □ Polytouch □ PitchBend □ ModWheel □ Volume □ Sustain □ All CC events

□ Lowest only □ Highest only

Click on the Learn button above the"MIN note" parameter (circled above) and play"C3"on your MIDI keyboard or other controller; the notes below C3 on the virtual keyboard become darkened, indicating that they are inactive. (Notes outside the specified MIN and MAX note range are always darkened this way.)



Piano and strings : Top Layer (MIDI In (OMNI))

Top Layer

Transpose 0 semitones

Channel mappings:

Source channels	☑ 1	☑ 2	☑ 3	☑ 4	☑ 5	☑ 6	☑ 7	☑ 8	☑ 9	☑ 10	☑ 11	☑ 12	☑ 13	☑ 14	☑ 15	☑ 16
Destination channels (click to change)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Keyboard split: MIN note: C3 MAX note: G8

Velocity In Min 1 Max 127; Out From 1 To 127 Scale

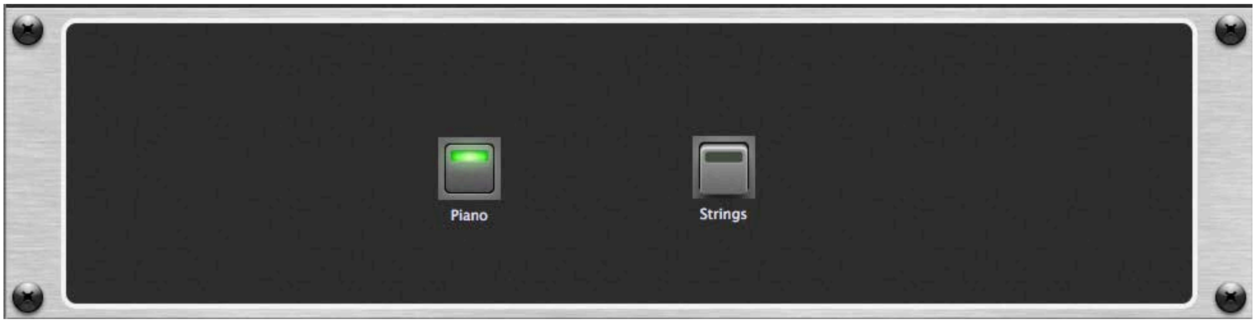
Event blocking: ☐ NoteOn ☐ NoteOff ☐ Aftertouch ☐ Polytouch ☐ PitchBend ☐ ModWheel ☐ Volume ☐ Sustain ☐ All CC events

Monophonic options ☐ Lowest only ☐ Highest only

Your keyboard split is complete! If you now play your keyboard controller, you'll only hear strings when Middle C or higher notes are played. (If you'd prefer hearing the strings on their ownwhen notes Middle C or higher are played instead of layering them with the piano, simply set the Max note for the "Bottom Layer" MIDI In block to B2.)

Now let's add widgets that allow the two sounds to be turned on and off individually. Switch to Performer View and enter Edit mode, then drag two LED Button widgets into the rack panel. Assign one to the "Bottom Layer"MIDI In block (controlling the piano sound) and set it to the "Bypass plugin" parameter, then assign the other to the "Top Layer"MIDI In block (to control the strings sound) and set it to the "Bypass plugin" parameter as well. Bypassing the MIDI In block instead of the synth plug-in means that a string of Note Off messages are sent when the block is bypassed, shutting off any notes still sounding when you mute the instrument.

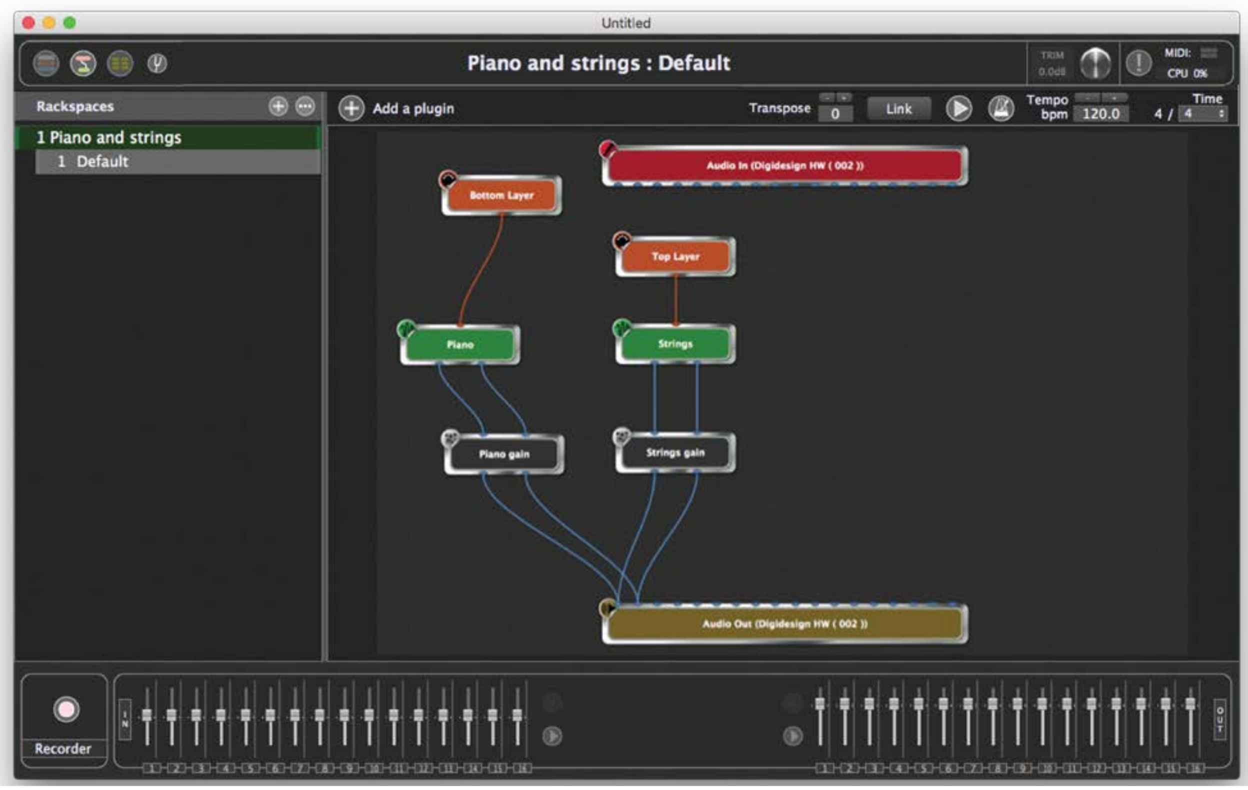
Click "Invert value" (in the left-hand Properties section) so that the "On" (button lit) position of the LED allows the associated plugin to be heard and the "Off" (button unlit) position causes the associated plugin to be bypassed. Caption each widget accordingly and test to make sure everything is working as expected. (Feel free to assign either or both widgets to physical devices on your MIDI controller if you like.) Your rackspace should now look like this:



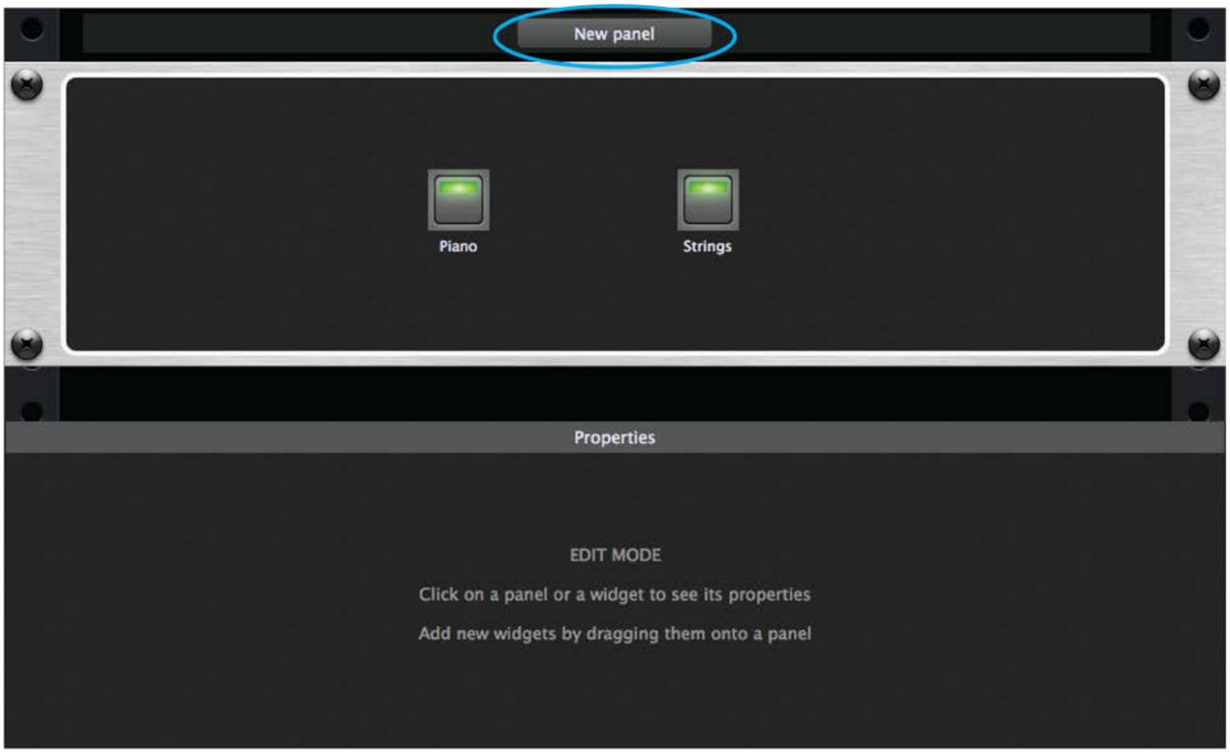
Next, we'll add a new panel and slider widgets to control the levels of the two sounds. (Rackspaces can contain multiple panels, and Gig Performer provides options to resize them if your laptop provides limited screen real estate.) Although some plugins offer master volume parameters, many do not, and so the best way to accomplish this is with the use of the Gain and Balance plugin bundledwith Gig Performer. Accordingly, return to Connections View and insert two Gain and Balance plugin blocks. Connect one between the audio outputs of your "Piano" plugin and your audio interface's output block, and rename it "Piano gain." Then connect the other between the audio outputs of your "Strings" plugin and your audio interface's output block, and rename it "Strings gain."

If you connect the audio outputs of multiple plugins to the input of another plugin, the signals will be mixed automatically. If you'd like to take control of that mix and create custom blends, simply insert one of Gig Performer's bundled Audio Mixer/Router plugins, or one of its Gain plugins (Gain Control, Gain and Balance, or Gain and Balance (m->s)) in-between, then adjust their volume controls as desired. These plugins can also be used as sends to effects such as reverb plugins. See the Configuring Audio Plugins tutorial in this User Guide for an example of how to do this.

Your screen should now look like this:



Return to Edit mode and click on the “New panel” button (circled below) that appears between the first panel and the Properties area below. (Depending upon the screen size of your laptop, you may need to resize your screen or scroll down below the current panel to see this button.)

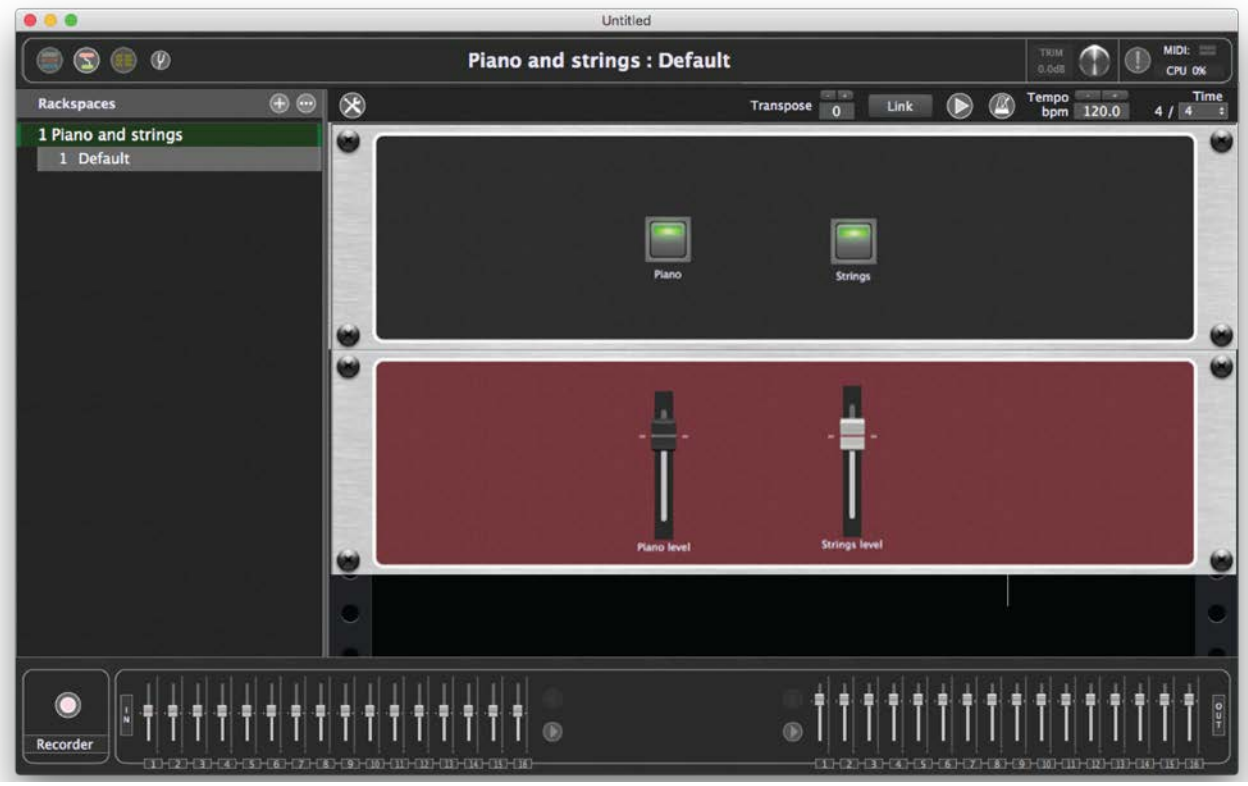


EDIT MODE

Click on a panel or a widget to see its properties

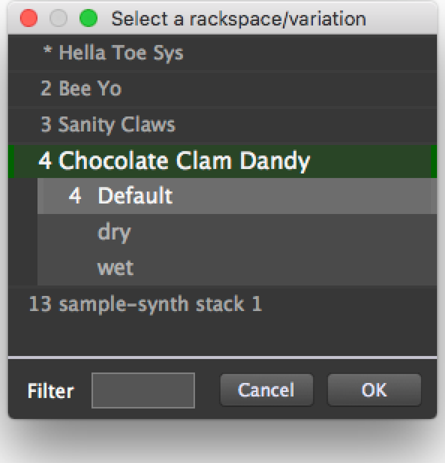
Add new widgets by dragging them onto a panel

Select "New 2U panel" ("U" stands for "Rack Units"; you can always change the size later if necessary) and then double-click on the panel. From the "Style" menu that appears under "Rack Unit Properties," select a custom color for the new panel. Next, drag one vertical black slider and one vertical white slider into the new panel. Assign the black slider to the "Gain" parameter of the "Piano gain" plugin (NOT the "Piano" plugin) and the white slider to the "Gain" parameter of the "Strings gain" plugin (NOT the "Strings" plugin). Finally, assign the black slider to control the "Piano gain" plugin's "Gain" parameter, and the white slider to control the "Strings gain" plugin's "Gain" parameter. Feel free to assign either or both widgets to physical devices on your MIDI controller if you like. (If you have an expression pedal connected to your controller, that would be an excellent way to fade the strings in and out—you can even substitute an Expression Pedal widget for the white slider.) Finally, caption the two sliders "Piano level" and "Strings level," respectively. (You may need to resize them to view the entire caption.) Listen and test to make sure everything is working as expected. Your Performer View screen should now look like this:



Want to cross-fade between two different sounds (such as the Piano and Strings in this tutorial), or two different effects? It's easily accomplished with Widget Groups. See an example of how to do this in the Configuring Audio Plugins tutorial.

15. **Add variations.**Finally, add a number of variations to the rackspace, following the procedures described earlier in this tutorial.For example, you might want to have one variation that provides piano sound only (with the "Strings" LED widget unlit) and another that provides both sounds (both LED widgets lit) at full volume, with a third variation that provides strings at half volume. Go on—experiment! The sheer flexibility of Gig Performer means that the sky is the limit.
16. **Load a variation.**Clicking a variation in the browser instantly loads it. However, in performance it might be more convenient to switch variations using the Quick Variation Finder. Press Cmd-J (Ctrl-J on Win) to open the Quick Variation Finder. Click on the desired variation of a rackspace and click the Close button to apply your selection and close the window.



17. Save your work.Last but by no means least, be sure to save your work by going to**File> Save As...**and typing in a filename. The saved file, with the extension .gig, contains completeinformation about all rackspaces and variations.

You can also opt to save just the current rackspace for later importation into another Gig Performer session by selecting "Export current rackspace..." in the File menu. Click here for more information.

Using Rig Manager

The Rig Manager is designed to speed workflow by eliminating the need to teach controllers to Gig Performer every time you use a different keyboard or controller setup. Rig Manager constructs a map of physical input devices like keyboards, and the controllersthey host, like sliders, buttons, faders, and knobs, to virtual controllers that are linked to widgets. Virtual controllers decouple the physical controller from the control paths in Gig Performer, which makes it easier to swap out different physical controllers while maintaining your usual controls.

Say, for example, you're sharing backline instruments with another performer and so have to use someone else's keyboard controller for your performance. Rig Manager's controller mapping means that you only need to map the new physical controllers to the existing virtual ones and all of your widget controls are reestablished. That can be accomplished in a few minutes around soundcheck. Similarly, if you use one controller for rehearsal and a different one for gigs, Rig Manager can make it much easier to switch between them. If you play multiple identical keyboards, Rig Manager simplifies distinguishing amongthose.

See the Reference section on the Rig Manager for additional details.

What's a rig?

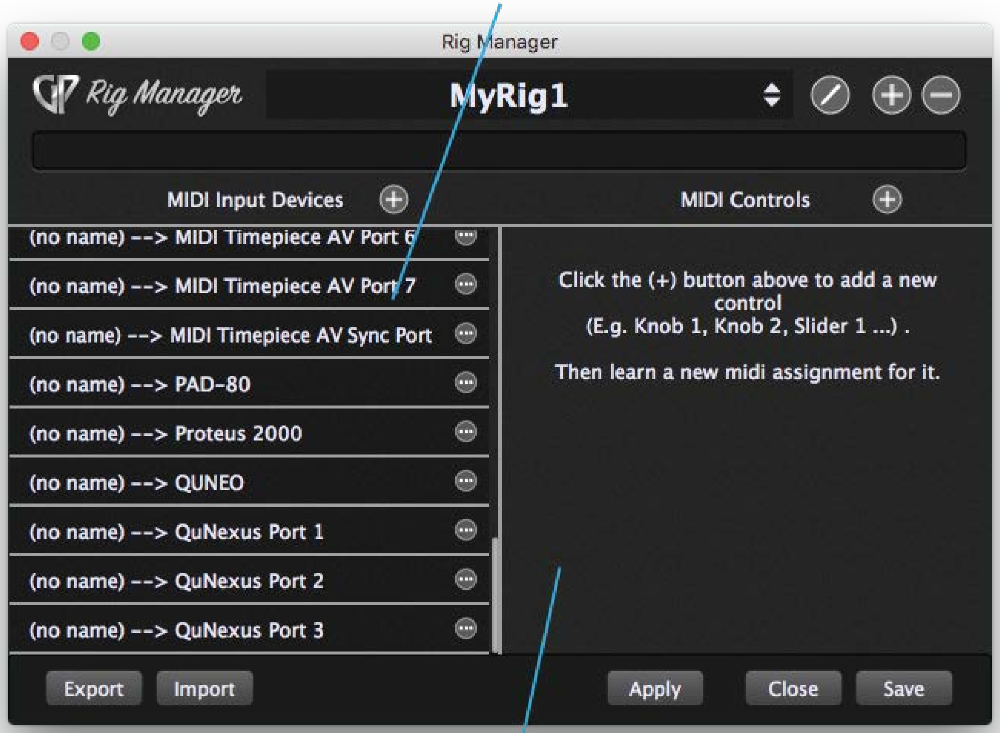
In Gig Performer, a "rig" is a collection of three sets of information: a set of physical MIDI controllers, a set of virtual controllers that can be attached to widgets, and a map that relates the two sets, thereby connecting your physical controller, through widgets, to parameter controls.

A MIDI Input Device is a physical controller, which often has multiple knobs, sliders, buttons, or other individual controllers on its surface, while MIDI Controls are the virtual controllers that get mapped to parameters. Once the lists of MIDI Input Devices and MIDI Controls are established, they appear as drop-down menus in fields where they are used, for example, as MIDI sources in plugins.

The following tutorial walks through adding MIDI Input devices and MIDI Controls, and linking them appropriately to let you quickly switch between a rehearsal setup and a performing setup.

1. PressCmd/Ctrl (Mac/Win) – R or choose the **Window>Rig Manager** command to open the Rig Manager.
2. The Rig Manager window will open showing a default configuration.

These are physical MIDI controllers that get global names



(no name) --> MIDI Timepiece AV Port 6
(no name) --> MIDI Timepiece AV Port 7
(no name) --> MIDI Timepiece AV Sync Port
(no name) --> PAD-80
(no name) --> Proteus 2000
(no name) --> QUNEO
(no name) --> QuNexus Port 1
(no name) --> QuNexus Port 2
(no name) --> QuNexus Port 3

Click the (+) button above to add a new control
(E.g. Knob 1, Knob 2, Slider 1 ...) .

Then learn a new midi assignment for it.

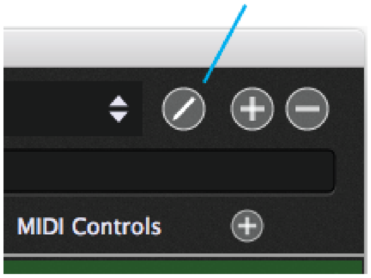
Add virtual MIDI controllers that connect to widgets here

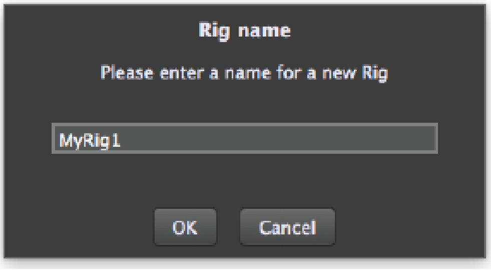
The main area of the window is divided between two columns. On the left are the MIDI Input Devices – physical control devices like keyboards, pad controllers, and control surfaces. In the screenshot above, none of the devices shown have been given global names.

The column on the right is MIDI Controls.

3. Click the Rename Rig button and enter a new name for your rig.

Click here to rename the rig





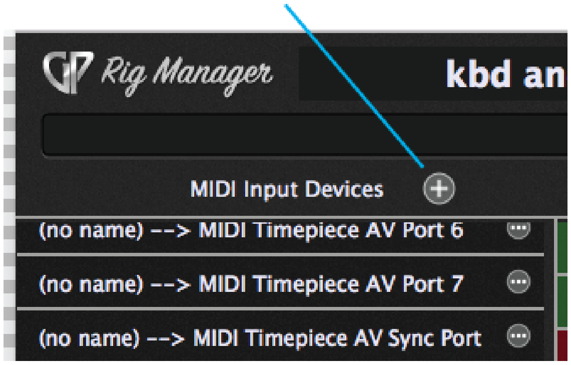
Rig name

Please enter a name for a new Rig

MyRig1

4. Let’s add another input device. Click the plus sign at the top of the MIDI Input Device list. If you needed to remove a device, you just would click the “three dots” button next to the device and choose **Remove this global name** from the popup menu.

Click here to add a new MIDI Input device

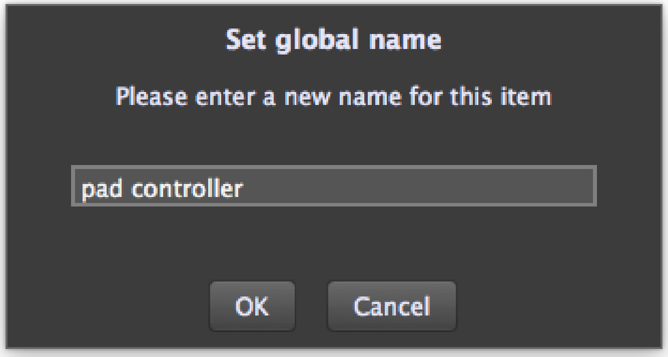


(no name) --> MIDI Timepiece AV Port 6

(no name) --> MIDI Timepiece AV Port 7

(no name) --> MIDI Timepiece AV Sync Port

5. In the dialog that appears, give the device a global name. This step is key to using the Rig Manager.

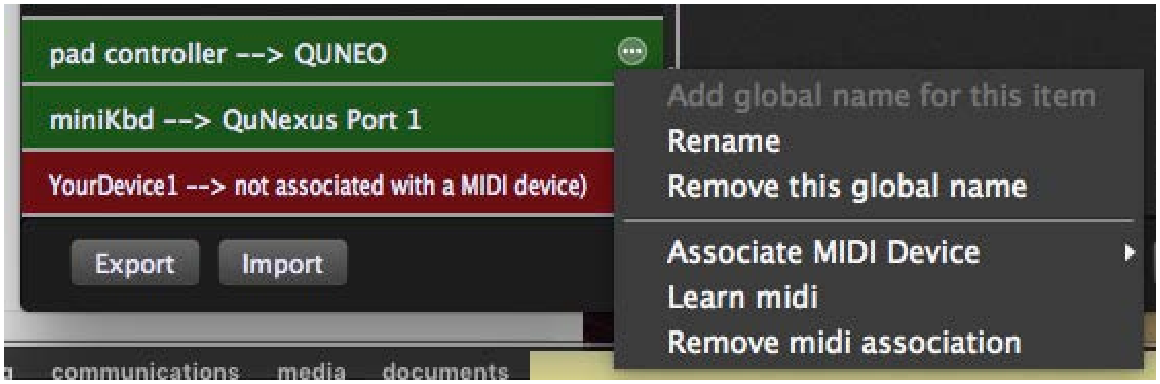


Set global name

Please enter a new name for this item

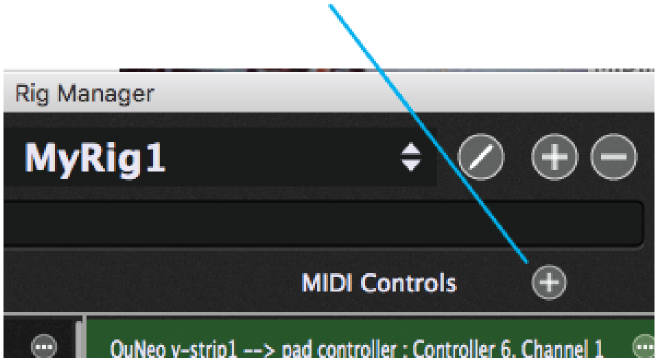
pad controller

6. The new device name needs to be linked to a physical MIDI controller. Click on the “three dots” button or right-click on the input device and choose a source device from the **Associate MIDI Device**submenu.

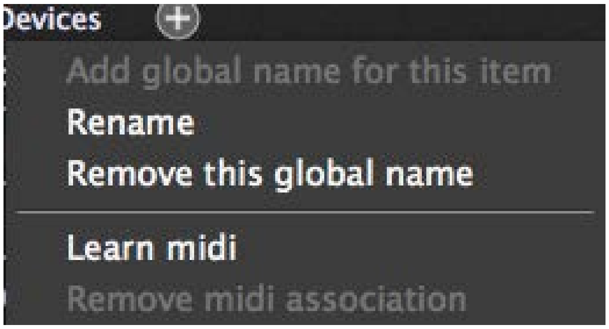


7. Now let's add a new MIDI control by clicking the plus sign at the top of the MIDI Controls column.In the dialog that appears, name the MIDI control.

Click here to add a new virtual controller



8. Now it remains to map the new virtual controller to a physical controller. Double-click the MIDI control you just created to put the control into Learn mode, or right-click it and choose **Learn MIDI**from the popup menu.

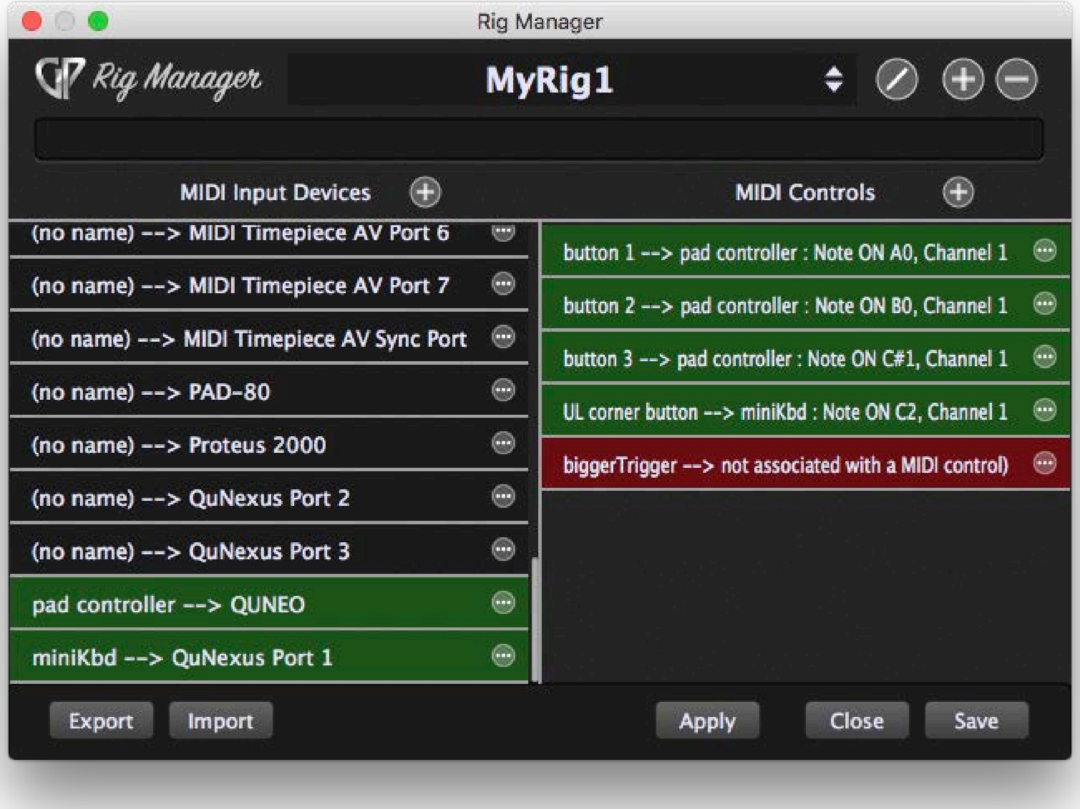


9. Move a knob, fader, or other physical controller and it will be mapped to the virtual controller.

The MIDI Control right-click menu also enables renaming the virtual control or removing it from the rig (by removing its name). Finally, you can disconnect a physical controller from a virtual one with the **Remove MIDI association**command.

10. Repeat the Learn MIDI process for each parameter you want controlled from your MIDI input device.

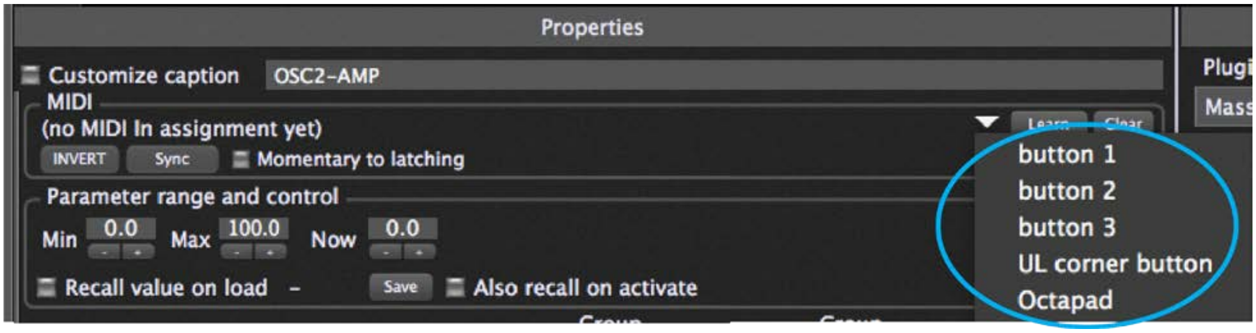
By now, Rig Manager should look something like this:



(no name) --> MIDI Timepiece AV Port 6
(no name) --> MIDI Timepiece AV Port 7
(no name) --> MIDI Timepiece AV Sync Port
(no name) --> PAD-80
(no name) --> Proteus 2000
(no name) --> QuNexus Port 2
(no name) --> QuNexus Port 3
pad controller --> QUNEO
miniKbd --> QuNexus Port 1

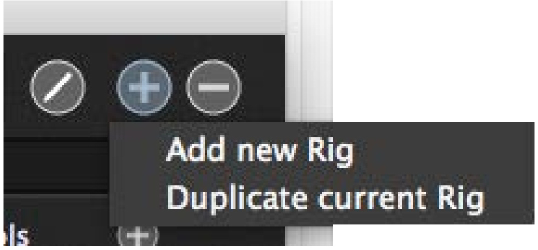
button 1 --> pad controller : Note ON A0, Channel 1
button 2 --> pad controller : Note ON B0, Channel 1
button 3 --> pad controller : Note ON C#1, Channel 1
UL corner button --> miniKbd : Note ON C2, Channel 1
biggerTrigger --> not associated with a MIDI control)

11. Connect the virtual controllers to widgets to control parameters. The MIDI controls that have been named and defined show up as sources, for example, in this list of sources to be mapped to a switch widget.



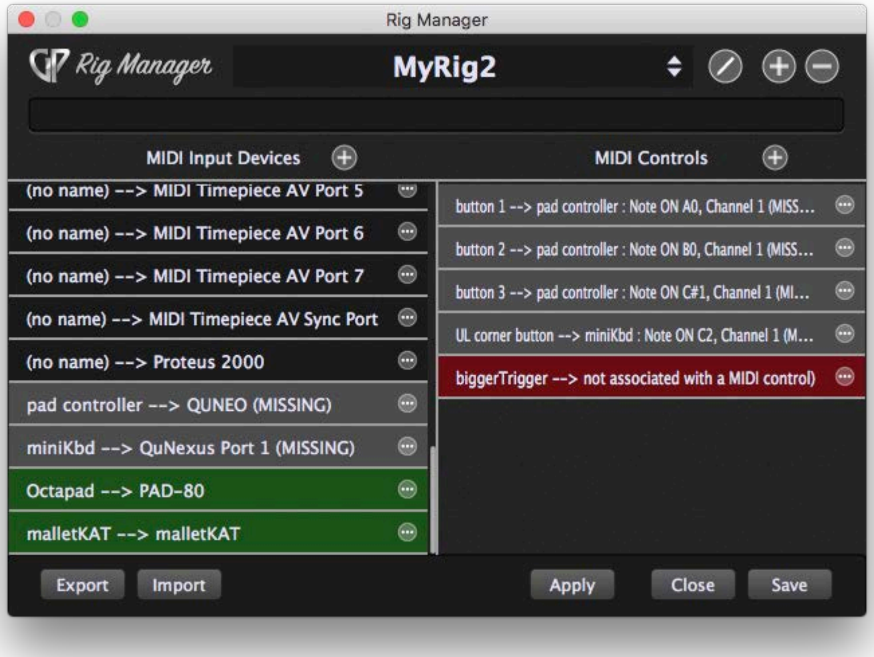
In the studio, QUNEO and QuNexus are the primary controllers. In performance, however, the controllers are a malletKAT and an old Roland PAD80. We will setup another rig for live performance.

12. Click the large plus button at the top of the window and choose **Duplicate current rig** to make a copy of the rig.



13. Rename the new copy MyRig2
14. Create two new MIDI Input devices and rename them. In this example, they are named “Octapad” and “malletKAT.”
15. Choose **Learn midi** for each of the new input devices and play on them to establish the link.

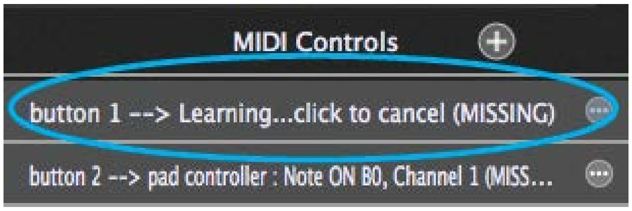
This performance rig now looks like this:



(no name) --> MIDI Timepiece AV Port 5
(no name) --> MIDI Timepiece AV Port 6
(no name) --> MIDI Timepiece AV Port 7
(no name) --> MIDI Timepiece AV Sync Port
(no name) --> Proteus 2000
pad controller --> QUNEO (MISSING)
miniKbd --> QuNexus Port 1 (MISSING)
Octapad --> PAD-80
malletKAT --> malletKAT

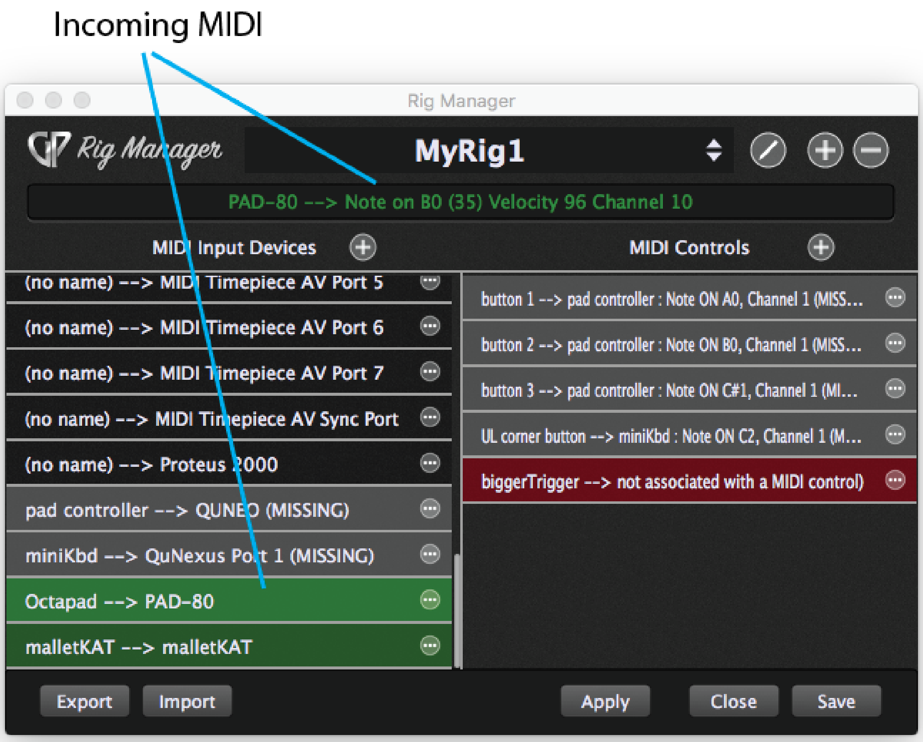
button 1 --> pad controller : Note ON A0, Channel 1 (MISS...
button 2 --> pad controller : Note ON B0, Channel 1 (MISS...
button 3 --> pad controller : Note ON C#1, Channel 1 (MI...
UL corner button --> miniKbd : Note ON C2, Channel 1 (M...
biggerTrigger --> not associated with a MIDI control)

16. Each MIDI Control must be trained once for a new physical controller. Double-click on the name of the MIDI control and then move a knob or other physical control. You can also enter Learn mode for a control by clicking the "three dots" button to the right of a control and choosing**Learn midi**.



button 1 --> Learning...click to cancel (MISSING)
button 2 --> pad controller : Note ON B0, Channel 1 (MISS...

17. When MIDI events from the physical control arrive, the item lights up in the MIDI Input Devices and Controls lists, and the events that came in will be displayed at the top of the window.


Incoming MIDI

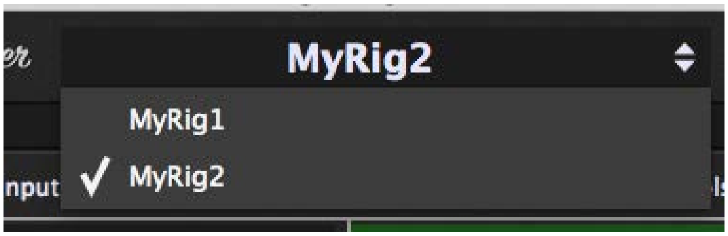
PAD-80 --> Note on B0 (35) Velocity 96 Channel 10

(no name) --> MIDI Timepiece AV Port 5
(no name) --> MIDI Timepiece AV Port 6
(no name) --> MIDI Timepiece AV Port 7
(no name) --> MIDI Timepiece AV Sync Port
(no name) --> Proteus 2000
pad controller --> QUNEO (MISSING)
miniKbd --> QuNexus Port 1 (MISSING)
Octapad --> PAD-80
malletKAT --> malletKAT

button 1 --> pad controller : Note ON A0, Channel 1 (MISS…
button 2 --> pad controller : Note ON B0, Channel 1 (MISS…
button 3 --> pad controller : Note ON C#1, Channel 1 (MI…
UL corner button --> miniKbd : Note ON C2, Channel 1 (M…
biggerTrigger --> not associated with a MIDI control)

18. Repeat the Learn midi process for each controller.
19. Save your work.

20. Switch back and forth between the studio and performance rigs by clicking on the arrowheads and choosing a rig from the dropdown menu.

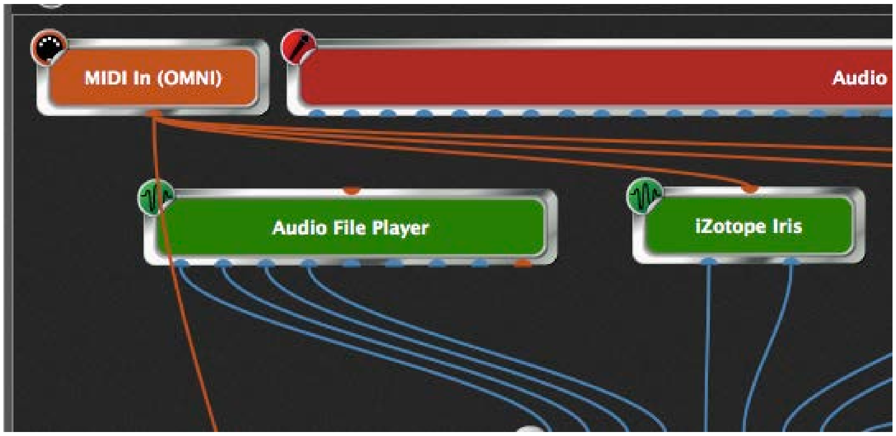


Note messages

Rig Manager handles remapping switch and continuous controls, which is the hard part of changing to a different controller. The easier part is redirecting the typical stream of note messages from your keyboard, pad controller, or other musical controller. That is handled with MIDI plugins in the Connections view, the other place where a change in controllers must be accommodated.

One method is to use the MIDI In (OMNI) plugin. Omni, in this case, means not simply that it responds to messages on any MIDI channel, but also that it “hears” all connected MIDI devices. When you unplug a keyboard at rehearsal and plug a different in for live performance, it makes no difference, since the plugin responds to note messages from any source. If you use MIDI In (OMNI) in your rackspaces, you need worry only about setting up rigs in Rig Manager for each set of controllers you use.

Caution: Be careful if you are using a virtual MIDI port, such as IAC on Mac OS X. Since the MIDI In (OMNI) module receives messages from all MIDI ports, sending a MIDI message out from a virtual MIDI port can create a feedback loop.



Just to be clear, while the Connections view is where note messages for pitch are routed, individual note messages can be assigned to virtual controllers in Rig Manager, so that notes can trigger actions or perform other control functions aside from playing pitches.

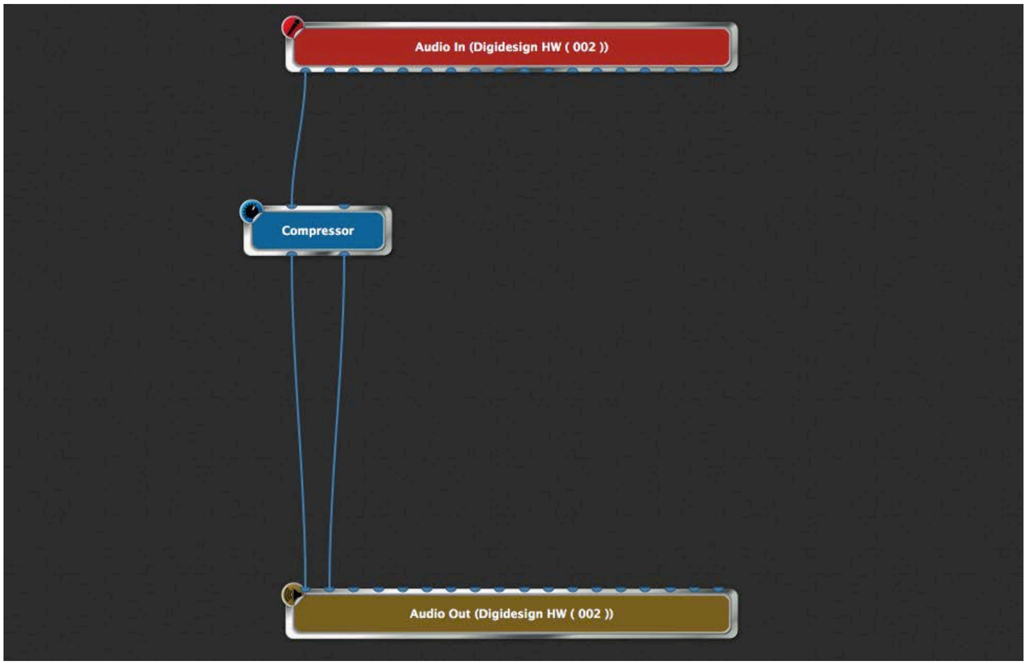
An alternative approach is create a rackspace with one of the controllers, say, the one used at rehearsal, then duplicate it, and modify the MIDI In block to change the MIDI Input source to the performance controller. Now you have rackspaces that are identical except for the controller used.

Configuring Audio Plugins

This tutorial describes how to add effect plugins to vocals and/or guitar.

1. **Launch Gig Performer**.If you have previously created a .gig file, it will automatically load (unless you have unchecked “Load last GIG file on startup”on theOptions>General Preferences tab). If so, go to File>New Gig to start with a clean slate.

2. **Create a signal chain for voice.**We’ll start by constructing a typical signal chain for live vocals. Since we won’t be using a MIDI controller for this tutorial, start by right-clicking on the MIDI In (Omni) block, then selecting “Delete” from the popup menu that appears. Then turn off all speakers and connect a microphone to an input of your audio interface, then enter Connections View and, using the procedures described in the previous tutorial, add the compressor plugin of your choice(because it is an effects processor, its block will be colored blue, as opposed to sound generating plugins, which are colored orange), rename it “Compressor,”and make audio connections to the red and brown interface blocks. Slowly turn up the volume of your sound system (to avoid feedback), then double-click on the plugin to open its editor and adjust its parameters until you are satisfied with the degree of compression. Close the editor to return to Connections View. The main areaof the screen should look something like this:

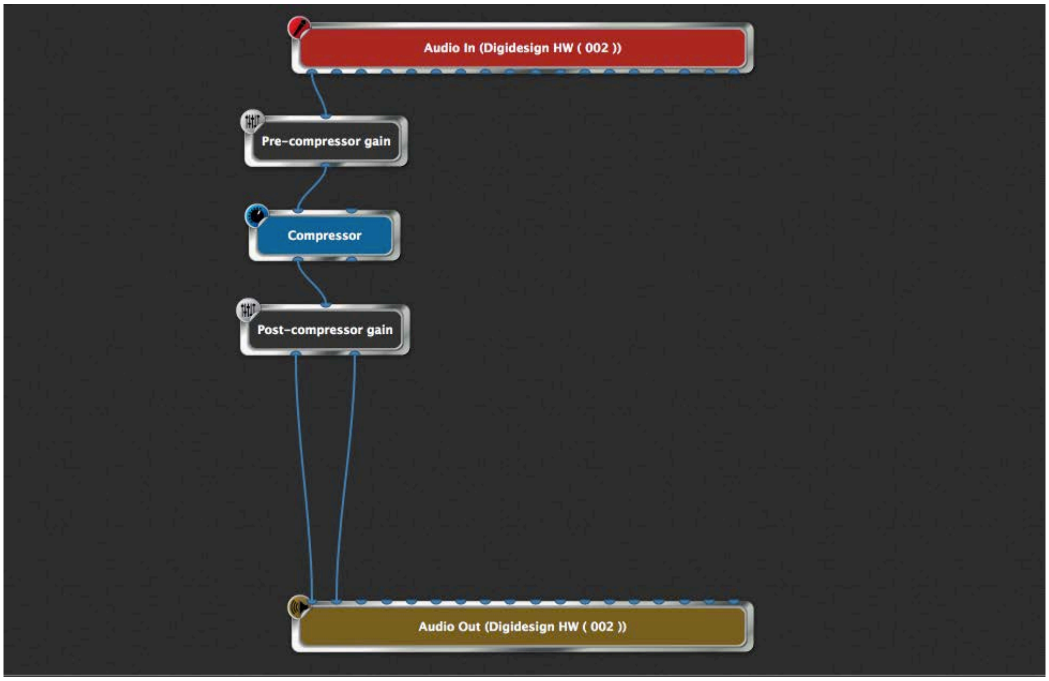

Audio In (Digidesign HW (002))
Compressor
Audio Out (Digidesign HW (002))

(Your screen may look slightly different, since the audio input and output blocks will reflect the name of your connected audio interface as well as the number of inputs and outputs it provides.)

Tip: If you're running Gig Performer on a Mac OSX system, the Audio MIDI Setup utility allows you to create Aggregate Devices, each consisting of multiple audio interfaces. Click here for more information.

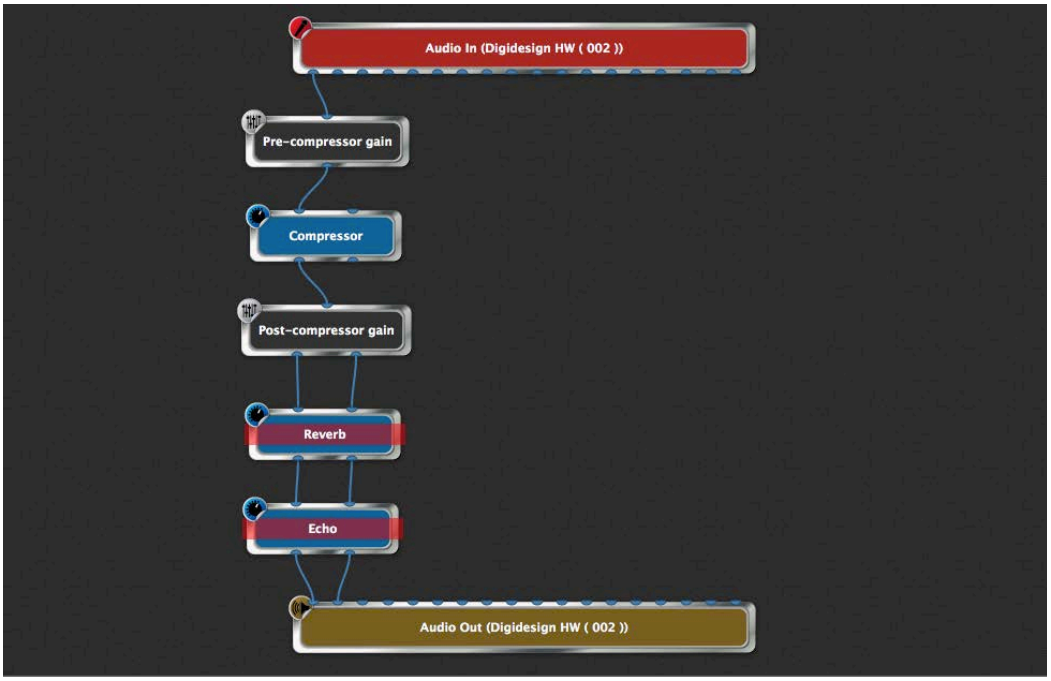
Note: If you have multiple audio interfaces connected to your system and you want to use an interface other than the one being displayed, choose the **Options>Audio setup** command and select the desired interface on the Audio I/O tab. You can even specify different interfaces for input and output.

3. **Add gain controls.** Next, for metering purposes, we'll add twobundled Gig Performer plugins. Insert a "Gain Control (mono)"plugin before the compressor and rename it "Pre-compressor gain," then insert a"Gain and Balance Control (mono to stereo)" plugin after the compressor and rename it "Post-compressor gain."(The latter has the benefit of converting a mono source such as a microphone to stereo, which will allow us to add stereo reverb and echo effects.) The main area of the Connections View screen should now look like this:


Audio In (Digidesign HW (002))
Pre-compressor gain
Compressor
Post-compressor gain
Audio Out (Digidesign HW (002))

Tip: There are two ways to modify connections: You can delete an existing connection by right-clicking on it and selecting "Remove this connection" (or by dragging it to any empty space) and then create a new connection. Alternatively, you can click near one end of a connection and drag it to a new pin.

4. **Add delay and reverb.**Let's complete the signal chain with the reverb and echo/delay plugins of your choice.Rename the two plugins"Reverb" and "Echo," respectively. The main area of the Connections View screen should now look like this:


Audio In (Digidesign HW (002))
Pre-compressor gain
Compressor
Post-compressor gain
Reverb
Echo
Audio Out (Digidesign HW (002))

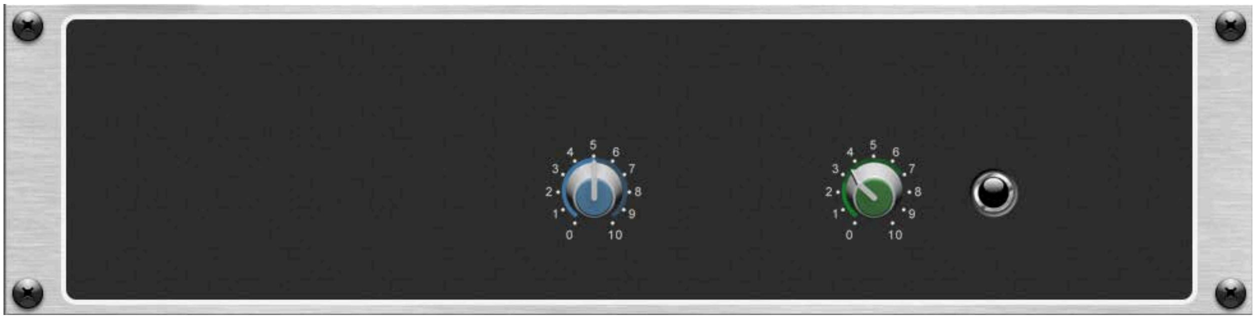
Note: Plugin blocks that are bypassed are overlaid with a red see-through rectangle, as shown in the illustration above.

Important note: There is a difference between bypassing a plugin and muting it.

For virtual instrument plugins, bypassing actually turns off processing for that plugin, thus stopping the sound from being generated. Muting, which is accomplished by inserting a Gain Control, Gain and Balance, Gain and Balance (m->s), or one of the audio mixer plugins and clicking the **Mute** button or setting its volume slider to 0, causes an audio signal to continue to be generated, but with the output silenced. If you're playing back a loop or click track during live performance but you don't want the audience to hear it for a period of time, mute the plugin instead of bypassing it. That way, it will continue to play (in your headphones, if so routed) until you're ready for the audience to hear it again.

In contrast, bypassing an effects plugin allows signals to pass through the plugin, but without modification.

5. **Add widgets to control the vocal sound.**It can be very helpful to be able to adjust the amount of reverb added to singers' voices during live performance, and you almost certainly will want to be able to bring echo in and outas needed. Accordingly, go to Performer View and click on the Widget Layout Designer icon to enter Edit mode. Drag a blue knob, a green knob, and a red LED into the panel and arrange them roughly this way:

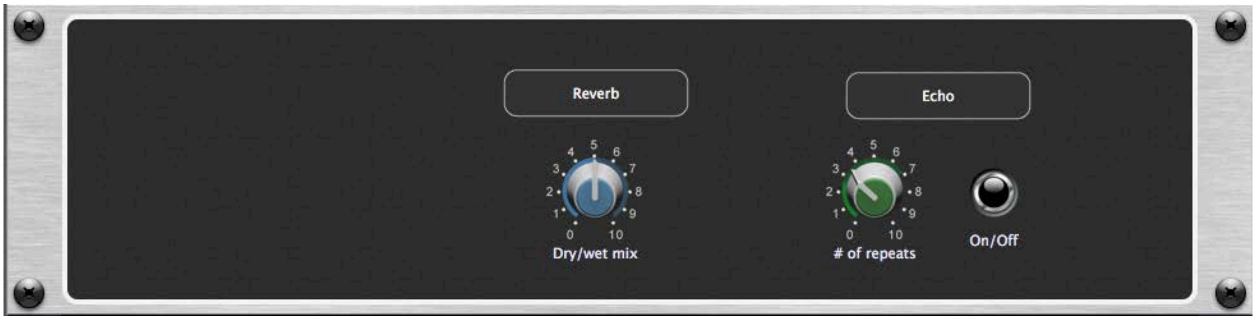


Add captions and assignments to these widgets as follows:

- Blue knob
  - Caption:"Dry/Wet mix"
  - Assignment: Dry/Wet parameter of your reverb plugin.
- Green knob
  - Caption: "# of repeats"
  - Assignment: Feedback parameter of your echo plugin. (This determines the number of repeats you hear.)
- Red LED

  - Caption:“On/Off”
  - Assignment: “Bypass plugin” parameter for your echo plugin. (Click on “Invert value”so that the plugin is bypassed when the LED is unlit and active when it is lit.)

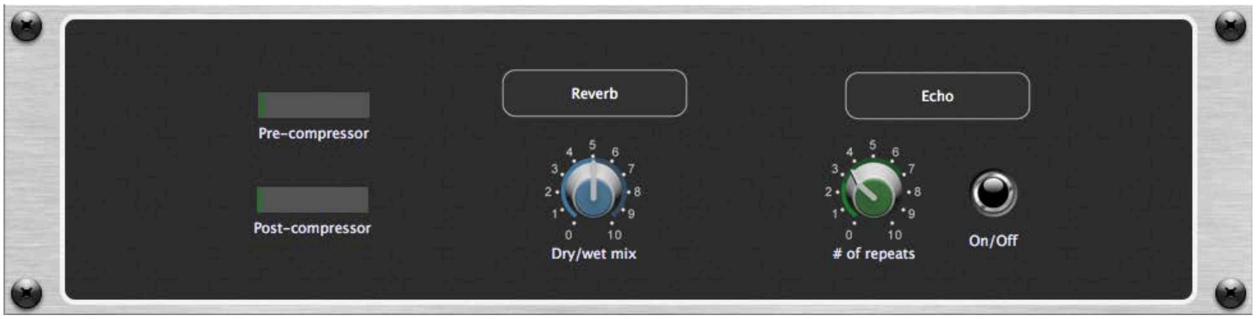
Finally, for purely cosmetic reasons, drag two Text label widgets into the panel. Position one over the blue knob and caption it “Reverb,” thenposition the other over the green knob and red LED and caption it “Echo.”Exit Edit mode and return to Performer View. Your panel should now look like this:



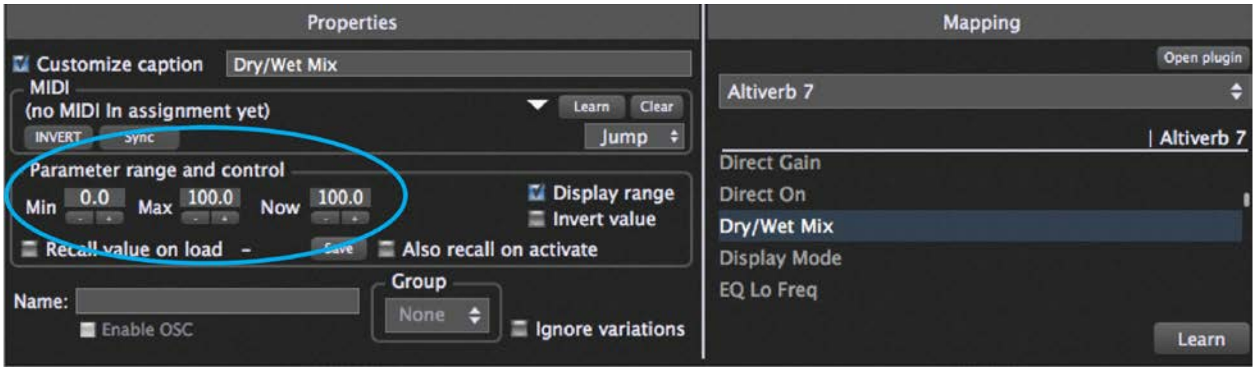
6. **Add metering.**Metering can be very helpful to verify proper gain structure and ensure that a signal is not being overloaded and distorted. The meter widgets provided by Gig Performer (available in both vertical and horizontal configurations) give you the ability to visuallymonitor levels at any point in the signal chain. To add metering, simply insert Gain Control (or Gain and Balance Control) plugins wherever you want to be able to view signal (as we did in step # 2 above) and assign meter widget(s) to their “Magnitude” parameter. Accordingly, drag two horizontal meter widgets into the left-hand side of the panel and add captions and assignments as follows:

- Top horizontal level meter
  - Caption: “Pre-compressor”
  - Assignment: “Magnitude” parameter for the Pre-compressor gain plugin.
- Bottom horizontal level meter
  - Caption: “Post-compressor”
  - Assignment: “Magnitude” parameter for the Post-compressor gain plugin.

Your panel should now look like this:



7. **Add scaling.**Most singers like to have at least some reverb on their voice at all times, but rarely will they want their sound to be swimming in reverb. This is where Gig Performer's scaling control comes in handy. It allows you to specify a minimum and maximum range for each widget, while still allowing full range of motion. To demonstrate, take your Reverb plugin out of bypass, then go back into Edit mode and click on the blue knob to select it. In the Properties section of the inspector, you'll see an area labeled "Parameter range and control" (circled):



☑ Customize caption: Dry/Wet Mix
MIDI: (no MIDI in assignment yet)
Jump
Parameter range and control: Min 0.0 Max 100.0 Now 100.0
☑ Display range
□ Invert value
□ Recall value on load
□ Also recall on activate
Name:
□ Enable OSC
Group: None
□ Ignore variations
Altiverb 7
| Altiverb 7
Direct Gain
Direct On
Dry/Wet Mix
Display Mode
EQ Lo Freq

Enter a Min(minimum) value of 30 and Max (maximum) value of 60. (The numbers can be typed in directly, or you can use the + and – buttons below to increment or decrement the value.) Now, when the blue knob widget is at its fully counterclockwise "0" position, you'll hear a dry/wet mix of 30% (which, for most plugins, will result in a small amount of reverb), while at its fully clockwise "10" position, you'll hear a dry/wet mix of 60% (which, for most plugins, will result in a significant, but not overwhelming amount of reverb. (Feel free to adjust the Min and Max values to taste!)

Similarly, an infinite number of echo repeats can be a very unpleasant thing—the sound can quickly spiral out of control—so set the Max value for the green knob widget to 70. That way, when the green knob is at its fully counterclockwise "0"position, you'll hear a single echo, while at its fully clockwise "10" position, you'll hear several—but not too many—echoes. Again, feel free to adjust to taste.

8. **Test, tweak, and save.**Test that all is working as expected by moving the blue and green knob widgets manually with mouse-dragging operations and turning the red LED on and off by clicking on it. (Take your echo plugin out of bypass before doing so.) If you have a MIDI controller or pedalboard connected to your system, you can associate each of these three widgets with physical devices such as knobs, sliders, buttons, and pads on your controller. Obviously the two knob widgets are best controlled from knobs and sliders, but the echo on/off widget (the red LED) can be controlled from buttons, pads, a pedal, or even a note being played! For example, you can assign it to the highest note on your keyboard—one which is probably rarely played in normal circumstances—and the echo will turn on only when that key is depressed.(Click here for an important note about using keys to control widgets.) Again, thanks to the flexibility of Gig Performer, the only limit here is your imagination. When you've got everything tweaked to your satisfaction, add any variations you want and name your rackspace—for the purposes of this tutorial, we'll call it "Vocal." Finally, be sure to save your work.

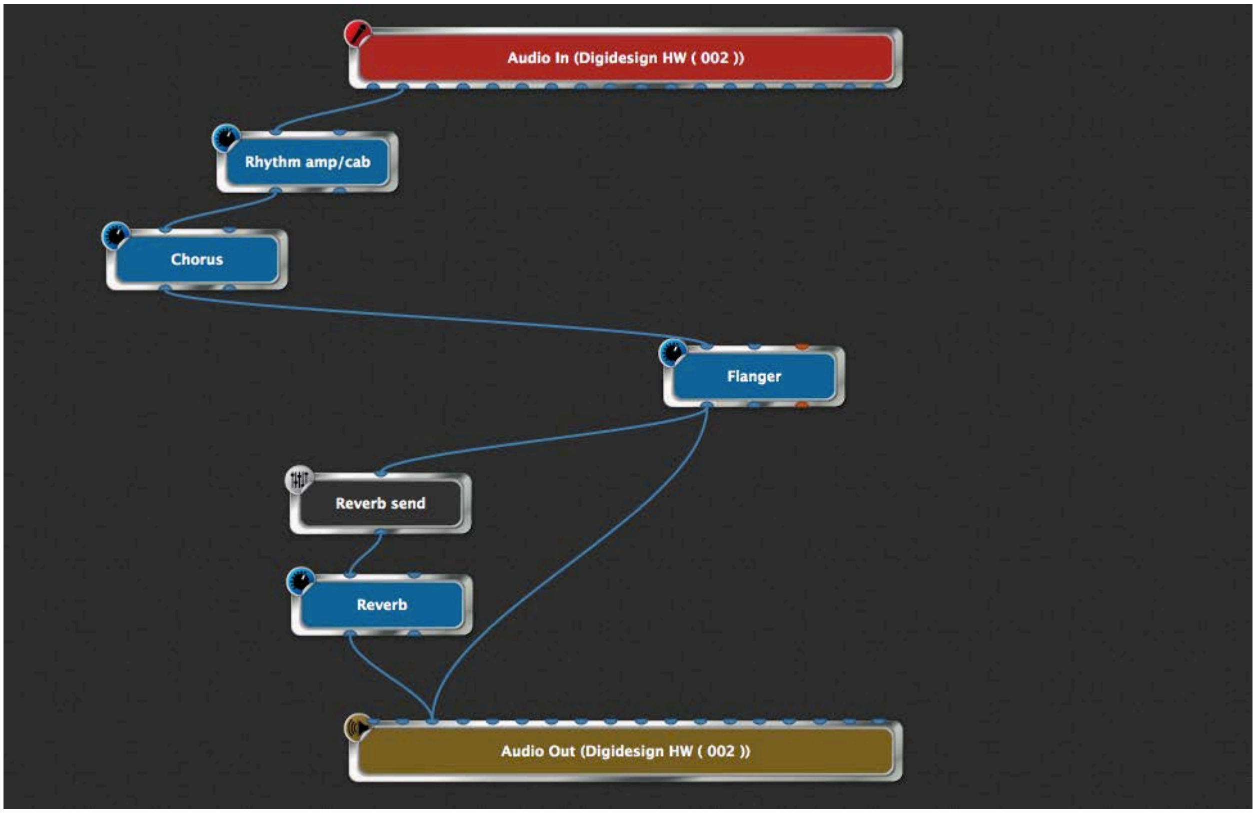
9. **Create a new instance.**Next, we're going to be constructing a signal chain for guitar, but before we do so, we'll be placing it in a separate instance. Instances are simply multiple copies of Gig Performer running simultaneously in the same computer. Their use allows each member of a band to have independent control over his or her ownrackspaceswithout affecting the rackspacesof other bandmembers.For example, a guitarist can freely switch between rhythm and lead settings without affecting the reverb and echo settings being applied to vocals.

   Go to **Options>Advanced Settings...** and make sure the "Enable multi-instance support" option is checked. (For more information, click here.) Then go to File>Create New Instance and type in a name. For the purposes of this tutorial, let's call the new instance "Guitar." A dialog box will appear informing you that you have created a new instance.

   Click OK to close the dialog and your new instance will appear—a complete new iteration of Gig Performer with no plugins, no connections, and no widgets.

10. **Create signal chains for guitar.**Now it's time to build a typical signal chain for guitar—actually, two of them. Begin byonce again deleting the MIDI In (OMNI) block and then connecting a guitar (via a DI box if necessary) to an input of your audio interface other than the one used for the microphone earlier in this tutorial.Enter Connections View andconnect that input toanamp/cabinet model

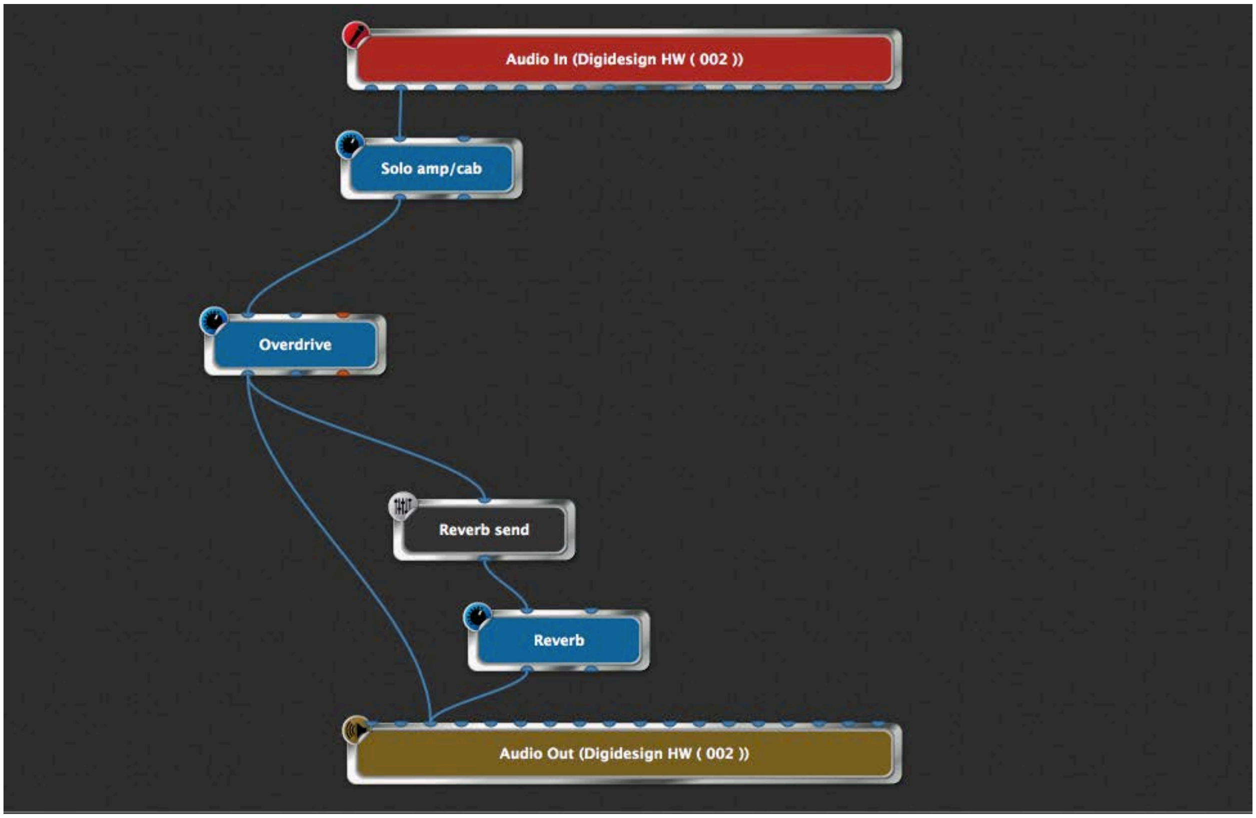
plugin, followed by chorusing andflanging plugins. Add a Gain Control pluginand the reverb plugin of your choice. Rename the amp/cabinet plugin "Rhythm amp/cab," the chorusing plugin "Chorus," the flanging plugin "Flanger," the Gain Control plugin "Reverb Send" (since it's taking on that function; see the text box below) and the reverb plug-in "Reverb." Connect the output of the Flanger plugin to the "Reverb send" plugin, and the output of the the "Reverb send" plugin to the Reverb plugin. Finally, connect an output from both the Flanger and Reverb plugins to one of your audio interface's outputs (choose an output other than the one being used for the microphone signal), as shown in the illustration below:


Audio In (Digidesign HW (002))
Rhythm amp/cab
Chorus
Flanger
Reverb send
Reverb
Audio Out (Digidesign HW (002))

Tweak the parameters in the various plugin editors until you get a sound you like (use the Gain Control to set the desired amount of reverb), then double-click on the rackspace name. In the Rackspace Properties dialog that appears, rename the rackspace "Rhythm," then click on Apply to close the dialog.

Gig Performer makes it easy to create **reverb sends**, where signal is routed to a reverb (or other type of signal processing plugin) from multiple sources: Simply insert a Gain plugin between the output of each source and the reverb plugin. The volume of each Gain plugin provides independent control over how much signal is sent to the reverb.

Next, we'll create a second rackspace for soloing. Since we're going to use the same audio connections (although we'll be substituting different plugins), choose **Edit>Duplicate the current rackspace**command (Cmd-Shift-D/Ctrl-Shift-D) to adda new rackspace, then double-click on its name. In the Rackspace Properties dialog that appears, change the name ofthis new rackspacefrom "Rhythm_1" to "Solo". In Connections View, substitute an amp/cabinet model if you prefer (you can also opt to simply use the same one you used for the Rhythm sound—Gig Performer allows you to use the same plugin multiple times, up to the limits of your RAM and CPU resources), followed by the distortion/overdrive plugin of your choice, then delete the flanging plugin. Rename the amp/cabinet plugin "Solo amp/cab" and rename the distortion/overdrive plugin "Overdrive." Your main Connections View screen should now look like this:


Audio In (Digidesign HW (002))
Solo amp/cab
Overdrive
Reverb send
Reverb
Audio Out (Digidesign HW (002))

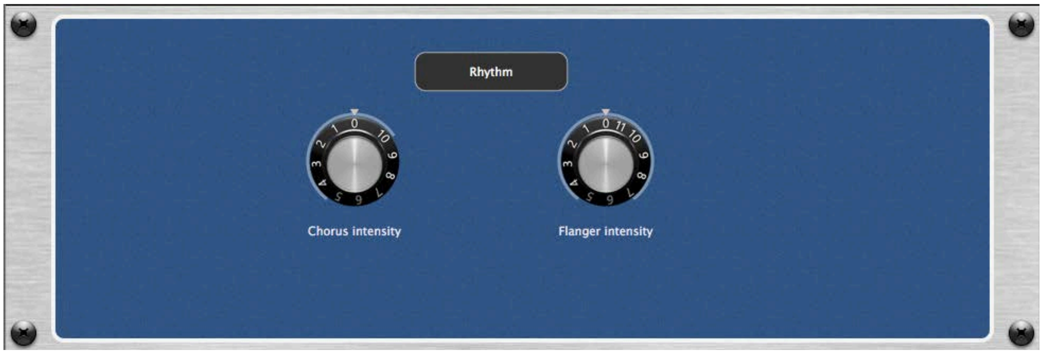
Guitarists especially will appreciate the handy full-screen chromatic tuner provided in Gig Performer, available at any time with the click of a mouse or the press of a MIDI footswitch—even with the touch of a button on a smartphone or tablet running an OSC app. Click here for more information.

11. **Add widgets to control the guitar sounds.**Go to Performer View and select the Rhythm rackspace, then click on the Widget Layout Designer icon to enter Edit mode. Drag an "Old school" knob and a "The 11" knob (our Spinal Tap version of the "Old school" knob—it goes to 11!) into the panel.

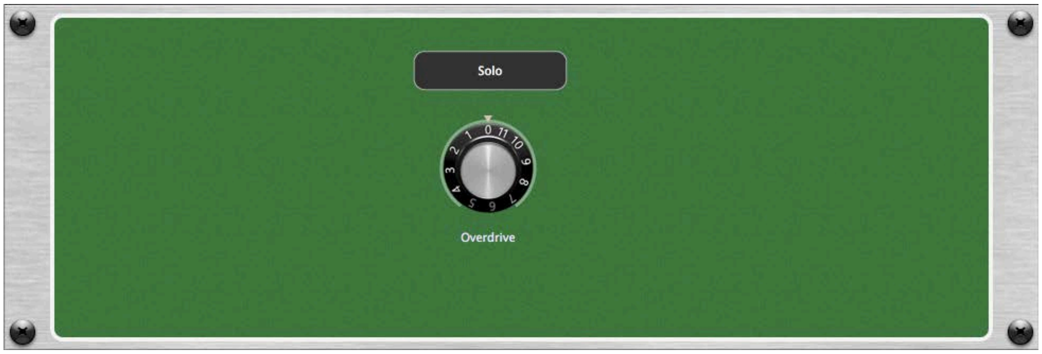
Add captions and assignments to these as follows:

- "Old school" knob
  - Caption: "Chorus intensity"
  - Assignment: Effect level parameter of your Chorus plugin.
- "The 11" knob
  - Caption: "Flanger intensity"
  - Assignment: Dry/Wet parameter of your Flanger plugin.

Apply widget scaling and MIDI controller assignments to taste, then, purely for cosmetic reasons, add a text label widget and caption it "Rhythm," then double-click on the panel and in the Rack Unit Properties menu that appears, change its "Style"(actually, its color) to blue. It should now look like this:



Now go to the Solo rackspace and drag a text label widget and an "11 knob" widget into the panel. Caption the "11" knob "Overdrive" and assign it to the Drive parameter of your Overdrive plugin. Apply widget scaling and MIDI controller assignments to taste, thencaption the text label widget "Solo" and change the rack panel "Style" (color) to green. It should now look like this:

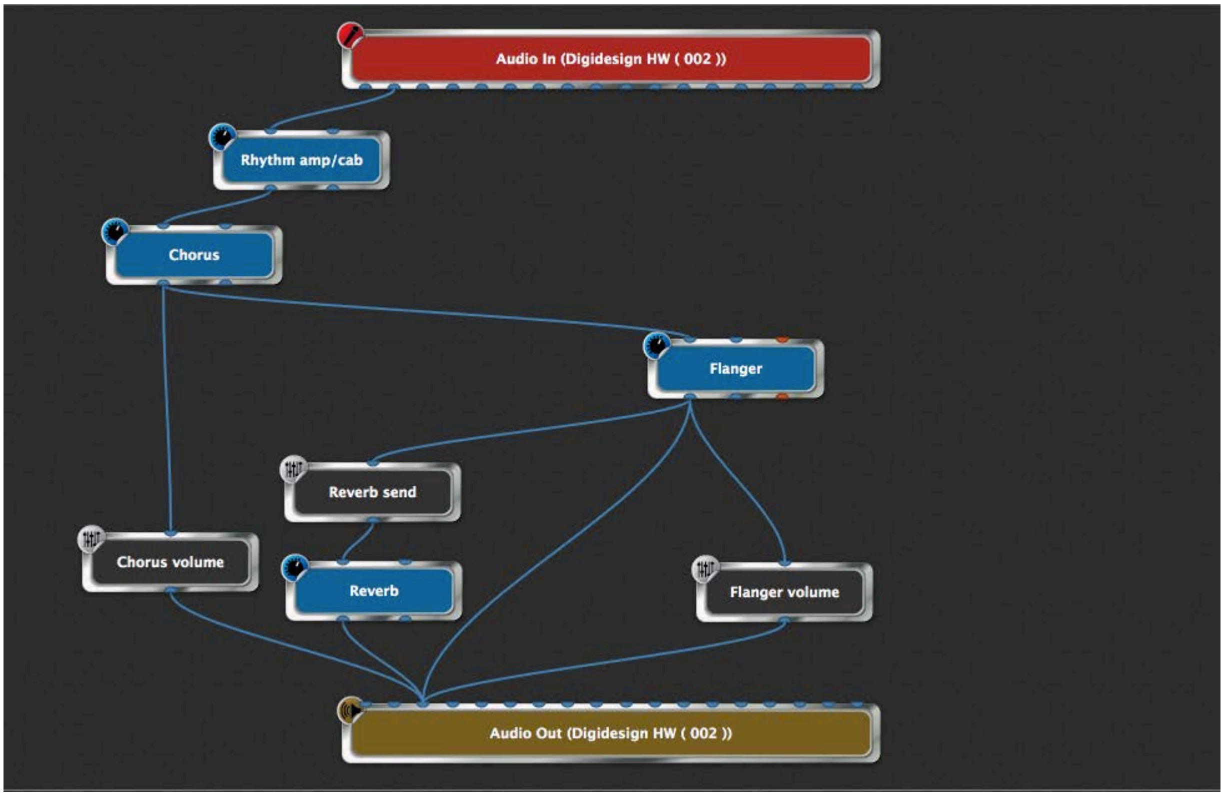


The advantage to putting different sounds into different rackspaces (as opposed to creating one complex rackspace with multiple signal chains and complicated switching) is that you can go from one to the other, glitch-free, with the use of simple MIDI program change messages transmitted via a MIDI controller, such as up-and-down switches on a pedalboard—or even via a smartphone or tablet running an OSC app. What's more, Gig Performer's Tail length function allows you to set the maximum time before a sound is cut off—up to 5 seconds—when you switch rackspaces—something that's particularly useful in situations like this, where reverb or echo is being applied to a sound. Try it!

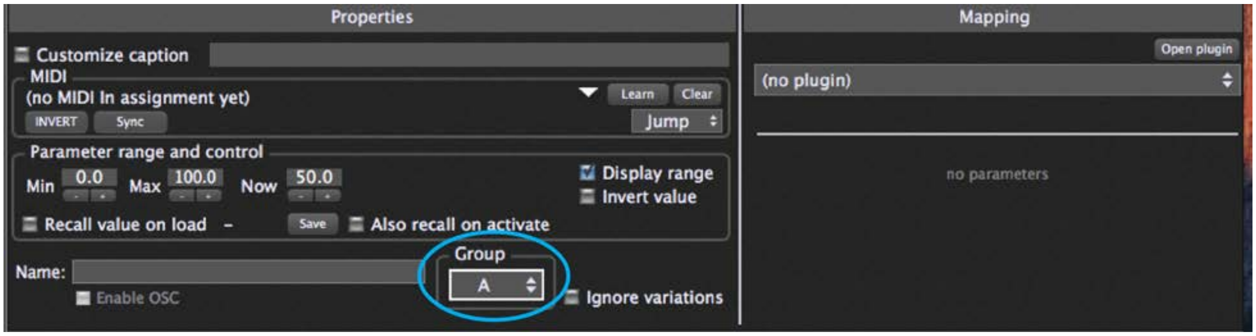
12. Create a widget group to crossfade between the chorus and flanging effects.Now it's time to explore Gig Performer's powerfulwidget groupfeature. The concept behind it is simple: Any widget assigned to a group can be controlled by any or all other widgets assigned to the same group. This enables you to change multiple parameters together.

(There are two other ways to control multiple parameters simultaneously: one is to assign the same MIDI control to multiple widgets, the other is using GP Script to control any number of parameters with a single widget.)

To demonstrate, go to Connections View, select the Rhythm rackspace, and add two Gain Controls, renaming one of them "Chorus volume" and the other one "Flanger volume." Connect the output of the "Chorus" plugin to the input of the "Chorus volume" plugin and the output of the "Flanger" plugin to the input of the "Flanger volume" plugin. Then connect the outputs of both the "Chorus volume" and "Flanger volume" plugins to the same output of your audio interface that you've been using for both the Rhythm and Solo rackspaces. Your main Connections View screen should now look like this:



Next, go to Performer View and move the two knob widgets to set moderate amounts of both Chorus intensity and Flanger intensity. Then go into Edit mode and drag a horizontal slider widget into the same panel, positioning it below the two knob widgets.Next, in the Properties section of the inspector, click on the dropdown menu below for the widget group (circled below) and select Group A. (Gig Performer provides 26 widget groups, lettered A through Z.)



□ Customize caption

MIDI

(no MIDI In assignment yet)

Jump

Parameter range and control

Min 0.0 Max 100.0 Now 50.0

☑ Display range

□ Invert value

□ Recall value on load –

□ Also recall on activate

Name:

□ Enable OSC

Group A

□ Ignore variations

(no plugin)

no parameters

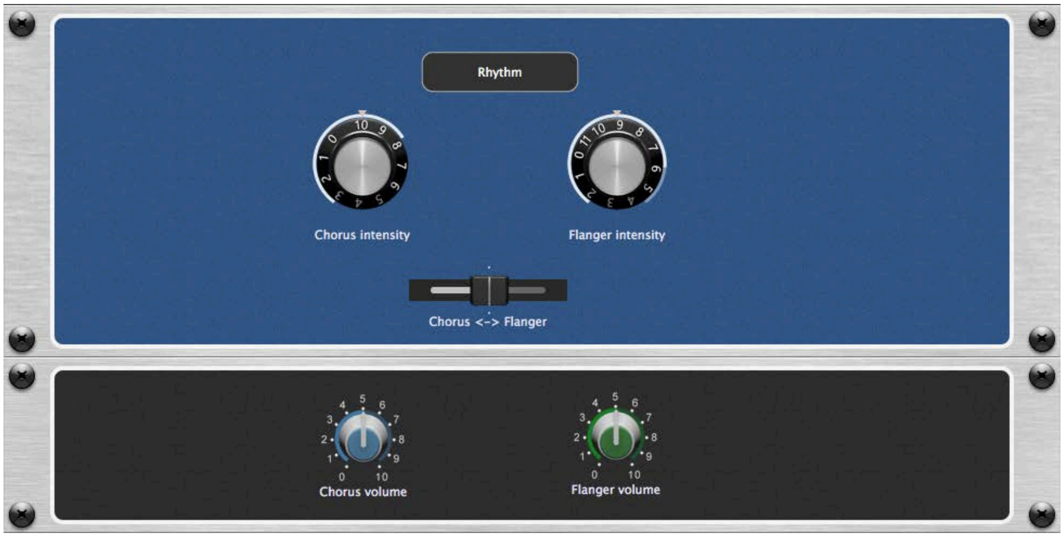
Caption the widget“Chorus <-> Flanger”but don’t assign it to any plugin; that’s because we’re going to use it as the “master” to control two other widgets we’ll be placing ina separate panel. Accordingly, click on the New Panel button beneath the currentpanel (depending on the screen size of your laptop, you may need to scroll down or resize the window to see this button). From the dropdown menu that appears, click on “Create a new 1U panel”. This is the smallest size panel provided by Gig Performer, but it will work fine for our purposes since we won’t actually need to see its widgets during live performance. Drag a blue knob widget into the left side of the new panel and a greenknob widget into the right side. Caption theblue knob“Chorus volume”and assign it to the Gainparameter for the“Chorus volume” plugin. Then caption the blue knob “Flanger volume” and assign it to the Gain parameter for the“Flanger volume” plugin. Next**—and this is important—**click on Invert Value (in the Properties section of the inspector) for the blueknob widget**only**.

Finally, assign both knob widgets to Widget Group A—the same group as the horizontal slider in the panel above. Because all three widgets are assigned to the same group, they will all move together. When the greenknob widget is turned up, it will, as expected, cause the volume of the flanging effect to increase, but turning up the blueknob widget will do the oppositebecause it is set to “Invert Value”: it will cause the volume of the chorus effect to decrease, and vice versa.

(Another method of achieving inversion of the blue knob’s action that provides greater precision is to use scaling: instead of clicking “Invert Value,” simply enter the maximum value in the Min field in the Parameter range and control (for example, set it to 84), and the minimum value in the Max field (say, 10). In this example, the blue knob will now go from a value of 84 when it is all the way down to a value of 10 when it is turned all the way up.)

As a result, when you move the horizontal slider from left to right, you’ll hear the chorusing effect crossfade into the flanging effect, and vice versa.Try it! If you want visual evidence that all is working correctly, go momentarily to Connections View and double-click on the “Chorus volume” and “Flanger volume” plugin blocks to open their editors, then return to Performer View. The editors will remain open and you’ll be able to see the effect of moving the horizontal slider.

Your final Rhythm rackspace should look like this:



If you find the two knobs in the small bottom panel visually distracting, Edit mode provides a couple of easy ways to fix that:

a) Right-click on the top panel, then select the “Convert to 4U panel” option from the drop-down menu. This will enlarge its size, thus hiding the panel below from view.

OR

b) Create one or more new panels and move them in-between your “master” blue panel and the small panel containing the blue and greenknob widgets.

13. **Test, tweak, and save.**Test that all is working as expected by moving widgets manually with mouse-dragging operations or from your MIDI controller. When you’re satisfied with all settings, be sure to save your work by choosing the **File>Save As...** commandand typing in a filename. Like all Gig Performer files, those saved when working within an instance are given the extension .gig and contain complete information about all rackspaces and variations. Bear in mind also that, because we created them in separate instances, the vocal signal chain created earlier in this tutorial continues to be completely active along with the guitar signal chain just created, and that making changes in one instance does not in any way affect other instances.

Important note: When multiple instances are open, be sure to exit from each one separately.

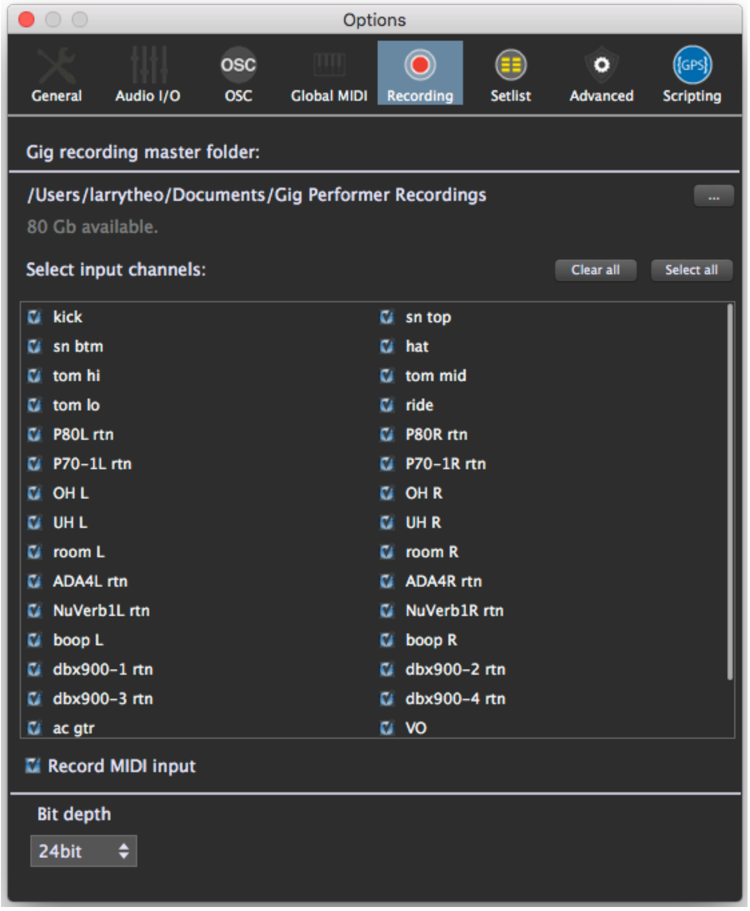
Capturing a Performance with the Audio Input and MIDI Recorders

The Audio Input Recorder is a basic utility to allow live performance to be captured raw, before it has entered your Gig Performer plugin network. Assuming you save your Gig Performer gig, you can make a quick capture recording, and in post-production use the WAV files that are captured in a DAW or other music or audio program. Also see Audio Input Recorder in the reference section.

A MIDI Recorder is also available that captures incoming data from each MIDI device to a track in a Standard MIDI File.Generally, a device corresponds to a physical MIDI port.

> **Tip:**You can see a list of MIDI devices known to Gig Performer by right-clicking any MIDI In module and moving the cursor to the **Change MIDI Input Device** command to see its associated drop-down menu.

1. Click the Recorder button or choose **Options>Recording Settings** to open the Recording Options dialog.



Gig recording master folder:

/Users/larrytheo/Documents/Gig Performer Recordings

80 Gb available.

Select input channels:

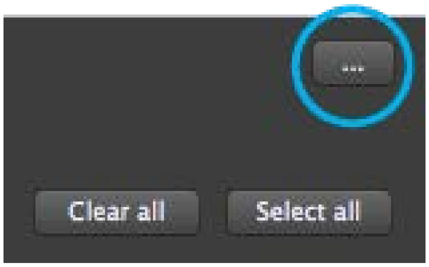
☑ kick ☑ sn top
☑ sn btm ☑ hat
☑ tom hi ☑ tom mid
☑ tom lo ☑ ride
☑ P80L rtn ☑ P80R rtn
☑ P70-1L rtn ☑ P70-1R rtn
☑ OH L ☑ OH R
☑ UH L ☑ UH R
☑ room L ☑ room R
☑ ADA4L rtn ☑ ADA4R rtn
☑ NuVerb1L rtn ☑ NuVerb1R rtn
☑ boop L ☑ boop R
☑ dbx900-1 rtn ☑ dbx900-2 rtn
☑ dbx900-3 rtn ☑ dbx900-4 rtn
☑ ac gtr ☑ VO

☑ Record MIDI input

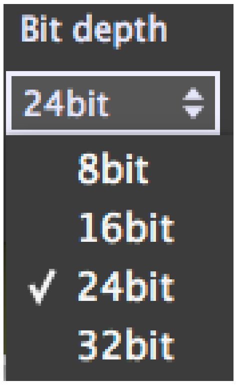
Bit depth

24bit

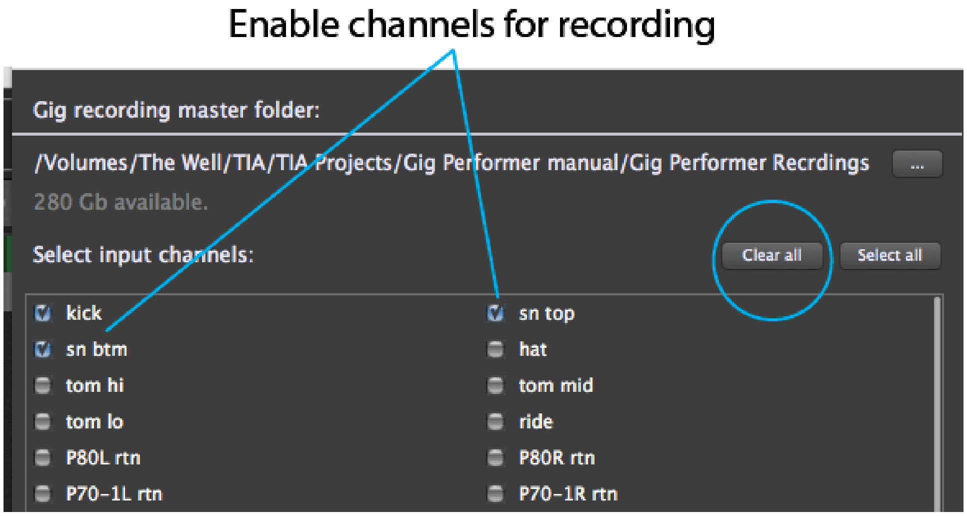
2. Click the Set Destination Folder button and navigate to a destination for the recorded files.



3. Choose a bit depth for the files from the menu in the lower left corner. The sample rate is set by the audio interface. Technically, the 32-bit setting uses floating point representation, whereas the other settings use fixed point.



4. We only want to record from a few inputs, so the fastest way to set up for that is to click the Clear All button to disable all inputs, then record-enablethechannels you want to capture.
5. If you want to record real-time MIDI data from your controllers, click the MIDI Record box. Data from all physical MIDI In ports known to Gig Performerwill be captured to a Type 1 Standard MIDI File, with data from eachport recorded onto its own track.


Enable channels for recording

Gig recording master folder:

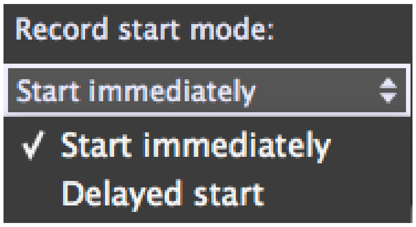
/Volumes/The Well/TIA/TIA Projects/Gig Performer manual/Gig Performer Recrdings

280 Gb available.

Select input channels:

☑ kick	☑ sn top
☑ sn btm	□ hat
□ tom hi	□ tom mid
□ tom lo	□ ride
□ P80L rtn	□ P80R rtn
□ P70-1L rtn	□ P70-1R rtn

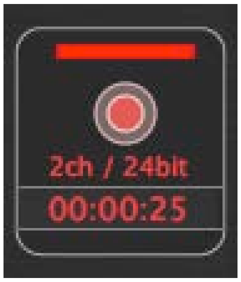
6. Make sure that Start Mode is set to **Start immediately**. Recording begins the moment the Start button is clicked.


Record start mode:

Start immediately

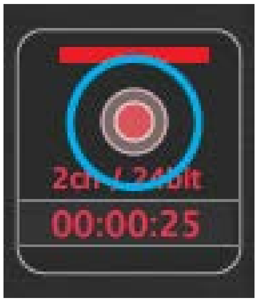
7. Click the Startbutton to begin recording. The Recorder button changes to a small record panel that tells you how many channels you are recording and at what bit depth, along with the current record time.

The Recorder button changes into a small record panel with information and a stop button.


2ch / 24bit

00:00:25

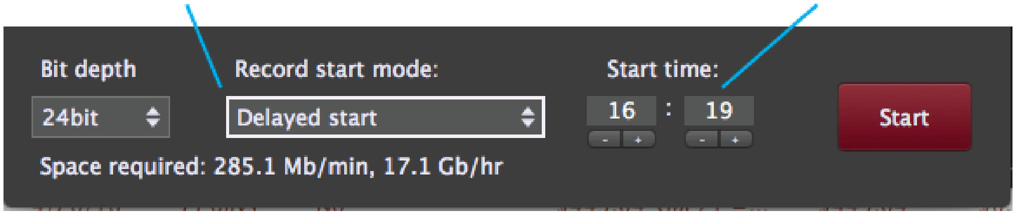
8. Click the Stop button in the record panel to stop recording.



9. Now let's try a recording set to start at some other time. For instance, we might want to set Gig Performer at sound check to begin recording at the show's scheduled start time. Set Record start mode to Delayed start. New Start time fields appear. Enter the time when you want the recording to start in hours and minutes.

Set to Delayed start mode

Delayed start time


Bit depth: 24bit

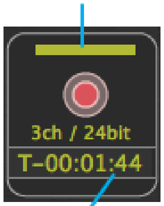
Record start mode: Delayed start

Start time: 16 : 19

Space required: 285.1 Mb/min, 17.1 Gb/hr

10. Click the Start button to start recording. The Recorder button changes to a countdown panel. The panel flashes green while it is counting down, then turns to red when it begins recording. When you are done recording, click the record stop button to stop recording.

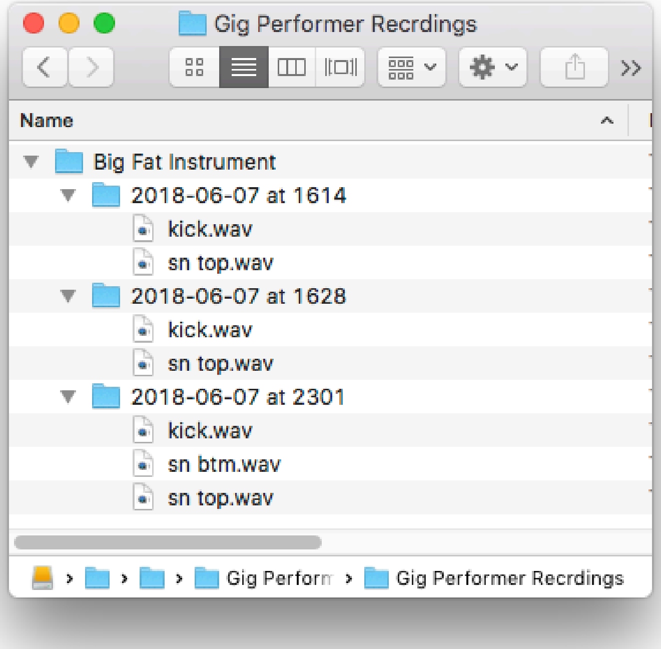
Flashes green while counting down



Countdown time

11. Your recordings are located in the destination folder chosen in the Recording Options dialog. Inside that top-level folder is a folder named for the current Rackspace, and inside of that are folders named with date and time of the

recordings.These folders contain the WAV files recorded at the indicated dates and times.



Big Fat Instrument
2018-06-07 at 1614
kick.wav
sn top.wav
2018-06-07 at 1628
kick.wav
sn top.wav
2018-06-07 at 2301
kick.wav
sn btm.wav
sn top.wav

Since each recording session will include the show date as part of the name, you can start a recording session every night of a tour without having to worry about documenting when the recordings were made.

The Audio Mixer/Router

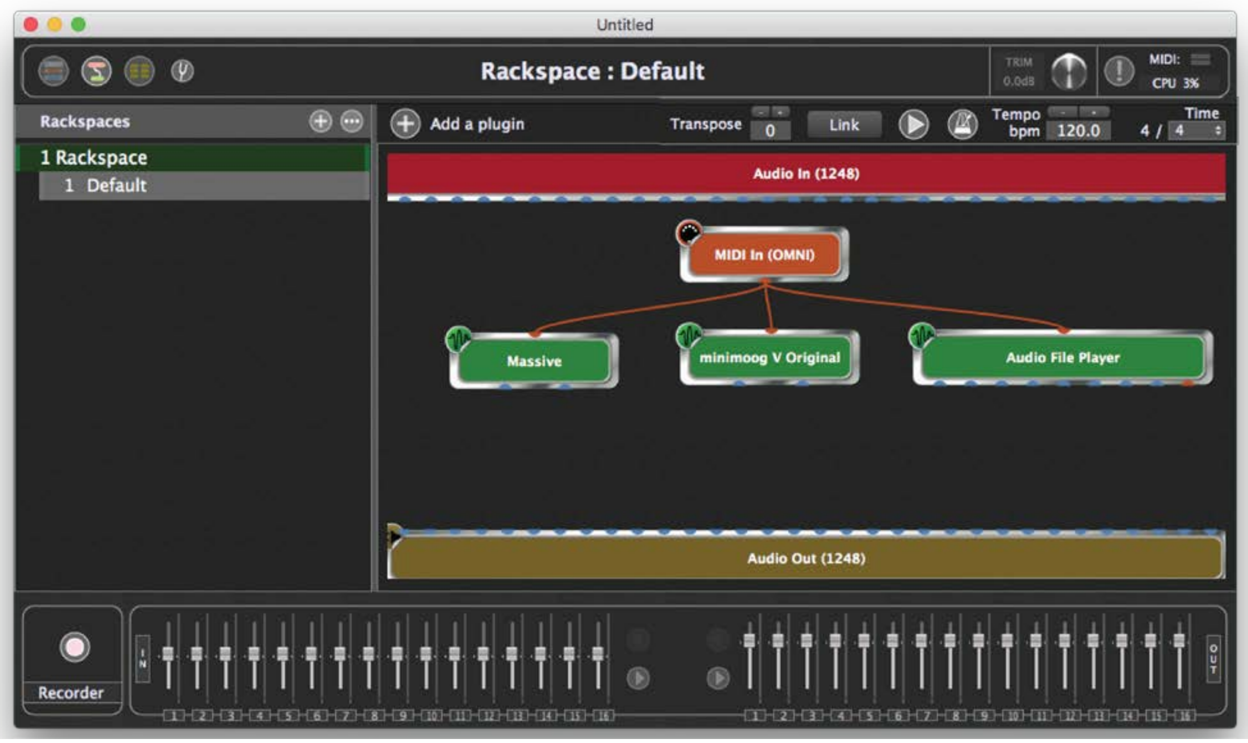
The Audio Mixer/Router is a flexible plugin that eliminates the need for a Gain plugin on the output of each source. Functions like level, mute, and output routing can be controlled from widgets so you can, for example, mute and unmute different sources, route channels to various effects devices connected to the outputs, or, of course, mix levels.

There are three versions of the plugin, withtwo, four or eight inputs. Each input hosts a stereo channel pair, or can be switched to mono operation. This yields between two and 16 input channels.

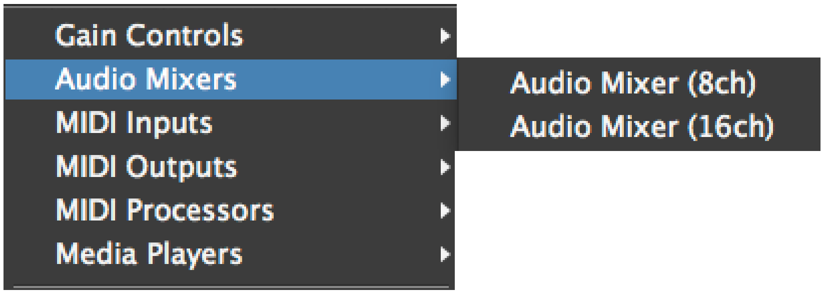
Each input has a fader, mute button, and solo button, and can be routed to one of two, four, or eight stereo output pairs, depending on which mixer plugin you're using. For more details, see Audio Mixer/Router in the Reference section.

Tutorial:

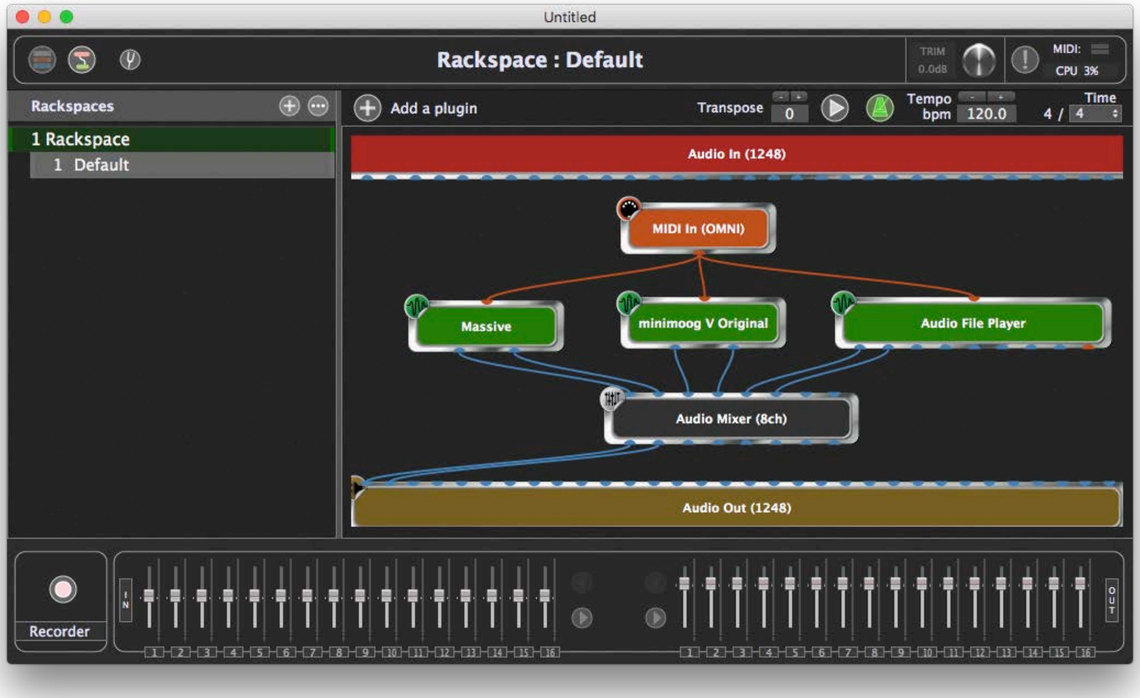
1. Make a new rackspace and add a few sound sources. Your screen should look something like this:



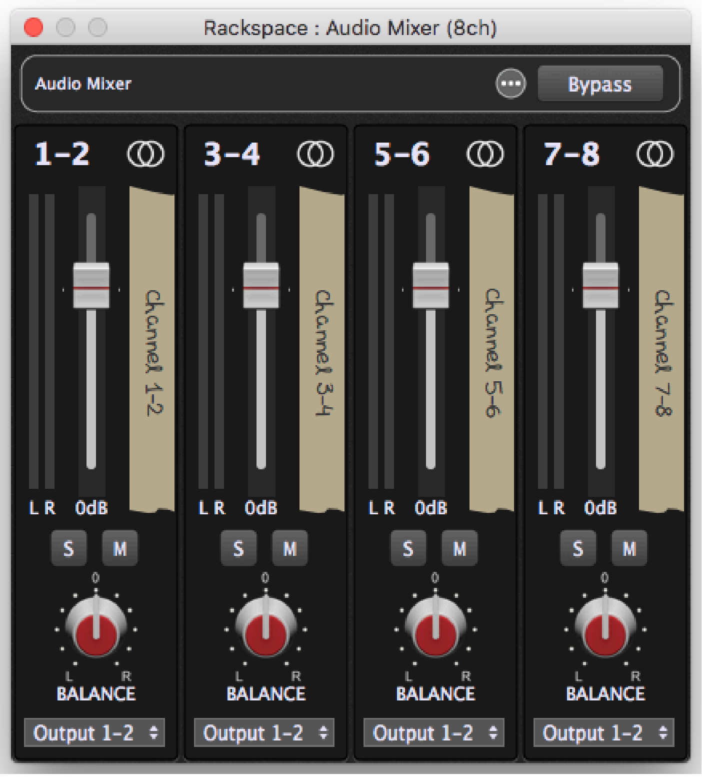
2. Now click the **Add a plugin** button and choose an eight-channel mixer.



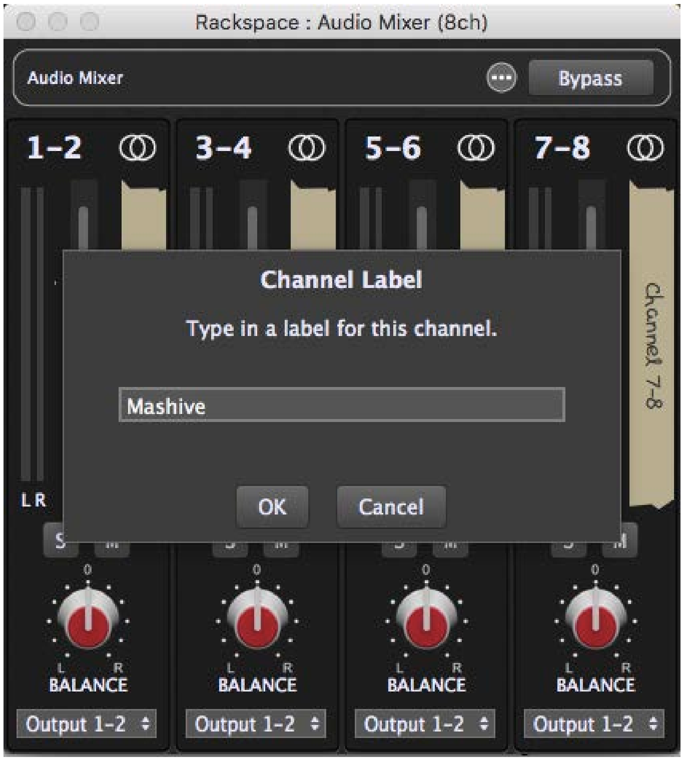
3. Drag from the outputs of the source on the left (in this example, Massive) to mixer inputs 1 and 2 to feed them into the mixer. To route stereo pairs, you can shift-drag, but you have to respect odd/even pair numbering: the left output from a plugin has to be dropped on an odd-numbered input of the mixer, or the right output dropped on an even-numbered input. Try shift-dragging from an output of one of the other sources. Shift-drag from output 1 of the mixer to an odd-numberedinterface output, and the outputs 1 and 2 will be connected for monitoring.



4. Double-click the mixer in the Connections view to open the editor for the plugin.



5. Let’s rename a channel. Double-click on a label to edit the name.

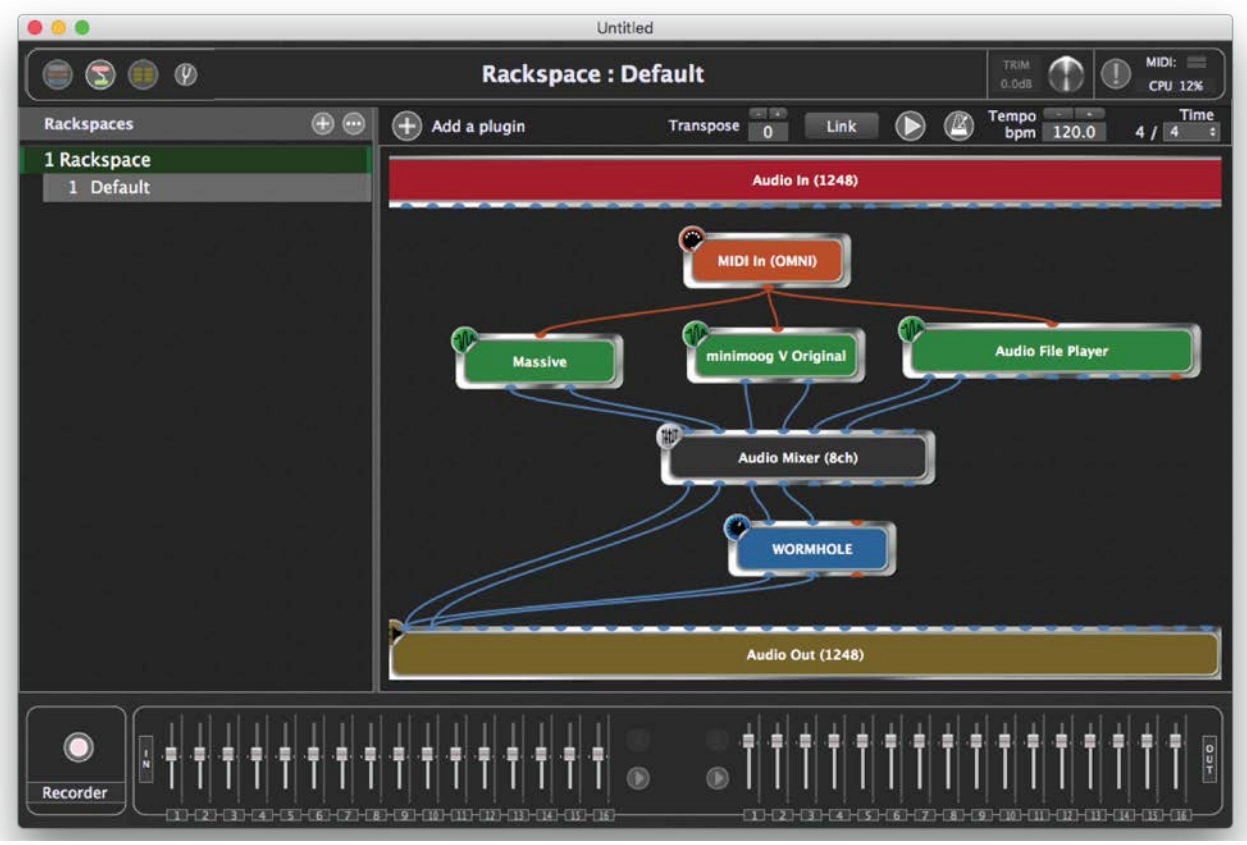


Channel Label

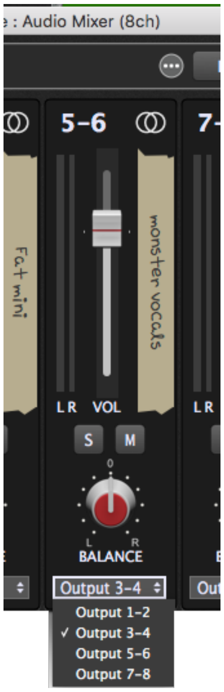
Type in a label for this channel.

Mashive

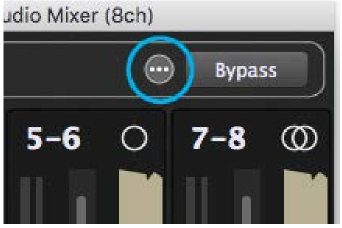
6. Now add an effect of some sort. Drag from mixer outputs 3 and 4 to the inputs of the effect, and from the outputs of the effect to inputs 1 and 2 of the interface. These will simply mix in equal amounts to other signals connected directly to those interface inputs.



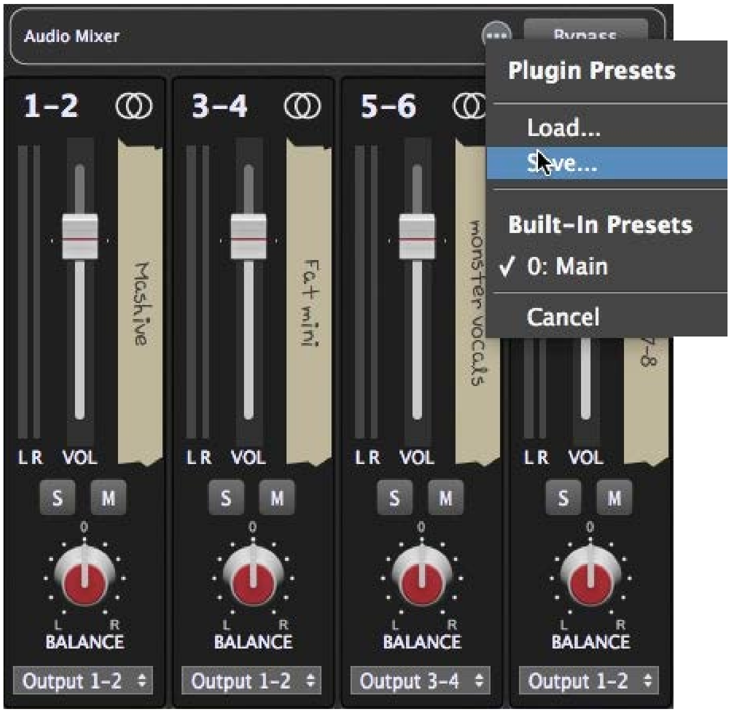
7. We can send the Audio File Player through our effects. Click at the bottom of the third channel strip and choose 3-4 as the outputs. Now the file player is feeding the effects, and the output of the effects is being summed with the outputs from the mixer.



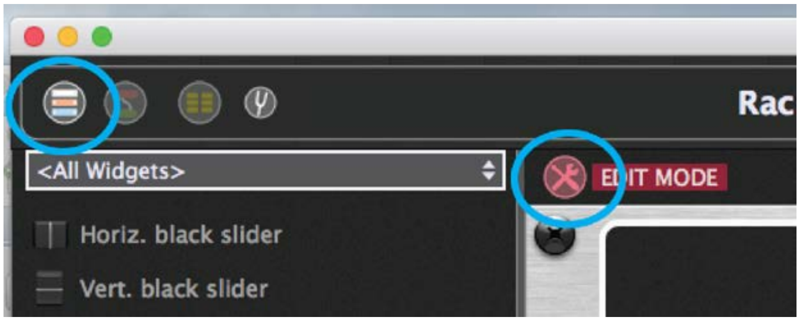
8. Now that our mixer is configured, we can save the setup as a preset. Click on the **Presets** menu button.



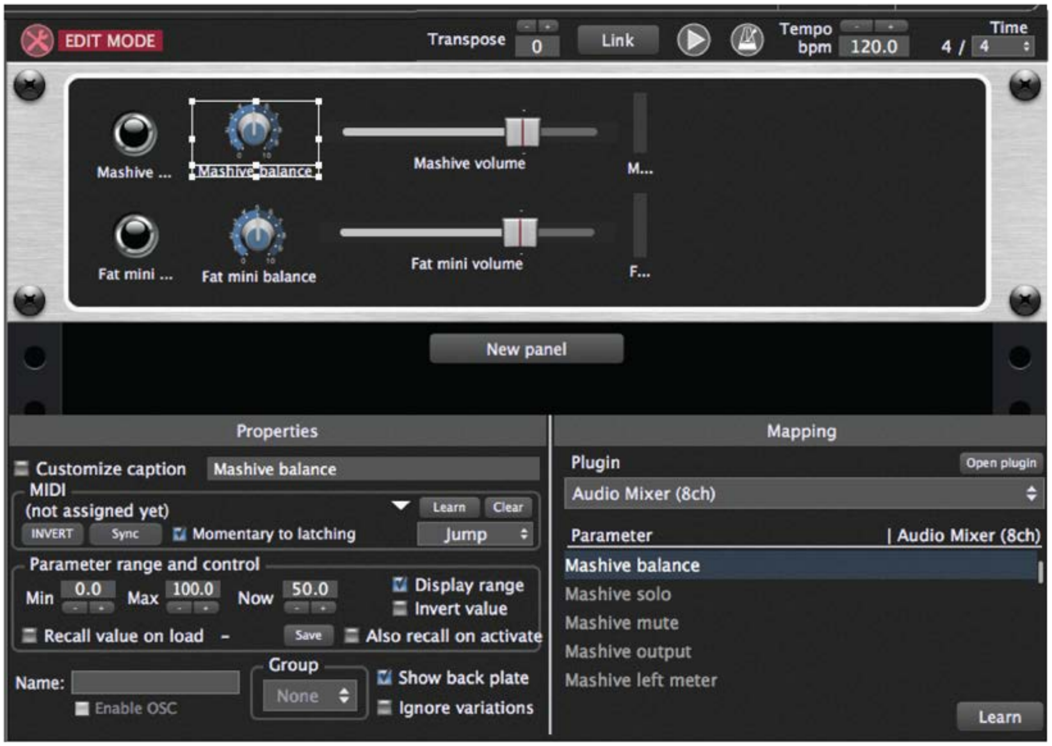
9. Choose **Save** from the popup menu and name your setup.



10. Now, let's attach a few widgets to mixer controls. Click the **Rackspace front view** button in the upper left corner so you can see the rack panels, then click the **Edit Mode** button to start editing the panel.



11. Now add some widgets. In the example below, there are mute buttons, balance controls, level faders, and meters for two inputs to the mixer. All except the meters can easily be mapped to physical controllers, such as on a keyboard.For more information on how to do that, see the Controlling Plugins From the MIDI Controller tutorial.



Transpose 0 · Tempo bpm 120.0 · Time 4 / 4

Properties

- □ Customize caption: Mashive balance
- MIDI: (not assigned yet)
- ☑ Momentary to latching
- Jump
- Parameter range and control: Min 0.0 · Max 100.0 · Now 50.0
- ☑ Display range
- □ Invert value
- □ Recall value on load -
- □ Also recall on activate
- Name:
- □ Enable OSC
- Group: None
- ☑ Show back plate
- □ Ignore variations

Mapping

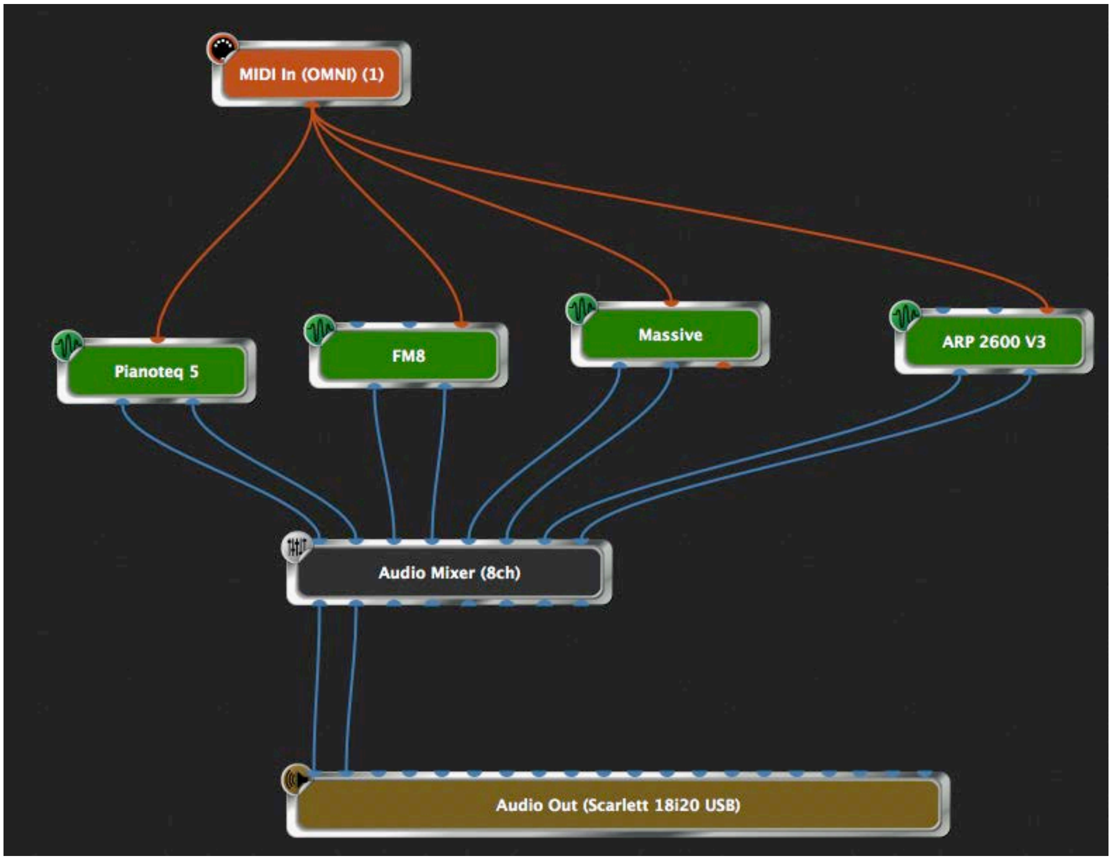
- Plugin: Audio Mixer (8ch)
- Parameter | Audio Mixer (8ch)
  - Mashive balance
  - Mashive solo
  - Mashive mute
  - Mashive output
  - Mashive left meter

12. Click the **Edit Mode button** to return to using the Rackspace and you are ready to mix.

Applications

There are many tasks that can be accomplished with the mixer/router plugin. Here are a few examples to get you going.

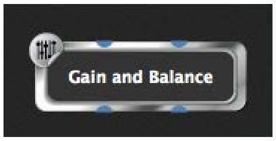
Simple Mixer

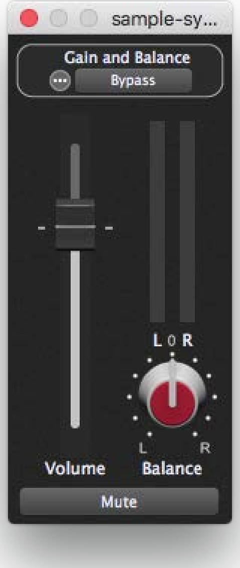


In this example we have four synths with stereo pair outputs feeding the mixer's eight inputs.Each inputhas gain and balance controls in theAudio Mixer plugin, so it is unnecessary to add gain and balance plugins on each individual synth output. All the inputs feeding the mixerin the figure are then routed to outputs 1 and 2 of the mixer plugin,which is connected to the first stereo pair of audio interface outputs.

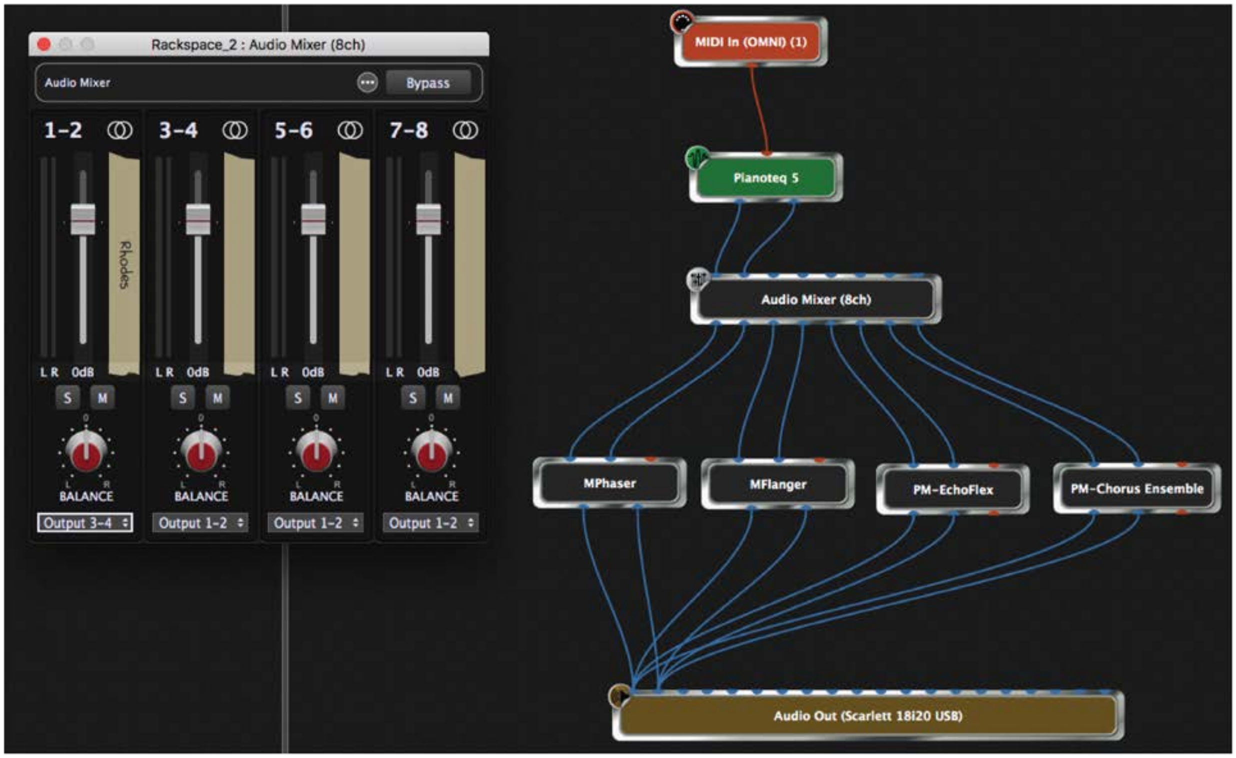
Each channel pair can be soloed or muted as desired and of course these can be controlledfrom front panel widgets (and so from your keyboard controller, once you map them to those widgets) as needed.

You can easily add a master volume for a mixer output pair, if you need, by feeding the mixer outputs to a Gain and Balance plugin. For a classic VCA-style approach, create a widget for the level of each output pair, then form them into a widget group and map them to a single physical knob or slider.





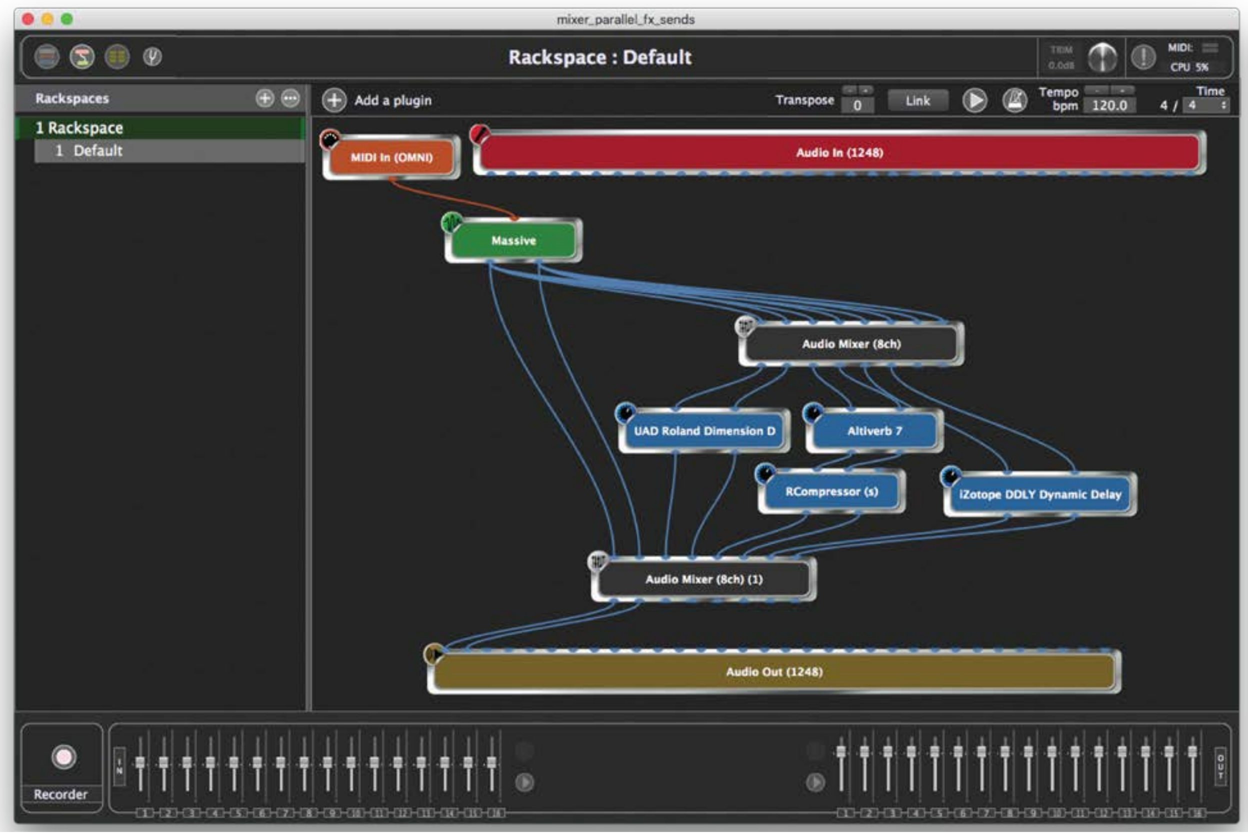
Input router



On the right, in the Connections view of a rackspace, you can see that we have a single synth (the Modartt Pianoteq) connected to the Audio Mixer. The four output pairs of the (eight-channel) Audio Mixer are connected to the inputs offour different effects. If you look atthe first channel strip in the mixer on the left, the Pianoteq is currently being routed to Output 3-4, which feeds the Flanger. Bychanging the output routing, you can quickly switch from one effect to another. Output routing also can be attached to a slider, knob, or switch widget, letting you control effects switching from your controller.

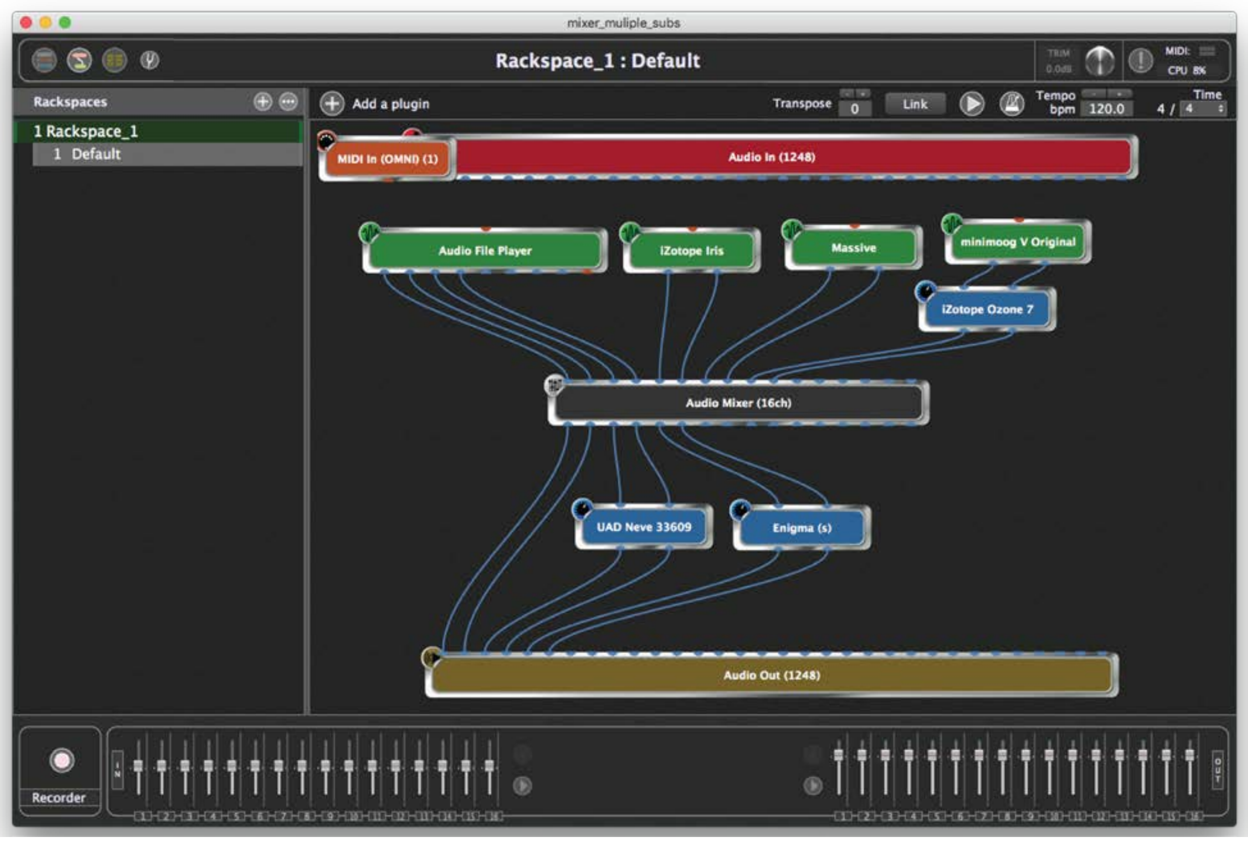
Feed multiple effects (aux sends)

Assigning the source to all of the inputs turns the mixer into a bank of aux sends. Each channel sends the source signal to one pair of outputs, with a separate level control for each. In this case, the outputs of the effects and the original synth source are all mixed at the end through another eight-channel mixer.

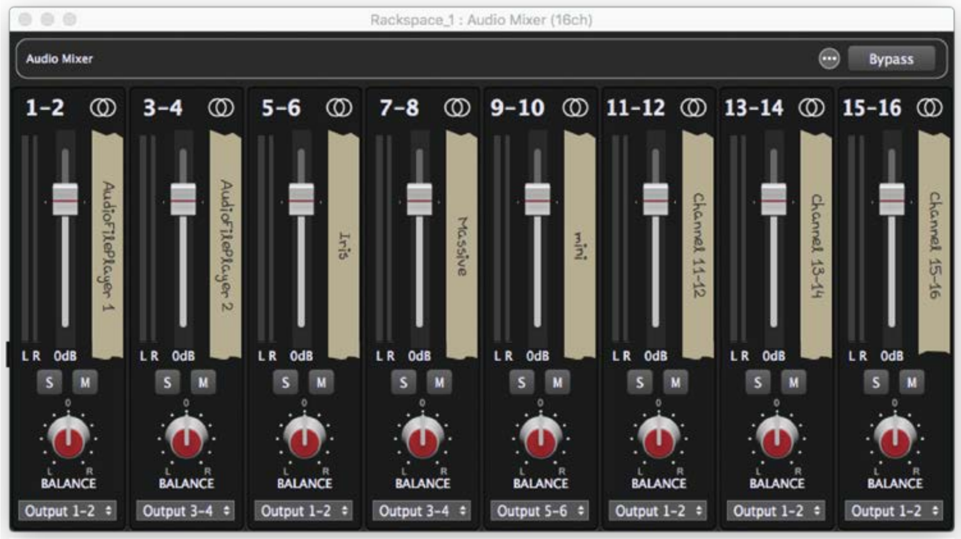


Multiple submixersin one plugin

Since inputs can be independently assigned to outputs, a single 16-channel mixer can be used as if it were a few small submixers in one package.

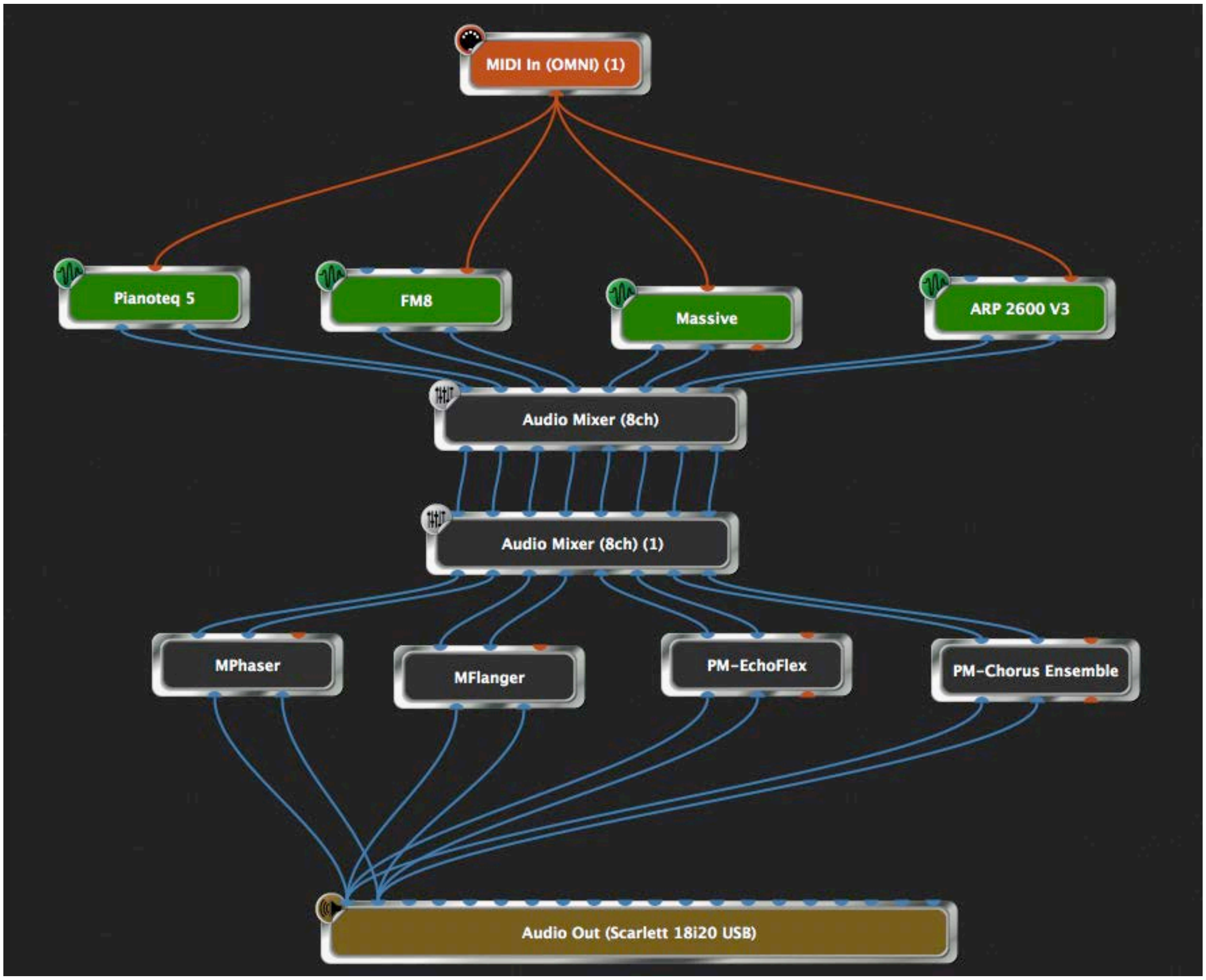


The first and third channel strips, channels 1-2 and 5-6, feed one mix, the second and fourth strips a different mix, and the fifth strip, for the moment, is routed out a third output pair that feeds an effect.


Rackspace_1 : Audio Mixer (16ch)
Audio Mixer
Bypass
1-2
3-4
5-6
7-8
9-10
11-12
13-14
15-16
AudioFilePlayer 1
AudioFilePlayer 2
Iris
Massive
mini
Channel 11-12
Channel 13-14
Channel 15-16
L R
0dB
S
M
BALANCE
Output 1-2
Output 3-4
Output 1-2
Output 3-4
Output 5-6
Output 1-2
Output 1-2
Output 1-2

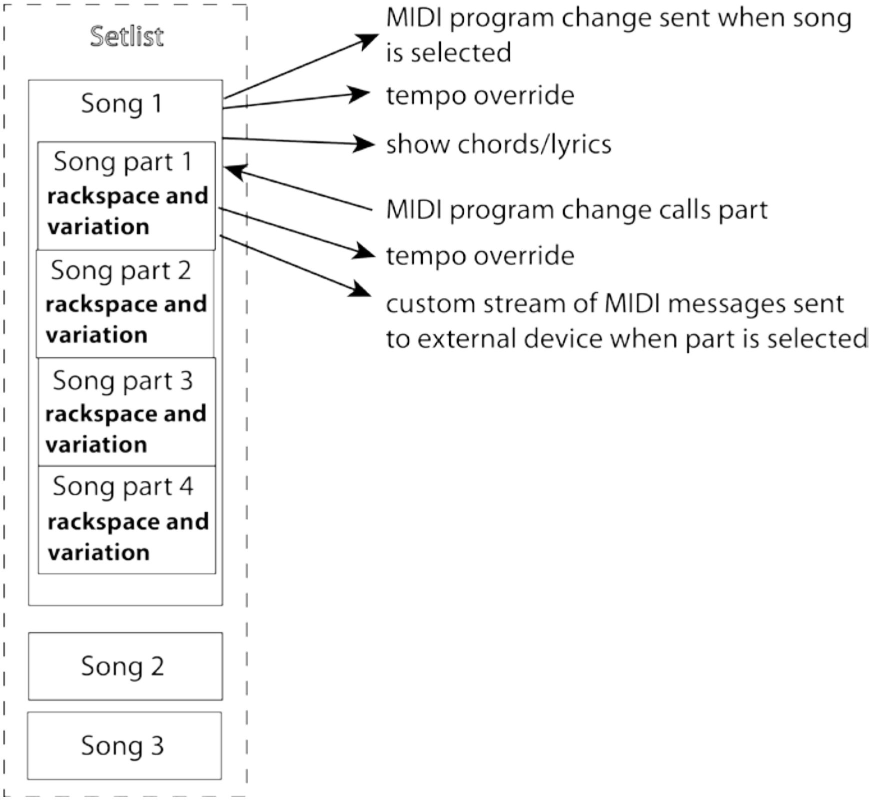
Mixer and Router Together

In this example, we can instantly route combinations of inputs to combinations of outputs soyou can route your piano to the phaser and chorus while routing your pads (ARP 2600) tochorus and echo. By using rackspace variations in Gig Performer you can instantly switchamong arbitrary combinations, using a controller to switch, if you like.



Songs, Song Parts, and Setlists

Gig Performer allows you to quickly and easily organize all of your rackspaces and variations to meet the needs of your performances, by letting you build setlists of your songs. Each song consists of song parts, such as verses, choruses, bridges, interludes, and so forth. A song part is basically a rackspace variation that is called when the part is selected, though there are other song part settings, as well.



Once you have created a number of songs, a setlist can be assembled by dragging songs into it from the song list. The order of songs in the setlist can be changed at any time. Setlists can be duplicated and modified, for gigs that vary only slightly from a usual song order, or a custom setlist can be created for a specific gig.

Gig Performer supports the ChordPro lead sheet file format with both an editor/viewer to enable creating, editing, and viewing ChordPro documents, and the ability to attach a ChordPro document to a song, so that selecting the song opens the ChordPro document to put the chords and lyrics right in front of you.

Of course, there are numerous other options and features that make Gig Performer setlists an extremely powerful and flexible way to navigate your rig through a performance.

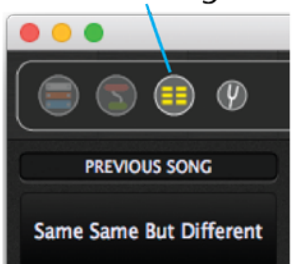
For more details, see Setlist/Songs View.

Let's walk through building and using a setlist.

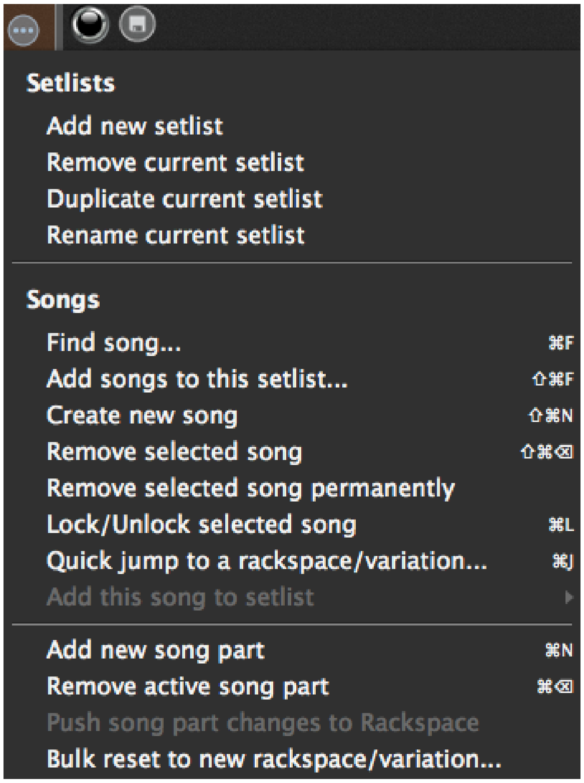
Since we have already described how to create rackspaces and variations, we will assume for this tutorial that you have already created all of the rackspaces and variations you will need. Of course, if that is not so, you can always pause while making a setlist, create any additional rackspaces and variations you still need, then resume your setlist process.

1. **Switch to Setlist/Song view.** Press **Cmd-Shift-L** (**Ctrl-Shift-L** on Windows) or click the **Setlist/Song view** button to change to the setlist/song view.

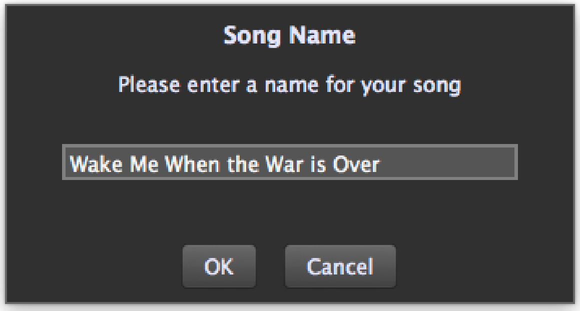
Click here to switch to Setlists/Song View



The setlists menu has commands for managing setlists, songs, and song parts. We will return to this menu often.


Setlists
Add new setlist
Remove current setlist
Duplicate current setlist
Rename current setlist
Songs
Find song... ⌘F
Add songs to this setlist... ⇧⌘F
Create new song ⇧⌘N
Remove selected song ⇧⌘⌫
Remove selected song permanently
Lock/Unlock selected song ⌘L
Quick jump to a rackspace/variation... ⌘J
Add this song to setlist
Add new song part ⌘N
Remove active song part ⌘⌫
Push song part changes to Rackspace
Bulk reset to new rackspace/variation...

2. **Create a new song.** Press Cmd-Shift-N (Ctrl-Shift-N on Win) or choose the **Create new song** command from the **Setlists** contextual menu. Enter a name in the dialog that appears.

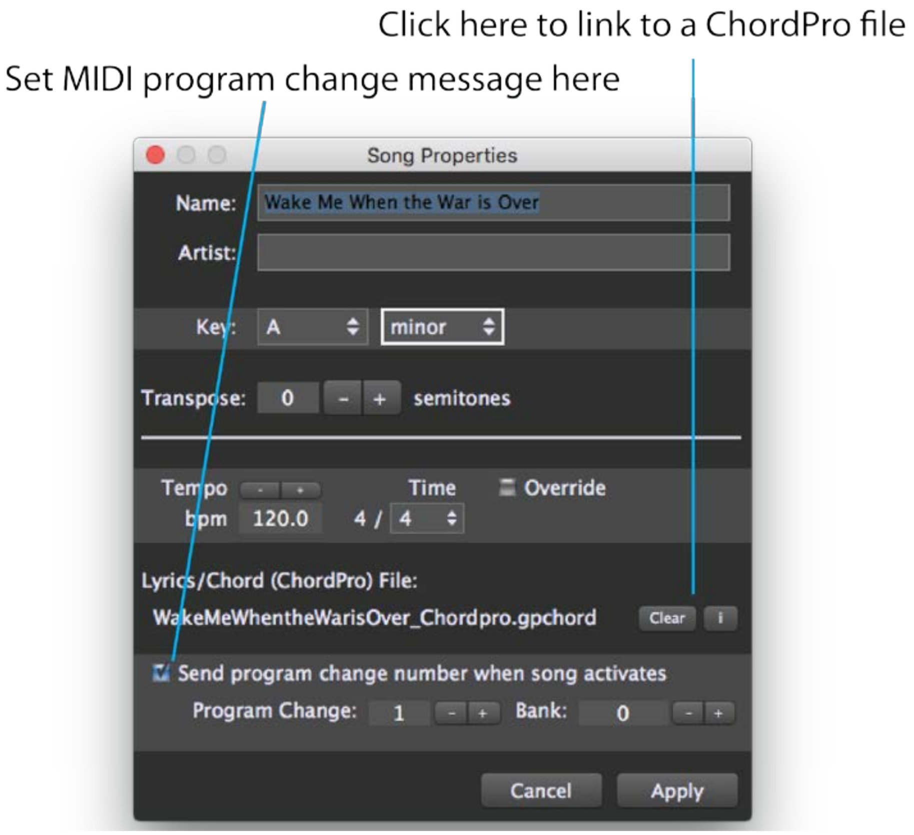


Song Name

Please enter a name for your song

Wake Me When the War is Over

3. **Set a MIDI program change to be sent.** External MIDI devices can be coordinated by a MIDI program change message that can be sent when a song is selected.The MIDI output device and port over which the program message is sent is set in the Options>Setlist settings.


Click here to link to a ChordPro file
Set MIDI program change message here
Song Properties
Cancel
Apply
Clear
i
- +

Name: Wake Me When the War is Over

Artist:

Key: A minor

Transpose: 0 semitones

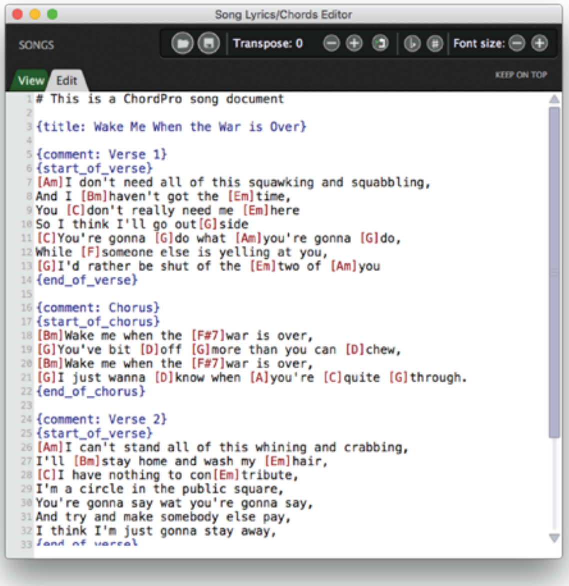
Tempo bpm 120.0 Time 4 / 4 Override

Lyrics/Chord (ChordPro) File: WakeMeWhentheWarisOver_Chordpro.gpchord

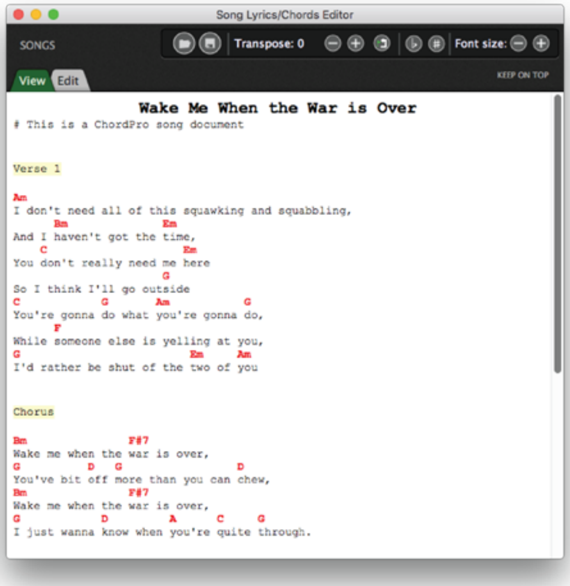
☑ Send program change number when song activates

Program Change: 1 Bank: 0

4. **Gig Performer supports the ChordPro format.** ChordPro is a simple text file format for creating lead sheets with song lyrics and chords. A ChordPro file can be linked to a song in Gig Performer, so that calling the song opens the ChordPro file. Gig Performer also has a text editor for creating, editing, and viewing ChordPro files.
5. **Select the song to which you want to attach a ChordPro file.**
6. **Open the ChordPro editor.**Type **Cmd-E** (**Ctrl-E** on Win) or choose the **Setlist/Song Settings** command from the **Window** menu to open the Song Lyrics/Chords Editor window.
7. **Enter the song data.** Enter lyrics and chord names for the song, as specified in the ChordPro documentation.
8. **View the document.** Click the View tab in the Song Lyrics/Chords Editor window to see how the document looks.

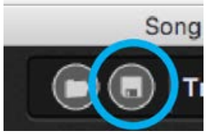


```
# This is a ChordPro song document

{title: Wake Me When the War is Over}

{comment: Verse 1}
{start_of_verse}
[Am]I don't need all of this squawking and squabbling,
And I [Bm]haven't got the [Em]time,
You [C]don't really need me [Em]here
So I think I'll go out[G]side
[C]You're gonna [G]do what [Am]you're gonna [G]do,
While [F]someone else is yelling at you,
[G]I'd rather be shut of the [Em]two of [Am]you
{end_of_verse}

{comment: Chorus}
{start_of_chorus}
[Bm]Wake me when the [F#7]war is over,
[G]You've bit [D]off [G]more than you can [D]chew,
[Bm]Wake me when the [F#7]war is over,
[G]I just wanna [D]know when [A]you're [C]quite [G]through.
{end_of_chorus}

{comment: Verse 2}
{start_of_verse}
[Am]I can't stand all of this whining and crabbing,
I'll [Bm]stay home and wash my [Em]hair,
[C]I have nothing to con[Em]tribute,
I'm a circle in the public square,
You're gonna say wat you're gonna say,
And try and make somebody else pay,
I think I'm just gonna stay away,
{end_of_verse}
```



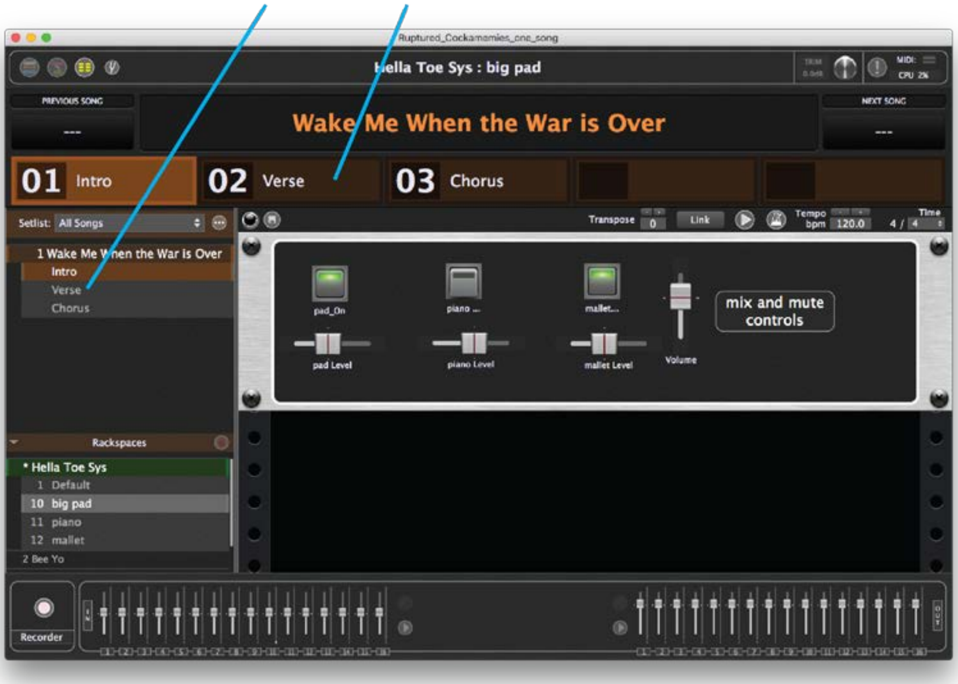
```
                 Wake Me When the War is Over
# This is a ChordPro song document


Verse 1

Am
I don't need all of this squawking and squabbling,
      Bm                Em
And I haven't got the time,
    C                    Em
You don't really need me here
                      G
So I think I'll go outside
C            G       Am            G
You're gonna do what you're gonna do,
      F
While someone else is yelling at you,
G                         Em      Am
I'd rather be shut of the two of you


Chorus

Bm               F#7
Wake me when the war is over,
G          D   G                 D
You've bit off more than you can chew,
Bm               F#7
Wake me when the war is over,
G            D         A      C      G
I just wanna know when you're quite through.
```

9. **Save document.** Click the Save to a file button in the Song Lyrics/Chords Editor to save your new ChordPro document.



Notice that there are already three parts in the song when it is created: Verse, Chorus, and Bridge. You can set the default parts created with a new song by choosing Setlist/Song Settings from the **Options** menu.

10. **Click on a song part to change it.** Switching to a different song part is as easy as clicking in the Setlist editor or on one of the song part tiles. The tiles can be useful when using the Lemur template or working on a Windows Surface tablet. Most commonly, a MIDI foot controllercalls up song parts by sendingMIDI program change messages set up in Gig Performer, as described below.

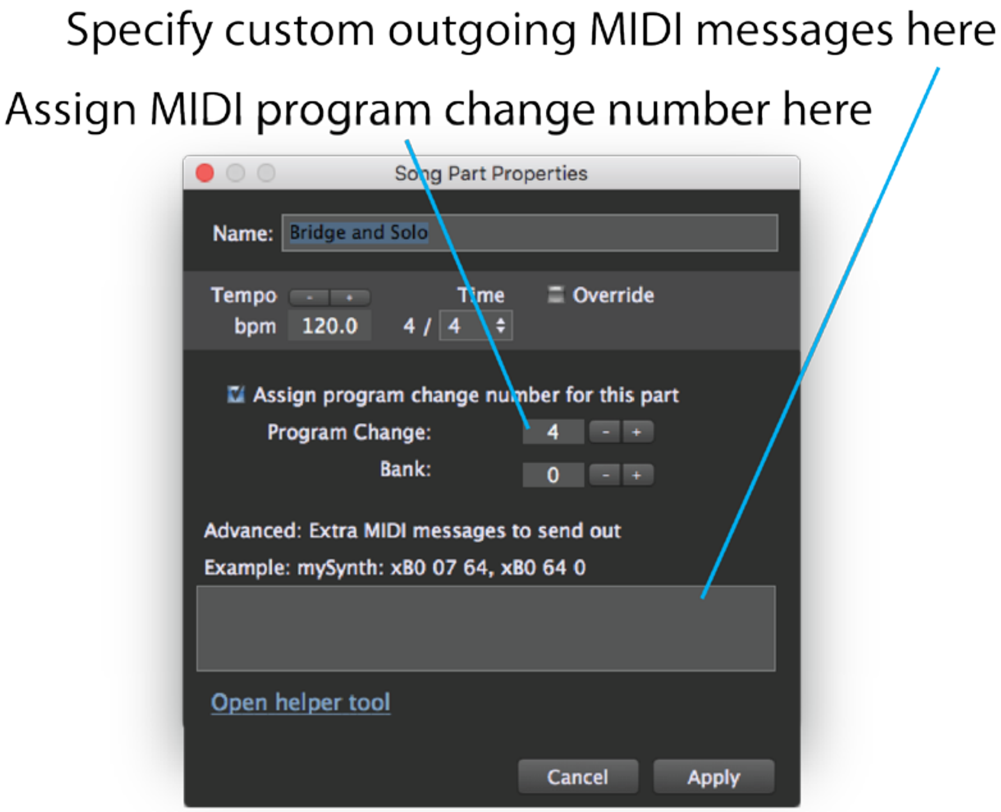


Click here or here to change song parts.

11. **Make a new song part.** Press **Cmd-N** or choose **Add new song part** from the **Setlists menu.** Enter a name in the **Song Part Name dialog** that appears. Click on the variation you want attached to the song part to select it.

Note: Gig Performer attaches the variation that is selected at the time you leave a song part and switch to another, so if you try different variations, be sure the correct variation for the song part is selected when you change to a different one.

12. **Assign a MIDI program change number to the part.** Song parts can be called by a MIDI program change message assigned to it in the dialog. Click the **Assign program change number for this part** box and enter a program change number and, optionally, a bank number, using the +/- keys or simply typing in the value.



Specify custom outgoing MIDI messages here

Assign MIDI program change number here

Song Part Properties

Name: Bridge and Solo

Tempo bpm 120.0

Time 4 / 4

□ Override

☑ Assign program change number for this part

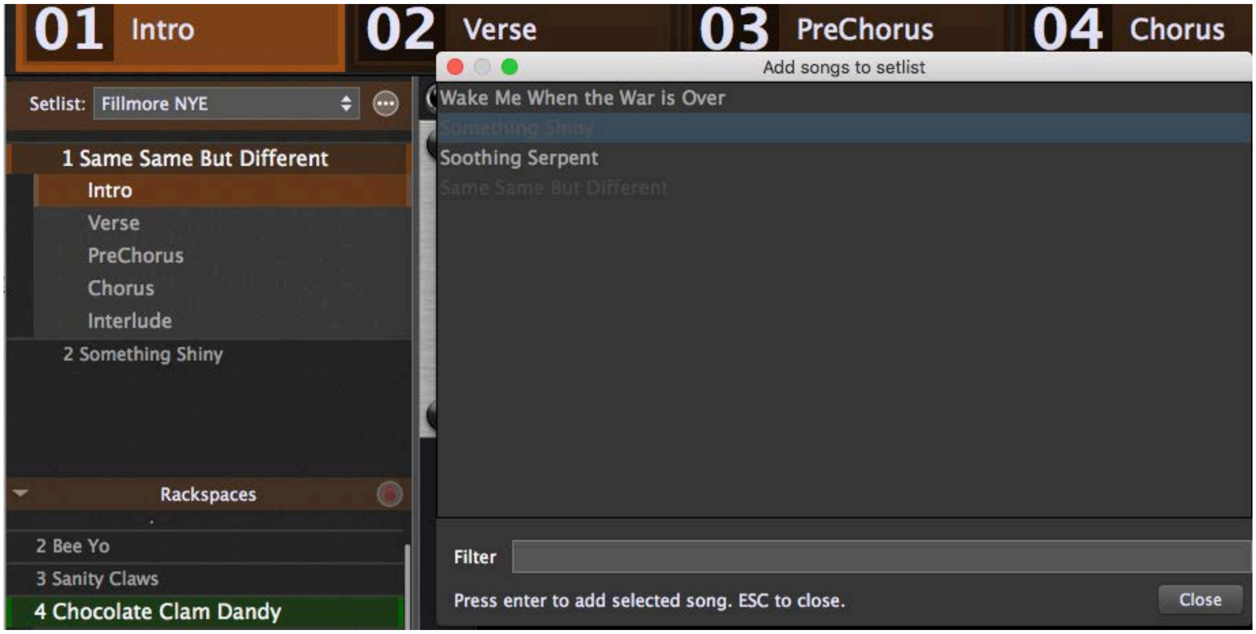
Program Change: 4

Bank: 0

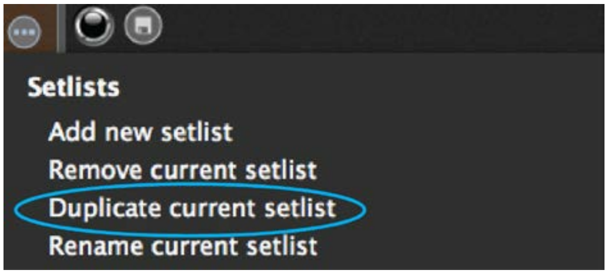
Advanced: Extra MIDI messages to send out

Example: mySynth: xB0 07 64, xB0 64 0

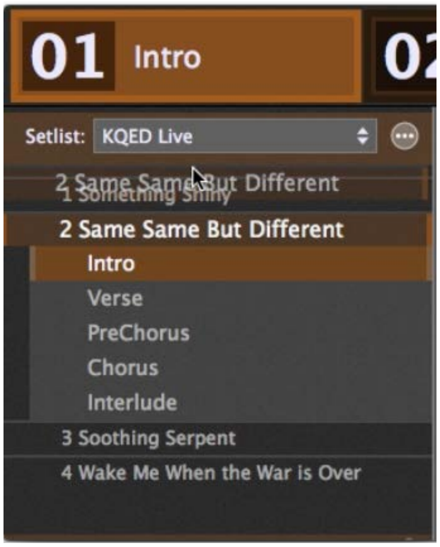
13. **Make more songs and song parts.** Repeat steps 2 and 3 to make some more songs and steps 4 through 6 to create more song parts.
14. **Create a setlist.** Choose the **Add new setlist** command in the **Setlists** contextual menu, or **New Setlist** from the **Edit** menu to create a new setlist. Enter a name in the **Setlist Name** dialog that appears.
15. **Add songs to the setlist.** Open the **Add songs to setlist** window by pressing **Cmd-Shift-F** (**Ctrl-Shift-F** on Win), by choosing the **Add songs to this setlist** command in the **Setlists** contextual menu or the **Add song(s) to setlist** command in the **Edit** menu. To add a song from the **Add songs to setlist** window to the setlist, select it in the **song list** window and press the **Enter** key. When you are finished adding songs, press the **Escape** key to close the song list window.



16. Choose the Duplicate setlist command from the Setlist contextual menu. Tomorrow's setlist will be just a little different than this. The quickest thing is to duplicate this setlist and then modify it. Enter a new name in the dialog that appears.



17. **Modify the setlist.** Add a few more songs. Rearrange the order of the songs by dragging them in the list.



01 Intro 02

Setlist: KQED Live

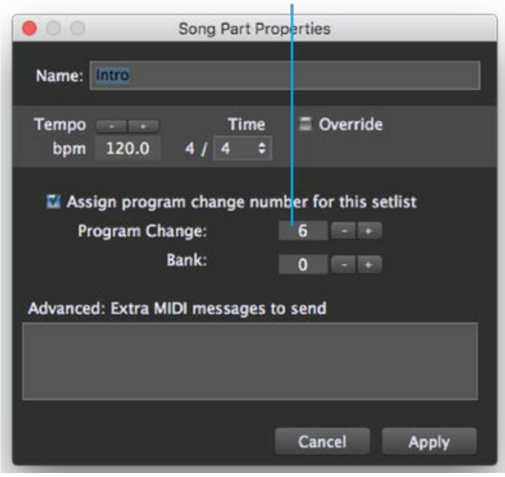
2 Same Same But Different
1 Something Shiny

2 Same Same But Different
- Intro
- Verse
- PreChorus
- Chorus
- Interlude

3 Soothing Serpent

4 Wake Me When the War is Over

18. **Adjust the song part program changes for consistency.** The program change assignment for a song part is specific to the setlist; the same song can have a different program change assignment in each setlist. This lets you have consistency even when the setlist changes from gig to gig. For example, say you set program change 1 to call the first song part of the first song in a setlist. The next gig might open with a different song, but rather than having to remember the correct program number for the first part of that song, you simply assign it to program change 1 in the second night's setlist. If you do that on every setlist, then you can start every show by calling program change 1, knowing it always will call up the first part of the first song no matter what song opens the set.



Reassign the program change for the new setlist.

Song Part Properties

Name: Intro

Tempo bpm 120.0 Time 4 / 4 □ Override

☑ Assign program change number for this setlist

Program Change: 6

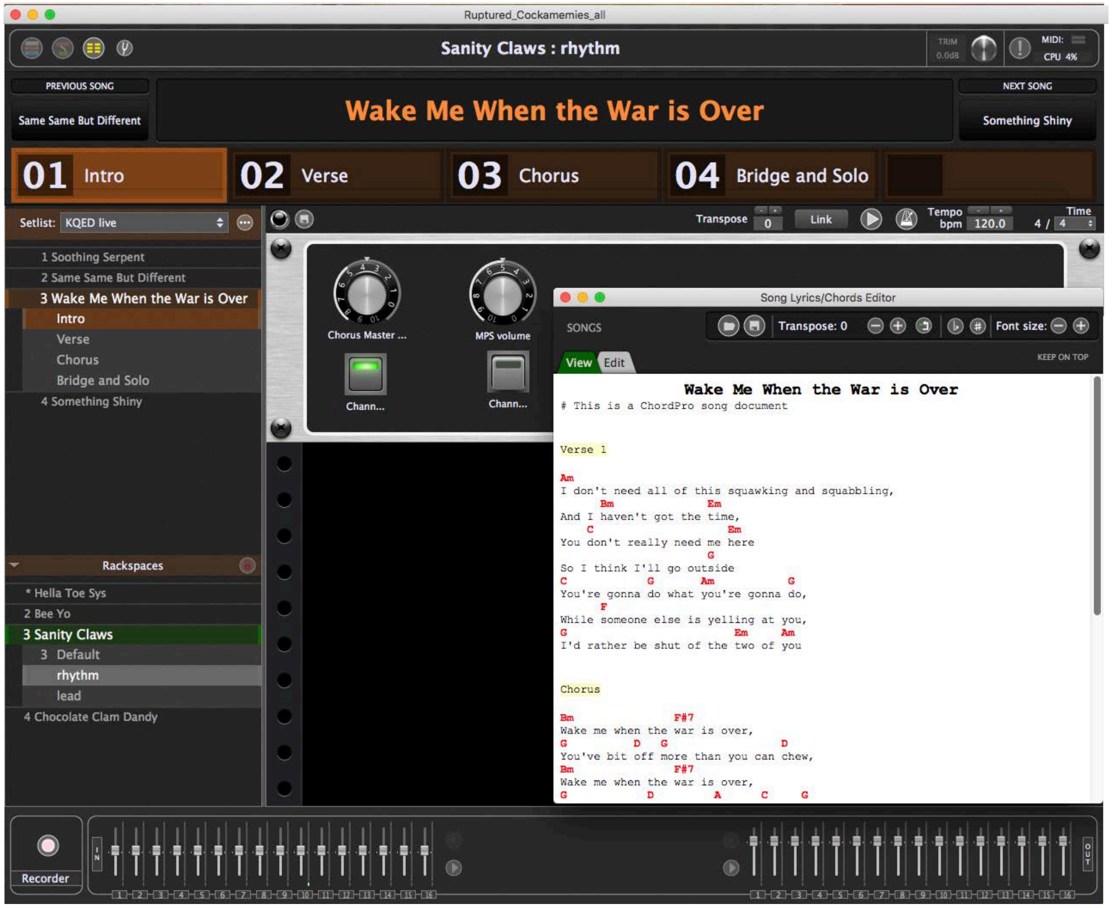
Bank: 0

Advanced: Extra MIDI messages to send

Now let’s use your new setlist.

19. **Press** Cmd-E (Ctrl-E) to open the **Song Lyrics/Chords editor**.

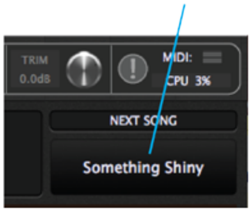
20. **Select a song with a ChordPro file attached.** The chords and lyrics will appear as Gig Performer switches to the first part in the song.



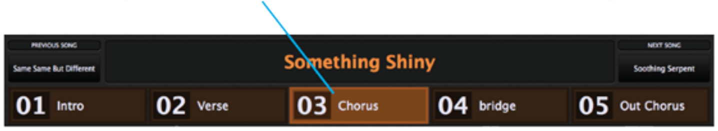
```
            Wake Me When the War is Over
# This is a ChordPro song document


Verse 1

Am
I don't need all of this squawking and squabbling,
      Bm               Em
And I haven't got the time,
    C                    Em
You don't really need me here
                      G
So I think I'll go outside
C            G       Am            G
You're gonna do what you're gonna do,
      F
While someone else is yelling at you,
G                         Em      Am
I'd rather be shut of the two of you


Chorus

Bm               F#7
Wake me when the war is over,
G          D   G                 D
You've bit off more than you can chew,
Bm               F#7
Wake me when the war is over,
G            D         A      C     G
```

21. **Click the Previous Song button to move up one song in the setlist.** Click a song part tile in the song bar to change to a different part.

Click here to switch to the next song in the setlist



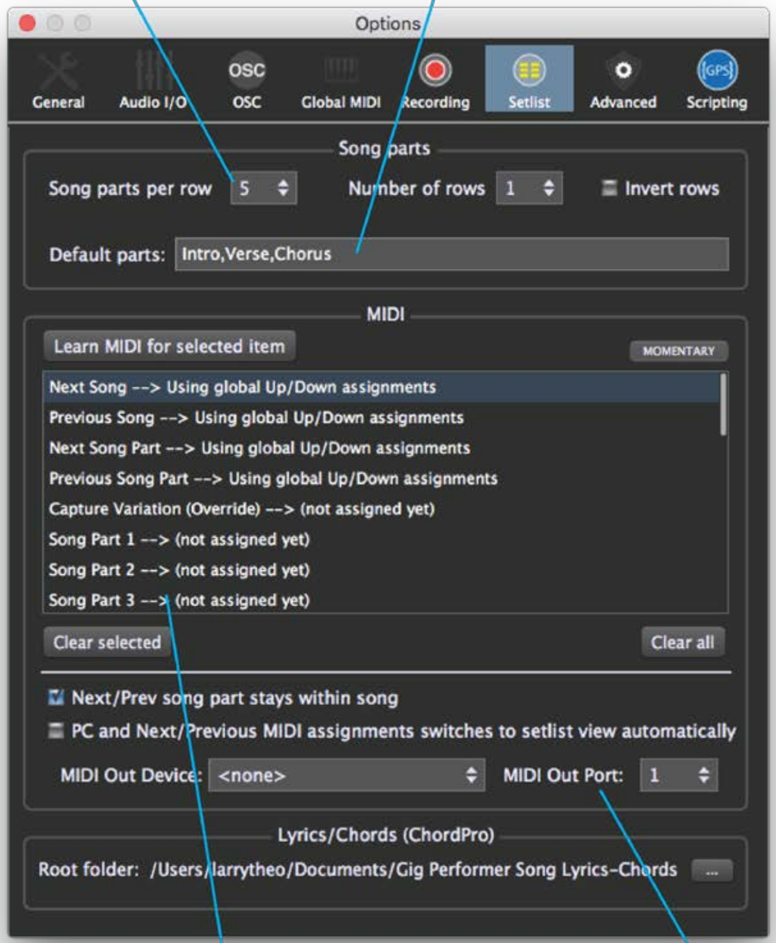
Click any of these tiles to switch to a different song part.



22. **Customize setlist operation in the Options>Setlist/Song Settings dialog.** Set the row configuration to match the layout of your MIDI foot or pad controller.

Configure song part tiles

Set default parts created with a new song



Song parts

Song parts per row 5 Number of rows 1 □ Invert rows

Default parts: Intro,Verse,Chorus

MIDI

Next Song --> Using global Up/Down assignments
Previous Song --> Using global Up/Down assignments
Next Song Part --> Using global Up/Down assignments
Previous Song Part --> Using global Up/Down assignments
Capture Variation (Override) --> (not assigned yet)
Song Part 1 --> (not assigned yet)
Song Part 2 --> (not assigned yet)
Song Part 3 --> (not assigned yet)

☑ Next/Prev song part stays within song
□ PC and Next/Previous MIDI assignments switches to setlist view automatically

MIDI Out Device: <none> MIDI Out Port: 1

Lyrics/Chords (ChordPro)

Root folder: /Users/larrytheo/Documents/Gig Performer Song Lyrics-Chords ...

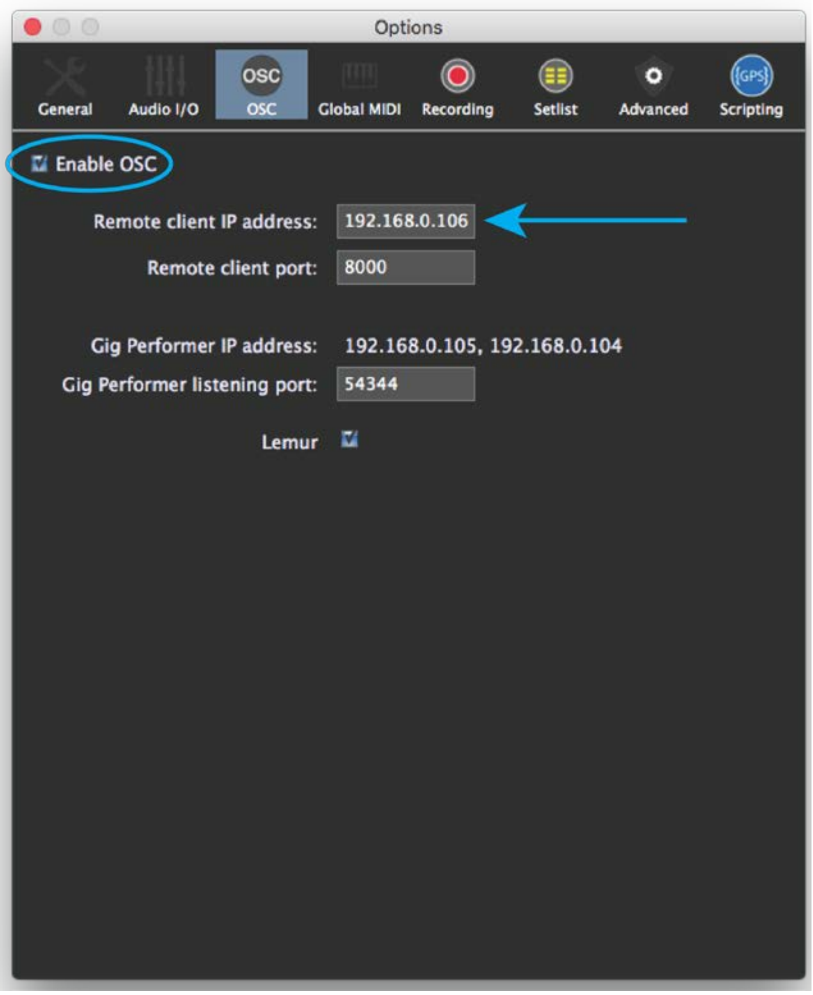
Assign MIDI controls to call songs and song parts

Set device and port for outgoing MIDI messages

Controlling Gig Performer Using OSC

As described in Appendix A of this user guide, OSC (Open Sound Control) is a communications protocol that allows you to view and move Gig Performer's widgets and access songs and song parts wirelessly from your iOS or Android tablet or smartphone. This tutorialwill show you how to use Gig Performer with an OSC app.

1. **Establish wireless communications.** For Gig Performer and an OSC app to work with one another, both your computer and the device running the app must be connected to the same wireless network, so ensure that they are communicating correctly.
2. **Establish OSC communications.** Launch Gig Performer and then go to **Options> OSC setup...**The dialog window shown below will appear.



☑ Enable OSC

Remote client IP address: 192.168.0.106

Remote client port: 8000

Gig Performer IP address: 192.168.0.105, 192.168.0.104

Gig Performer listening port: 54344

Lemur ☑

Click on the **Enable OSC** button (circled in the image) so that it is checked in blue. The IP address of the device running the OSC app is displayed in the **Remote client IP address** box (indicated by the arrow).

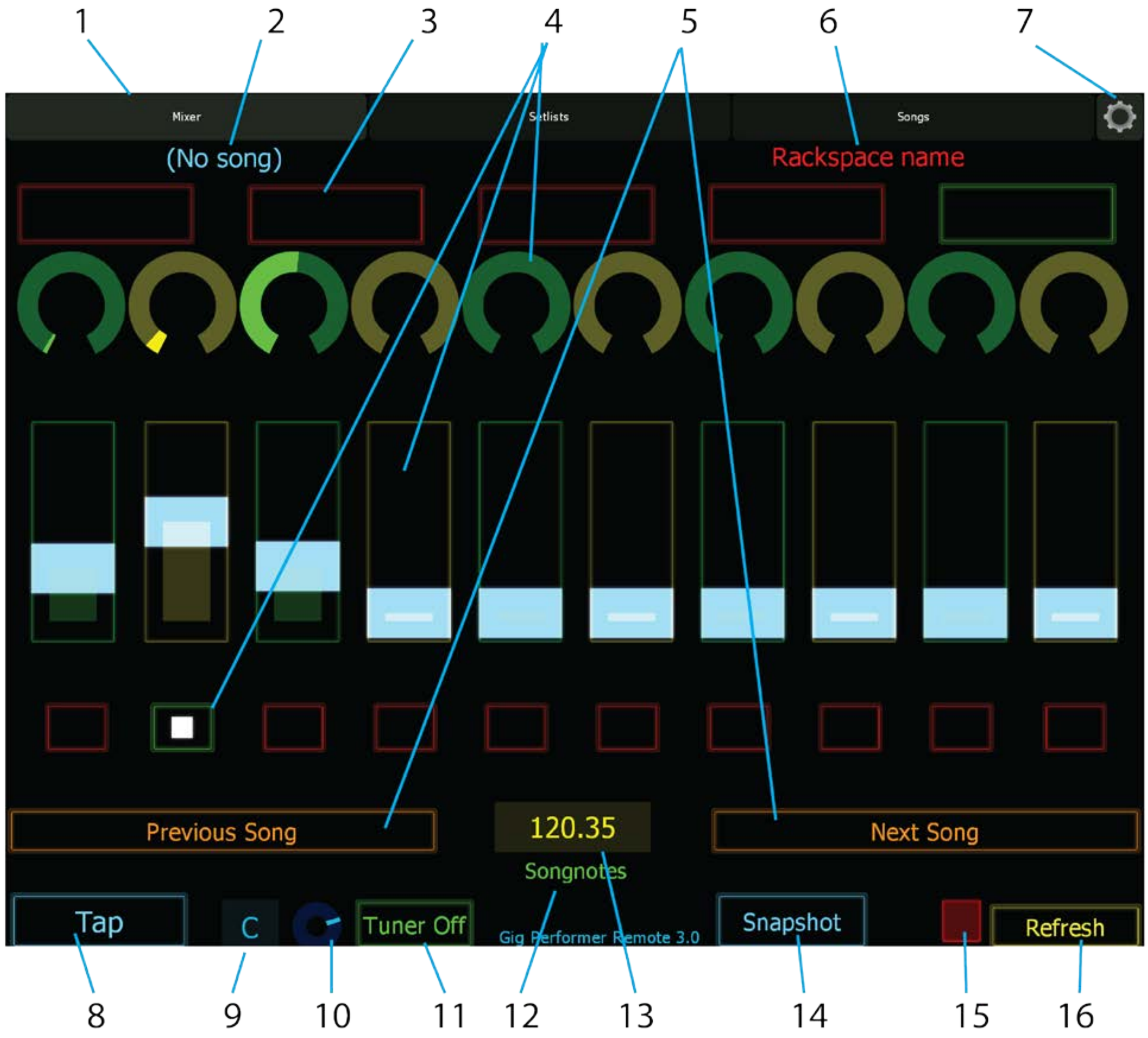
The port number of the OSC app is displayed in the **Remote client port** box. Note that the "Gig Performer IP address" value cannot be changed, as this is permanently set. If you're using the OSC app Lemur or another OSC application that allows detailed control, check the "Lemur" box.If you're using TouchOSC or any other app with a more basic OSC implementation, leave it unchecked.

Note: If you're running an OSC app on multiple devices—say a tablet and a smartphone—and you want to use both devices onstage (or if you want to give one to your FoH engineer to have access to your controls), there's a simple way to accomplish that, using something called Broadcast Mode. Assuming you're using standard class C addressing (with IP numbers beginning with 192.168.1.x), simply replace the last value of the OSC app's IP address with 255 (i.e., 192.168.1.255) and restart Gig Performer. Now all devices whose IP addresses are on the same LAN (i.e., those devices with IP addresses all starting with 192.168.x) and are listening on the OSC app's port number will communicate simultaneously with Gig Performer. Click here to access our Knowledge Base article with more information.

1 **When done configuring options, close the dialog and quit Gig Performer.**
2 **If you're using Lemur or TouchOSC, download a Gig Performer template.** For your convenience, we have created templates that make it easy to control Gig Performer widgets from the popular OSC apps Lemur and TouchOSC. If you are using either of these apps, click here to download the Lemur file, or click here to download the TouchOSC file. Note that these templates merely provide a starting point for your own OSC programming and you certainly are not required to use them—in fact, we encourage you to construct your own custom OSC environments.

The Lemur and TouchOSC Templates

The Mixer page of the Lemur template (as viewed on a tablet or smartphone) looks like this:



1. **Template page tabs –** Touch these tabs to switch between the Lemur template's Mixer, Setlists, and Songs pages.
2. **Current song name-** When a song is loaded, the name is displayed here.
3. **Song part tiles –** When a song is loaded, these tiles show the names of song parts. Clicking on a tile loads the indicated part.

4. **Lemur controls-** These rows of knob, fader, and button controls get mapped to widgets in your rackspaces.

5. **Previous/Next Song buttons-** Touch one of these to move to the next or previous song in the current setlist.

6. **Current rackspace name-** Each song part is associated with a rackspace variation. The name of the current rackspace is displayed here.

7. **Settings menu-** Touch here to drop down a menu that allows you to load and save projects, plus access Lemur settings.

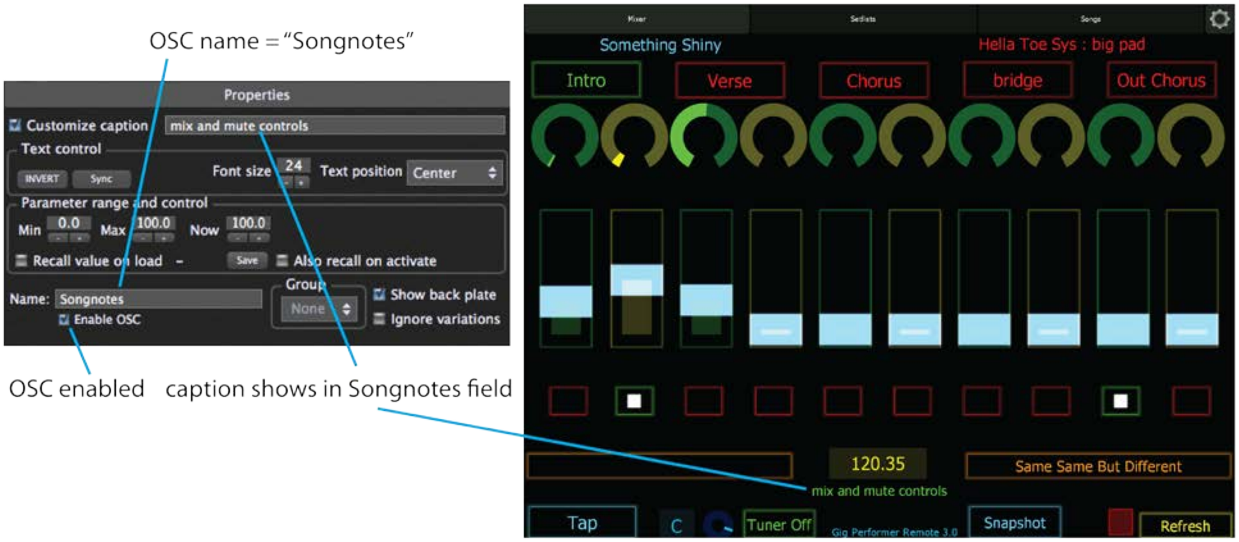
8. **Tap Tempo button -**The Tap Tempo button is always active and has the highest priority, so it can be tapped at any time to set a tempo. This can be useful for syncing delays to the tempo of a live performance.

9. **Tuner current note display –** When the tuner is on, the note currently detected from the input is displayed here.

10. **Tuner current intonation display –** When the tuner is on, this display shows how sharp or flat the note currently detected from the input actually is from the note displayed.

11. **Tuner on/off button –** Touch this button to bring up Gig Performer's Tuner view for tuning. Touch it again to hide the tuner.

12. **Songnotes display –**A small display for a reminder or any short message you would like to have shown on the Lemur screen. Create a label on a rack panel in Gig Performer, check the **Customize caption box** in the Widget Properties area. and enter as the caption the text you want displayed in Lemur. In the **OSC Name field**, enter "Songnotes", and then check the **Enable OSC box.** The caption for that label will now appear in Lemur in place of the Songnotes label.



OSC name = "Songnotes"

Properties

☑ Customize caption: mix and mute controls

Text control: INVERT, Sync; Font size 24; Text position Center

Parameter range and control: Min 0.0, Max 100.0, Now 100.0

▤ Recall value on load – Save ▤ Also recall on activate

Name: Songnotes

☑ Enable OSC

Group: None

☑ Show back plate

▤ Ignore variations

OSC enabled

caption shows in Songnotes field

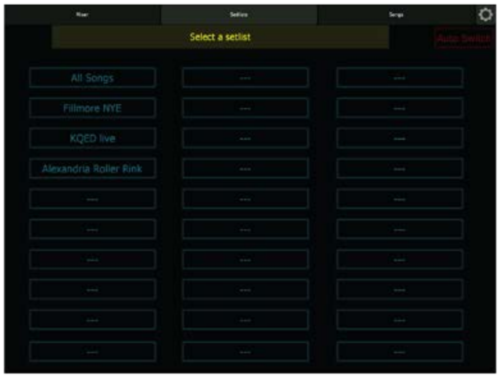
13. **Tempo field –** Displays Gig Performer's current tempo.

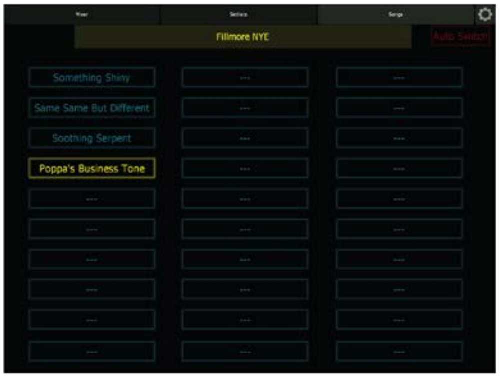
14. **Snapshot button –** This is the same as the **Capture variation to part buttonjust** above the rack in the Setlist/Song view. It captures any changes you may have made to widget settings since recalling the rackspace variation associated with the song part, and stores the changes with the part, so they are recalled each time that part is.

15. **Snapshot indicator –** When a snapshot is taken, this indicator turns red to show that the variation has been edited. Touching the red button recalls the variation as stored. Changes to widget values are lost.

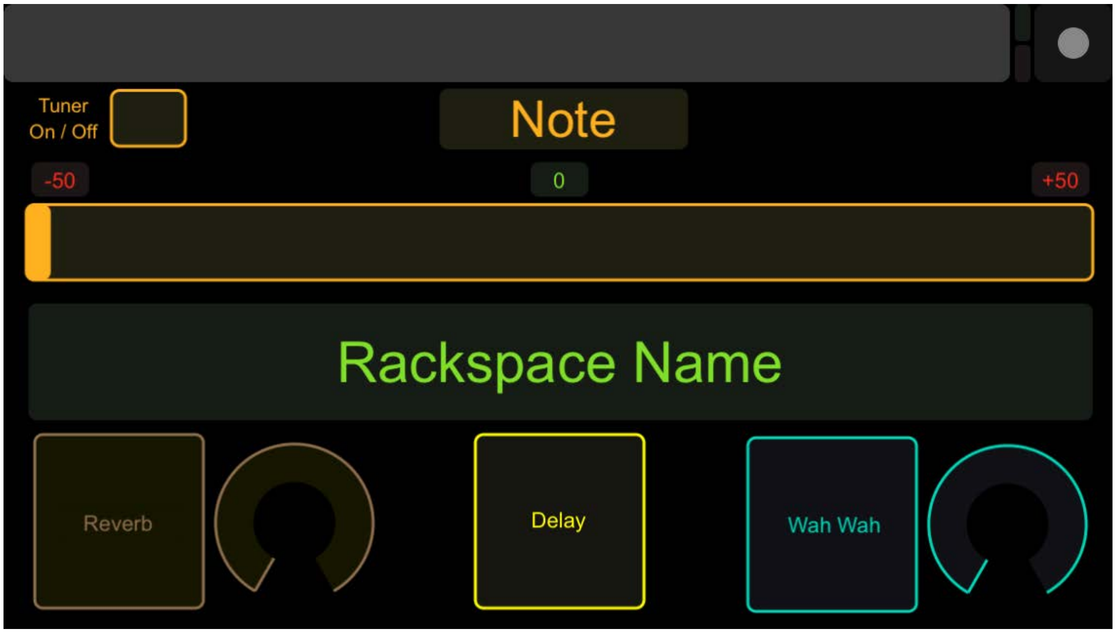
16. **Refresh button –** Touch this button after making changes to a Gig Performer rackspace to update the Lemur display with the changes.

The Setlists and Song pages in the Lemur template offer quick touch access to, of course, your setlists and songs.





The TouchOSC template, as viewed in an iPhone (the only device it can run on), looks like this:



Note that our Lemur template has been designed for general use, while the TouchOSC template has been designed specifically for guitarists. (An instructional video about using the Lemur template can be found here, and an instructional video about using the TouchOSC template can be found here.) Both templates display the currently selected rackspace and variation prominently; the Lemur template also displays the caption of a text label in the "Songnotes" field.

If you'd like to experiment with either template, relaunch Gig Performer (remember, you quit it in step 3 above) and then touch the Tuner On/Off button in the template. Gig Performer should immediately go toTuner View. (If for some reason the Tuner On/Off button doesn't work, double-check that the directions in steps #1 and #2 above have been followed precisely to ensure correct bidirectional communications between the device running your OSC app and the computer running Gig Performer.)

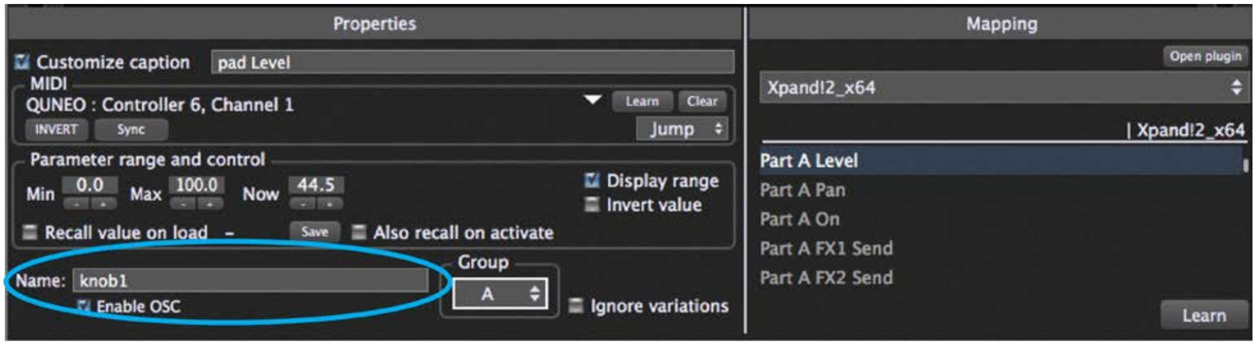
In Lemur, if you have a microphone or instrument connected to one or more inputs of your audio interface, the virtual dial to the left of the Tuner On/Off button will move and the note number to the left of that knob will change to reflect the note coming in on the channel(s) Gig Performer is currently "listening" to. In TouchOSC, the box at top center will display the note coming in, and the slider below it will show the tuning offset in cents, with the center position indicating that you are exactly in tune.

Touch the Tuner On/Off button again to return Gig Performer to Performer View. If you are using Lemur, touch the "Tap" button presented by the template several times. You should see theTempo display in the upper-right hand corner of Gig Performer's screen change bpm value as you do so; the shorter the intervals between taps, the higher (faster) the bpm value.

3 **Give your widgets the appropriate OSC names.** One of the requirements of OSC is that each object in both the host computer and remote device must have matching unique case-sensitive names. You can make up any names you like, as long as they are comprised of ASCII characters only, with no spaces. If you're running Lemur or TouchOSC, here are the OSC names used in the templates we've created:

Object	Lemur template	TouchOSC template
Rotary knobs (Reverb andWahWah in Touch OSC template)	"Knob1" through "Knob 10"	"knob1" and "knob2"
Sliders / drawbars	"Slider1" through "Slider10"	(none)
Buttons (Reverb, Delay, WahWah in Touch OSC template)	"Button1" through "Button 10"	"button1", "button2" and "button3" respectively
Text label caption	"Songnotes"	(none)

To use the templates, simply put Gig Performer into Edit mode, select a widget, and in the Widget Properties inspector, check the Enable OSC box, enter one of the OSC names in the "Name" field (circled in blue), **then press Enter (this is important)**.



☑ Customize caption: pad Level

MIDI
QUNEO : Controller 6, Channel 1
Jump

Parameter range and control
Min 0.0 Max 100.0 Now 44.5
☑ Display range
□ Invert value
□ Recall value on load –
□ Also recall on activate

Name: knob1
☑ Enable OSC
Group: A
□ Ignore variations

Xpand!2_x64
| Xpand!2_x64
Part A Level
Part A Pan
Part A On
Part A FX1 Send
Part A FX2 Send

For example, if you're using the Lemur template, add a knob widget and name it "Knob1"; if you're using the TouchOSC template, name the knob widget "knob1".

Then take Gig Performer out of edit mode and move the knob widget with your mouse. As you do so, the identically named knob in the Lemur or TouchOSC screen changes the same way. Because OSC is bi-directional (meaning that changes can be made in either the device running the OSC app or the host computer running the target software—in this case, Gig Performer), the opposite occurs too: You can move the onscreen knob in Lemur or TouchOSC and the knob widget in Gig Performer changes the same way.

Do the same with a switch or LED button widget, changing its OSC name to Button1 (for use with the Lemur template) or button1 (for use with the TouchOSC template). Exit Edit mode and test to make sure that touching the corresponding button in your tablet or smartphoneturns the associated Gig Performer widget on or off, and vice versa.

7 **Make your existing rackspaces OSC-compatible.**It's easy to make any rackspace OSC-compatible: All you have to do is assign OSC names to your widgets that match the names being used in the environments you create in your OSC app editor. For example, load any of the rackspaces created in the Controlling Plugins From Your Midi Controller and Configuring Audio Plugins tutorials presented earlier in this User Guide, go into edit mode, and select each widget in turn. The only change you have to make to each is in the "Name" field in the Properties area of the inspector and checking the Enable OSC box below it; all MIDI device and plugin parameters remain the same. The only difference is that you can now move these widgets remotely from the tablet or smartphone running your OSC app, and vice versa (i.e., any widget adjustments you make from your laptop or from a connected MIDI controller are reflected in the tablet/smartphone screen).

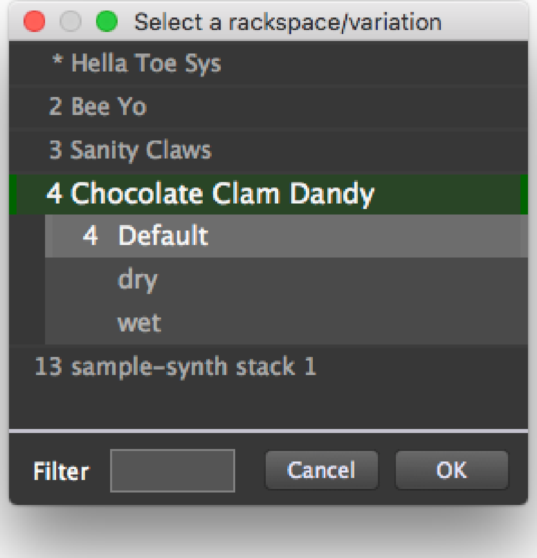
Please note that this tutorial only scratches the surface of what can be accomplished when using an OSC app with Gig Performer. Again, the Lemur and TouchOSCtemplates provided on our website make good jumping-off points towards constructing your own custom OSC environments. Consult the documentation of your OSC app and Appendix A in this User Guide for more information.

REFERENCE

Quick Rackspace/Variation Finder

The Quick Rackspace/Variation Finder is a fast way to access a specific rackspace and variation, great for jamming and improv performances, or assembling setlists. It is available everywhere in the program:

- Using key shortcut Cmd-J (Ctrl-J on Win)
- In the **Edit menu**
- In the **More ("...")** menu

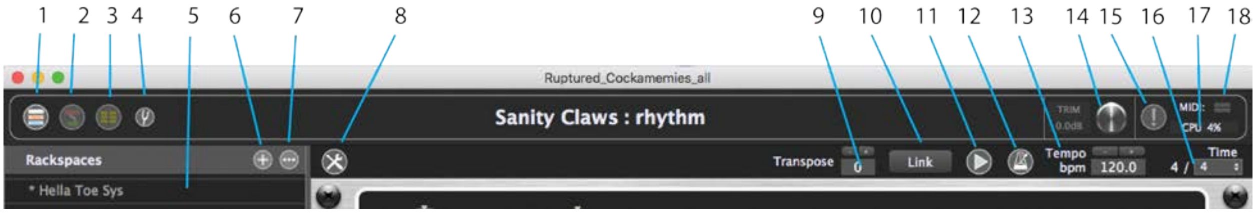


- To switch to a different rackspace, open the Quick Rackspace/Variation Finder, click the variation you want to select it, and click the OK button. Or simply double-click the variation in the list.
- Search terms can be entered in the **Filter** field to help find the variation you want.

Toolbars

The Connections and Performer views share top and bottom tool bars with key global functions. The top tool bar is a bit different in Setlist/Songs View.

Top Toolbar



1. **Performer view–**Click here to show Performer View.

2. **Connections view–**Click here to show Connections View.

3. **Setlist/Song view-** Click here to show Setlist/Songs View.

4. **Tuner icon –** Click here to toggle Tuner View (the icon turns red).

5. **Browser-**In Performer and Connections views,the browser shows a list of all rackspaces and variations that have been created. The current rackspace is outlined in green. Click on a rackspace in the list to select it. Double-click on a rackspace in the list to open the Rackspace Properties dialog. Drag rackspaces up and down in the list to reorder them. The browser looks different in Setlist/Song view, though the rackspace/variations list is still displayed. For more detail, see Setlist/Songs View.

6. **Add rackspace ("+") button –** Click here to add a new rackspace and open the Rackspace Properties dialog.

7. **Contextual menu –** Click on the button with the ellipsis (three dots) to drop down a menu of operations appropriate to the browser display: (...) "More" Menu (Manage Rackspaces/Variations) in Performer and Connections views, and "More..." Menu (Manage Setlists) in Setlist/Song view.

8. **Action button –** In Performer view, click this button to enter Edit Mode. In Connections view, click this button to add a plugin to the current rackspace. In Setlist/Song view, click this button to capture the current variation into the currently selected Song part.

9. **Transpose –** Enter a global transposition value that will be applied to the outputs of all MIDI In blocks by clicking on the field and entering a number of semitones directly, or using the increment (+) and decrement (-) buttons to set the number of semitones.

10. **Link button-** Click this button to activate or deactivate Ableton Link support.

11. **Play/Stop button–** Click to start playback, click again to stop it.When playing, all plugins that provide playback control (for example, sample, loop, or beat playback plugins) play back. When disengaged ("stop"), they are all put in a stopped state.

12. **Audio metronome on/off –** Click to turn on the audio metronome, click again to turn it off. See Audio metronome for more information.

13. **Tempo display–**Shows the current tempo being used by all connected plugins (for example, sample/loop/beat-playback plugins, as well as those generating echoes, delays, and reverbs). You can override this by entering a new value, or by using the increment/decrement buttons below the display. The tempo can also be affected by song and song part tempos. For more information see Tempo and Time Signature.

14. **Global output volume (Trim) knob –** Use this to temporarily change the overall volume of your Gig. An increase in volume is highlighted in red and a decrease is highlighted in blue. Double-click on the knob to reset its value to 0dB. The Trim knob can be remotely controlled; map it to a MIDI controller in the Global MIDI Assignments section of the Global MIDI tab of the Options dialog.

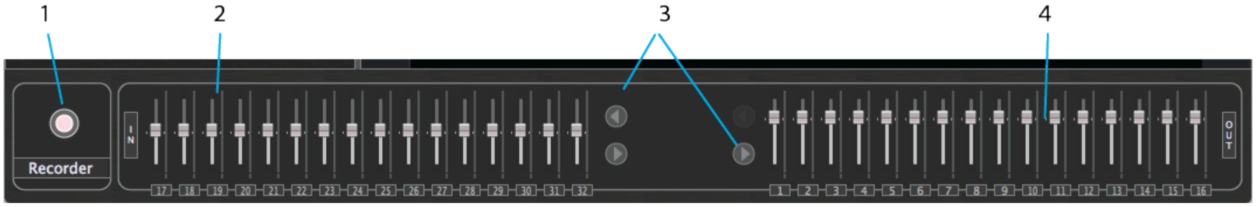
15. **MIDI Panic button –**Click this to issue "turn all notes off" messages on all MIDI channels, along with a Sustain OFF message and a Center PitchBend message, thus stopping stuck notes in their tracks. Double-click this button to reset the audio engine.

16. **Time signature display–**Shows the current time signature being used by all connected plugins (for example, sample/loop/beat-playback plugins, as well as those generating echoes, delays, and reverbs). You can reset this by clicking on the dropdown menu, which enables you to enter any time signature, up to 99/64.

17. **CPU usage indicator –**Shows the percentage amount of CPU overhead currently being used by Gig Performer for audio processing. (Actually, Gig Performer itself is only responsible for a very small percentage; most of the CPU overhead is taken up by the plugins currently loaded in memory.) It is normal for this value to change in real time as incoming MIDI data is received, and as widgets are moved.

18. **MIDI activity indicator –**Flickering here indicates the presence of MIDI data. The top bar lights green when incoming MIDI messages are received (via a MIDI In block), while the bottom bar flickers when MIDI messages are being transmitted (via a MIDI Out block). If the "Sync with external clock" option is selected for the current rackspace, a thin yellow bar above the two flickers to indicate the presence of incoming MIDI Clock data and also serves as a visual tempo indicator. Open the MIDI Monitor window if you want to see detailed information about the MIDI messages being received.

Bottom Toolbar



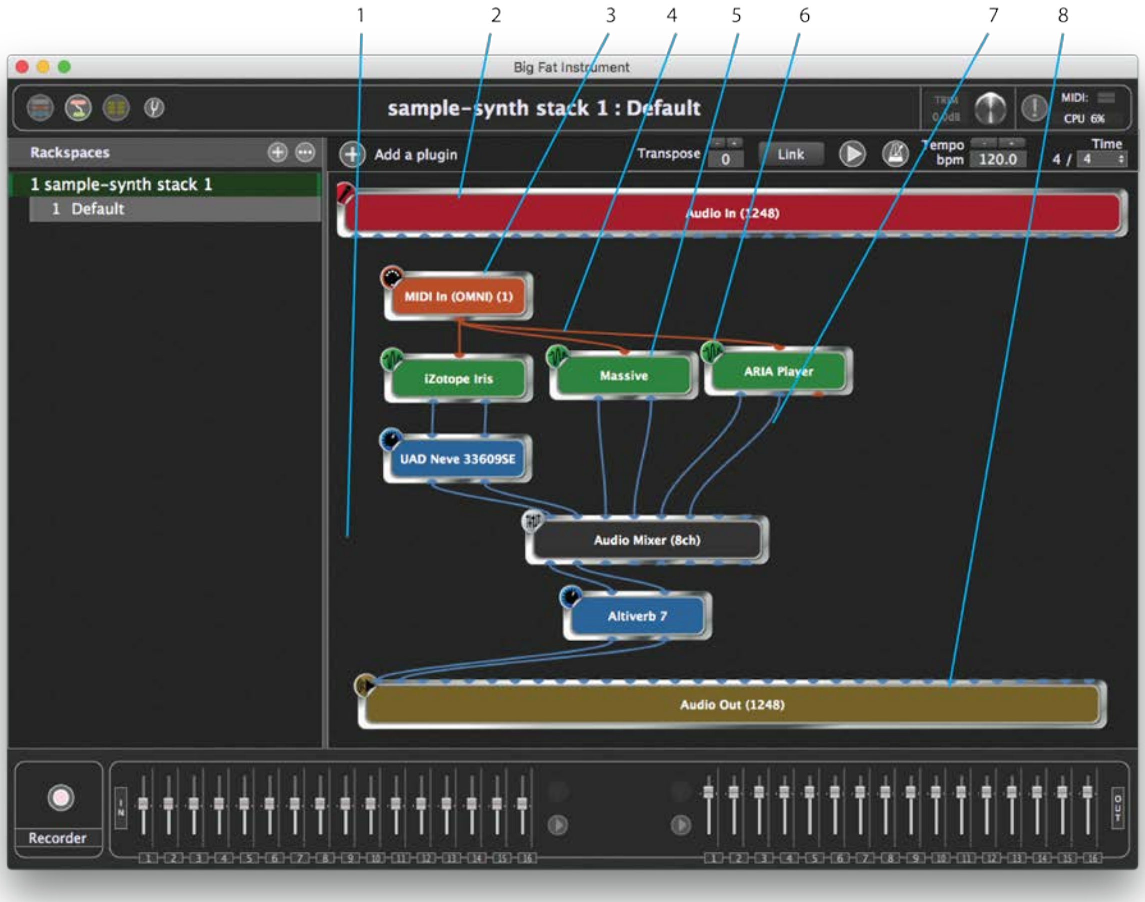
1. **Recorder button –** Click this to open the Recording Options dialog.

2. **Interface input levels –**A set of meters and sliders that allow you to view and adjust the levels of your audio interface's various inputs. Flickering here indicates the presence of input signal. Double-click a fader to set it to 0 dB.

3. **Next/Previous channel group buttons –**Gig Performer displays up to 16 input or output interface channels at a time. If your interface has more than 16 I/O connections, the remaining inputs and outputs are accessed by clicking the Next or Previous button to scroll the display.

4. **Interface output levels –** A set of meters and sliders that allow you to view and adjust the levels of your audio interface's various outputs. Double-click a fader to set it to 0 dB.

Connections View



1. **Plugin connections area –**This large area of the screen is used to display your plugins and their interconnections. Drag a plug into move or reposition it. Double-clicking a plugin opens its plugin editor.

2. **Audio interface input module –** Gig Performer automatically shows inputs for audio interface device(s) known to the operating system.

3. **A MIDI In device –** Gig Performer inserts a MIDI In (OMNI) plugin by default. Other MIDI devices can be added in the same fashion as any other plugin. MIDI plugins are colored orange.

4. **MIDI connections –** Shown as orange lines between from MIDI Out ports to MIDI In ports.

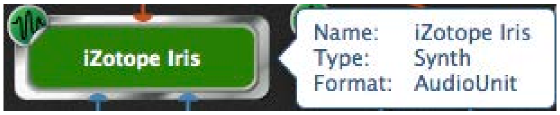
5. **Audio plugin –**Virtual instruments are colored green, audio processors are colored blue, Gig Performer included plugins are colored black.

6. **Plugin edit button –** Clicking this button opens the editor for the plugin. A plugin editor also can be opened by double-clicking the plug-in or right-clicking and choosing **Edit plugin** from the menu that pops up.

7. **Audio connections –** Shown as blue lines from audio outputs to audio inputs.

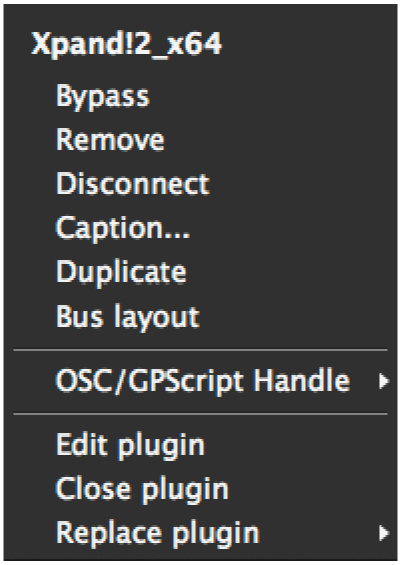
8. **Audio interface output module –** Gig Performer automatically shows outputs for audio interface device(s) known to the operating system.

Hovering the cursor over a plugin brings up a tool tip with basic information:



Name: iZotope Iris
Type: Synth
Format: AudioUnit

The tooltip can be disabled by unchecking the Show plugin information when mouse is over the plugin block box on the General preferences tab. Right-clicking on any plugin opens a popup menu that allows you to do the following:



Bypass- Places the plugin in bypass mode. Click here for more information.

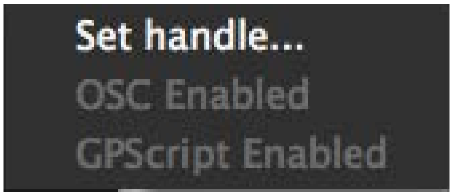
Remove– Deletes the plugin. This is the only way to remove a plugin.

Disconnect- Removes all connections to and from this plugin.

Caption –Changes the displayed name (useful if you are using the same plugin more than once).
Duplicate – Creates a duplicate of the plugin.

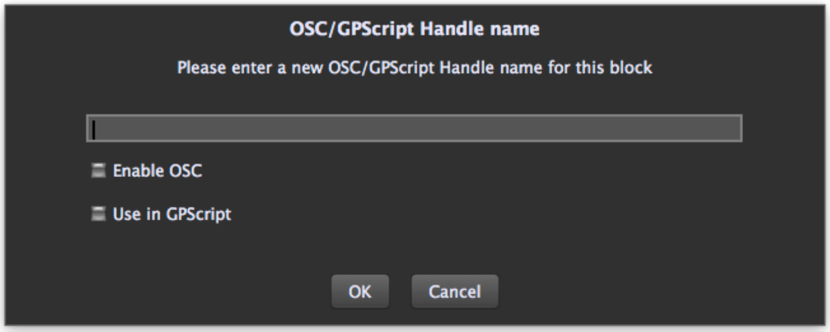
Bus Layout – Opens the Channel and Bus Layout dialog.

OSC/GPScript Handle– Allows you to set ahandle to be used by GPScript, OSC, and/or the Advanced Extra MIDI messages to send out feature, as well as to enable OSC and/or GPScript.



Set handle...
OSC Enabled
GPScript Enabled

Set handle – Enter a name in this field that can be used to identify this plugin in OSC or GPScript code, or in the Advanced: Extra MIDI messages to send feature in the Song Part Properties dialog. Only alphanumeric characters and underbars are allowed.



OSC/GPScript Handle name

Please enter a new OSC/GPScript Handle name for this block

□ Enable OSC

□ Use in GPScript

Enable OSC –When a handle name has been entered, this checkbox enables OSC functionality for this plugin. This box does the same thing as the **OSC Enabled** command, and checking this box causes the command to appear checked, as well.

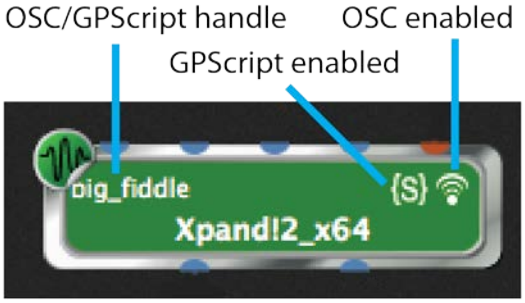
Use in GP Script -When a handle name has been entered, this checkbox enables GPScript functionality for this plugin. This box does the same thing as the **GPScript Enabled** command, and checking this box causes the command to appear checked, as well.

OSC Enabled - When an OSC/GPScript handle name has been assigned, choosing this command enables OSC functionality for this plugin. This box does the same thing as the

Enable OSC checkbox in the **Set handle dialog**, and choosing this command causes that box to appear checked, as well.

GPScript Enabled - When an OSC/GPScript handle name has been assigned, choosing this command enables GPScript functionality for this plugin. This box does the same thing as the **Use in GPScript** checkbox in the **Set handle dialog**, and choosing this command causes that box to appear checked, as well.

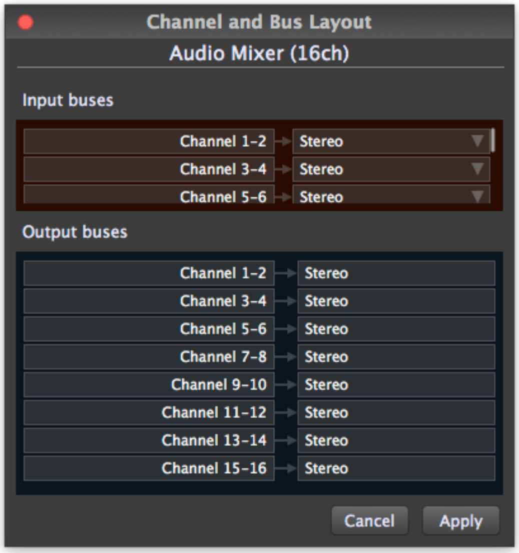
Assigning a handle, enabling OSC for a plugin, and enabling GP for a plugin are all indicated by symbols on the plugin in the Connections view.



Edit plugin–Opens the plugin's editor.

Replace plugin– Choose a plugin from the menu that pops up to replace the current plugin with another.

Channel and Bus Layout dialog



Channel and Bus Layout

Audio Mixer (16ch)

Input buses

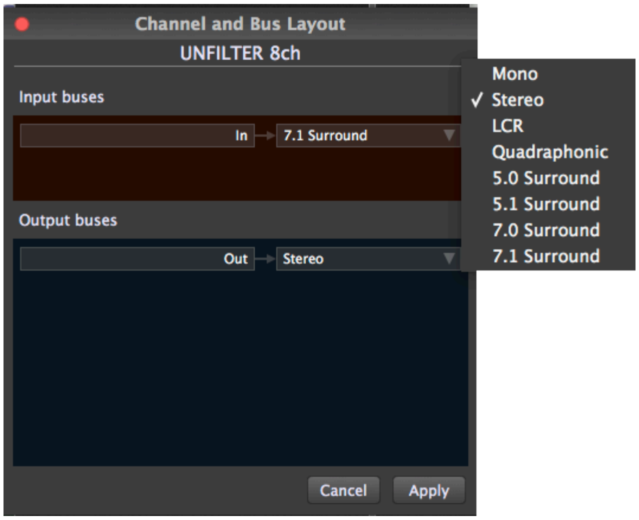
Channel 1-2	Stereo
Channel 3-4	Stereo
Channel 5-6	Stereo

Output buses

Channel 1-2	Stereo
Channel 3-4	Stereo
Channel 5-6	Stereo
Channel 7-8	Stereo
Channel 9-10	Stereo
Channel 11-12	Stereo
Channel 13-14	Stereo
Channel 15-16	Stereo

Each plugin supports some configurations of inputs and outputs, usually stereo, sometimes mono, sometimes even multichannel. In a DAW environment, the DAW deals with configuring these input and output buses. In Gig Performer, you can access a plugin's bus options directly in the Channel and Bus Layout dialog.

Many plugins will simply show a stereo output bus, but some plugins, such as the eight-channel Zynaptiq Unfilter shown here, offer many more configurations.



Channel and Bus Layout

UNFILTER 8ch

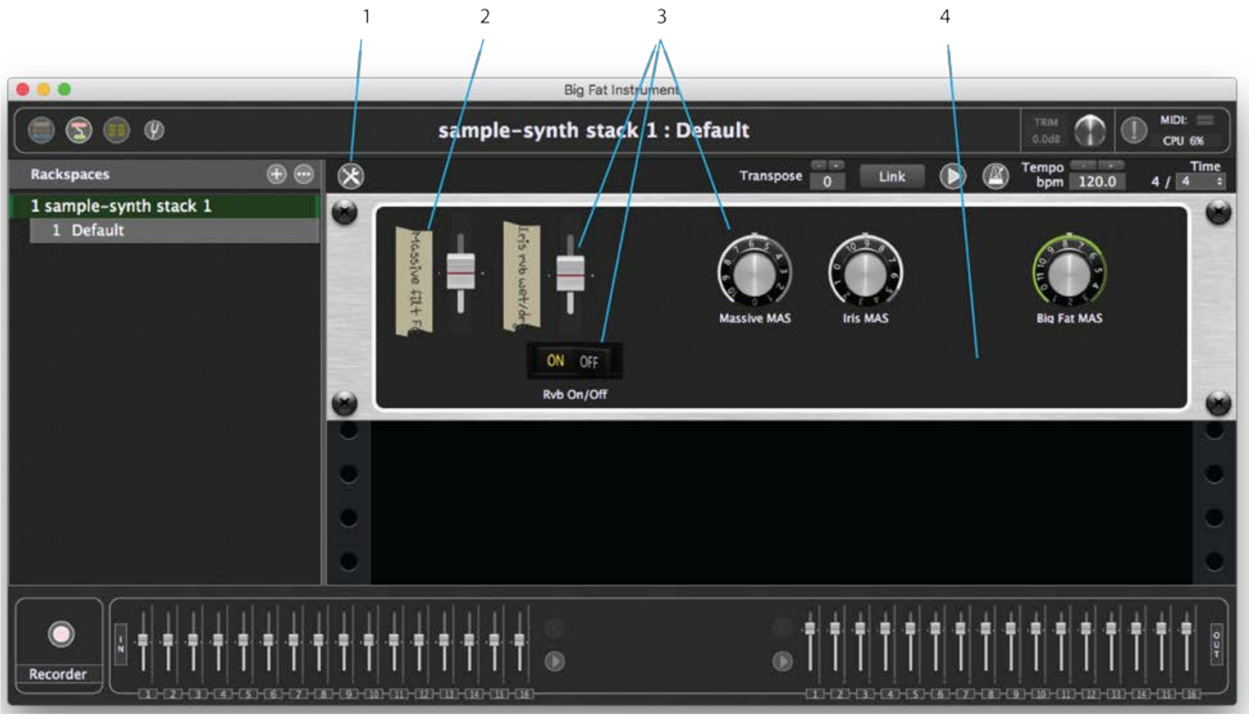
Input buses

In → 7.1 Surround

Output buses

Out → Stereo

Performer View



1. **Layout Designer (Edit Mode) button –** click this to enter Edit Mode and configure widgets on your panels.

2. **Label widget –** Label widgets have no function; they are for display only.

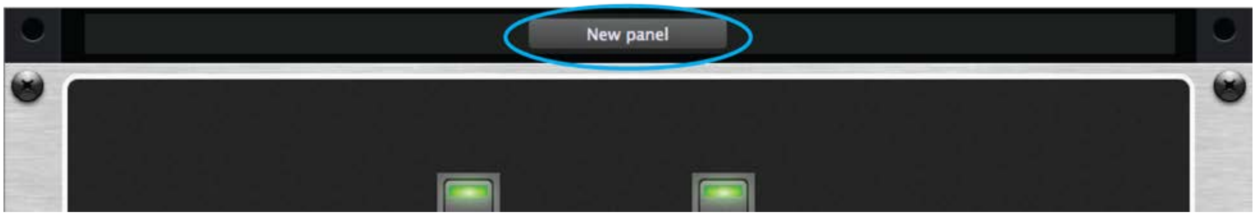
3. **Control widgets –** Knobs, switches, faders, and other controls attach to plugin parameters to give you fast and easy control over the parameters you adjust in performance.

4. **Rackspace –** One or more panels containing the widgets in the current rackspace and variation. (You may need to scroll down if there are many panels being used.)

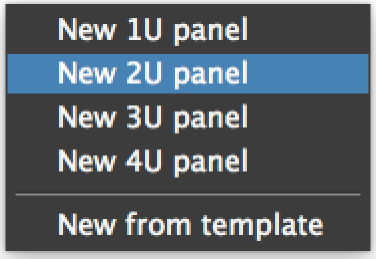
Add Panel

To add a new panel to the rack:

1 Click the **Edit Mode button**.
2 Click the **New Panel button** above the first panel in the rack.

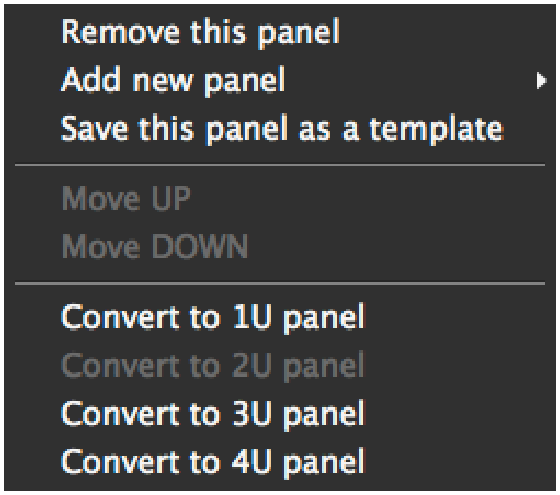


3 Choose a panel height from the popup menu.



Right-Click Menu Panel Functions

Right-clicking on a rack panel pops up a menu with several important functions for managing panels.



Remove a Panel

To delete a panel from the rack:

1. Right-click the panel you want to remove and choose **Remove this panel** from the popup menu.
2. Click **OK** in the confirmation dialog to delete the panel, or **Cancel** to exit the operation without deleting the panel.

Add new panel

Adds a new panel of the size chosen from the drop-down menu.

Save this panel as a template

Panel layouts can be saved as templates. The widgetsand their layout are saved, as well as MIDI mappings. When you load a template, the control mappings are intact on the source side, but, on the target side of the map, connections to parameters must be made. To load a template, click the **New** panel button and choose the **New from template** command.

Moving a panel within the rack

Right-click on the panel you want to move and choose the **Move UP** or **Move DOWN** command from the popup menu

Sizing

To change the height of a panel, right-click on the panel and choose a desired sizebetween 1U (rack unit) and 4U from the popup menu.

Setlist/Songs View

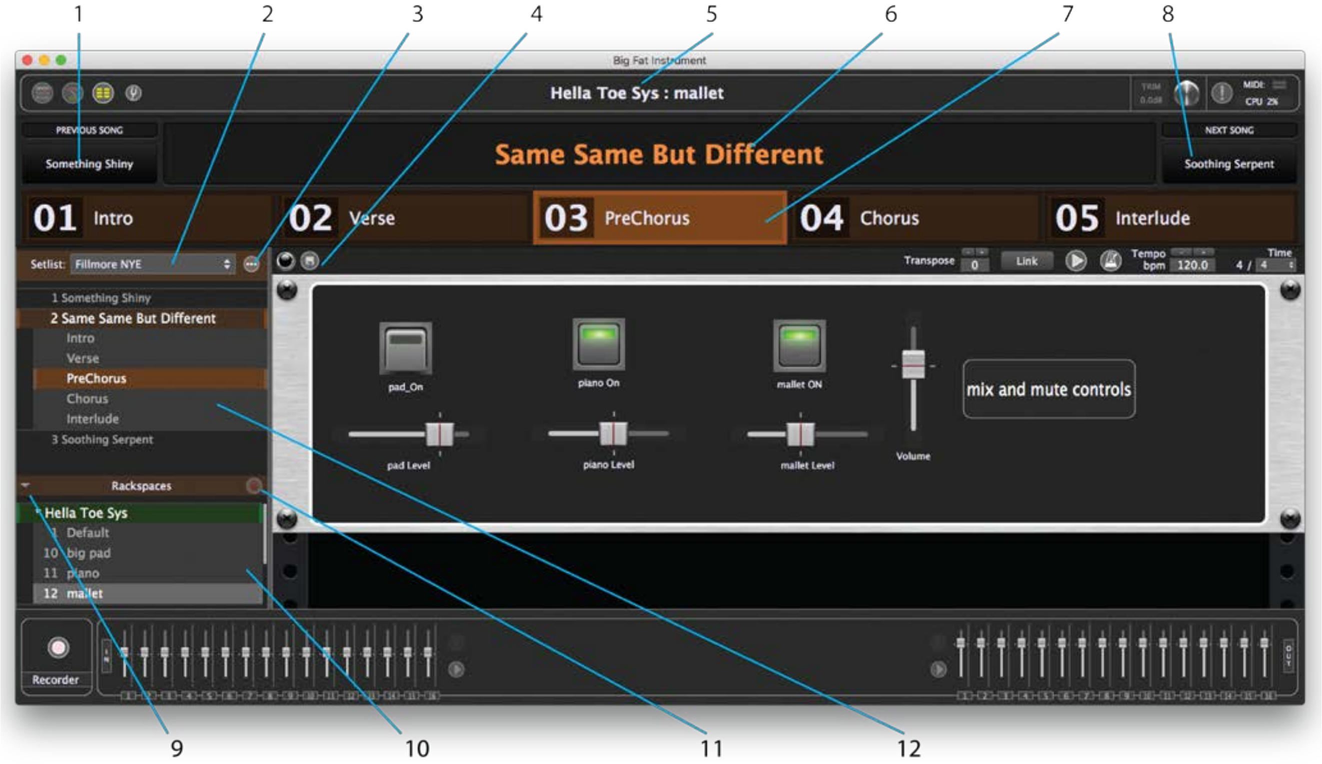
A setlist is a collection of songs.

Songs:

- Consist of an arrangement of song parts.
- Each can have its own key, transpose value, tempo, and time signature.
- Each can be linked to a ChordPro document for lyrics and chords.
- Can send a MIDI program change message out when selected.
- Can be loaded by selecting in a setlist, in the Quick Song Finder or using Previous/Next buttons that can be mapped to incoming MIDI messages.

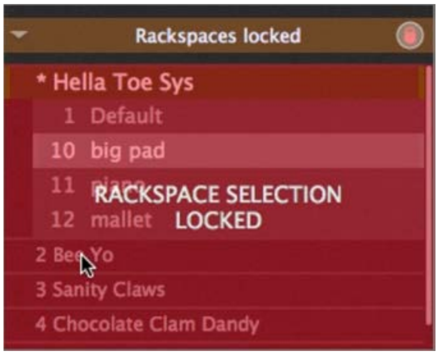
A song part:

- Is linked to a rackspace variation
- Can have its own key, transpose value, tempo, and time signature.
- Can be called by an incoming MIDI program change message.
- Can store edits to the variation it uses.
- Can send out a string of MIDI messages when it is selected.



1. **Previous Song button –**Click here to move to the next higher song in the setlist.
2. **Setlists drop-down menu –** Select a setlist to play from this menu. The All Songs setlist is always present in the list.
3. **Manage Setlists contextual drop-down menu –**This menu contains numerous commands for managing song parts, songs, and setlists.It is covered in detail in the Manage Setlists Contextual Menu section below.
4. **Capture variation button –** Click this button to save edits you have made to a variation into the currently selected song part only. The changes are not saved to the variation in the rackspace browser. This is a quick way of customizing a variation and saving it. Once the button is clicked, it offers options for what to do next. See Capture Variation Edits.
5. **Active rackspace and variation**
6. **Song name**
7. **Song part tile bar –** Click on any ofthese large, highly visible buttons to instantly switch between song parts. Tiles can be configured to visually match the layout of a MIDI foot pedal controller. The Gig Performer Lemur template also contains tiles, to make it easy to select songs by touch. Selecting a song part does one or more of these things: load the assigned variation, send out an assigned MIDI program change message,
8. **Next Song button -** Click here to move to the next lower song in the setlist.

9. **Rackspace/Variation Show/Hide button –** Click here to show or hide the rackspace browser.
10. **Rackspace browser –** Rackspaces and variations can be selected here, to map to song parts or on the fly. There is no rackspace management in the Setlists-Songs view. For that you must enter Performer or Connections view.
    NOTE: Performer and Connections views access only stored versions of variations. Edits performed in Setlist view are not available in those views.
11. **Lock Song button –**This is a safety feature to prevent accidental changes. When this button is clicked, changes to rackspaces assignments in songs are not allowed. Attempting to change a rackspace produces an error screen.



Rackspaces locked

* Hella Toe Sys

1 Default

10 big pad

11 piano

12 mallet

RACKSPACE SELECTION LOCKED

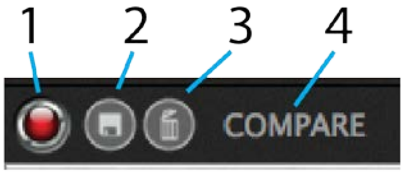
2 Bee Yo

3 Sanity Claws

4 Chocolate Clam Dandy

12. **Setlist editor –** Setlists can be accessed, songs can be added to a setlist from the Add Songs to Setlist window, and reordered by dragging. Many important functions for the setlist editor are found in the Manage Setlists Contextual Menu.

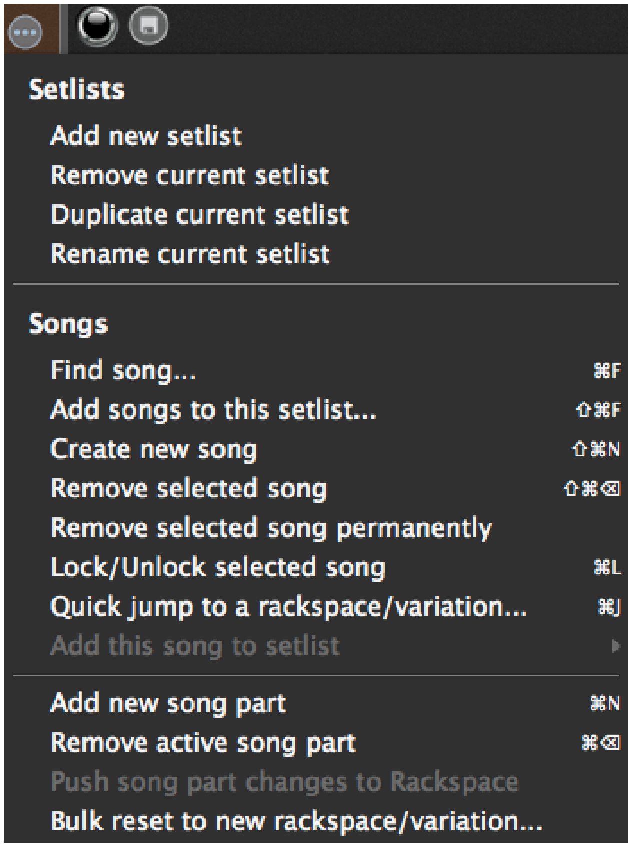
Capture Variation Edits



1. **Edit captured indicator–** When this indicator is red, it indicates that widgets have been edited since the variation was recalled, and the edited variation stored with the selected song part.
2. **Capture variation button –**Click this button to save edits you have made since calling this variation into the currently selected song part only. The changes are not saved to the variation in the rackspace browser.
3. **Revert to stored variation button –** Click this button to abandon changes you've made and reload widgets values from the stored variation.

4. **Compare button -** This momentary button, when held down, plays the stored variation. Releasing it plays the variation with all of the widget changes you've made since loading the variation. This provides instant comparisons.

"More..." Menu (Manage Setlists)


Setlists
Add new setlist
Remove current setlist
Duplicate current setlist
Rename current setlist
Songs
Find song... ⌘F
Add songs to this setlist... ⇧⌘F
Create new song ⇧⌘N
Remove selected song ⇧⌘⌫
Remove selected song permanently
Lock/Unlock selected song ⌘L
Quick jump to a rackspace/variation... ⌘J
Add this song to setlist ▸
Add new song part ⌘N
Remove active song part ⌘⌫
Push song part changes to Rackspace
Bulk reset to new rackspace/variation...

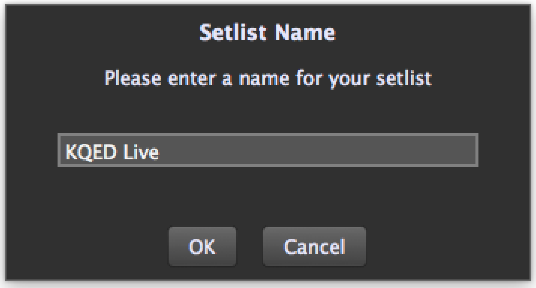
Setlists

Add new setlist–Opens a setlist name dialog and creates a new setlist in the browser.

Remove current setlist–Removes the current setlist from the gig document.

Duplicate current setlist– Opens a setlist name dialog and creates a copy of the current setlist in the setlist editor. This command is grayed out when the All Songs list is active.

Rename current setlist– Opens a setlist name dialog and allows renaming of the current setlist.



Setlist Name

Please enter a name for your setlist

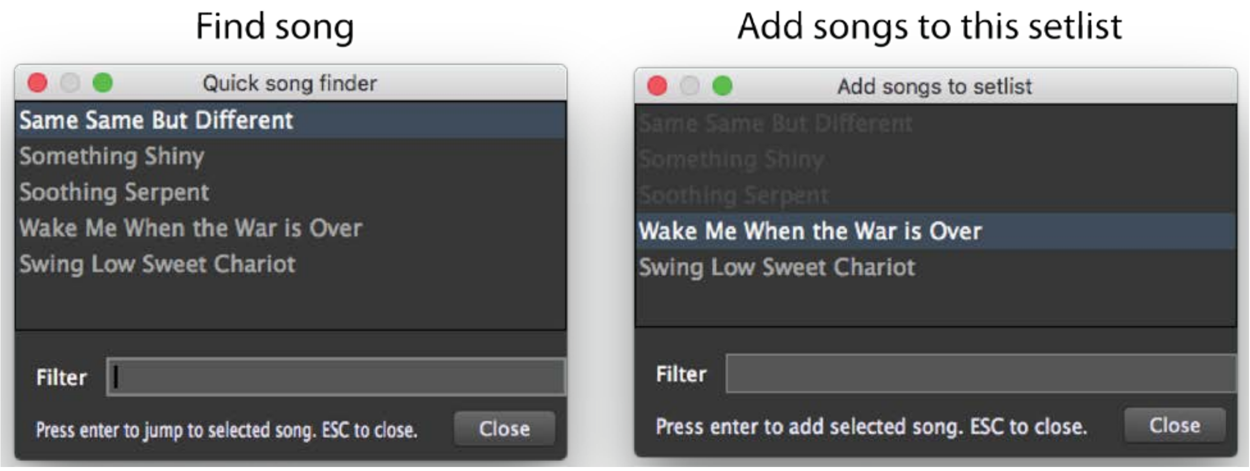
KQED Live

Songs

Find song– Choose this command to open the **Quick song finder window** displaying all of the songs currently in the setlist. To add a song from the list in the window to the current setlist, select the song you want to add and press the Enter key. Press the Escape key or click the Close button when you have added all the songs you want in this setlist. (Cmd-F/Ctrl-F on Win)

Add songs to this setlist –This command also opens the **Quick song finder**, but lists all available songs, graying out the names of all songs in the list that are already in the current setlist. (Cmd-Shift-F/Ctrl-Shift-F on Win).

Filter field – Find a song by entering search terms in this field to filter the view.



Find song

Quick song finder

Same Same But Different
Something Shiny
Soothing Serpent
Wake Me When the War is Over
Swing Low Sweet Chariot

Press enter to jump to selected song. ESC to close.

Add songs to this setlist

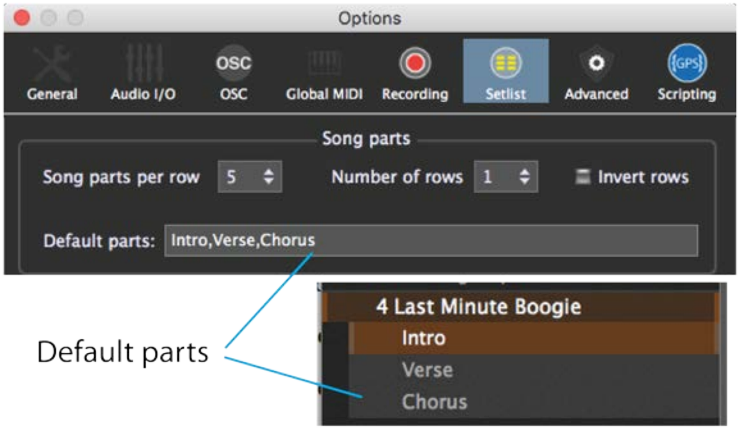
Add songs to setlist

Same Same But Different
Something Shiny
Soothing Serpent
Wake Me When the War is Over
Swing Low Sweet Chariot

Press enter to add selected song. ESC to close.

Create new song– Opens a song name dialog and creates a new song, containing some default parts, in the current setlist. The new song is also added to the All Songs list.

The default parts created for a new song can be set in the Setlist/Song Settings tab of the Options dialog.



Song parts

Song parts per row 5

Number of rows 1

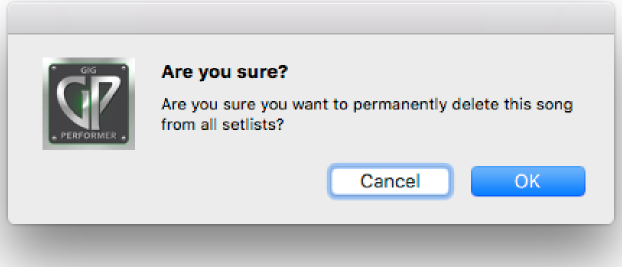
☐ Invert rows

Default parts: Intro,Verse,Chorus

4 Last Minute Boogie
Intro
Verse
Chorus

Remove selected song– Removes the selected song from the current setlist.

Remove selected song permanently– Removes the selected song from all setlists and the gig document entirely.

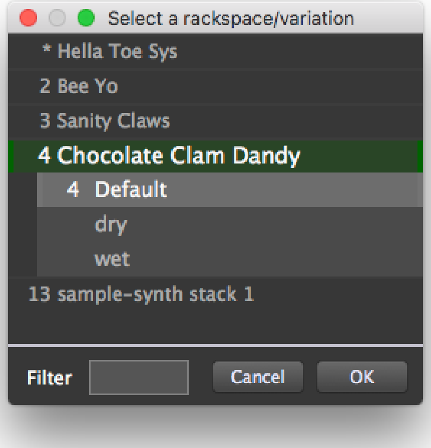


Are you sure?

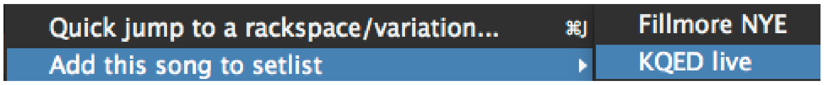
Are you sure you want to permanently delete this song from all setlists?

Lock/Unlock selected song– Choosing this command blocks changes to rackspace assignments in songs.This protects the songs from accidental modification. Clicking on a rackspace in the browser when the song is locked produces an error message.

Quick jump to a rackspace/variation– This command opens the Quick Rackspace/Variation Finder window, allowing you to immediately switch to any rackspace and variation.



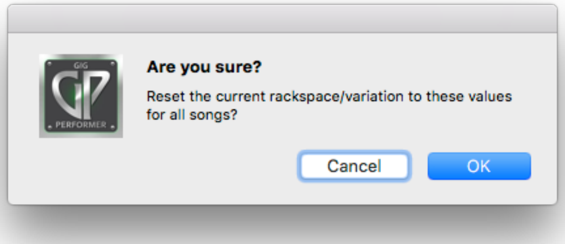
Add this song to setlist– This command is another way to add a song to a setlist. Select the song in the browser, then choose this command, picking a target setlist from the drop-down menu.



Add new song part – Opens a part name dialog and creates a new song part in the song currently selected in the browser.

Remove active song part – Removes the currently selected song part in the setlist editor. The part is removed from all occurrences of the song in all setlists.

Push song part changes to Rackspace – This command overwrites a stored variation with any changes made to widgets and then stored to a song part using the Capture Variation Edits button. After changes have been written back to the original variation, the red indicator next to the button goes out, and the Compare and Revert buttons disappear.



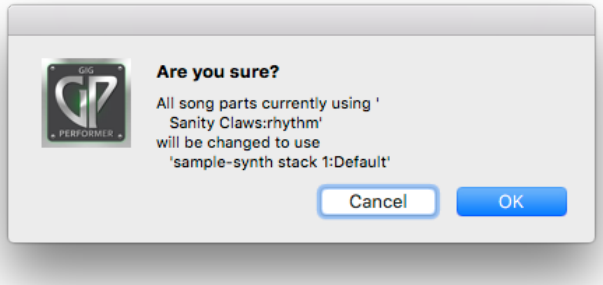
Are you sure?

Reset the current rackspace/variation to these values for all songs?

TIP: If you edit values on the fly and like the changes, this is how you save those changes. If you want to keep the original variation as well, make a duplicate of it before clicking the Capture Variation Edits button.

Bulk reset to new rackspace variation – This command enables replacing a rackspace variation, everywhere it occurs, with a different variation. This makes it fast and easy to change all of your piano sounds in one gesture when you get a new, more glorious piano instrument or sound.

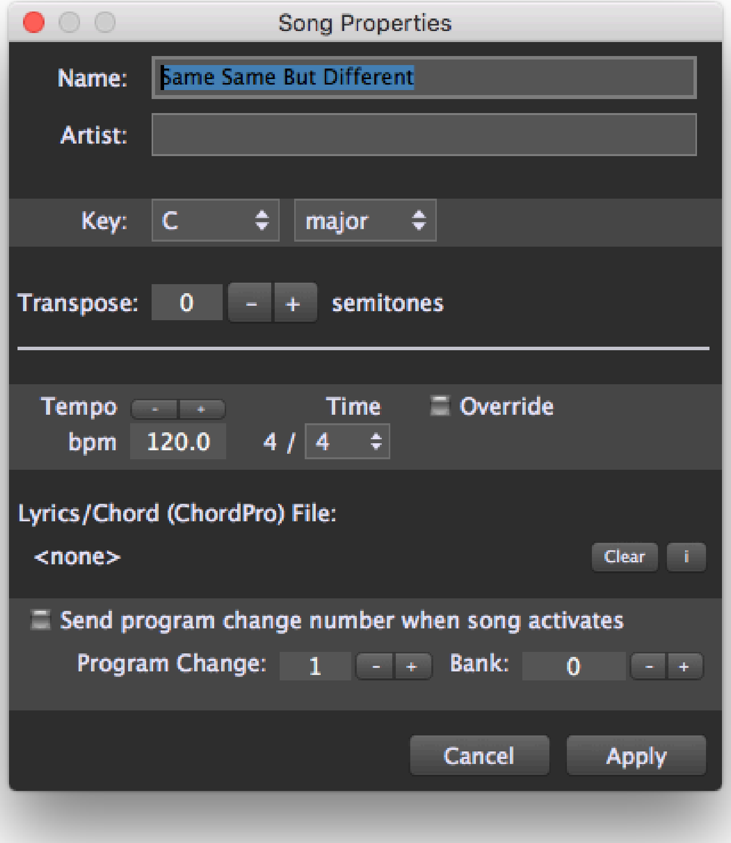
1. Select the rackspace you want to replace, either by clicking on it in the rackspace browser or by clicking on a song part that uses it.
2. Choose the Bulk reset to new rackspace variation command. The Rackspace Quick Finder will open.
3. Choose the rackspace you want to use as a replacement and click the OK button.
4. After accepting the change in the confirmation dialog, all instances of the old variation will be replaced with the new variation.



Are you sure?

All song parts currently using '
Sanity Claws:rhythm'
will be changed to use
'sample-synth stack 1:Default'

Song Properties



Song Properties

Name: Same Same But Different

Artist:

Key: C major

Transpose: 0 semitones

Tempo bpm 120.0 Time 4 / 4 □ Override

Lyrics/Chord (ChordPro) File:
<none>

□ Send program change number when song activates

Program Change: 1 Bank: 0

Name – The song name.

Artist – The artist who performed the song.

Key – Choose the **key center** and quality (**major/minor**) of the song from the drop-down menus. This value has no function other than displaying the key of the song for you.

Transpose – Enter a transpose value in semitones using either the + (plus) and – (minus) keys or by typing the value directly into the field.

Tempo – Sets a tempo for the song. Check the **Override box** to make this tempo active.

Time (signature) – Set the desired time signature for the song by clicking on the numerator and entering a value, and choosing a value for the denominator from the drop-down menu.

Override – When checked, this causes the song tempo shown here to overwrite the global tempo. Note that a tempo specified in the song part properties dialog, if enabled by its Override box, has higher priority and will override the song tempo.

Lyrics/Chord (ChordPro) File – A file in ChordPro format can be attached to the song in this area. When the song is selected, the attached ChordPro file will be displayed, if the Song Lyrics/Chords editor is already open.

NOTE: Files can be loaded only from the root directory specified in the Setlist/Song Settings tab of the Options dialog. The default path (on Mac OS X and Windows) is: ~/Documents/Gig Performer Song Lyrics-Chords. If the file you want to use is not in that directory use the**File>Import ChordPro file** command.

NOTE: A ChordPro document can only open automatically on song selection if the Song Lyrics/Chords editor is already open when the song is selected.

"i" (Information)button – Opens a dialog stating that ChordPro documents must be opened in the SongLyrics/Chords Editor.

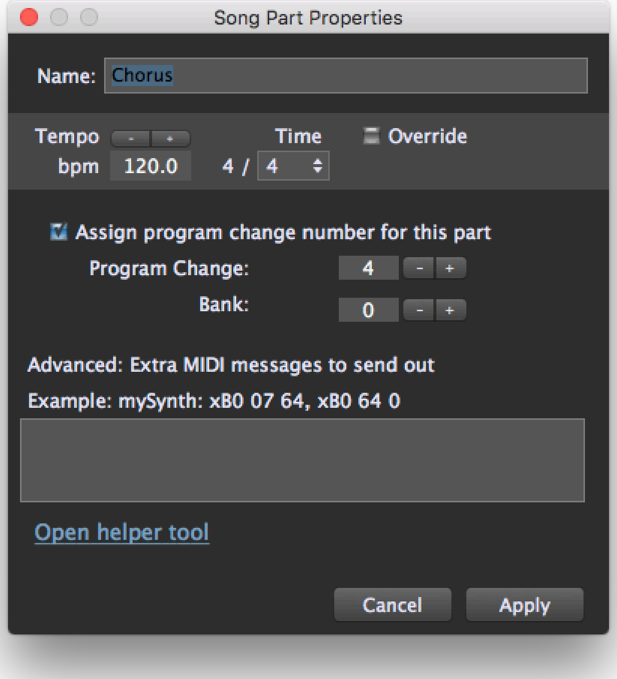
Clear button – Clears the field of any file that may be attached.

Send program change number when song activates – A MIDI program change number and bank can be set in this area using the +/- keys or by entering the value directly. When the song is selected, the program change number is sent out using the MIDI Device and Port selected in the Setlist/Song Settingsdialog.

Apply/Cancel buttons – When you are done editing, click the **Apply** button to keep the current settings and close the dialog, or the **Cancel** button to abandon all edits and close the dialog.

Song Part Properties

A song part consists of a rackspace and variation, plus a collection of properties. The variation may be as stored, or it may have been edited just for that song part. Songs are mostly assemblies of song parts.



Song Part Properties

Name: Chorus

Tempo bpm 120.0

Time 4 / 4

☐ Override

☑ Assign program change number for this part

Program Change: 4

Bank: 0

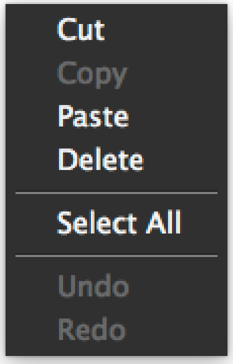
Advanced: Extra MIDI messages to send out

Example: mySynth: xB0 07 64, xB0 64 0

Name – Enter a song part name here.
Tempo +/-, text field– The tempo of the song part can be set using the + (plus) and – (minus) keys above the tempo field, or by clicking in the field and editing it directly.
Time signature -Set the time signature by clicking in the numerator and entering a value for beats per bar, then selecting a denominator from the drop-down menu.
Override check box – If this is box is checked, when the song part becomes active, the tempo set here for the song part will become the tempo while the part plays, overriding any song or global tempo. Song part tempo is the highest priority.
Assign program change number for this setlist checkbox – If this box is checked, the program change assigned here is sent out to the device and port set in the Setlist/Song Settings tab of the **Options** window.
Program change number–Sets the incoming program change message that will make this variation active. Use the + (plus) and – (minus) keys, or click in the field and enter a value directly.
Bank– Sets the incoming program banknumber that will make the variation active. Use the**+ (plus)** and **– (minus)** keys, or click in the field and enter a value directly. A program change message is also required to call a variation.
Advanced: Extra MIDI messages to send. For each song part, you can specify an arbitrary sequence of MIDI events to be injected into any MIDI In or MIDI Out block. Enter MIDI

messages as hexadecimal values, decimal values, or a combination of both.Plugins can be addressed in this field by preceding the MIDI messages with a plugin's OSC/GPScript handle, if it has one assigned. Similarly, MIDI messages can be directed to a specific external MIDI device by preceding the MIDI messages with external MIDI device's name.

Right-clicking is this field invokes a drop-down menu:



Cut – Cuts selected text out of the field.

Copy –Copies select text to the clipboard.

Paste – Pastes text from the clipboard into the field.

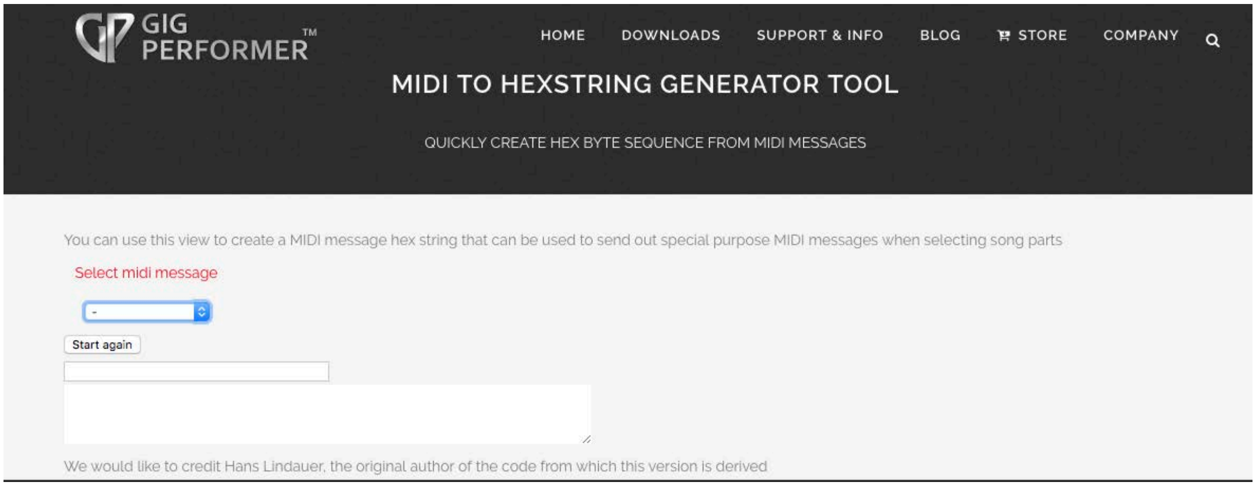
Delete – Deletes selected text from the field.

Select All – Selects the entire contents of the field.

Undo – Undo the last action.

Redo – Redo the action of the last Undo command.

Open helper tool – This link opens an online tool for converting MIDI messages to hexadecimal values.



MIDI TO HEXSTRING GENERATOR TOOL

QUICKLY CREATE HEX BYTE SEQUENCE FROM MIDI MESSAGES

You can use this view to create a MIDI message hex string that can be used to send out special purpose MIDI messages when selecting song parts

Select midi message

-

Start again

We would like to credit Hans Lindauer, the original author of the code from which this version is derived

Ways to call a song part

- Click on tile in the part bar at the top of the Setlist/Song view.
- Click on the part name in the Setlist browser.
- Select with a MIDI program change message assigned in the Song Part Properties dialog.
- Use the up/down arrows to move around the setlist to the desired variation.

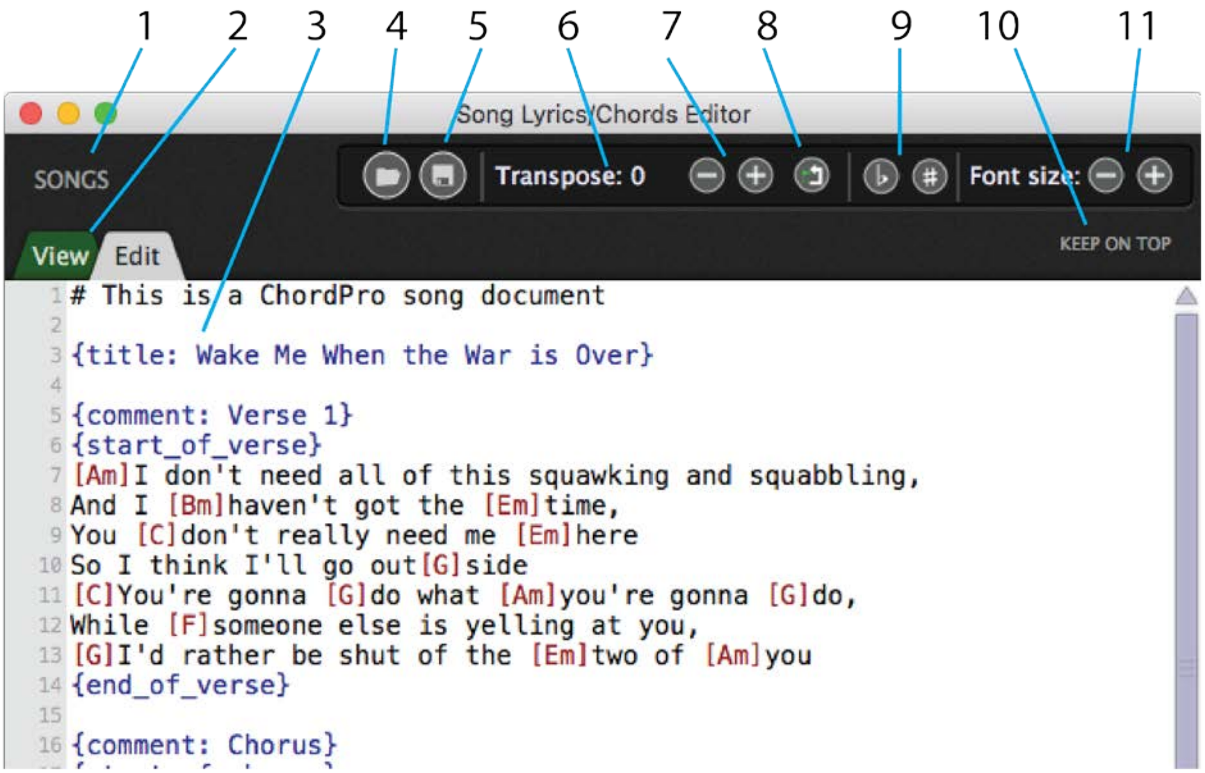
ChordPro Support

Gig Performer supports the ChordPro lead sheet file format, a simple text-based structure for storing and presenting lyrics and chords for a song. ChordPro files can be created, edited, and imported in Gig Performer. A ChordPro file can be linked to a song in the Song Properties dialog so that the lead sheet opens when the song is selected.

For more information on ChordPro, see: https://www.chordpro.org. The file format specification is at: https://www.chordpro.org/chordpro/ChordPro-File-Format-Specification.html.

Song Lyrics/Chords Editor

ChordPro files are created and modified in the **Song Lyrics/Chords Editor**:



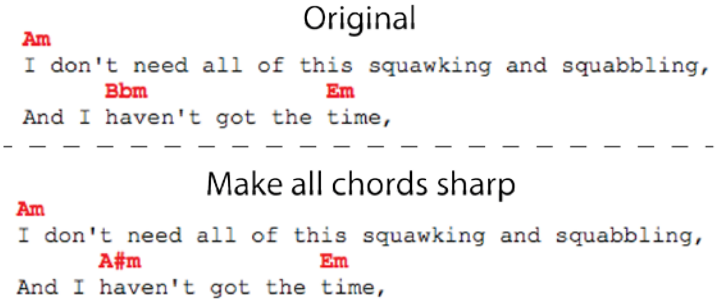
```
# This is a ChordPro song document

{title: Wake Me When the War is Over}

{comment: Verse 1}
{start_of_verse}
[Am]I don't need all of this squawking and squabbling,
And I [Bm]haven't got the [Em]time,
You [C]don't really need me [Em]here
So I think I'll go out[G]side
[C]You're gonna [G]do what [Am]you're gonna [G]do,
While [F]someone else is yelling at you,
[G]I'd rather be shut of the [Em]two of [Am]you
{end_of_verse}

{comment: Chorus}
```

1. **Song list button** – Click this to switch toSong List view.
2. **View/Edit tabs**– These tabs switch the editor between Edit mode, where ChordPro coding is done, and View mode, which interprets the code and displays the result.
3. **Editing area** – A simple text editor for entering lyrics, chords, and ChordPro tags.
4. **Load from file button** – This button brings up an Open dialog for loading gpchord (Gig Performer Chord Pro) files. Note that files can only be opened if they are in the root directory specified in the Setlist/Song Settings tab of the Options dialog. The default, on both Mac OS X and Windows, is: ~/Documents/Gig Performer Song Lyrics-Chords.

> **IMPORTANT NOTE**: A gpchord file can be loaded into the song lyrics/chords editor ONLY from this root directory. To load a ChordPro file not made by Gig Performer, or not stored in the root directory, use the File>Import ChordPro file command.

> **TIP:** To place a chord over the middle of a word, simply insert the chord in its brackets in the middle of the word, immediately in front of the letter you want the chord to appear over, with no spaces. See the word “outside” in the example above.

5. **Save to file button –** Saves the current contents of the editor to a gpchord file. Files can be saved anywhere, but can only be opened directly from the root directory.
6. **Transpose amount -** This display shows the current transposition in semitones. It is not editable. Transposition only affects the display of chord names in the ChordPro document.
7. **Increase/decrease transposition amount buttons –** Use these buttons to set the transposition amount for chord display in the Chord Pro document.
8. **Save transposition button –** Clicking this commits the transposition and makes it permanent in the ChordPro file.
9. **Make all chords sharp/flat buttons –** When there are chords whose root contains an accidental, choosing one of these commands causes those chords to be displayed enharmonically.



Original

```
Am
I don't need all of this squawking and squabbling,
     Bbm             Em
And I haven't got the time,
```

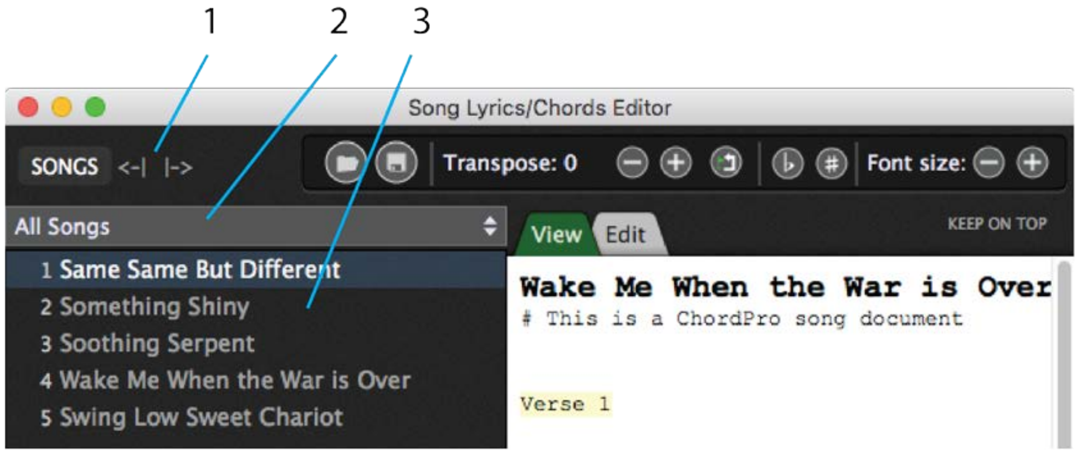
Make all chords sharp

```
Am
I don't need all of this squawking and squabbling,
     A#m             Em
And I haven't got the time,
```

10. **Keep on top button-**When this button is clicked the Song Lyrics-Chords Editor window is always in the foregroundand cannot be obscured by another window. When engaged, the button turns green.
11. **Font size larger/smaller buttons –** These buttons enlarge or shrink the font size on everything in the View mode. Clicking this button when you are in Edit mode will switch you to View mode, where you can size the font.

Song List view

When the song list button is clicked, a song browser appears to allow selecting the song for which you want to work on a ChordPro document. Click the **Songs** button again to close the song list.

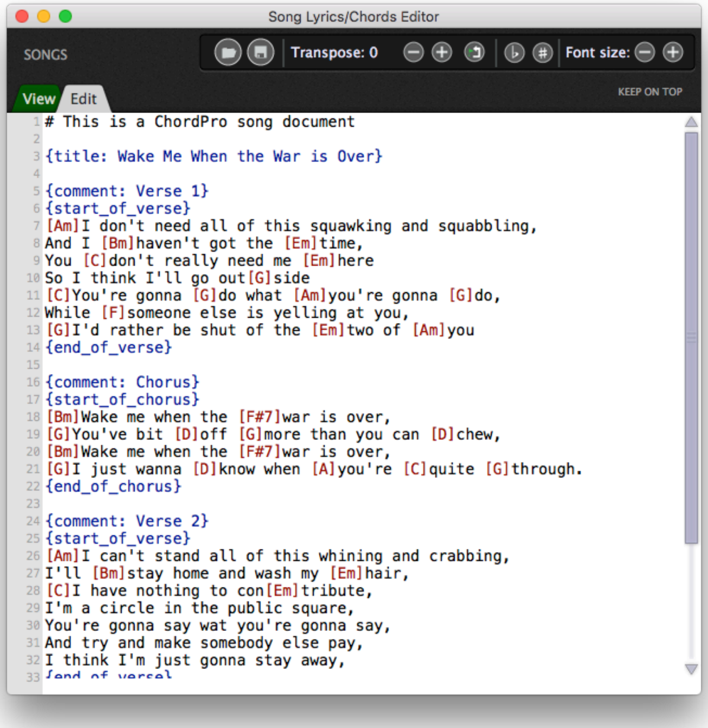

1
2
3
Song Lyrics/Chords Editor
SONGS
<-|
|->
Transpose: 0
Font size:
All Songs
View
Edit
KEEP ON TOP
1 Same Same But Different
2 Something Shiny
3 Soothing Serpent
4 Wake Me When the War is Over
5 Swing Low Sweet Chariot

```
Wake Me When the War is Over
# This is a ChordPro song document


Verse 1
```

1. **Narrow/widen song list buttons –** The song list width can be sized using these buttons.
2. **Setlist chooser –** A drop-down menu for choosing a setlist to work on.
3. **Song list –** Shows the songs in the active setlist. Clicking on a song displays the linked ChordPro file, if there is one assigned.

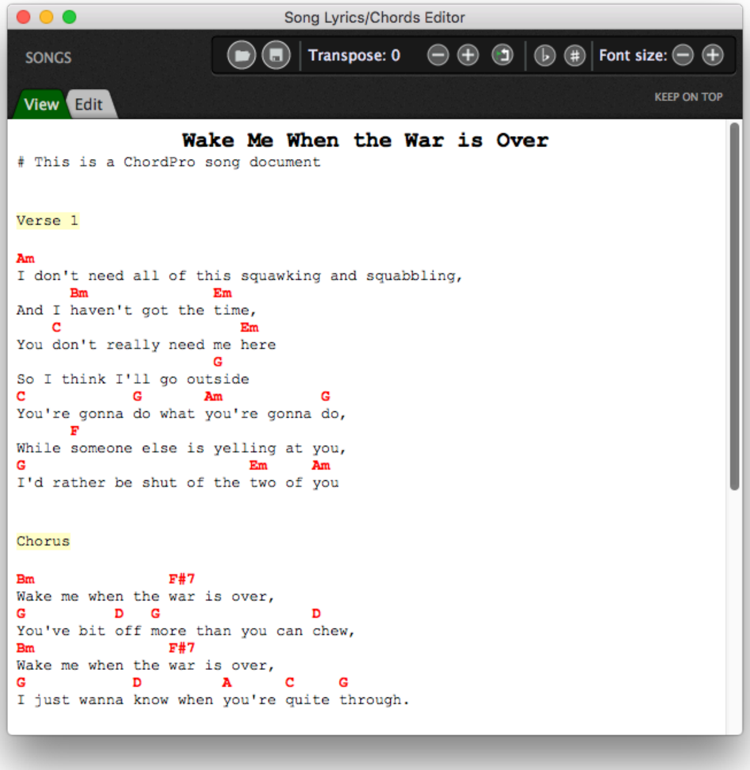
To create a new ChordPro document:

1. Choose the **Window>Show Lyrics/Chords** command to open the **Song Lyrics/Chords Editor**.
2. Click the **Edit** tab to switch to the **Edit** view.



```
# This is a ChordPro song document

{title: Wake Me When the War is Over}

{comment: Verse 1}
{start_of_verse}
[Am]I don't need all of this squawking and squabbling,
And I [Bm]haven't got the [Em]time,
You [C]don't really need me [Em]here
So I think I'll go out[G]side
[C]You're gonna [G]do what [Am]you're gonna [G]do,
While [F]someone else is yelling at you,
[G]I'd rather be shut of the [Em]two of [Am]you
{end_of_verse}

{comment: Chorus}
{start_of_chorus}
[Bm]Wake me when the [F#7]war is over,
[G]You've bit [D]off [G]more than you can [D]chew,
[Bm]Wake me when the [F#7]war is over,
[G]I just wanna [D]know when [A]you're [C]quite [G]through.
{end_of_chorus}

{comment: Verse 2}
{start_of_verse}
[Am]I can't stand all of this whining and crabbing,
I'll [Bm]stay home and wash my [Em]hair,
[C]I have nothing to con[Em]tribute,
I'm a circle in the public square,
You're gonna say wat you're gonna say,
And try and make somebody else pay,
I think I'm just gonna stay away,
{end_of_verse}
```

3. Enter lyrics and chords and tag them in accordance with the ChordPro format.
4. Click the **View** tab at any time to check that your entry and tagging are correct.



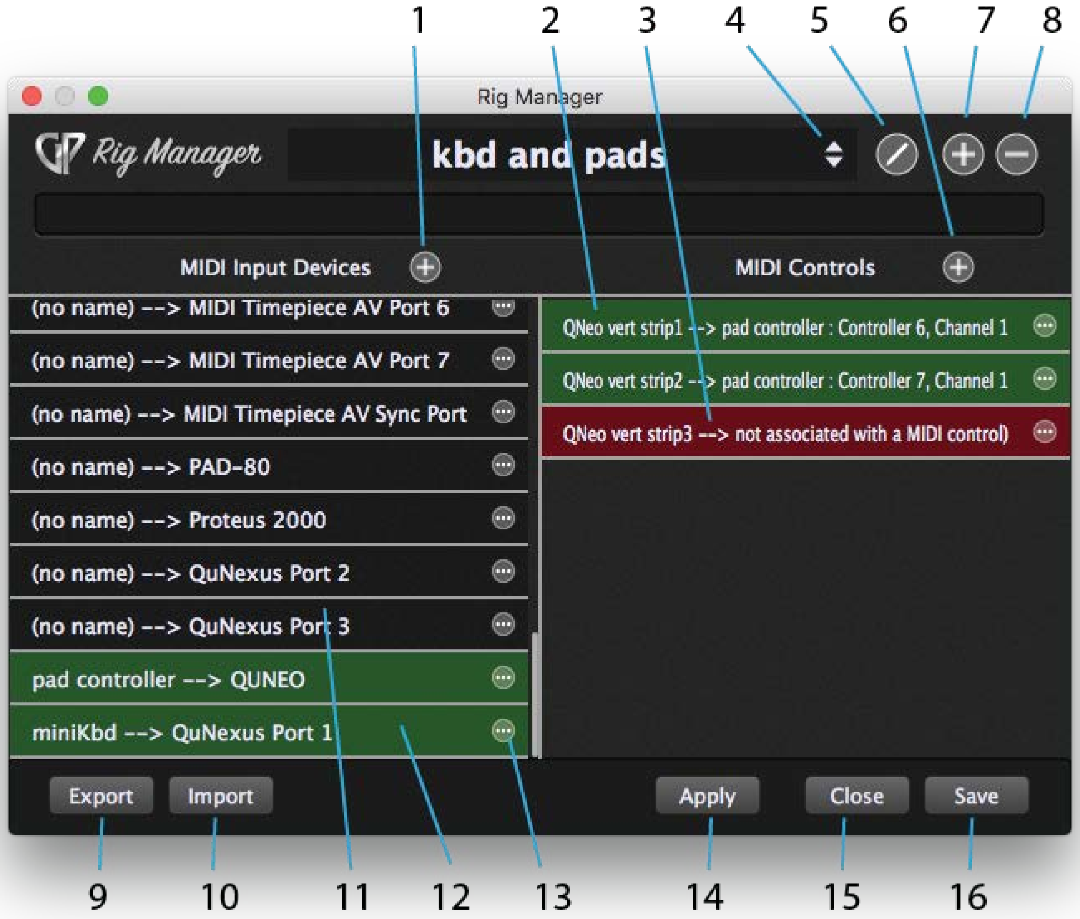
```
                   Wake Me When the War is Over
# This is a ChordPro song document


Verse 1

Am
I don't need all of this squawking and squabbling,
      Bm              Em
And I haven't got the time,
    C                    Em
You don't really need me here
                      G
So I think I'll go outside
C            G       Am           G
You're gonna do what you're gonna do,
      F
While someone else is yelling at you,
G                         Em     Am
I'd rather be shut of the two of you


Chorus

Bm               F#7
Wake me when the war is over,
G          D   G                 D
You've bit off more than you can chew,
Bm               F#7
Wake me when the war is over,
G            D         A      C     G
I just wanna know when you're quite through.
```

5. Click the Save to file button, name the file and save it. The file is saved as a .gpchord file in the root directory specified in the Setlist/Song Settings tab of the Options dialog.

IMPORTANT NOTE: A gpchord file can be loaded into the song lyrics/chords editor ONLY from this root directory. To load a ChordPro file not made by Gig Performer, or not stored in the root directory, use the File>Import ChordPro file command.

Rig Manager

> TIP: Double-clicking an item in a Rig Manager list immediately puts that item into MIDI Learn mode. To map a different controller to your system, just double-click an item, move the control on the new controller, and then move on to a the next item.



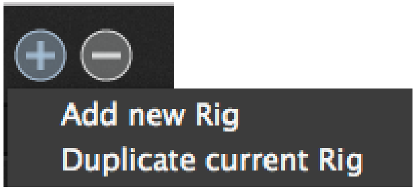
kbd and pads

(no name) --> MIDI Timepiece AV Port 6
(no name) --> MIDI Timepiece AV Port 7
(no name) --> MIDI Timepiece AV Sync Port
(no name) --> PAD-80
(no name) --> Proteus 2000
(no name) --> QuNexus Port 2
(no name) --> QuNexus Port 3
pad controller --> QUNEO
miniKbd --> QuNexus Port 1

QNeo vert strip1 --> pad controller : Controller 6, Channel 1
QNeo vert strip2 --> pad controller : Controller 7, Channel 1
QNeo vert strip3 --> not associated with a MIDI control)

1. **Add MIDI Input Device button -** Click this to add a physical controller to the MIDI Input Devices list.
2. **Assigned MIDI Control –** Green background indicatesthat the indicated virtual controller is mapped to a plugin parameter. On the far left is the name assigned to the virtual controller, for instance “QNeo vert strip1”. The arrow indicates that the virtual controller is mapped to the MIDI Input Device, channel and controller message shown.

3. **Unassigned MIDI Control –** This virtual controller ("QNeo vert strip3") was created, so the parameter control is in place, but the red background indicates it has yet to be mapped to a physical controller.
4. **Rig list –**Clicking this drops down a menu of currently available rigs. Choose one from the list.
5. **Rename this rig –** Click this to customize the name of a rig.
6. **Add a new MIDI Control–** Click this to add a named MIDI control to the list. The new control needs to be assigned to a physical controller to be used.
7. **Add a new rig–** Click this button to duplicate the rig or add a new one.

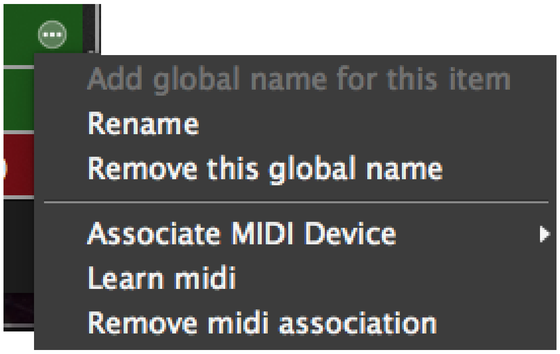


8. **Remove this rig –** Click this button to remove the current rig. A confirmation dialog offers you the chance to cancel, if you change your mind.
9. **Export Rigbutton –** Click this to save the state of the rig to a new file that can be moved to a new computer and then imported.
10. **Import Rig button –** Click this to load an existing rig file.
11. **Unnamed MIDI Input Device –** Gig Performer recognizes the control device, but no global name has been defined for it yet.
12. **Named MIDI Input Device –** This device has been given a global name by clicking on the Options button and choosing **Add global name for this item**.
13. **Options button –** Clicking this button pops up the MIDI Input Options menu.
14. **Apply button –** Click this to apply changes you have made.
15. **Close button –** Click this to close the Rig Manager view.
16. **Save button –** Click this to save the state of the rig to the current rig file. Use this when you will be opening the saved file on the same computer it was created on.

TIP: Each entry in the Rig Manager's **MIDI Input Devices** list will be recorded to a separate track by the MIDI Recorder when it is used.

MIDI Input Options menu

The MIDI Input Options menu pops up when the Options button is clicked for a MIDI input device in the list.



Add global name for this item – Choose this command, when it is available, to give a global name (alias) to a MIDI input device.Once a device has been named, this command is grayed out.

Rename – Choose this to change the global name of the device.

Remove this global name – This command removes the global name, returning the device to "(no name)" status.

Associate MIDI Device – You can keep a global name as it is but change the device it applies to by associating the device in the list with a different physical device. Choose this command and a submenu pops up with all of the available MIDI devices. Choose the one you want from the list.

Learn midi – Often, the fastest way to associate a device in the list with a physical device is to choose this command and then move a control on the physical device. The global name will be mapped to the physical device detected.

Remove midi association– Choosing this command retains the global name but disconnects it from the MIDI device to which it was mapped when the command was chosen.

Steps to Make a Rig

See Using Rig Manager for a full tutorial. Here are the basic steps to using Rig Manager:

1. Create one or more MIDI input devices and give them global names.
2. Create one or more MIDI controls and give them global names.

3 Link the MIDI controls to physical controllers on the MIDI input device.

4 In the Widget Editor, choose a global MIDI control to link to the widget.

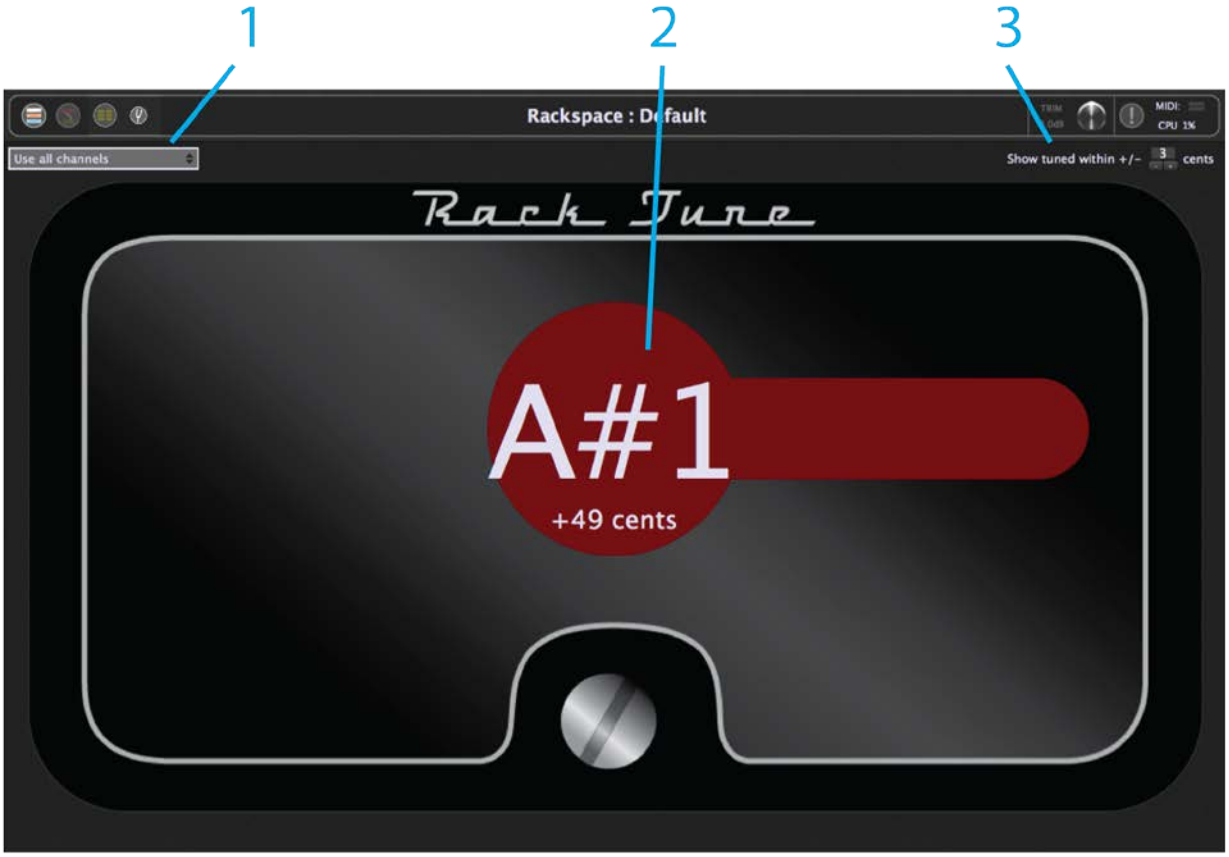
5 When all controls have been mapped, save your rig.

6 Duplicate and modify the rig or build a new one for a different use, for instance, performing versus rehearsing. Name the second rig.

7 Switch between rigs as needed to accommodate different MIDI input devices.

Musical Utilities

Tuner



1. **Use all channels –**Click here for a dropdown menu if you want to havethe Tuner "listen" to signal coming from a specific input channel of your connected audio interface. The default is to have it listen to signals coming from all channels.

2. **Tuning indicator –** Shows whether input pitch is sharp or flat, with an exact amount in displayed in cents.

3. **Show tuned within –** Allows you to specify a range within which the main display changes from red (out of tune) to green (in tune). Available values are +/- 1 cent to +/- 5 cents; default is +/- 3 cents.

Metronome

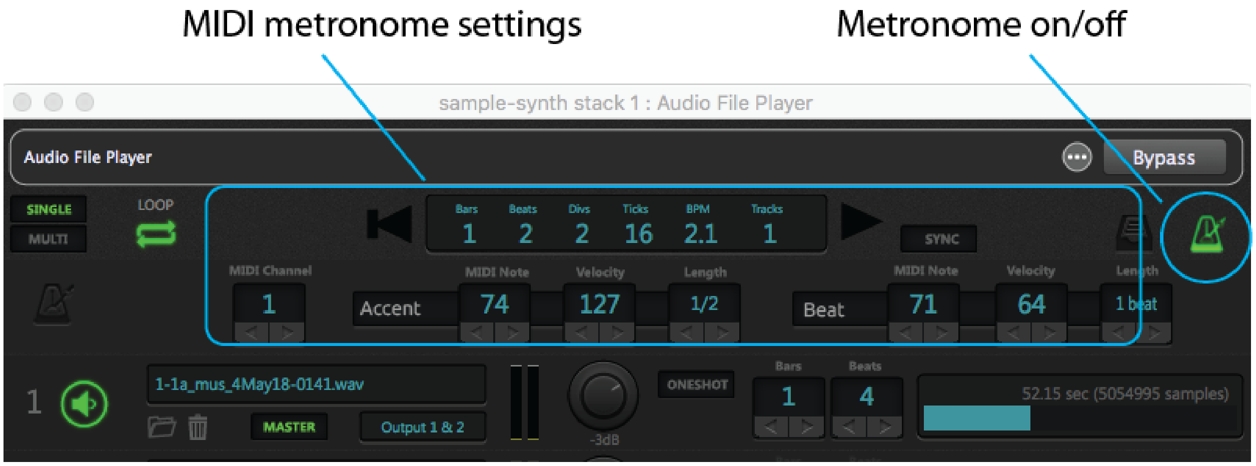
Gig Performer offers two separate metronomes for different purposes.

MIDI Metronome

Included in the Audio File Player is a MIDI metronome optimized for working with short loops. Tell the MIDI metronome the musical length of a loop and it will generate MIDI notes for a metronome.The MIDI metronome does not do tempo detection and is unrelated to the audio metronome.

The MIDI metronomeplays when the master audio file starts playing. MIDI channel and note settings for the metronome are accessed in the Audio File Player by clicking the metronome on/off button. Naturally, the MIDI Out from this plugin needs to drive MIDI In on a MIDI sound source plugin.

For more detail on the MIDI metronome, see the Audio File Player section.



<u>Using the MIDI metronome</u>

1. Load a file into the player in Single mode, or multiple files in Multi mode. In Multi, the longest file will be designated as the Master and used to calculate the metronome.

2. Click the Metronome On/Off button to activate the metronome.

3. Enter the duration of the loop in bars into the Bars field on the lane of the Master track, by typing it in directly or using the +/- arrows beneath the field.

4. Enter the number of beats per bar in the loop into the Beats field on the lane of the Master track, by typing it in directly or using the +/- arrows beneath the field.

5. Choose or add a plugin as the sound source for the metronome. Choose the sound you want to use for the metronome and set a MIDI channel for it. Don't forget to make audio connection from the sound source to your interface outputs.

6. In the Connections view, connect the MIDI Out port of the Audio File Player to the MIDI In port of the plugin you are using to generate the metronome sound.

7. Define the MIDI note, velocity, and duration (Length) to be used for both the Accent and Beat metronome sounds.

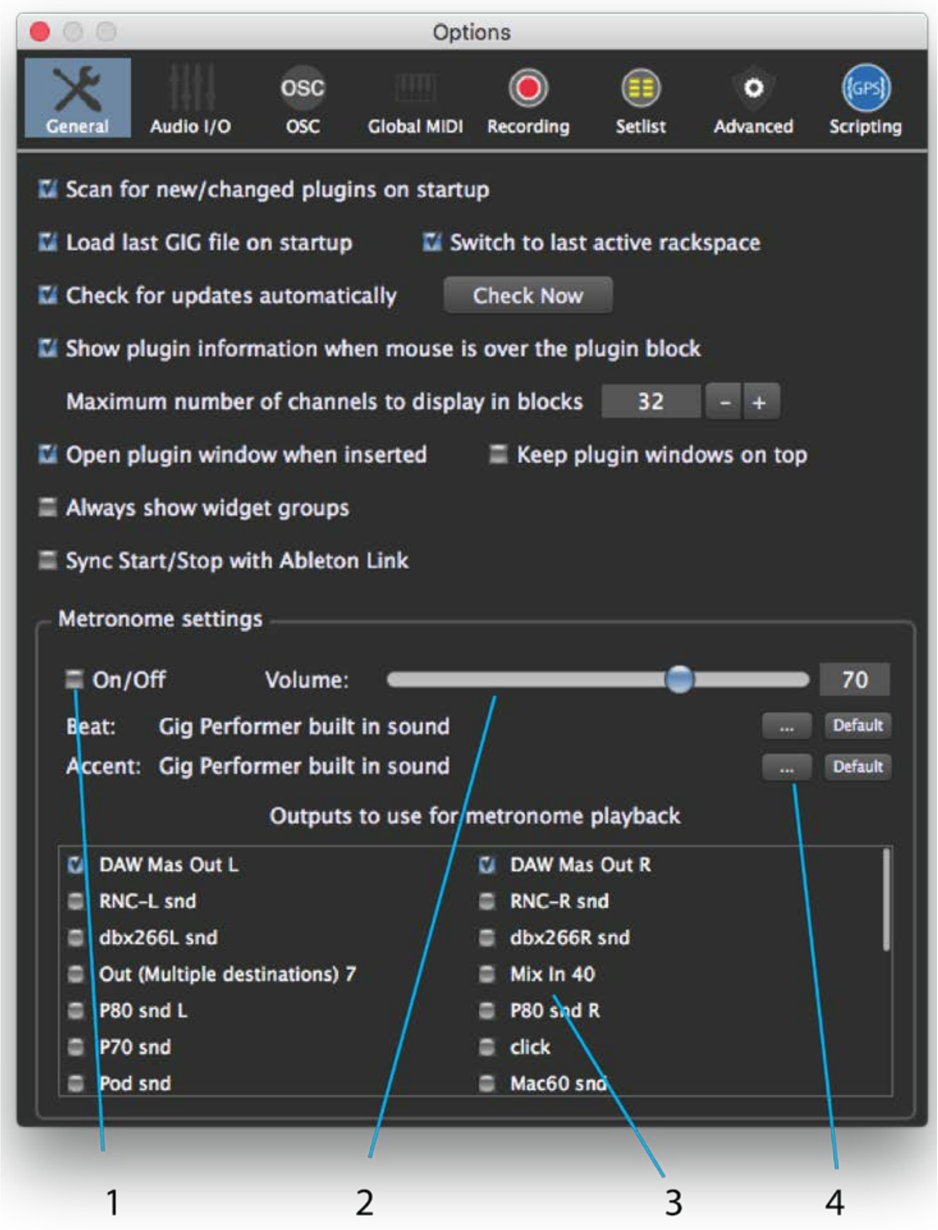
8. Set the MIDI Channel field to the channel number of the metronome sound.

9. Click the Play/Pause button and the metronome will play, along with the audio files.

Note: The MIDI metronome assumes the loop contains an exact number of bars, with the beginning of the file being bar 1, beat 1. If this is not the case, the metronome may not sync correctly with the Master track.

Audio Metronome

Setup for the audio metronome is done on the General tab of Gig Performer's preferences. Choose the **Options>Preferences** command to open the Options dialog, or,on Mac OS X, press Cmd-, (comma).



☑ Scan for new/changed plugins on startup

☑ Load last GIG file on startup ☑ Switch to last active rackspace

☑ Check for updates automatically

☑ Show plugin information when mouse is over the plugin block

Maximum number of channels to display in blocks 32

☑ Open plugin window when inserted □ Keep plugin windows on top

□ Always show widget groups

□ Sync Start/Stop with Ableton Link

Metronome settings

□ On/Off Volume: 70

Beat: Gig Performer built in sound

Accent: Gig Performer built in sound

Outputs to use for metronome playback

☑ DAW Mas Out L ☑ DAW Mas Out R
□ RNC-L snd □ RNC-R snd
□ dbx266L snd □ dbx266R snd
□ Out (Multiple destinations) 7 □ Mix In 40
□ P80 snd L □ P80 snd R
□ P70 snd □ click
□ Pod snd □ Mac60 snd

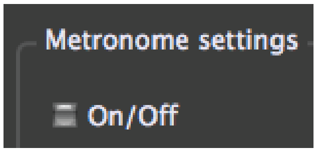
1. **Metronome on/off button** - Click this to activate the metronome.
2. **Metronome playback volume –** To set the volume of the metronome, drag the slider or click in the field and enter the value directly.

3. **Metronome output routing –** Check boxes for all outputs to which you want the metronome routed.
4. **Metronome sound select area –** Click on the "…" button to select a sound file (aif, wav, or mp3).

The audio metronome is simple to use:

1. Set the outputs to which you want the metronome routed in the General prefs tab.
2. Set the desired global tempo
3. Set the desired time signature
4. Activate the metronome.

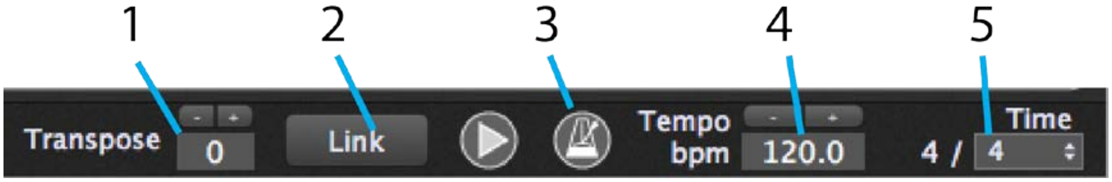
The metronome on/off buttons in the top toolbar and on the General prefs tab are identical; changing one changes the other, as well.



The first beat of each bar is accented, regardless of the specified global time signature

Tempo and Time Signature

Tempo and time signature for the metronome can be set globally in the upper toolbar.



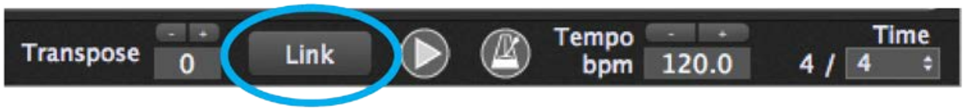
1. **Global transpose –** This value is the number of semitones added to or subtracted from the output of each MIDI In block.
2. **Ableton Link enable –** Click this button to synchronize Gig Performer's tempo to an Ableton Link tempo.
3. **Metronome on/off –** This button activates and deactivates the audio metronome
4. **Global tempo–**This is the tempo of Gig Performer's audio metronome. Click in the field and enter a tempo or use the +/- keys above it.
5. **Global time signature –**Click on the numerator and enter the number of beats per measure, click on the denominator and choose the number of beats per measure from the dropdown menu.

Note: There are tempo settings in the Song Properties and Song Part Properties dialogs, as well as the global tempo, and these settings are accompanied by an Override check box that makes that tempo active. Selecting a song that has a tempo specified and the Override box checked forces Gig Performer to change the global tempo setting to the specified song tempo. A song part with a tempo specified and the Override box checked is the highest priority and overrides both global and song tempos when the part is selected. When the Override box is not checked, a specified tempo is inactive and has no effect.

Ableton Link support

Ableton Link is a system for synchronizing multiple music applications, which may be running on different devices, by providing unified, real-time tempo control. Link-enabled programs find each other automatically when they are on the same network. Any "player" in the network can change the tempo and the others follow. No configuration of any kind is necessary, only that it be enabled.

To enable Gig Performer to sync to Ableton Link, click on the **Link** button that is always in the upper toolbar.



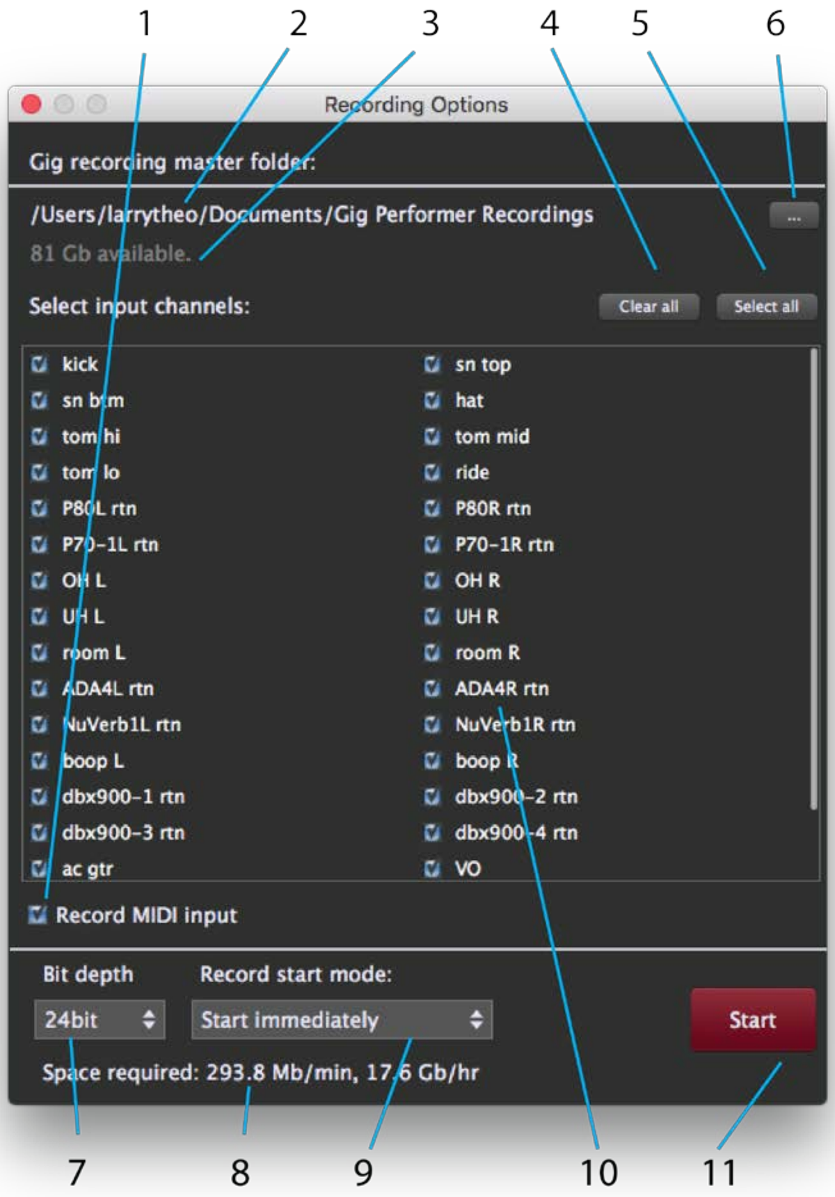
Recorders

Audio Input Recorder

The Audio Input Recorder is a simple means of capturing audio arriving at your audio interface's inputs, before it has passed through any of the plugins. By capturing the audio and saving your gig, you can replicate the performance and modify it in post-production. To access the Audio Input Recorder, just click the **Recorder** button in the bottom toolbar.

Files are recorded in WAV format. Each time you start the recorder, a folder is created within the target folder and named with the date of the recording. All files from that recording are placed within that folder. This makes it very easy to keep straight recordings made at different performances.

A delayed start feature lets you set a time for recording to automatically begin, so that the recorder can be configured and tested at soundcheck, then set to start recording at the proper time.



Gig recording master folder:

/Users/larrytheo/Documents/Gig Performer Recordings

81 Gb available.

Select input channels:

☑ kick
☑ sn btm
☑ tom hi
☑ tom lo
☑ P80L rtn
☑ P70-1L rtn
☑ OH L
☑ UH L
☑ room L
☑ ADA4L rtn
☑ NuVerb1L rtn
☑ boop L
☑ dbx900-1 rtn
☑ dbx900-3 rtn
☑ ac gtr

☑ sn top
☑ hat
☑ tom mid
☑ ride
☑ P80R rtn
☑ P70-1R rtn
☑ OH R
☑ UH R
☑ room R
☑ ADA4R rtn
☑ NuVerb1R rtn
☑ boop R
☑ dbx900-2 rtn
☑ dbx900-4 rtn
☑ VO

☑ Record MIDI input

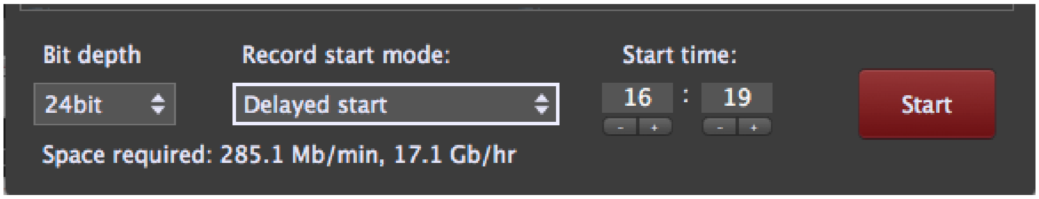
Bit depth: 24bit

Record start mode: Start immediately

Space required: 293.8 Mb/min, 17.6 Gb/hr

1. **Record MIDI Input –** When checked, MIDI input from all sources is recorded at the same time as audio. Each MIDI In block is recorded as a separate track in a multitrack (Type 1) Standard MIDI File. These files are easily imported into just about any DAW with MIDI capabilities or other MIDI sequencing software.
2. **Target folder display** – All files are recorded into this folder.

3. **Available space display –** Shows the amount of recording space available on the disk on which the target folder is located.
4. **Clear All button –** Clicking this button deselects all inputs, so that no channels will be recorded. When you only need to record a few channels, it can be quickest to deselect all and then select only the ones you need. Similarly, when setting up recording for a new situation, it may be most efficient to deselect all inputs and enable those you need.
5. **Select All button**– Clicking this button enables all channels for recording. When you want to record most or all channels, it can be quickest to select all channels and then deselect the ones you do not need.
6. **Set target folder button**– Clicking this brings up a file navigator that enables you to choose a target folder.
7. **Bit depth field**– Click this to select 8, 16, 24, or 32-bit (floating point) depth for recording.
8. **Recording space required display**– Shows the amount of disk space required to record the selected channels per minute or per hour. This display adjusts as you select or deselect channels.
9. **Record Start mode field –** Click on this and choose Start immediately to begin recording as soon as the red Start button is clicked, or choose Delayed start and enter a start time in the minutes and seconds fields, as seen below.



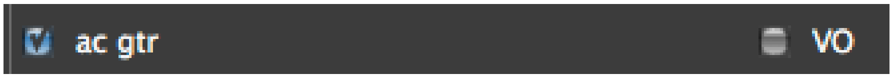
Bit depth: 24bit

Record start mode: Delayed start

Start time: 16 : 19

Space required: 285.1 Mb/min, 17.1 Gb/hr

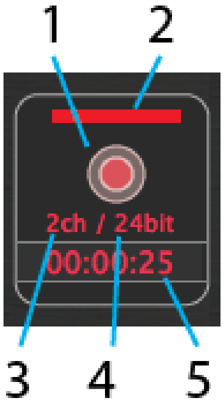
10. **Channel record enable checkboxes –** Click a checkbox to enable a channel for recording. A channel is record-enabled when the box is blue and checked; recording is disabled for the channel when the button is gray and unchecked.



☑ ac gtr

□ VO

11. **Record Start button –**Clicking this button immediately begins recording of record-enabled channels. The button changes state when recording:



1. **Stop button –** Click this to stop recording.
2. **Recording beacon –** Flashes red while Gig Performer is recording audio.
3. **Number of channels being recorded**
4. **Bit depth (resolution) of recording**
5. **Elapsed recording time**

MIDI Recorder

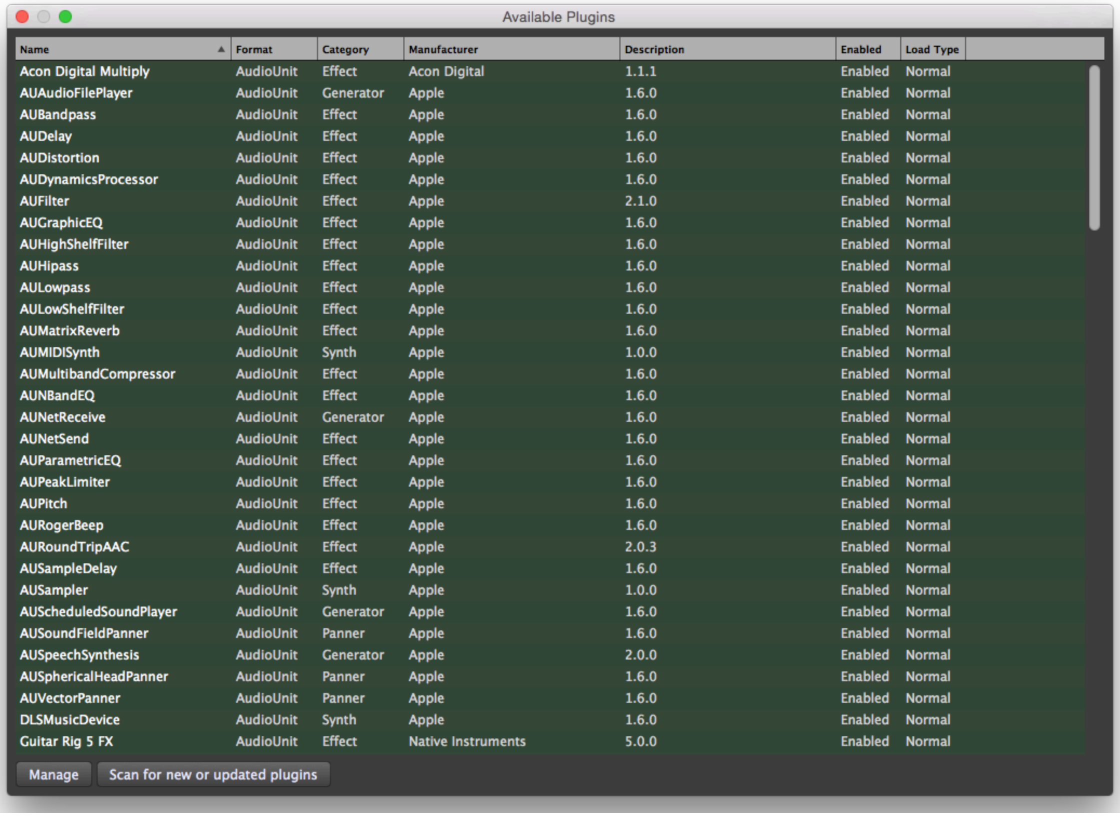
☑ Record MIDI input

Gig Performer makes it as easy to capture incoming MIDI as audio. Check the RecordMIDIInput box in the **Recording Options screen** and incoming MIDI events from all MIDI devices known to Gig Performer recorded, each device to its own track in a Type 1 Standard MIDI File. Generally, each MIDI device corresponds to a physical MIDI port.

You can see a list of MIDI devices known to Gig Performer by right-clicking any MIDI In module and moving the cursor to the **Change MIDI Input Device** command to see its associated drop-down menu. Another place to see this list is the **MIDI Input Devices** column of the Rig Manager.

MIDI Recorder captures tempo, as well.

Plugin Manager



Name	Format	Category	Manufacturer	Description	Enabled	Load Type
Acon Digital Multiply	AudioUnit	Effect	Acon Digital	1.1.1	Enabled	Normal
AUAudioFilePlayer	AudioUnit	Generator	Apple	1.6.0	Enabled	Normal
AUBandpass	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUDelay	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUDistortion	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUDynamicsProcessor	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUFilter	AudioUnit	Effect	Apple	2.1.0	Enabled	Normal
AUGraphicEQ	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUHighShelfFilter	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUHipass	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AULowpass	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AULowShelfFilter	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUMatrixReverb	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUMIDISynth	AudioUnit	Synth	Apple	1.0.0	Enabled	Normal
AUMultibandCompressor	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUNBandEQ	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUNetReceive	AudioUnit	Generator	Apple	1.6.0	Enabled	Normal
AUNetSend	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUParametricEQ	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUPeakLimiter	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUPitch	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AURogerBeep	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AURoundTripAAC	AudioUnit	Effect	Apple	2.0.3	Enabled	Normal
AUSampleDelay	AudioUnit	Effect	Apple	1.6.0	Enabled	Normal
AUSampler	AudioUnit	Synth	Apple	1.0.0	Enabled	Normal
AUScheduledSoundPlayer	AudioUnit	Generator	Apple	1.6.0	Enabled	Normal
AUSoundFieldPanner	AudioUnit	Panner	Apple	1.6.0	Enabled	Normal
AUSpeechSynthesis	AudioUnit	Generator	Apple	2.0.0	Enabled	Normal
AUSphericalHeadPanner	AudioUnit	Panner	Apple	1.6.0	Enabled	Normal
AUVectorPanner	AudioUnit	Panner	Apple	1.6.0	Enabled	Normal
DLSMusicDevice	AudioUnit	Synth	Apple	1.6.0	Enabled	Normal
Guitar Rig 5 FX	AudioUnit	Effect	Native Instruments	5.0.0	Enabled	Normal

The Plugin Manager lets you view all of the plugins installed in your system, categorized by name, format (AudioUnit or VST), category (Effect, tone Generator, Synth, Panner, etc.), manufacturer name, and version name.

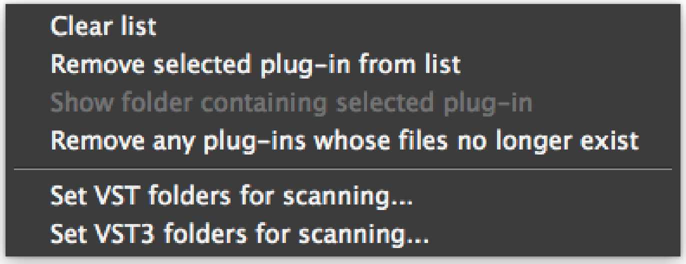
To access the Plugin Manager, press Cmd-Shift-P or choose **Plugin Manager…** from the Window menu. Click on "Name," "Format," "Category," or "Manufacturer" in the top line to sort the list by that category. Click that category a second time to reverse the order of the list.

Right-clicking on any plugin name opens a pop-up menu that allows you to Enable or Disable any plugin, or to specify its Loading type ("Normal" or "Delayed"). If you encounter third-party plugins that regularly fail to reinitialize successfully (i.e., they don't launch or display with the last saved value when recalling a .gig file), try designating them here as "Delayed." (They will be overlaid with a light green see-through bar.) Gig Performer will then slightly delay their loading time the next time it is launched, which may help solve the problem.

Clicking on the **Scan for new or updated plugins** button(to the right of the **Manage** button) causes Gig Performer to immediately scan your system for any newly installed or updated plugins; any found will be added to the Plugin Manager list.

Important note: Plugins labeled as "Disabled" in the Plugin Manager are overlaid with a red see-through bar and will not appear in the list of available plugins when you right-click in the Plugin Connections area. This enables you to reduce clutter if you are sure you won't need to access particular plugins when creating a rackspace.

Clicking on the **Manage** button (in the bottom right-hand corner) opens the following popup menu:


Clear list
Remove selected plug-in from list
Show folder containing selected plug-in
Remove any plug-ins whose files no longer exist
Set VST folders for scanning...
Set VST3 folders for scanning...

Clear list -Clears the listing of all plugins.

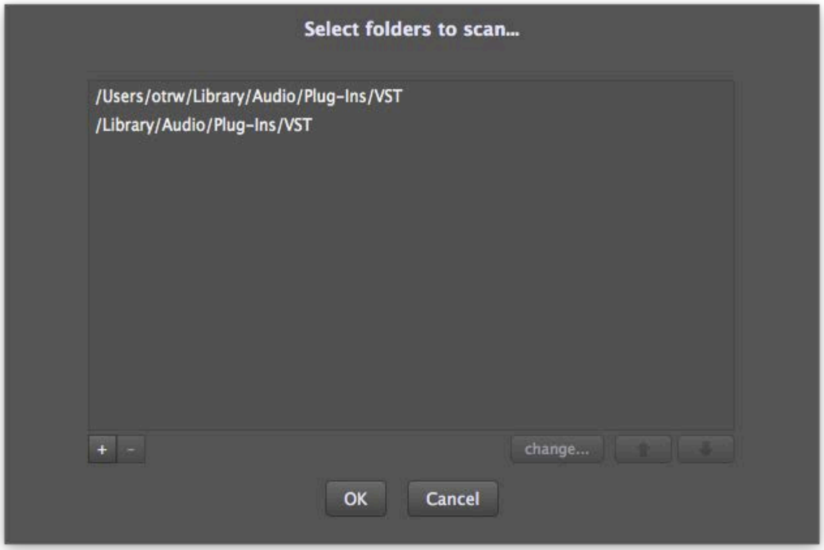
Note: While this does not actually erase any plugins from your computer, it does remove them from Gig Performer's memory, so a new scan is required after this operation. Click on the "Scan for new or updated plugins" button next to the Manage button to begin the scan.

Remove selected plugin from list - Select a plugin first to make this operational.

Show folder containing selected plugin- Select a plugin first to make this operational.

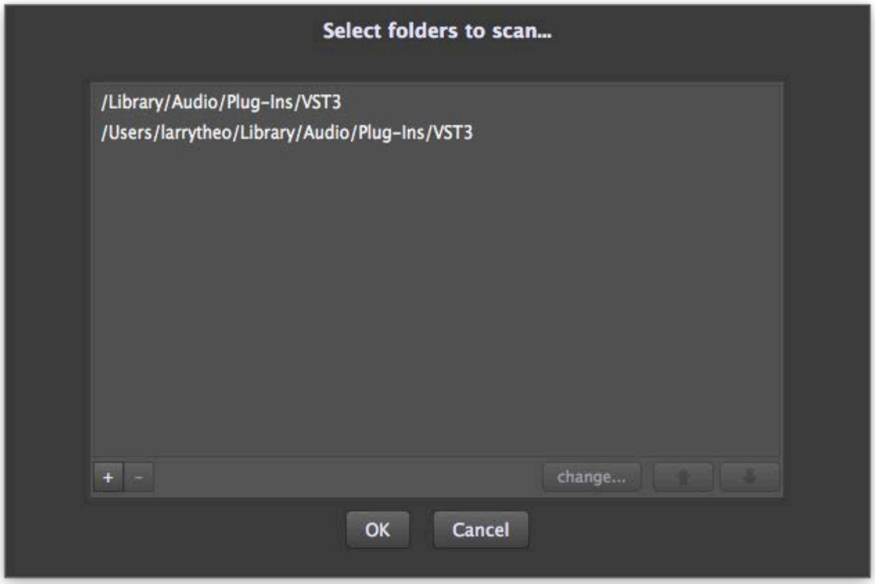
Remove any plugins whose files no longer exist - Use this to clean up your plugins list.

Set VST folders for scanning... –Opens the window shown below, which allows you to change, add, or delete the VST plugin folders to be scanned. (See Troubleshooting for more information.)


Select folders to scan...

/Users/otrw/Library/Audio/Plug-Ins/VST
/Library/Audio/Plug-Ins/VST

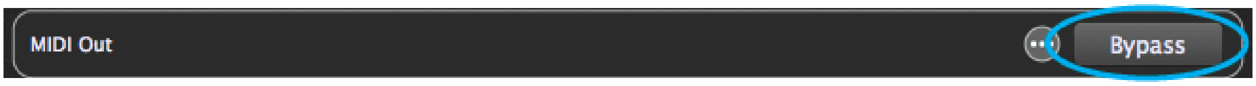
Set VST3 folders for scanning... – Opens a window that allows you to change, add, or delete the folders to be scanned for VST3 plugins. (See Troubleshooting for more information.)


Select folders to scan...

/Library/Audio/Plug-Ins/VST3
/Users/larrytheo/Library/Audio/Plug-Ins/VST3

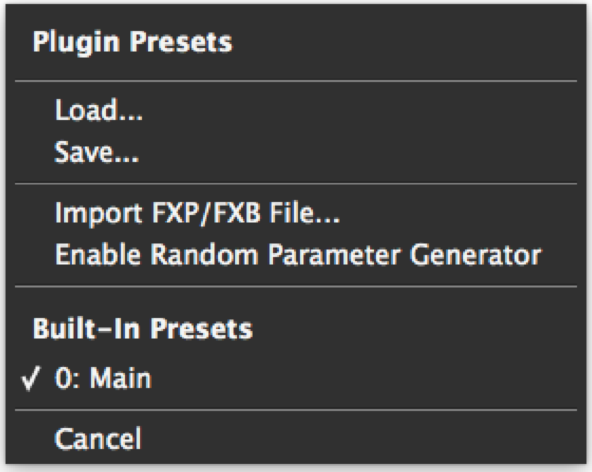
Bundled Plugins

Note: Like all third-party plugins, the plugins supplied with Gig Performer include Bypass and "..." ("three dots" or "More...") buttons in the upper right-hand corner, as shown circled here:



- Clicking on the Bypass button causes the plugin to be muted (not bypassed). Click here for more information.

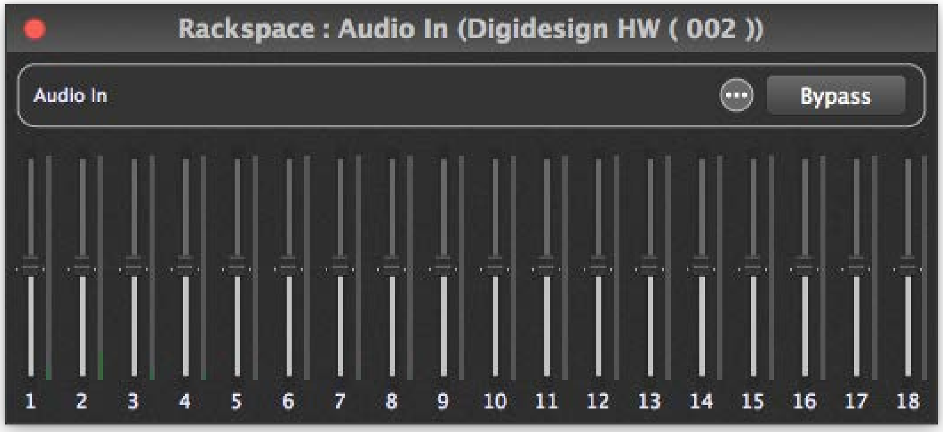
- Clicking on the "..." button causes the Plugin Presets pop-up menu to appear:



This allows you to load factory-created presets, or to load or save presets you create yourself.

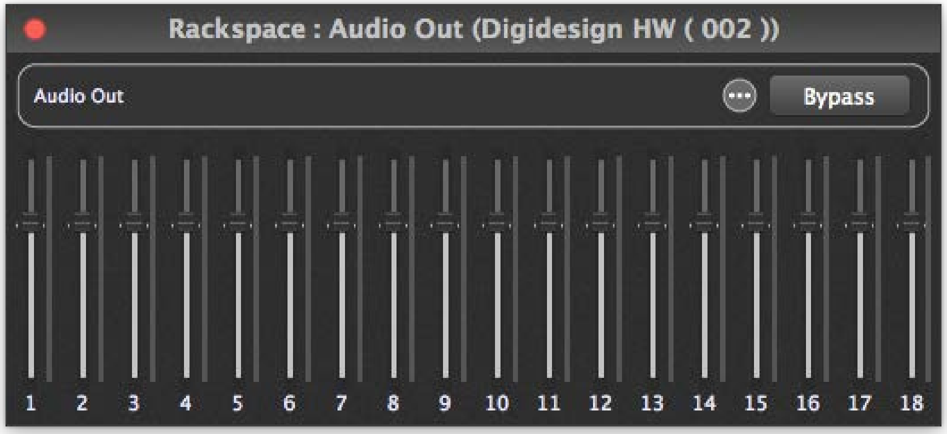
- Double-clicking any gain or balance control in a supplied plugin causes it to return to zero. This does not work on widgets.

Audio In



The Audio In plugin represents the current connected physical interface used to receive audio. The actual number of sliders displayed is a function of the capabilities of the interface connected to your system. For example, in the graphic above, 18 sliders are shown since those are the number of inputs (16 individual, plus stereo) provided by the connected Digi 002 interface.

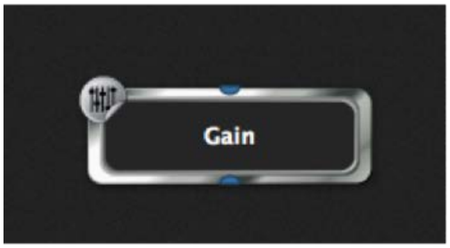
Audio Out

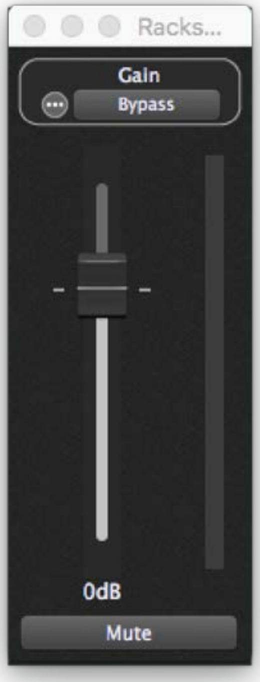


The Audio Out plugin represents the current connected physical interface used to send audio. The actual number of sliders displayed is a function of the capabilities of the interface connected to your system. For example, in the graphic above, 18 sliders are shown since those are the number of outputs (16 individual, plus stereo) provided by the connected Digi 002 interface.

Important note: Audio In and Audio Out plugins cannot be removed from your rackspace. However, you can specify different audio interfaces for Audio In and Audio Out.

Gain Control





This plugin provides a single audio input pin and a single audio output pin. It allows the gain of a mono audio signal to be adjusted.Clicking on the **Mute** button or setting the **Volume** slider to 0 causes the plugin to be muted (not bypassed). Click here for more information.

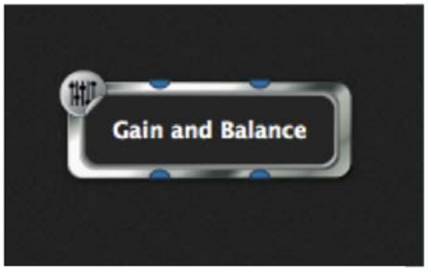
Gain Control widget control

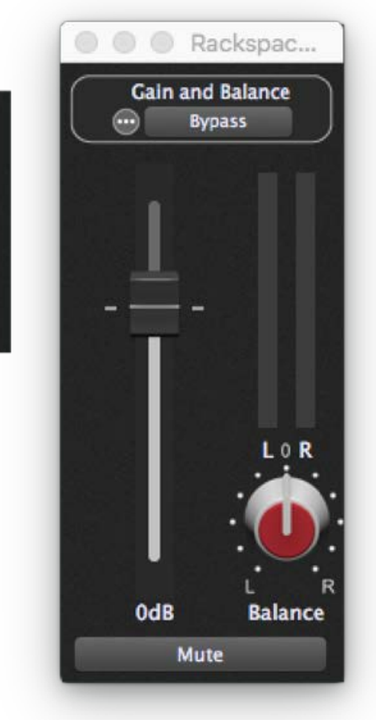
The following parameters for the Gain Control plugin can be widget-controlled:

- Gain
- Balance
- Magnitude

The magnitude parameter is used for level metering. Click here for more information.

Gain and Balance





This plugin provides stereo audio input pins and stereo audio output pins. It allows the gain and Left/Right balance of a stereo audio signal to be adjusted.Clicking on the **Mute** button or setting the **Volume** slider to 0 causes the plugin to be muted (not bypassed). Click here for more information.

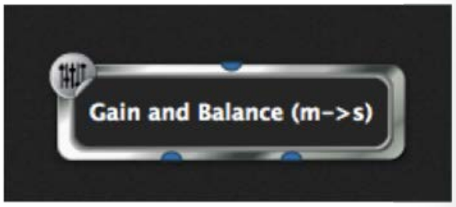
Gain and Balance widget control

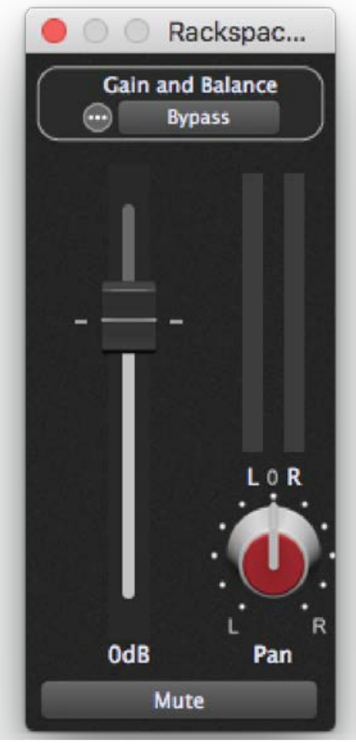
The following parameters for the Gain and Balance plugin can be widget-controlled:

- Gain
- Balance
- Channel 1 Magnitude
- Channel 2 Magnitude

Magnitude parameters are used for level metering. Click here for more information.

Gain and Balance (m->s)





This plugin provides a single audio input pin and stereo audio output pins. It converts a mono audio signal to stereo and allows its gain to be adjusted. Clicking on the **Mute** button or setting the **Volume** slider to 0 causes the plugin to be muted (not bypassed). Click here for more information. A balance control enables the incoming mono source to be pannedacross Left/Right channels.

Gain and Balance (m->s) widget control

The following parameters for the Gain and Balance (m->s) plugin can be widget-controlled:

- Gain
- Balance
- Channel 1 Magnitude
- Channel 2 Magnitude

Magnitude parameters are used for level metering. Click here for more information.

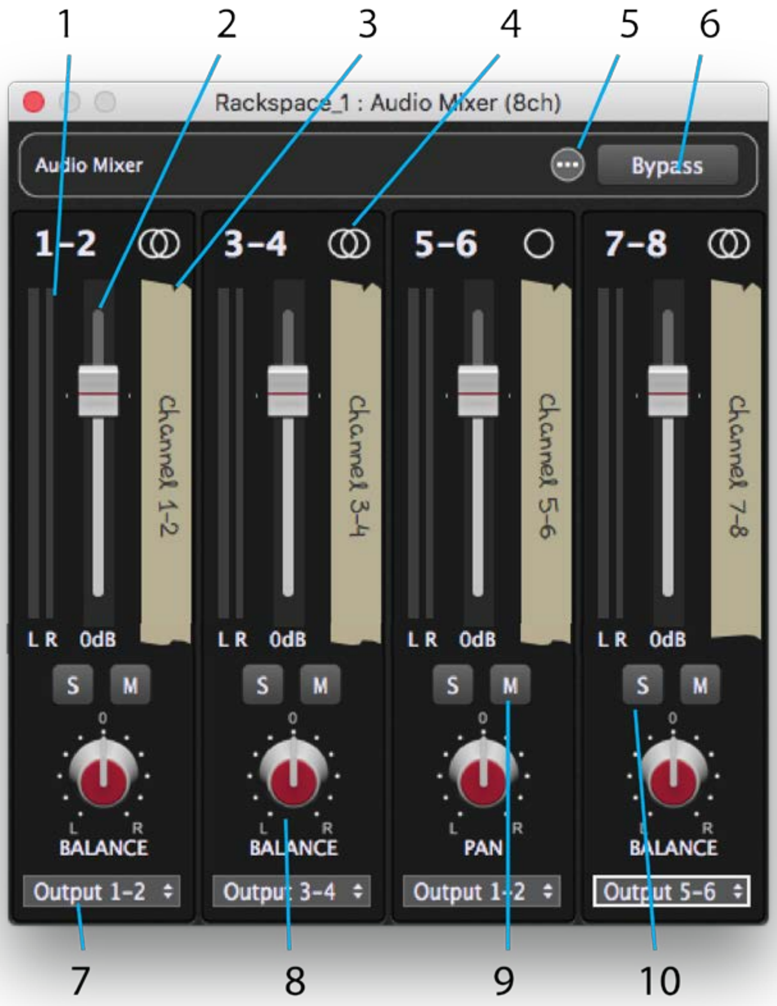
If you connect the audio outputs of multiple plugins to the input of another plugin, the signals will be mixed automatically. To create custom blends, insert one of the Audio Mixer/Router plugins and balance as desired. Audio Mixer/Router plugins can also be used to create **reverb sends**, where signal is routed to a reverb (or other type of signal processing plugin) from multiple sources. Simply insert an Audio Mixer/Router plugin and route the output of each source to its inputs and a pair of outputs to the reverb plugin.

Audio Mixer/Router

The Audio Mixer/Router is a simple but powerful function available as a four, eight, or 16-channel plugin. Inputs and outputs are in stereo pairs, so the four-channel plugin has two channel strips, the eight-channel plugin four channel strips, and the 16-channel version eight strips.

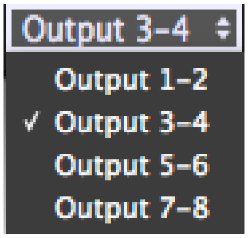
Each channel strip can be switched between mono (left channel only) or stereo input, and assigned to any of the four output channel pairs. All channel strip functions can be attached to widgets.

Because the Audio Mixer/Router combines several basic functions, with widget control of any of them, it can be configured to meet pretty sophisticated needs. For applications ideas and information, see Audio Mixer applications.



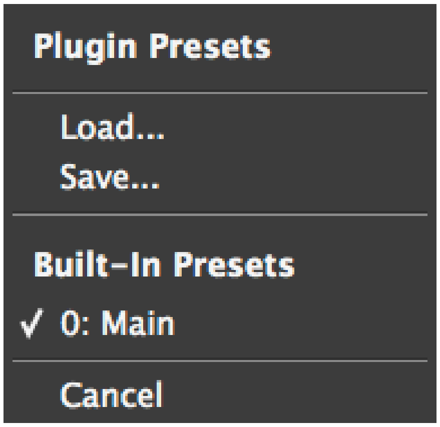
1. **Channel level meter –** Displays the post-fader output level from the channel.
2. **Channel level fader –** Adjusts level from the channel to the selected output pair. Option/Alt-click the fader to return it to the0 dB position. Control-drag to adjust level in 0.1 dB steps.
3. **Channel label –**To edit a channel name, double-click the “tape strip.”
4. **Mono/Stereo channel switch –** Click this icon to toggle the channel strip between mono (left channel) and stereo input.
5. **Presets menu –** Mixer/router setups can be stored as presets. Click this button for a popup menu with commands for storing and loading mixer/router presets.
6. **Mixer bypass –** Bypasses the mixer. Inputs 1 and 2 are routed to outputs 1 and 2; other inputs are not routed to outputs.

7. **Output channel select –** Click this field to choose an output pair for a strip from the dropdown menu.



8. **Channel Balance/Pan knob –** For stereo channels, this is a left/right balance control, while on mono channels it is a pan control. Option-click (Mac) or Control-click (Win) on this control to return it to center.
9. **Channel Mute button –** Mutes the channel.
10. **Channel Solo button –** Solos the channel, effectively muting all others.

Presets menu



Load preset– Loads a previously stored mixer preset.

Save preset– Saves the current state of the mixer as a preset file with a .gpp_internal extension.

Built-in Presets – Whatever preset information the individual plugin makes accessible to Gig Performer is displayed here. Often, there will be only one preset showing, which represents the preset currently running in the plugin.

Cancel –This simply exits the menu, just as clicking outside of the menu does.

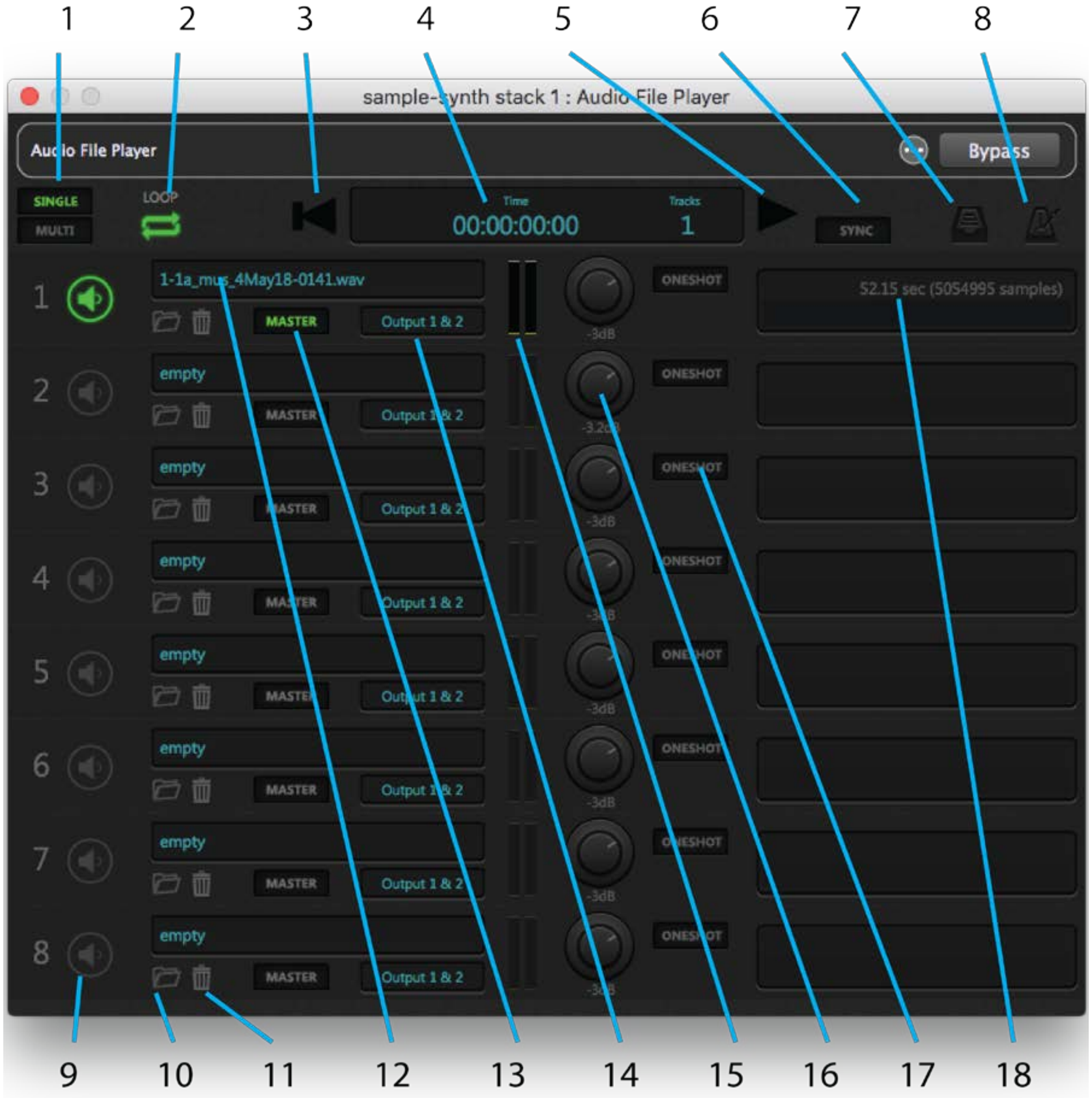
Widget Control

The following settings for each channel can be attached to widgets:

- **Volume**
- **Balance/Pan**
- **Solo**
- **Mute**
- **Output**
- **Left meter**
- **Right meter**

The plugin **Bypass** button can also be attached to a widget.

Audio File Player



The Audio File Player plugin provides a MIDI input pin,eight audio output pins, and a MIDI output pin. It allows you to load and play up to eight audio files (in AIF, WAV or MP3 formats), either one at a time or simultaneously. You can use this plugin as a backing track player or for looping short samples, switching seamlessly from one to another while running. Up to eight audio output channels are available, singly or in pairs.

The Audio File Player plugin provides eight "lanes" for playback. You can think of these as roughly equivalent to tracks on a DAW. Its editor provides controls that enable the following advanced operations:

1. **Single/Multi** – Switch between playing back one audio file at a time versus multiple files simultaneously.

2. **Loop** – Turns looping on or off for all lanes.

3. **Go To Beginning** – Instantly go to the beginning of all files.

4. **Time/Tracks display** – Displays the current time, in hours/minutes/seconds/hundredths of a second, as well as the number of tracks (channels) loaded with audio files.

5. **Play/Pause** – Start or pause file playback. (**Note**: The "pause" function does not return the file to the beginning; it simply pauses it wherever it is. Clicking on the button a second time resumes playback from the point at which it was stopped.)

6. **Sync**–When turned on (lit green), Gig Performer will play and pause the Audio File Player automatically when its button is engaged. This allows you to begin playback of audio files at the same time you start a drum machine emulation plugin, for example.

7. **Load Full Directory** – Loads a full directory of audio files at once, up to the number of available (empty) lanes. If, for example, you have six empty lanes and you try loading eight files using this button, only the first six files will get loaded, into lanes 3 – 8. With this feature, you can create a directory with all of the files you will need, then load the entire player with them at the gig by invoking only a single command.

8. **Metronome**– Clicking on this icon provides a number of additional controls that allow you to create a MIDI click track in sync with the tempo of your Master track. See the MIDI metronome information for more details.

In addition, the Audio File Player editor offers the following lane-specific controls:

9. **Play Enable**–Enables or mutes playback of the file loaded into the lane.

10. **Load New Audio File** (file cabinet icon) – Click this to load an audio file into the lane.

11. **Remove Audio File** (trashcan icon) – Click this to delete the audio file from the lane.

12. **File name**– Displays the name of the file currently loaded in the lane.

13. **Master indicator** – When lit, indicates that the file in the lane is being used as the Master, determining the playback length of all other files. Note that this is not a parameter that can be changed; instead, Gig Performer automatically designates the lane loaded with the longest audio file as the Master. If you load other lanes with new files, the Master designation may change accordingly (if the new files are longer than the one in the currently designated Master lane).

14. **Output assignment** – Allows you to set the output for that lane's playback. A stereo file should ideally be set to a pair of outputs (i.e., Output 1&2, Output 3&4, etc.) but if it is sent to a mono output, the two channels will automatically be mixed to mono. Conversely, sending a mono file to a pair of outputs will result in the same signal being routed to both outputs.

15. **Level indicators** – Shows the current playback level for that lane.

16. **Volume control** – Allows you to adjust the volume of the lane.

17. **Oneshot on/off** – Allows you to turn looping off for that lane, even if the master Loop button is on (lit). Clicking on the Oneshot button (causing it to light) during playback will cause the file in that lane to play to the end and then stop. If you then turn Oneshot for that lane off again, the file will resume playback when the file loaded into the Master lane ends.

18. **File duration**– Displays the duration in seconds and samples of the file currently loaded in the lane.

Basic Audio File Player Operation

To load audio files file, do any of these:

- Click the Load New Audio File button in a lane and navigate to the desired file.

- Drag and drop the file from the OS X Finder or Windows Explorer onto the lane into which you want it loaded.

- Click the Load full directory button and navigate to a source folder of audio files to be loaded in the Audio File Player.

The Audio File Player allows files of different sample rates to be loaded into separate lanes and played simultaneously.

To play audio files:

To play a single audio file, be sure the Single/Multi button is set to single and that the lane is play-enabled, then click the Play/Pause button to start playback.

To play multiple files simultaneously

1. Load all of the files you want to play.
2. Set the Single/Multi button to Multi. This allows you to enable or mute multiple lanes.
3. Set muting for tracks you don't want to hear.
4. Click the Play/Pause button to start playback.

Things to note about the Audio File Player

- The Audio File Player designates the longest file loaded as the Master, whether or not it is play-enabled. The duration of the Master track becomes the loop length when loop playback is enabled.

- The MIDI metronome is generated based on the length of the Master track, and information you supply about the number of bars in the loop and the number of beats in each bar.

- Lanes can be muted and enabled in real-time while the player is in playback. As this is muting, not stopping playback, all files always stay in sync with the master track's location, so if a lane is muted during playback and then enabled 10 seconds later, it resumes playing 10 seconds later than the point at which it was muted.

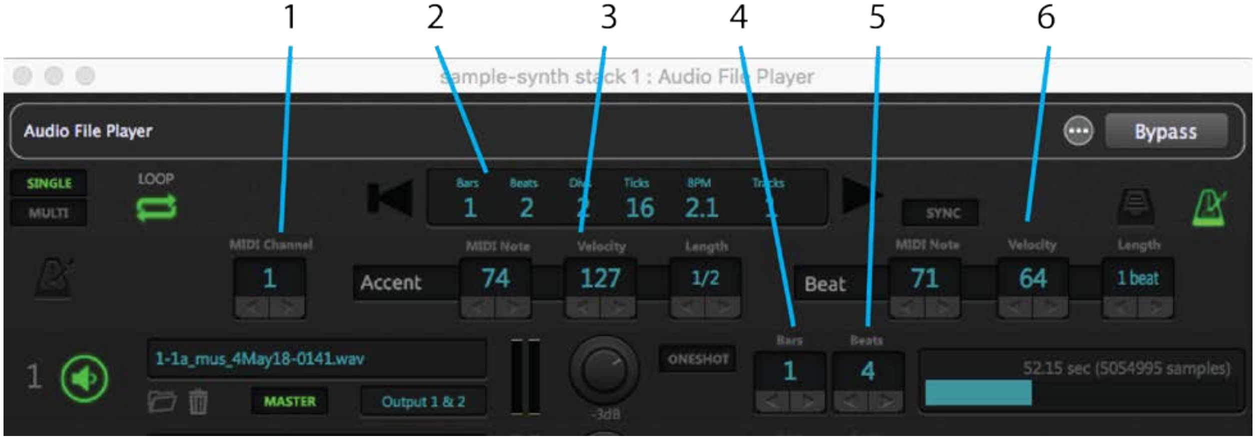
- When a file is muted, you will still be able to see "ghost" meter activity for it.

- Beneath the file duration on the right of each lane is a bar that turns blue as the file plays. The length of the bar represents the full duration of the file in that lane and the progress through it of the play cursor. So, while working in Multi mode, in 10 seconds, the blue bar in a lane with a 20 second-long file is filled to one half, where as the blue bar for a two-minute loop running at the same time in another lane is only filled 1/12th of the way across, just a small chunk.

- To move the playhead to locations in the Master file other than the start, create a knob or slider widget and map it to the Playhead position parameter of the Audio File Player.

MIDI Metronome

The metronome section of the Audio File Player dialog (shown below) offers a number of additional controls that allow you to create a MIDI click track in sync with the tempo of your Master track. The idea is that you define the length in bars of a loop and the number of beats in a bar, and the MIDI metronome generates MIDI notes for Accent and Beat as defined in the dialog. The MIDI metronome does not do tempo detection, and is unrelated to the audio metronome.



1. **MIDI Channel –** Allows you to enter the MIDI channel you want to use for the click.
2. **Detailed time display** –Automatically derives the tempo from the Master track and displays it in BPM, along with bars/beats/divisions/ticks (the latter is 24 per quarter note, same as MIDI clock).
3. **Accent MIDI Note/Velocity/Length** – Allows you to set the MIDI note you want to use for the clock, along with its velocity and length.
4. **Loop length**– For a metronome to be generated you must specify the length in bars of the loop on the Master track. This field is not part of a time signature or bars/beats location.
5. **Beats per bar**– For a metronome to be generated you must specify the number of beats in each bar of the loop on the Master track. This field is not part of a time signature or bars/beats location.

6. **Beat MIDI Note/Velocity/Length** –Allows you to set the MIDI note you want to use for the accent (played on the first beat of each bar), along with its velocity and length.

> **Note:** MIDI click tracks work wellwhen loops are perfect, as will be the case with most commercially purchased loops. However, if the loop in your Master track is not perfect, or if the Master track contains an audio file with numerous tempo changes or an inexact rhythm (such as might occur with a live drummer who sometimes slips ahead of or behind the beat), you're probably better off creating an audio click track in your DAW and loading it into one of the lanes so it can play back alongside of the Master track in perfect synchronization.

Audio File Player widget control

The following parameters for the Audio File Player plugin can be widget-controlled:

Play/Pause allows you to start or pause playback from a widget.

Play from beginning causes playback to start from the beginning in response to a widget movement.

Play from marker causes playback to start from the marker position (see below) in response to a widget movement.

LOOP On/Off allows you to turn looping on or off from a widget.

Playhead position allows you to change the playback start point from a widget. Using a continuous widget (such as a knob or a slider) for this purpose enables you to seek particular points or even to "scrub" one or more tracks while they are playing.

Marker position allows you to change the marker position from a widget.

Set Marker now allows you to set the marker position "on the fly" from a widget.

Click Generation On/Off allows you to use a widget to turn the MIDI click track on or off.

Play mode (single/multi) allows you to use a widget to switch between Single and Multi playback.

Lane 1 – 8 Playing allows you to use a widget to start playback from the specified lane.

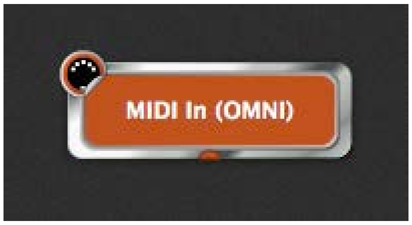
Lane 1 – 8 Oneshot On/Off allows you to use a widget to turn Oneshot mode on or off for the specified lane.

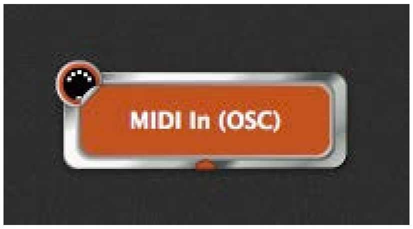
Lane 1 – 8 Volume allows you to use a widget to adjust the volume of the specified lane.

Lane 1 – 8 Output Assignment allows you to use a widget to change the output assignment for the specified lane.

> We recommend the use of on-off widgets, such as one of the LED buttons, for real-time control over most Audio File Player parameters. However, **Playhead position**, **Marker position**, **Lane 1 – 8 Volume**, and **Lane 1 – 8 Output Assignment** are better controlled from continuous widgets such as knobs and sliders.

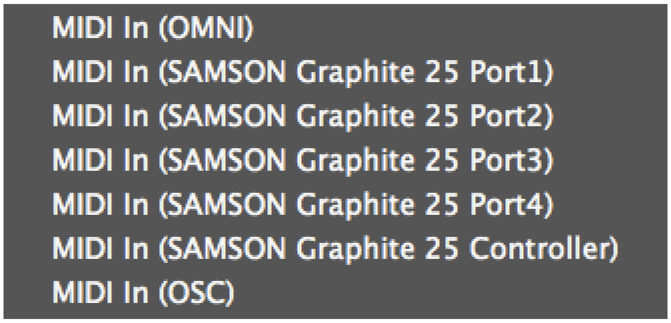
MIDI In (OMNI) and MIDI In (OSC)





These plugins provide a single MIDI output pin and are used to receive incoming MIDI messages and pass them on to other plugins. They are identical, except in one respect: the MIDI In (OMNI) block receives messages from allexternal MIDI devices such as connected MIDI controllers, while the MIDI In (OSC) block receives MIDI messages only from OSC sources. (For more information on OSC, click here.)

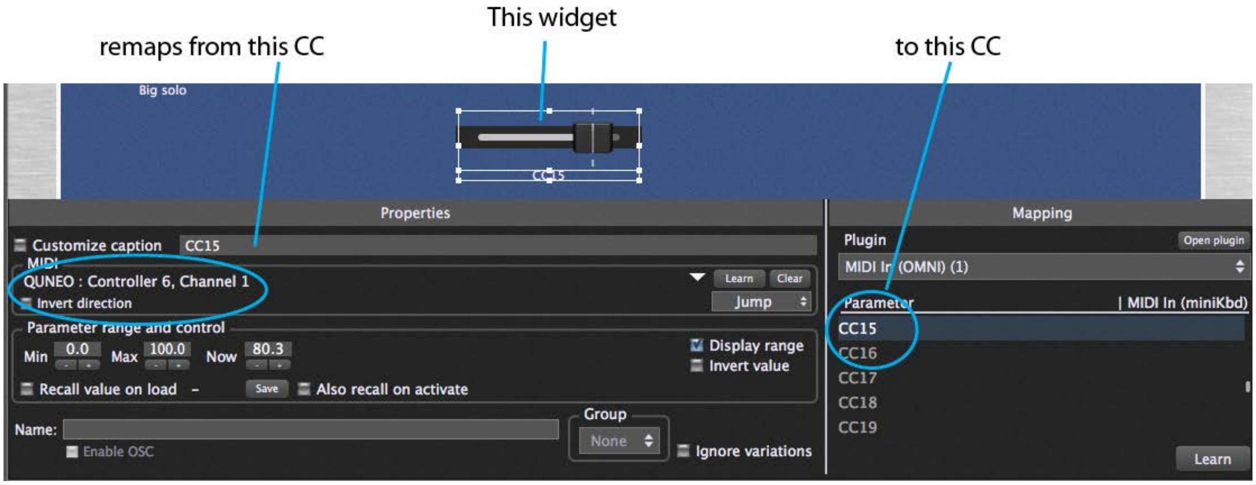
MIDI In (OMNI) is a generic plugin that“listens” to all incoming MIDI messages coming from all connected controllers, on all channels and on all ports. It is the default MIDI input block in any new rackspace, although it can be deleted if not needed (for example, if you are using plugins to process incoming audio signal only). If one or more MIDI controllersare connected to your system, Gig Performer automatically generates additional controller-specific MIDI In blocks when you right-click in the Plugin connections area and select “MIDI Inputs,” as shown below:


MIDI In (OMNI)
MIDI In (SAMSON Graphite 25 Port1)
MIDI In (SAMSON Graphite 25 Port2)
MIDI In (SAMSON Graphite 25 Port3)
MIDI In (SAMSON Graphite 25 Port4)
MIDI In (SAMSON Graphite 25 Controller)
MIDI In (OSC)

These can (and, in most cases, should) be substituted for the generic MIDI In (OMNI) plugin. Because they"listen" to specific MIDI controllers and/or ports, their usage is generally preferable, especially if your rig contains multiple keyboards or controllers.

Like other plugins, a MIDI In block has numerous parameters that can be controlled by widgets. These parameters include any MIDI note or continuous controller message. This makes it possible to remap MIDI messages coming from a physical controller without using the MIDI Filter plugin, like this:

1. In Rack Edit mode, create a new widget.
2. In the widget properties area, choose a MIDI In block for the plugin in the Mapping section.
3. Choose a note or continuous controller from the Parameter list in the Mapping section.
4. Click the Learn button in the MIDI section and move the physical controller.



This widget

remaps from this CC

to this CC

Big solo

CC15

Properties

☐ Customize caption CC15

MIDI

QUNEO : Controller 6, Channel 1

☐ Invert direction

Jump

Parameter range and control

Min 0.0 Max 100.0 Now 80.3

☑ Display range

☐ Invert value

☐ Recall value on load –

☐ Also recall on activate

Name:

☐ Enable OSC

Group None

☐ Ignore variations

Mapping

Plugin

MIDI In (OMNI) (1)

Parameter | MIDI In (miniKbd)

CC15

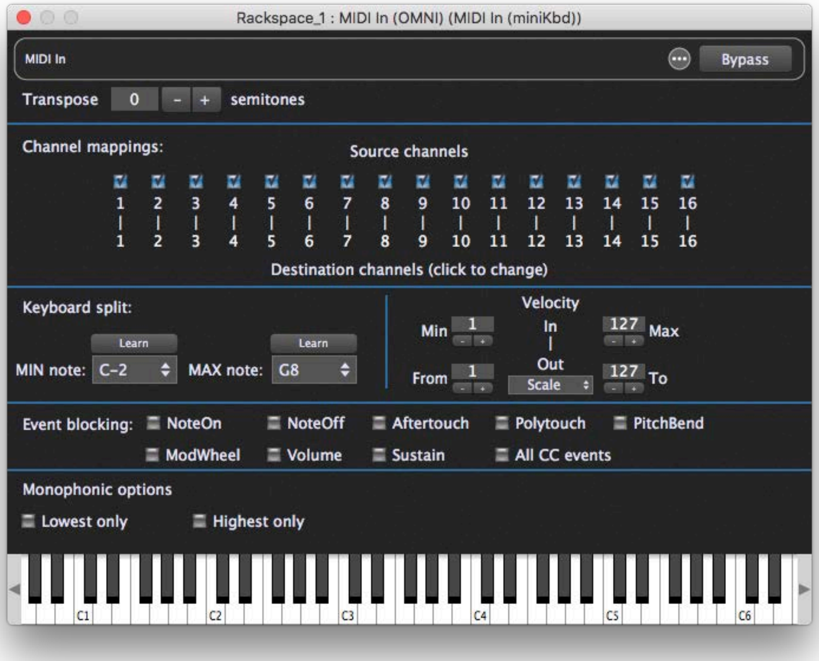
CC16

CC17

CC18

CC19

Incoming messages from the physical controller will now get remapped to the note or continuous controller you chose. Of course, this incoming CC could as easily be remapped to pitchbend, aftertouch, or mod wheel messages.



Rackspace_1 : MIDI In (OMNI) (MIDI In (miniKbd))

MIDI In

Transpose 0 semitones

Channel mappings:

Source channels

☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Destination channels (click to change)

Keyboard split:

MIN note: C-2 MAX note: G8

Velocity

Min 1 In 127 Max

From 1 Out 127 To

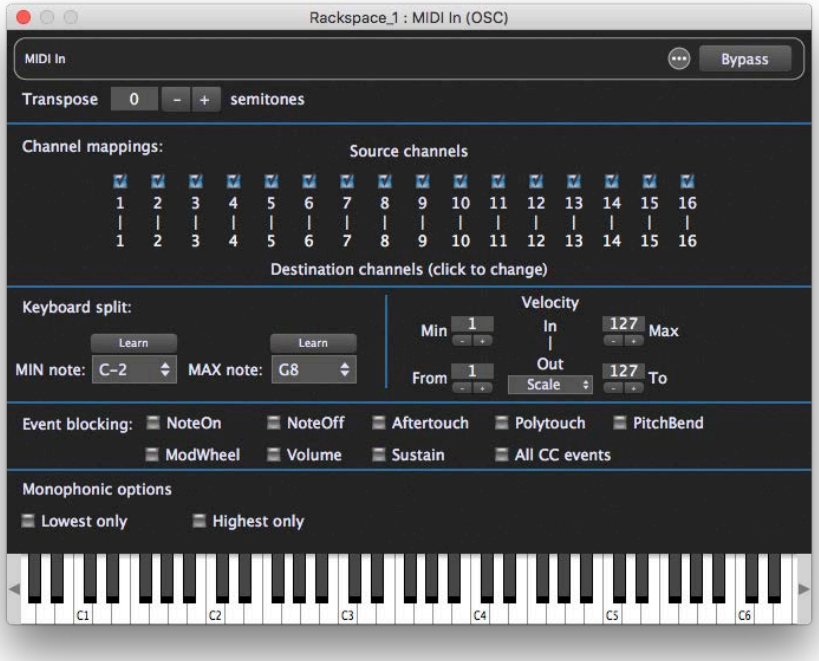
Scale

Event blocking: □ NoteOn □ NoteOff □ Aftertouch □ Polytouch □ PitchBend

□ ModWheel □ Volume □ Sustain □ All CC events

Monophonic options

□ Lowest only □ Highest only



Rackspace_1 : MIDI In (OSC)

MIDI In

Transpose 0 semitones

Channel mappings:

Source channels

☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Destination channels (click to change)

Keyboard split:

MIN note: C-2 MAX note: G8

Velocity

Min 1 In 127 Max

From 1 Out 127 To

Scale

Event blocking: □ NoteOn □ NoteOff □ Aftertouch □ Polytouch □ PitchBend

□ ModWheel □ Volume □ Sustain □ All CC events

Monophonic options

□ Lowest only □ Highest only

The MIDI In plugin editors provide controls that enable the following advanced operations:

Key transposition

Transpose incoming note messages over a range of +/- 48 semitones.

Channel blocking and mapping

Specified MIDI channels can be blocked (uncheck the box) or rerouted to a different channel (click on the destination channel and enter a different number).

Keyboard splits

Create keyboard splits by specifying MIN and MAX note range values. Notes below the MIN value and/or above the MAX valueare blocked; only notes between those values are allowed through. The MIN and MAX values can be entered in one of these ways:

- Click on the MIN note or MAX note field and choose the desired note from the dropdown menu, OR
- Click the Learn button above the MIN note or MAX note field and play the desired note on your controller. Click the Learn button again to complete the setting.
- Click the Learn button above the MIN note field and shift-click the desired low note on the plugin's keyboard, or click the Learn button above the MAX note field and option/alt-click the desired high note on the plugin's keyboard. Remember to click the Learn button again after the note is learned to exit Learn mode.

It is easy to see the valid split range because the keyboard in the MIDI In block plugin shows the physical key being pressed when you play your controller, rather than the output value, which might be transposed.

Velocity layering

Velocities below the MIN value and/or above the MAX value are blocked; only velocities between those values are allowed through. If the same value is entered for MIN and MAX, all incoming notes will have their velocity changed to that value. This allows you to create velocity layers by specifying whether notes are heard depending on how hard (or soft) you press a key.

Velocity scaling

By selecting “Scale” in the dropdown box and entering “From” and “To” values, the velocity of all incoming note messages areadjusted so as to reduce their dynamic range (i.e., softer notes will be heard louder, or louder noteswill be heard more softly).

Using velocity layering with velocity scaling

The MIDI In plugin editor’s velocity range control is particularly useful if you want to have multiple sounds available but hear only some when you play softly and others when you play harder. If connected to a piano plugin, for example, you can opt to only hear the piano sound when you play the keyboard forcefully.

In addition, after you have defined a velocity range, you may want to change the actual velocities that are sent to your sound-generating plugin. For example, you may have created a piano sound that you only want to be heard if the velocity value is greater than 80. However, you may still want the piano to respond as if it was being played quietly. In such a case, set the Min/Max velocity values to 80 and 127 respectively so that you only hear the piano when you hit the keys hard. Then choose Scale and set the From/To values to 1 and 50 respectively. As a result, notes received with a velocity of 80 will produce notes with velocity of 1, while notes received with a velocity of 127 will produce notes with a velocity of 50. Notes in-between those values will be mapped somewhere in between.

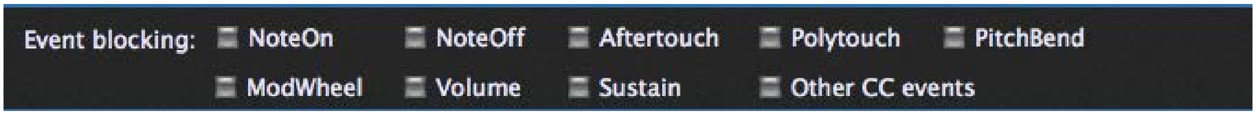
Velocity constraining

By selecting “Constrain” in the dropdown box, any incoming note messages with a velocity below the “From” value are given that value exactly; at the same time, any incoming note messages with a velocity above the “To” value are given that value exactly. For example, if you enter a “From” value of 60, and a “To” value of “80,” all incoming note messages with velocities of 60 or less are given a velocity of 60 exactly, while all incoming messages with velocities of 80 or more are given a velocity of 80 exactly; only velocity values from 61 – 79 are preserved. Constraining allows you to create hard velocity splits.

Event Blocking: Note and CC blocking

Sometimes it is desirable to block certain note or continuous controller messages coming from a controller. Although the MIDI Filter plugin is capable of this kind of filtering and much more, MIDI In blocks include an event blocking section that conveniently provides basic blocking capabilities to reduce the number of situations requiring the MIDI Filter plugin.

A common scenario is your controller has one or more knob, button, or other physical controls that are not being used, and so not assigned to a plugin parameter. If this unassigned control accidentally gets nudged, it could create an unintentional change to a plugin value. Blocking the MIDI messages coming from that control prevents this accidental modification.



Event blocking: □ NoteOn □ NoteOff □ Aftertouch □ Polytouch □ PitchBend □ ModWheel □ Volume □ Sustain □ Other CC events

The Other CC events box, when checked, blocks all continuous controller messages except for the three that have their own boxes: ModWheel (CC1), Volume (CC7), and Sustain (CC64).

Monophonic options

The Monophonic options area at the bottom of the editor filters the note messages from the controller down to a monophonic stream of notes.

The Lowest only box, when checked, produces a mono part extracted from the lowest note as played on the instrumemt.

The Highest only box, unsurprisingly, produces a mono part extracted from the highest note as played on the instrumemt.

These options can be useful for doubling melodies, or extracting a bass part from what might have been written as a piano part.Most often, users choose to check one or the other of these boxees, but you could check both, in which case you would be extracting lines from both the top and bottom of what is being played.

Playing plugins without an external MIDI controller

A virtual keyboard is provided to allow sound-generating plugins to be played even when an external MIDI controller is not connected.

MIDI In widget control

The following parameters for MIDI In plugins can be widget-controlled:

- Bypass plugin
- Channel 1 – 16 Output

- MinNote (minimum note)
- MaxNote (maximum note)
- Transpose
- StopAllNotes
- StickyNoteChannel
- StickyNoteVelocity
- Note On C-2 through G8 (MIDI note numbers 0 – 127)
- MinNoteVelocity (minimum note velocity)
- MaxNoteVelocity (maximum note velocity)
- MinScaledNoteVelocity (minimum note velocity)
- MaxScaledNoteVelocity (maximum note velocity)
- OutputVelocityMode
- PC (Program Change)
- Continuous controller (CC) 0-127
- Pitchbend
- Aftertouch
- MonoLowestOnly
- MonoHighestOnly
- BlockNoteOn
- BlockNoteOff
- BlockPitchbend
- BlockAftertouch
- BlockPolytouch
- BlockControllers
- BlockModWheel
- BlockVolume
- BlockSustain

Channel 1 – 16 Output allows you to change MIDI channel blocking or remapping in real time.

MinNote/MaxNote allow altering the MinNote or MaxNote settings in real time.

Transpose allows you to alter the key transposition in real time.

Caution: Do not change this setting while holding notes, or stuck notes may result. If you do get a stuck note, don't panic—just click on the Panic icon to turn all notes off.

“StopAllNotes”issues a “turn all notes off” message, along with a sustain OFF message and a Center PitchBend message, same as the Panic icon. One usage is to turn off MIDI notes when switching variations. Although you normally wouldn’t do this (because the whole point of variations is to allow you to continue holding notes down while changing plugin parameter values), it is useful if you want to use variations to actually change the underlying sound you’re creating, in which case any notes from the previous sound need to stop immediately once you change variations. Just associate this parameter with a widget (typically, a button), and it will trigger automatically when you change variations.

Sticky notes allow you to trigger loops or sample playback from a widget. It works by sending Note On events only, as if you played a note but didn’t release it. To use sticky notes, create an on-off widget (a switch or button) and assign it to the MIDI In plugin and a Note On parameter. Then add a second continuous (knob or slider) widget and assign it to the MIDI In plugin as well, but to the StickyNoteVelocity parameter. When you activate the switch or button widget, a note will play continuously in the connected sound-generating plugin, with the knob widget determining the velocity of the MIDI note message. (You can also add a widget assigned to StickyNoteChannel to select the MIDI channel.) When the button widget is turned off, an appropriate note off event will be sent to the plugin to stop the sound.

MinNoteVelocity/MaxNoteVelocity and **MinScaledNoteVelocity/ MaxScaledNoteVelocity** allow you to alter these settings, respectively, in real time.

OutputVelocityMode allows you to change between Scaled and Constrained velocity.

PC (Program Change)allows widget movements to send MIDI program change messages to plugins to call up different presets. Note that doing so prevents Patch Persist from working and can result in stuck notes depending on the plugin implementation. *For those reasons, this feature is not recommended for normal use.*

Continuous controller (CC) 0-127 – Incoming MIDI messages to the widget are remapped to the specified CC number.

Pitchbend – Incoming MIDI messages to the widget are remapped to pitchbend messages

Aftertouch – Incoming MIDI messages to the widget are remapped to mono aftertouch messages.

The following parameters switch the indicated checkboxes:

Monophonic options

MonoLowestOnly

MonoHighestOnly

Event blocking

BlockNoteOn

BlockNoteOff

BlockPitchbend

BlockAftertouch

BlockPolytouch

BlockControllers

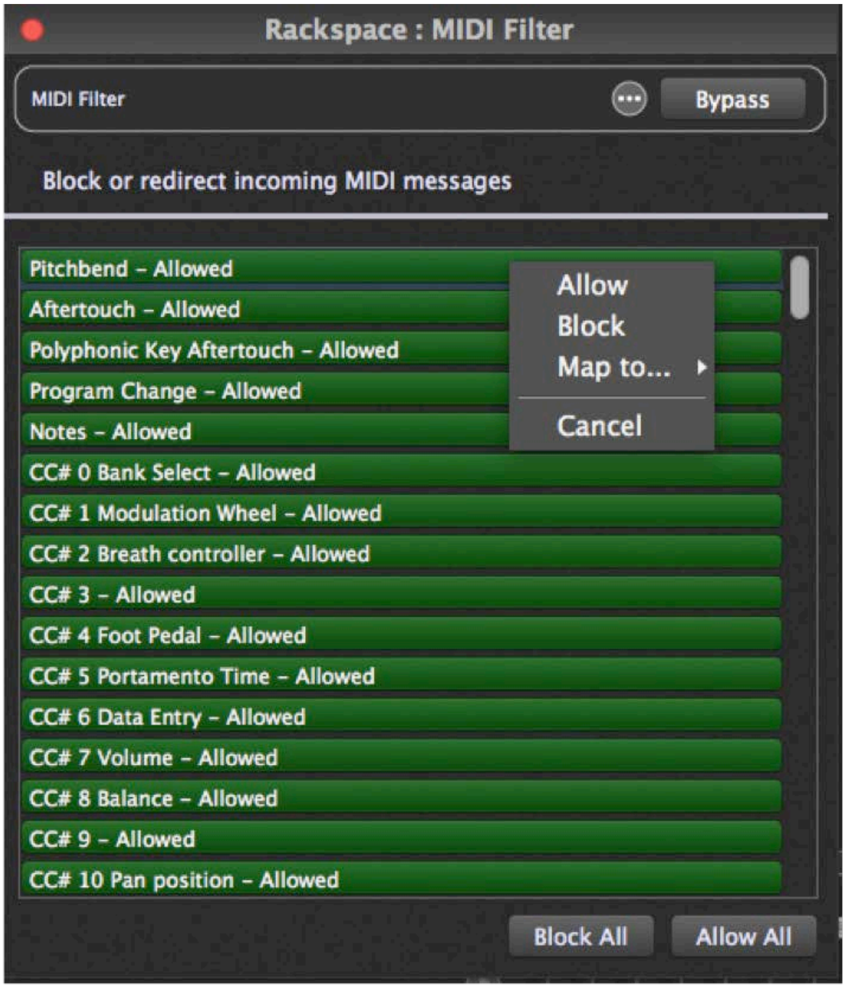
BlockModWheel

BlockVolume

BlockSustain

Note: If you have an audio interface or MIDI controller connected to your system that provides a MIDI output, Gig Performer will expose that output with a corresponding MIDI Out block. Click here for more information.

MIDI Filter



Block or redirect incoming MIDI messages

Pitchbend – Allowed
Aftertouch – Allowed
Polyphonic Key Aftertouch – Allowed
Program Change – Allowed
Notes – Allowed
CC# 0 Bank Select – Allowed
CC# 1 Modulation Wheel – Allowed
CC# 2 Breath controller – Allowed
CC# 3 – Allowed
CC# 4 Foot Pedal – Allowed
CC# 5 Portamento Time – Allowed
CC# 6 Data Entry – Allowed
CC# 7 Volume – Allowed
CC# 8 Balance – Allowed
CC# 9 – Allowed
CC# 10 Pan position – Allowed

This plugin provides a single MIDI input pin and a single MIDI output pin. It allows specified MIDI messages to be blocked and/or remapped. (MIDI note messages can be blocked, but not remapped.)

Clicking on any line in the MIDI Filter plugin editor causes a popup menu to appear in which you can make one of three choices:

Allowed	Messages of this type are passed through.
Blocked	Messages of this type are blocked.
->	Messages of this type are remapped (converted into messages of another type).

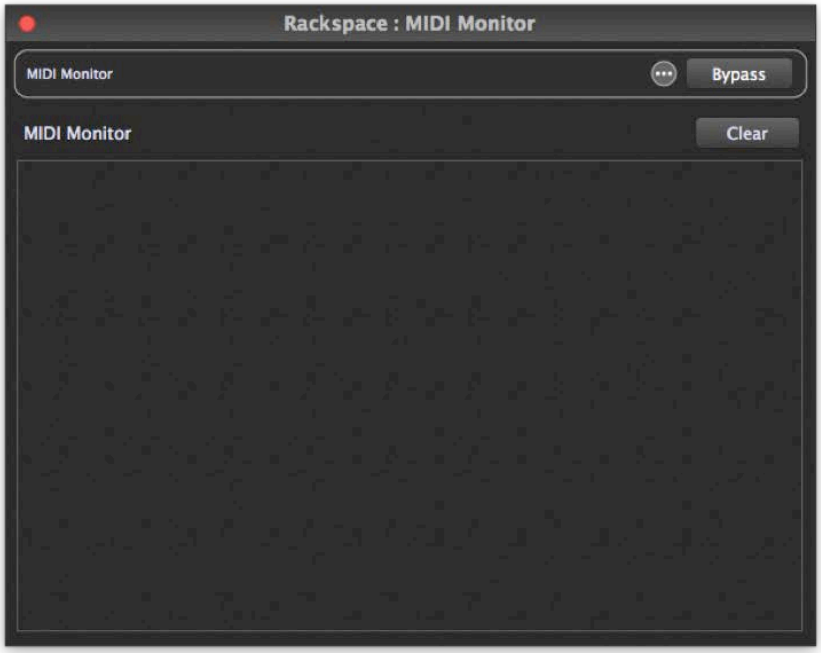
Typically inserted between a MIDI In plugin and a sound-generating plugin, common usages include blocking sustain pedal messages, converting modulation wheel messages into volume messages, and mapping aftertouch messages to mod wheel messages.

There is also an option to block all MIDI messages. If you enable this option, everything, including Notes, will be blocked. This can be very useful if you use multiple variations and want to allow MIDI messages to go to some plugins but not others, depending on which variation is selected. You should, however, exercise caution when doing this in real time as you could get left with a hanging note if you block notes before the Note Off message is sent out. If this occurs, don't panic—just click on the Panic icon to turn all notes off.

MIDI Filter widget control

Every MIDI Filter plugin parametercan be widget-controlled. When using on-off widgets (switches or LED buttons), the selected parameter will be allowed when in the ON position, and blocked when in the OFF position. The use of continuous widgets (knobs and sliders) allows you to scroll through all remapping options in the in-between settings. (The minimum setting will allow the selected message, while the maximum setting will block the message.)

MIDI Monitor

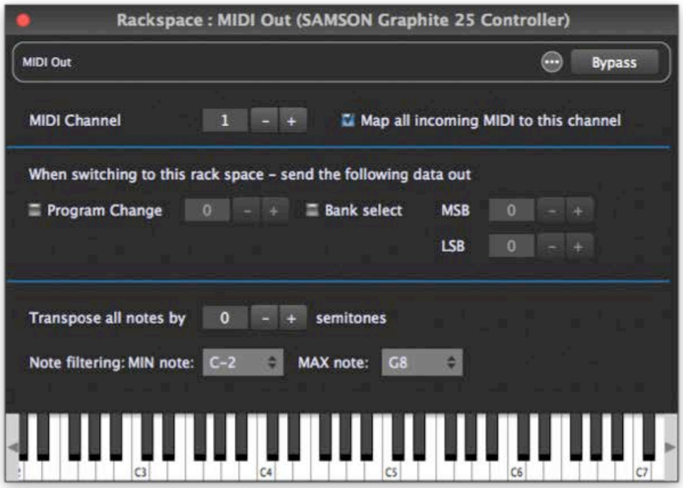


This plugin provides a single MIDI input pin and a single MIDI output pin. It displays MIDI messages from theplugin block to which it is connected and is used primarily for diagnostic purposes.

<u>MIDI Monitor widget control</u>

The MIDI Monitor plugin provides no parameters that can be widget-controlled.

MIDI Out


Rackspace : MIDI Out (SAMSON Graphite 25 Controller)

MIDI Out

MIDI Channel 1 - + ☑ Map all incoming MIDI to this channel

When switching to this rack space – send the following data out

☐ Program Change 0 - + ☐ Bank select MSB 0 - +

LSB 0 - +

Transpose all notes by 0 - + semitones

Note filtering: MIN note: C-2 MAX note: G8

This plugin is created only for connected MIDI controllers that offerone or more MIDI outputs. It provides a single MIDI input pin and is usedto send outgoing MIDI messages to external devices such as synthesizers or samplers.

The MIDI Out editor provides controls that enable the following advanced operations:

MIDI Channel sets the channel of the MIDI output.

Map all incoming MIDI to this channel turnsoff channel mapping so that MIDI data coming from a sequencer retainsits channel information. Note that program change transmission (described below) is still associated with the specified MIDI channel even if this box is checked.

Program Change and **Bank Select** allowGig Performer to automatically transmit a program change and/or Bank Select message every time a rackspace containing the MIDI Out block is activated so that each of your external MIDI devices (keyboard and rackmount synths, drum machines, etc.) load with the correct sounds and are instantly ready for use.

Transpose all notes by allows you to alter the key transposition of the MIDI output.

Note filtering allows you to set up keyboard splits for each MIDI Out block. Notes below the MIN value and/or above the MAX value are blocked; only notes between those values are allowed through.

MIDI Out widget control

The following parameters for the MIDI Out plugin can be widget-controlled:

- ChannelNumber
- MinNote (minimum note)
- MaxNote (maximum note)
- Transpose
- CC #0 – 127
- MapChannel
- PC (Program Change)

ChannelNumber allows you to alter the MIDI output channel number in real time.

MinNote/MaxNote allow you to alter the MinNote or MaxNote settings in real time.

Transposeallows you toalter thekey transposition in real time. **Caution**: Do not change this setting while holding notes, or stuck notes may result. If you do get a stuck note, don't panic—just click on the Panic icon to turn all notes off.

CC allows you to transmit the specified CC (control change) messages as the widget is moved.

MapChannel causes all MIDI events being transmitted to be mapped to a specific MIDI channel.

PC (Program Change) allows widget movements to send MIDI program change messages to external MIDI tone generators in order to call up different presets.

Widget and Rack editing (Edit Mode)

About Widgets and Racks

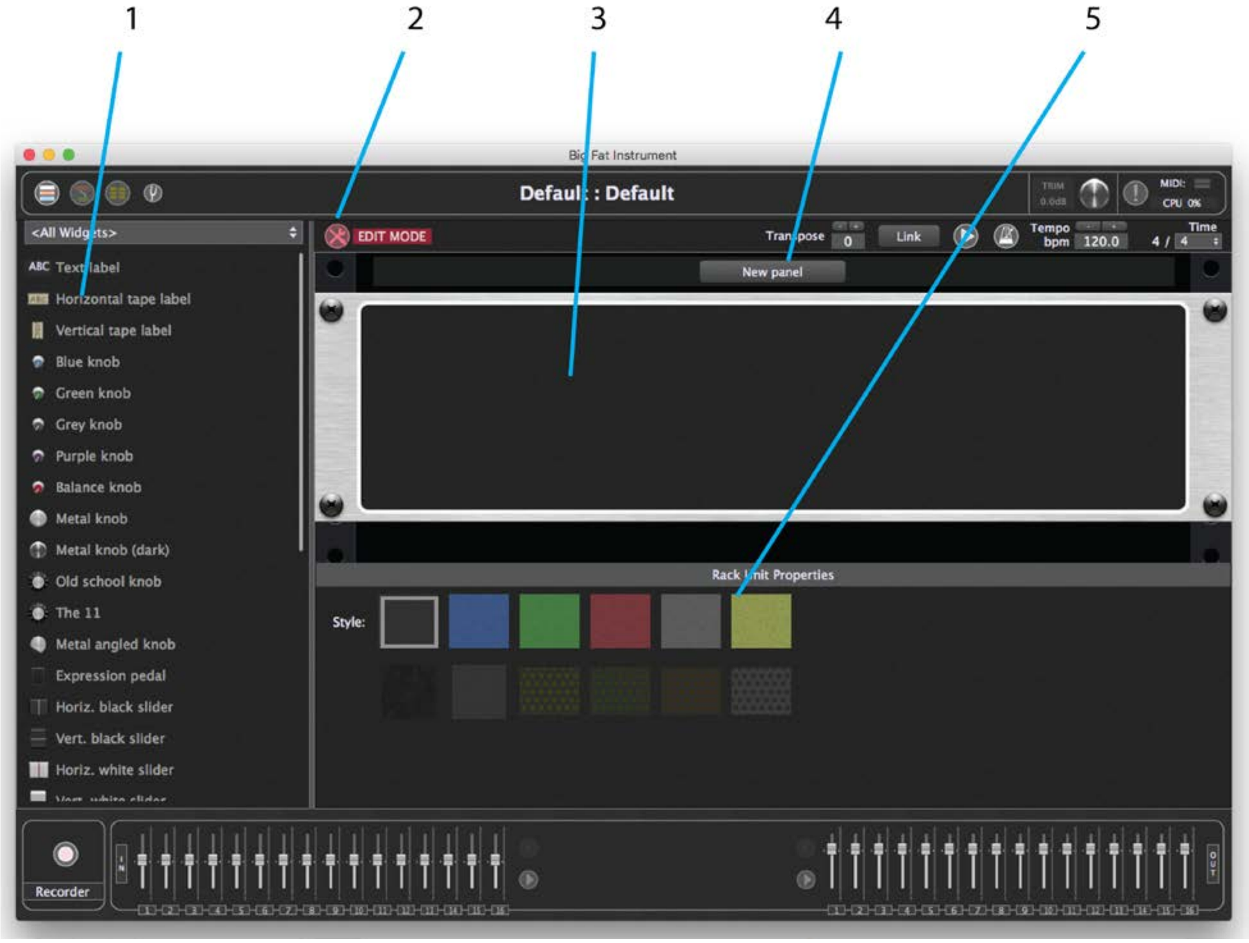
A widget is an onscreen control associated with a plugin parameter. A rack is simply a graphic space within which groups of widgets can be placed and labeled. Users typically create different panels to organize their controls in ways that are simple to identify and deal with in live performance.

A widget can be controlled onscreen or by linking it a physical control on a keyboard, control surface, or other MIDI or OSC controller. Once this mapping of parameters to widgets is done, Rig Manager makes it easy to quickly put all of these assignments under the control of a different physical controller, such as when it is necessary to play someone else's master keyboard for a gig.

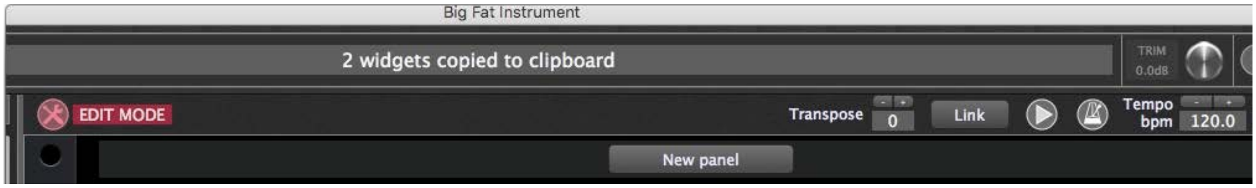
A widget assignment to a parameter will override a global physical assignment to the same parameter.

Edit Mode screen

When you click the **Edit Mode** button, you enter into rack edit if a panel is selected, or widget edit if a widget is selected. Shown here first is rack edit mode. A widget can be added by dragging a widget name from the widget types list to a panel, or by copying and pasting an existing widget. A confirmation message flashes when you copy a widget or panel. MIDI associations are preserved whenever you copy and paste a widget or a panel.



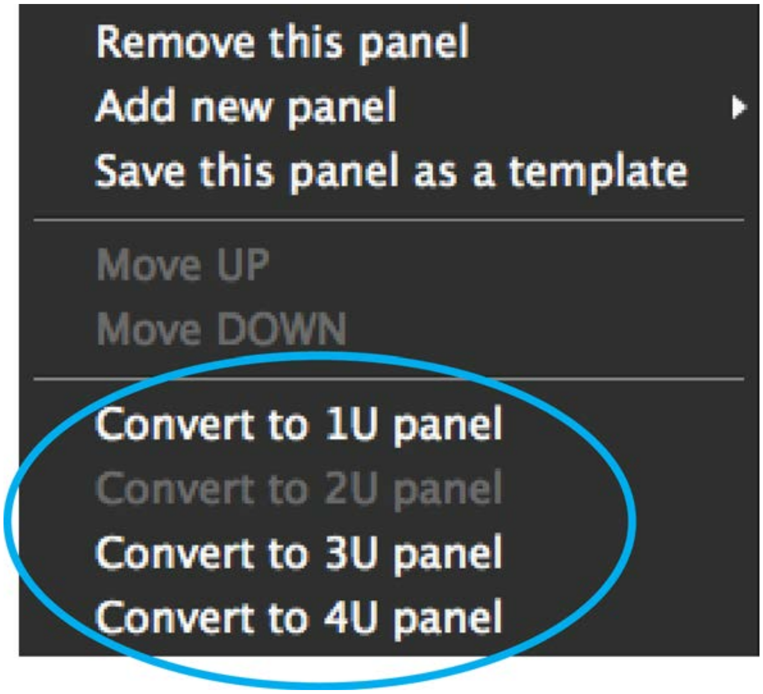
1. **Widget types list**– To add a widget to a panel, simply drag it from this list to the panel. To add multiple copies of a widget at once, type the number of copies you want while dragging the widget from the widget browser. A message will flash at the top of the window to confirm the number of copies that will be made.



2 widgets copied to clipboard

2. **Edit Mode button**– Click this to enter Edit Mode or to exit to Performer mode and use Gig Performer
3. **Panel area –** Panels host all of the widgets. Panels can be set to any of four heights by right-clicking in the panel area. Choose the height you need to

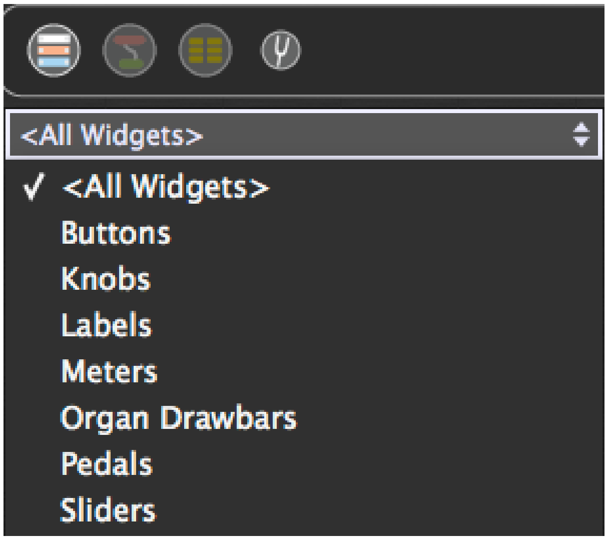
accommodate widgets. For more detail on this menu, see Rackpanel right-click menu.



4. **New Panel button –** Click here to create a new, blank panel in the rack.
5. **Panel backgrounds –**To change a panel's background, click the panel to select it and then click one of the textures displayed in this area.

Widget types list

This list shows all of the available types of widgets. A dropdown menu at the top allows you to filter the list to just one type of widget at a time.



Widget types include:

Labels and indicators– Inactive widgets that simply allow labeling

Knobs of different colors and designs (including our Spinal Tap “The 11”—it goes to 11!) **–** These are all active widgets that provide continuous control.

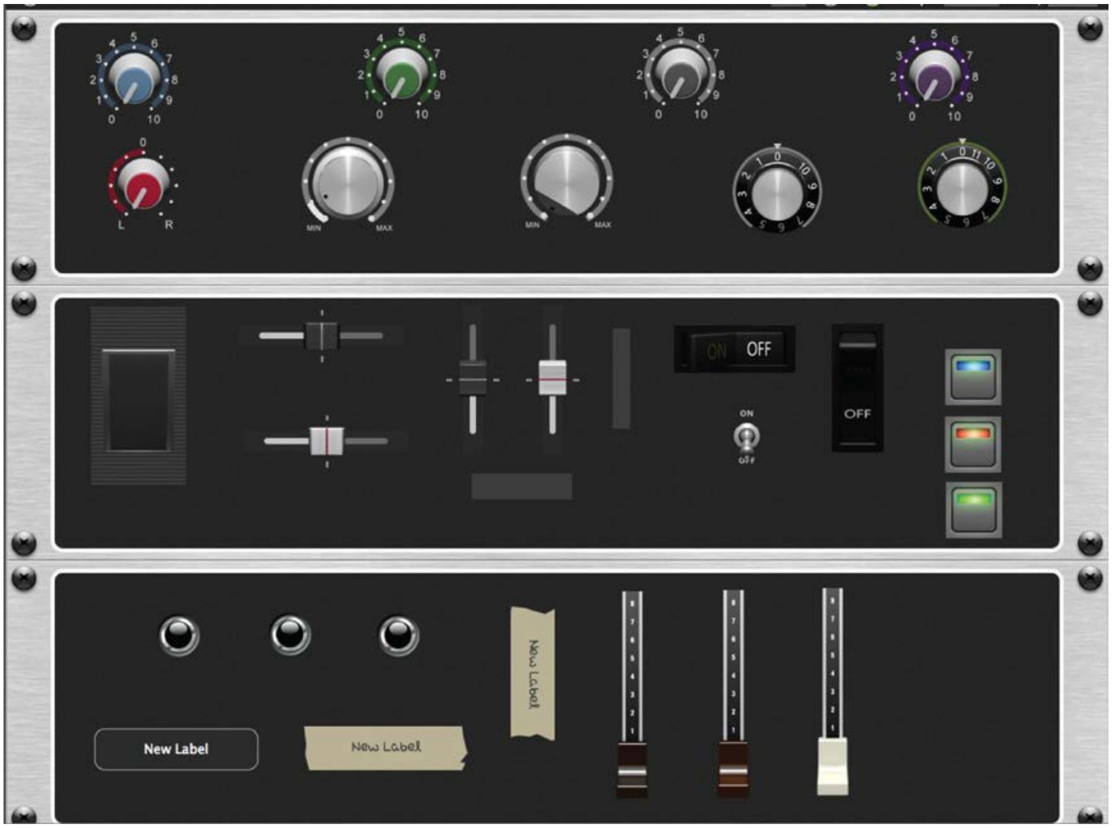
Expression pedal – Another continuous control, visually emulating the movement of an expression pedal.

Sliders and organ drawbars of different colors and designs – Active widgets that provide continuous control.

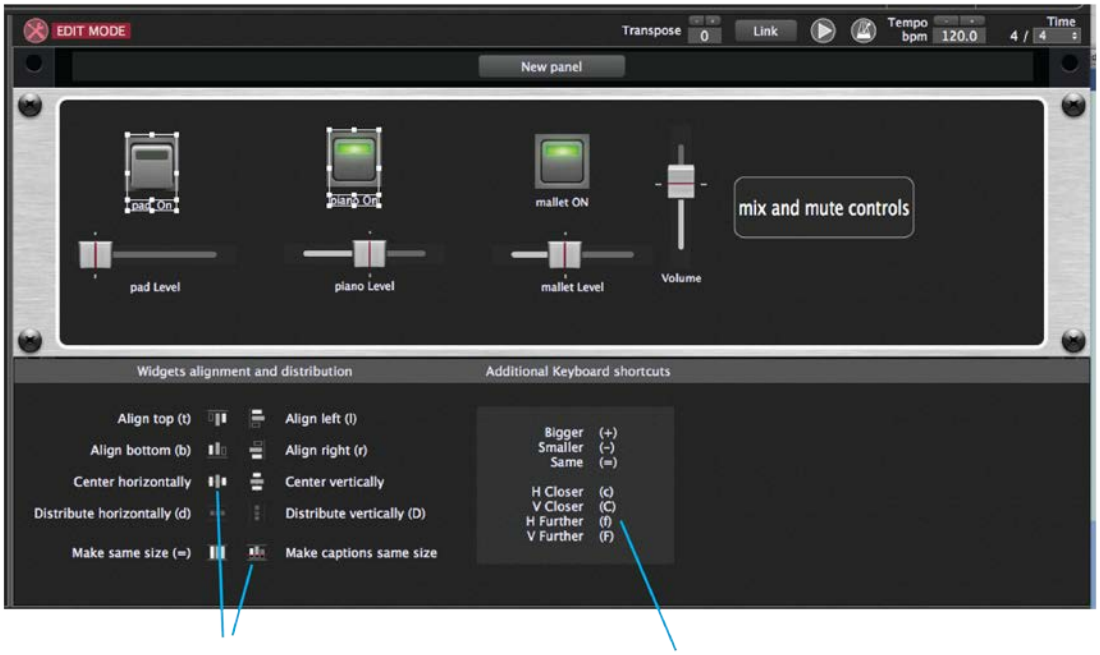
Meters (one vertical and one horizontal) –Used to provide continuous level metering.

Buttons and switches of different colors and designs. **–** All of these active widgets provide simple on-off control.

Here is what all of the widgets look like:



Widget Alignment and Distribution Tools



Alignment and distribution buttons

Widget sizing shortcuts

When multiple widgets are selected, the Widget Properties Inspector shows alignment and distribution tools.

Widgets alignment and distribution

Click the associated button or use the displayed key shortcut to execute the alignment or distribution function on selected widgets.

Align top (t)**–** Align the tops of the selected widgets.

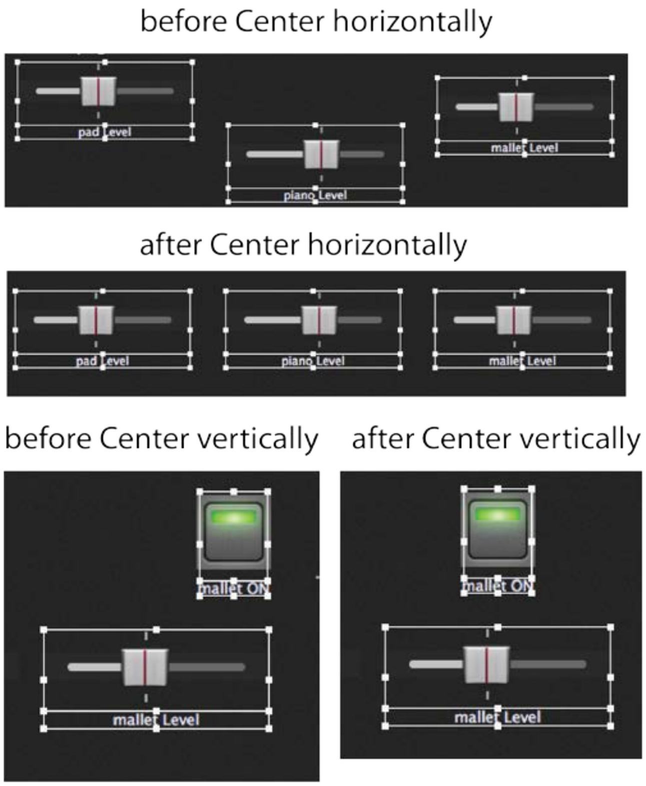
Align bottom (b)**-** Align the bottoms of the selected widgets.

Align left (l) **-** Align the left sides of the selected widgets.

Align right (r) **-** Align the right sides of the selected widgets.

Center horizontally - Aligns selected widgets so that all are lined up on the X axis at the same value of Y.

Center vertically - Aligns selected widgets so that all are lined up on the Y axis at the same value of X.


before Center horizontally

after Center horizontally

before Center vertically

after Center vertically

Distribute horizontally (d)**–** Selected objects are moved to have even horizontal spacing between them.

Distribute vertically (D)**–** Selected objects are moved to have even vertical spacing between them.

Make same size (=) **–** This button makes all selected widgets the same size as the largest one of them.

Make captions same size – This button changes the size of the caption text areas for all selected widgets to matchthe one with the largest text area.

Additional Keyboard shortcuts

These also apply to selected widgets.

Bigger (+) – Grow selected widgets.

Smaller (-) – Shrink selected widgets.

Same (=) – Make all selected widgets the same size as the largest one selected.

H closer (c) – Move the selected widgets closer to each other horizontally (along the X axis).

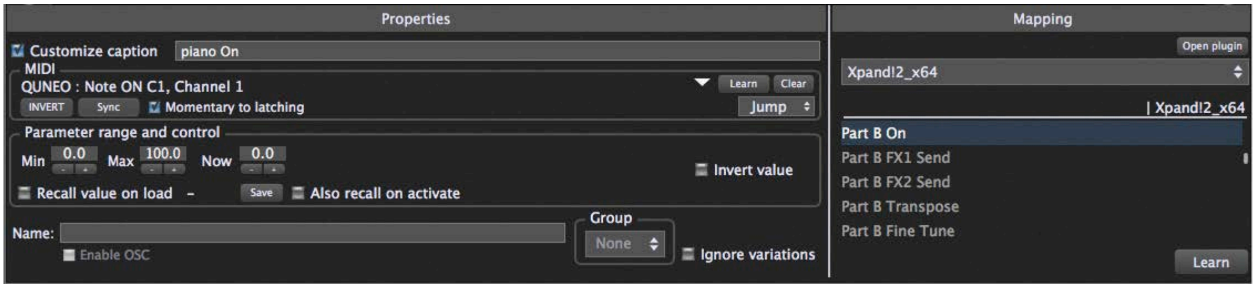
V Closer (C)– Move the selected widgets closer to each other vertically (along the Y axis).

H further (f)– Move the selected widgets further from each other horizontally (along the X axis).

V Further (F)– Move the selected widgets further from each other vertically (along the Y axis).

Widget Properties inspector (Edit mode)

Adding or double-clicking on a widget in Edit mode opens the Widget Properties inspector:



Properties

☑ Customize caption: piano On

MIDI

QUNEO : Note ON C1, Channel 1

☑ Momentary to latching

Jump

Parameter range and control

Min 0.0 Max 100.0 Now 0.0

□ Invert value

□ Recall value on load –

□ Also recall on activate

Name:

□ Enable OSC

Group: None

□ Ignore variations

Mapping

Xpand!2_x64

| Xpand!2_x64

Part B On

Part B FX1 Send

Part B FX2 Send

Part B Transpose

Part B Fine Tune

Properties section

Customize caption – Check to have the widget display a custom caption, then type in a new name. If unchecked, the default caption will be defined by the selected parameter. Normally the caption will be assigned the parameter name only. However, if you hold the Shift key down while selecting a parameter, the caption will include the plugin name as well as the parameter name. If you also hold the Option/ALT key down, the caption will include the plugin name only.

Text label widgets are capable of displaying the current "Now" value. To do so, simply assign it to a plugin and parameter, then check "Customize caption" and enter the term "[value]" (make sure the word "value" is inside square brackets) along with, optionally, a text label name. Whenever that same parameter is altered by another widget, the Text label widget will display the current value. For example, if you have a slider widget controlling the Gain of a plugin, as you move the slider, the value inside the Text label will change.

You can also use a Text label widget to act as a very cool visual cue (to remind you of pretty much anything) when switching between variations. To do so, check "Customize caption" and enter any name, but don't assign the Text label widget to any plugin (select "(no plugin" in the Mapping section of the inspector), then enter a "Now" value. The lower the value, the dimmer the label will be; the higher the value, the brighter it will be!

Learn– Click on this button to have Gig Performer learn the physical controller you want to associate with the widget. When actively in Learn Mode, the button is lit red; to exit Learn mode, click on it again and it will return to its normal gray color.

Clear– Click on this to clear the current controller assignment.

Jump/Catch–This drop-down box allows you to choose between Jump and Catch modes. When a widget is in Jump mode, moving the associated physical controller will cause the widget to immediately jump to the position of the controller. When a widget is in Catch mode, the widget will not respond to the associated physical controller until the controller's value matches the current value of the widget, at which point the widget will respond.

Treat momentary as latching – (*Displayed only when editing an on-off widget i.e., a switch or a button*). When checked, the widget remains in its ON position until you click on it (or touch the associated physical controller), and vice versa. This is useful when you want to have a widget carry out an operation until stopped; i.e., staying at a fast Leslie speed until you are ready to return to a slow Leslie speed.)

Parameter range and control–Each widget has a range of 0 to 100. *Note that this is a percentage, not an absolute value.* For example, if the plugin parameter it is controllinghas an absolute range from 0.0 to 1.0, then when the widget is at its "100" setting it is actually changing that parameter to 1.0 (i.e., 100% of 1.0).

By entering **Min** and **Max** values other than 0 and 100, you can scale the range of the widget movement. For example, if you set Min to 10 and Max to 20, then the associated plugin parameter will change from 0.1 to 0.2.Finer resolution can be accomplished by entering decimal point values, in tenths of a decimal. Scaling can be inverted by setting the MIN value to a higher number than the MAX value.

You can also directly enter MIDI data values in the Min and Max fields, by adding the letter 'm' at the end of the value. For example, if you type in "64m," Gig Performer will interpret it as an exact value of 50.4 (roughly 50% of the maximum MIDI value of 127). The **Now** field simply indicates the current setting of the widget.

Recall value on load– Gig Performer normally restores all widgets to the position they were last in when a .gig file is saved. Checking this button allows you to take a snapshot of the current position of the selected widget so that it is that position that gets recalled and loaded instead. This can be useful if you are playing, for example, a song where the volume of your piano always starts softly but gets quite loud by the end. When you come back to that song again, you don't want the piano to still be loud—you want it to always start soft. The concept is that, even if you play around with widgets during the course of your live performance, when you nextreload a .gig file, your originally stored values will return.

Snap– (*Displayed only when "Recall value on load"is checked*.) Click on this button to take a snapshot of the current widget position

Also recall on activate– Gig Performer normally restores all widgets to the position they were last in when a rackspace is activated.Checking this button allows you to take a snapshot of the current position of the selected widget so that it is that position that gets loaded instead when the rackspace is again activated. The concept is that, even if you play around with widgets during the course of your live performance, when you nextreturn to a rackspace, your originally stored values will return.

Display range – When checked, the widget shows the Min and Max values as a white shadow around the outside of the knob or slider.

Invert- When checked, the widget does the opposite of the expected operation; i.e., a switch or button will cause a parameter to go to minimum when in the ON (lit) position, and vice versa.

Sync – Pressing this button engages bidirectional communication between Gig Performer and the controller mapped to the control. This lets physical controls on control surfaces reflect changes executed in Gig Performer, as well as Gig Performer widgets showing changes made on the physical control to which it is linked. If

Widget groups– Each widget can only be associated with a single plugin parameter. However, the 26 Widget Groups provided by Gig Performer enable you to control multiple parameters from any or all of the widgets assigned to that group. Clicking below the words "Widget Group" opens a dropdown menu with the letters A - Z. All widgets with the same letter are grouped together so that moving one widget will cause all the other widgets in that group to move as well, adjusting various plugin parameters simultaneously.

Note that you can use each widget's Min/Max range controls to specify how much each individual widget affects its associated parameter. You can even make some parameters go in reverse, either by clicking on that widget's "Invert value" button or by setting the MIN value to a higher number than the MAX value. This makes it possible to implement special effects like crossfades, for example.

Name– Allows you to set a namefor use in GP Script and/or OSC.

Enable OSC –Check this box to make the Name field also be available to OSC. For more information on OSC, click here.

Show back plate– (*Displayed only when editing a knob or toggle switch.*) When checked, "back plate" information such as numbers or an "on/off" legend is displayed.

Ignore variations – When checked,the selected widget will be excluded from variation changes and will instead remain at its current value. This is useful if you have, for example, several different variations thatchange the widget settings controlling effects such as phasing or flanging, but you want the widget controlling overall volume to remain constant.

Mapping section

Plugin – Click in this box for a dropdown list of plugins currently being used by the rackspace, then select the plugin you want the widget to access.

Open plugin – (*Displayed only when a plugin is selected.*) Click on this button to open the editor for the selected plugin.

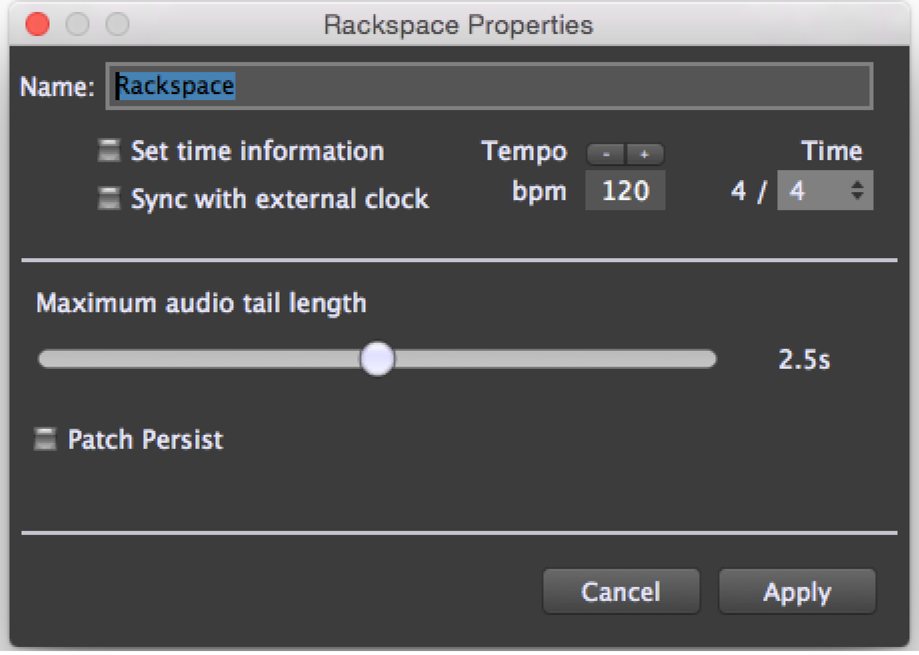
Parameter – (*Displayed only when a plugin is selected.*) A list of available controllable parameters is displayed here. Select the one you want to have the widget control. Alternatively, you can press the **Learn** button.

Learn – (*Displayed only when a plugin is selected.*) Click on this button to have Gig Performer learn the plugin parameter you want to associate with the widget. When actively in Learn Mode, the button is lit red; to exit Learn mode, click on it again and it will return to its normal gray color.

> **Note:** Assignments of widgets to physical controllers in the **Widget Properties** area overrides assignments made in the **Options>Global MIDI Assignments** window.

Rackspace Properties

Double-clicking on a rackspace name opens a dialog that displays rackspace properties.



Rackspace Properties

Name: Rackspace

□ Set time information

□ Sync with external clock

Tempo bpm 120

Time 4 / 4

Maximum audio tail length 2.5s

□ Patch Persist

Name – To rename the rackspace, type a new name here.

Set time information – Check this box if you want the rackspace to have its own tempo value, then enter the desired bpm (beats per minute) and time signature. If this box is unchecked, the rackspace will use the global tempo information set in the Performer View Tempo Display.

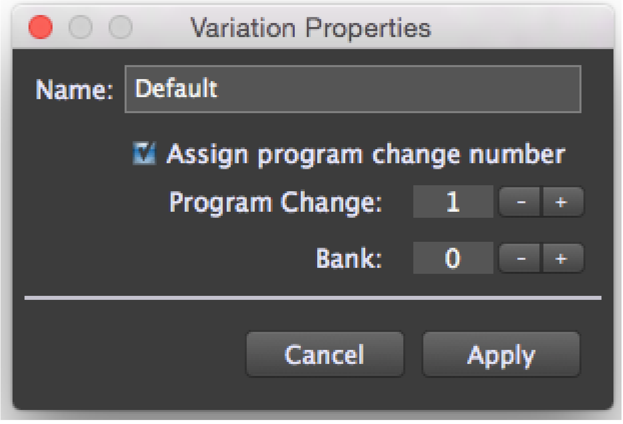
Sync with external clock – Check this box if you want the rackspace to adjust its tempo to incoming MIDI clock messages (24 ticks per quarter note), thus enabling echo/delay or arpeggiator plugins capable of receiving MIDI clock to automatically synchronize.

Tail length – Sets the maximum time before a sound is cut off when you switch rackspaces. Maximum length is 5 seconds. Often known as an audio "tail," this feature is useful when using effects such as reverb or echo.

Patch Persist – If enabled (checked), whenever you switch to another rackspace, notes that you are holding down (either on your keyboard or via your sustain pedal) will continue sounding until you release the keys or pedal. (This actually works with any number of rackspaces, up to the limits of your system's available RAM and CPUresources, so you could conceivably keep the sound of five or more rackspaces going simultaneously.) If disabled (unchecked), whenever you switch from one rackspace to another, the sound is immediately silenced even if you are holding notes down or have pressed the sustain pedal.

Variation Properties

Double-clicking on a variation name opens a dialog that displays variation properties.



Name: Default

☑ Assign program change number

Program Change: 1

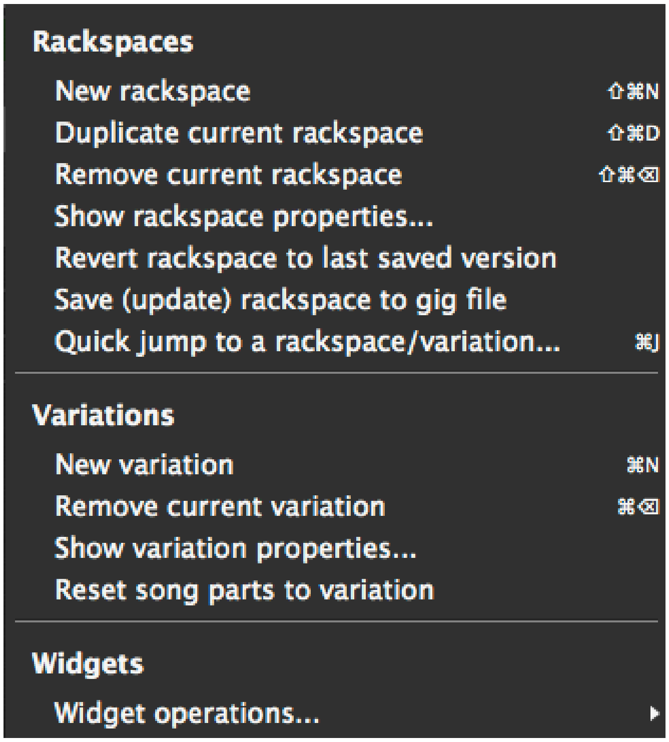
Bank: 0

Name – To rename the variation, type a new name here.

Assign program change number – When checked, allows you to specify the program change number (and, optionally, the bank change number) used to switch to this variation. If a program change number has been assigned, that number will appear to the left of the variation name. If there is only one variation of the rackspace, the program change number will also appear to the left of the rackspace name; if there is more than one variation, an asterisk will appear there instead.

If unchecked, Gig Performer will assign program change numbers in ascending order; i.e., the first variation in the first rackspace will be called up when program change 1 is received; the first variation in the next rackspace will be called up when program change 2 is received, etc.

(...)“More” Menu (Manage Rackspaces/Variations)


Rackspaces
New rackspace ⇧⌘N
Duplicate current rackspace ⇧⌘D
Remove current rackspace ⇧⌘⌫
Show rackspace properties...
Revert rackspace to last saved version
Save (update) rackspace to gig file
Quick jump to a rackspace/variation... ⌘J
Variations
New variation ⌘N
Remove current variation ⌘⌫
Show variation properties...
Reset song parts to variation
Widgets
Widget operations... ▸

Clicking on the “three dots” button in the rackspace list in Performer or Connections View opens the “More” menu:

Rackspaces

New rackspace– Creates a new rackspace.

Duplicate current rackspace– Duplicates the current rackspace.

Remove current rackspace– Deletes the current rackspace.

Show rackspace properties...–Opens the Rackspace Properties dialog.

Revertrackspace to last saved version– Reverts the current rackspace to the last version saved. (**Important:** Click OK in the resultant “Are you sure?” dialog only if you are sure that you want to do this, because all changes made since the file was last saved will be permanently lost. If you want the opportunity to save your work first, click Cancel, then go to **File>Save**or**File>Save As...**)

Save (update) rackspace to gig file– Saves the selected rackspace to the currently open .gig file.

Quick jump to a rackspace/variation – Opens the Quick Rackspace/Variation Finder

Variations

New variation– Creates a new variation.

Remove current variation– Deletes the current variation.

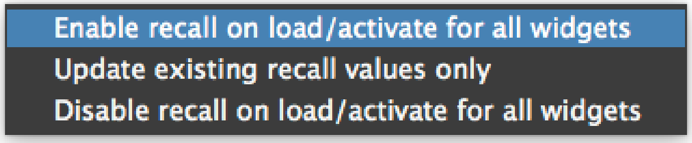
Show variation properties...– Opens the Variation Properties dialog.

Reset song parts to variation –Writes the current settings to every song part using the active variation. Use this command when you update a sound and want to ensure the newest version is used everywhere the sound occurs.

Widgets

Widget Operations submenu

This submenu deals with how widget values are handled when a gig is loaded.



Enable recall on load/activate for all widgets
Update existing recall values only
Disable recall on load/activate for all widgets

Enable recall on load/activate for all widgets–When this command is chosen, the current values of all widgets are stored for recall when the gig is loaded. (Individual widget values can be stored using the “Recall value on load” and “Also recall on activate” buttons in the Parameter range and control section of the Widget Properties inspector.)

Update existing recall values only– When this command is chosen, the stored values of any widgets set to have values recalled on load are updated with the current values of the widgets. This is useful if, for example, you have edited a gig loaded with stored values and now want to save the changes, as it saves the values of all edited widgets at once.

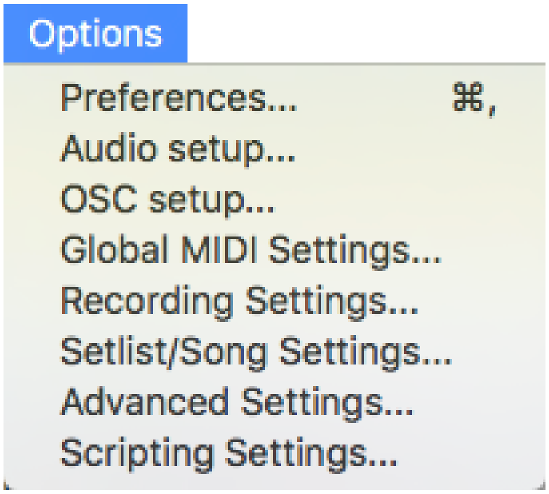
Disable recall on load/activate for all widgets–When this command is chosen, global recall of widget values is disabled, and stored values will be recalled only for widgets with recall enabled by one or both checkboxes in the Parameter range and control section of the Widget Properties inspector.

Tuner View

This button opens Gig Performer’s Tuner View for details on the tuner, see Tuner View.

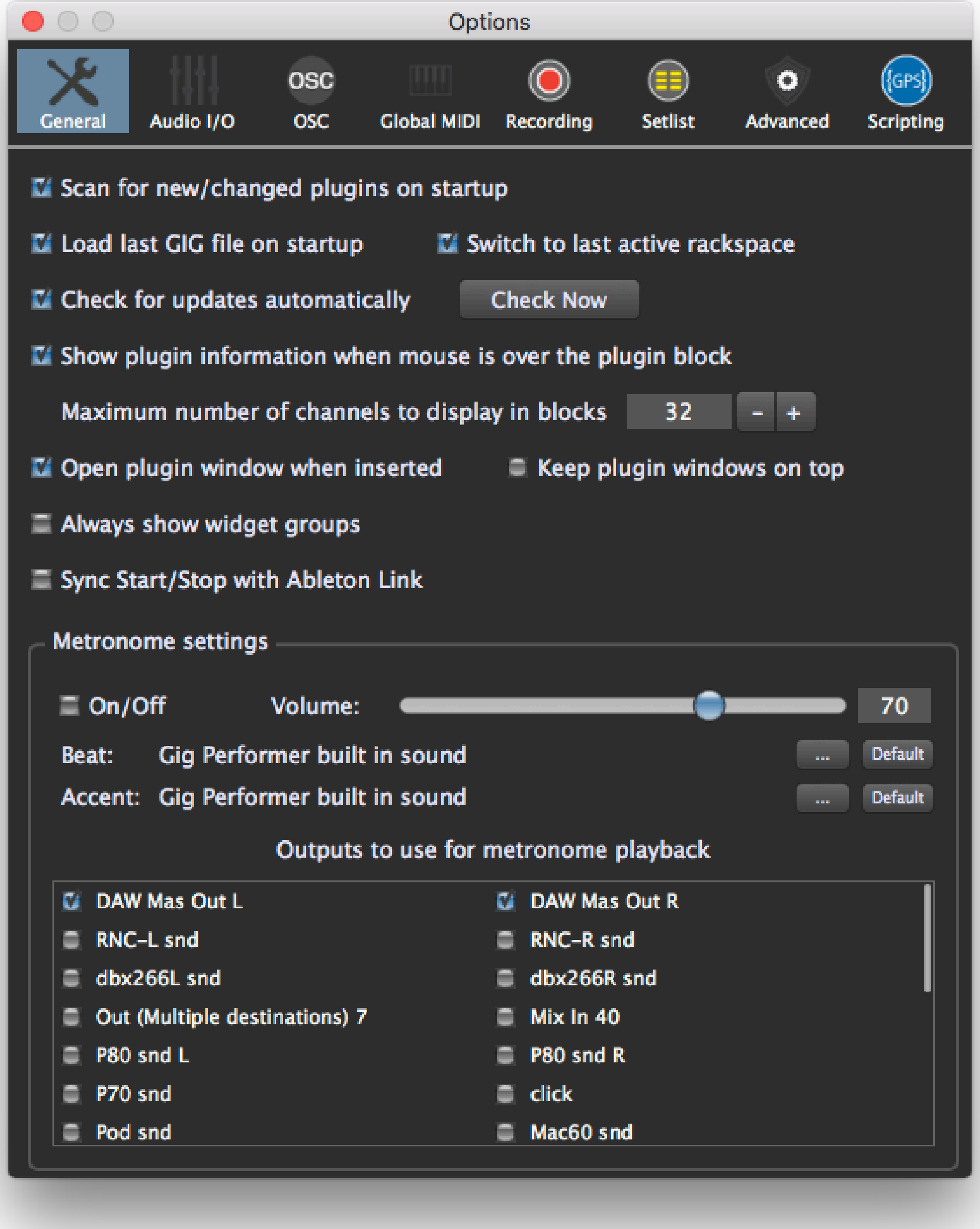
Preferences

Gig Performer's preferences tabs are accessible from the Options menu.



General

The General tab appears when the Preferences command is chosen in the Options menu, or when its shortcut is used.



☑ Scan for new/changed plugins on startup

☑ Load last GIG file on startup ☑ Switch to last active rackspace

☑ Check for updates automatically

☑ Show plugin information when mouse is over the plugin block

Maximum number of channels to display in blocks 32 − +

☑ Open plugin window when inserted □ Keep plugin windows on top

□ Always show widget groups

□ Sync Start/Stop with Ableton Link

Metronome settings

□ On/Off Volume: 70

Beat: Gig Performer built in sound

Accent: Gig Performer built in sound

Outputs to use for metronome playback

☑ DAW Mas Out L	☑ DAW Mas Out R
□ RNC-L snd	□ RNC-R snd
□ dbx266L snd	□ dbx266R snd
□ Out (Multiple destinations) 7	□ Mix In 40
□ P80 snd L	□ P80 snd R
□ P70 snd	□ click
□ Pod snd	□ Mac60 snd

Scan for new/changed plugins on startup–When enabled, Gig Performer will check for new or updated plugins every time it starts. It's generally a good idea to leave this enabled unless you're certain that your plugins have not been updated since you last used Gig Performer. If you're wondering why your latest plugin or update isn't appearing, not scanning for plugins is the most probable cause. (See Troubleshooting for more information.)

Load last GIG file on startup–When enabled, Gig Performer will load the last .gig file that was used.

Switch to last active rackspace–When enabled, Gig Performer will switch to the rackspacethat was active when the .gig file was last saved. Otherwise, the first rackspace will be activated.

Check for updates automatically–Unless you're actually on tour, it's probably useful to keep this option enabled so that you are informed of updates.

Note: Updating any critical-path software while on tour or otherwise engaged in a production project is strongly discouraged because it can introduce instability and/or bugs.

Check Now –Click on this button to manually check for updates. **Note:** This requires an internet connection.

Show plugin information when mouse is over the plugin block–When enabled, moving the mouse over blocks in the connection view will cause a popup tooltip to appear with information about the plugin.

Maximum number of channels to display in blocks–This option prevents plugins from showing more than a specified number of audio channels. Some plugins have a large number of audio channels, most of which may never be used, making their blocks unwieldy to view and causing them to take up a lot of space in Connections View. Note that the value entered here does not apply to the audio interface Audio In and Audio Out blocks.

Open plugin window when inserted–When checked, the plugin editor window will open automatically whenever a plugin is inserted. Disable this feature if you prefer to create many plugin blocks first and edit them all later.

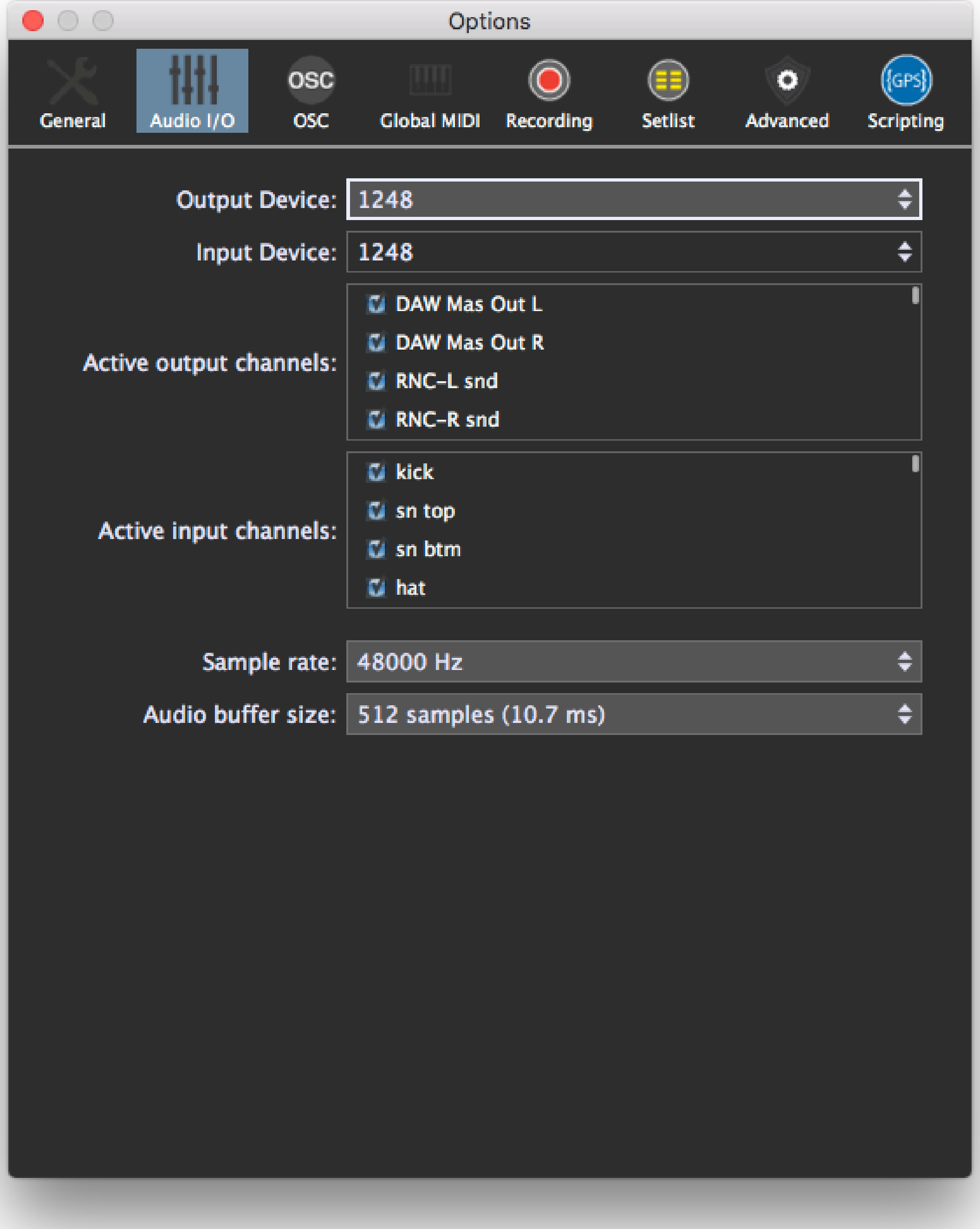
Keep plugin windows on top–When enabled, open plugin window editors always appear on top of other windows

Always show widget groups–When you gang multiple widgets together, the widget group to which they belong (lettered A through Z) is displayed on the top left of the widget while editing. If you would like that widget group name to stay on while NOT editing, enable this option.

Audio metronome settings

The lower half of the General tab contains settings for the audio metronome. These settings are detailed in the Audio Metronome section.

Audio I/O setup



Output Device: 1248

Input Device: 1248

Active output channels:
☑ DAW Mas Out L
☑ DAW Mas Out R
☑ RNC-L snd
☑ RNC-R snd

Active input channels:
☑ kick
☑ sn top
☑ sn btm
☑ hat

Sample rate: 48000 Hz

Audio buffer size: 512 samples (10.7 ms)

Output Device– Select the audio output device you want to use. If you're running Gig Performer on a Mac OSX system and have created any Aggregate Devices (consisting of multiple interfaces) in the Audio MIDI Setup utility, they will appear here as options as well.

Input Device– Select the audio input device you want to use.If you're running Gig Performer on a Mac OS X system and have created any Aggregate Devices (consisting of multiple interfaces) in the Audio MIDI Setup utility, they will appear here as options as well.

Note: Changing the audio output or input device will cause your entire .gig file to be reloaded immediately, so we recommend making such changes immediately after creating a new .gig file. If you are changing your audio configuration (device, sample rate, buffer size, etc.), it can be useful to start Gig Performer with the SHIFT key held down, which temporarily prevents Gig Performer from loading your previous .gig file.

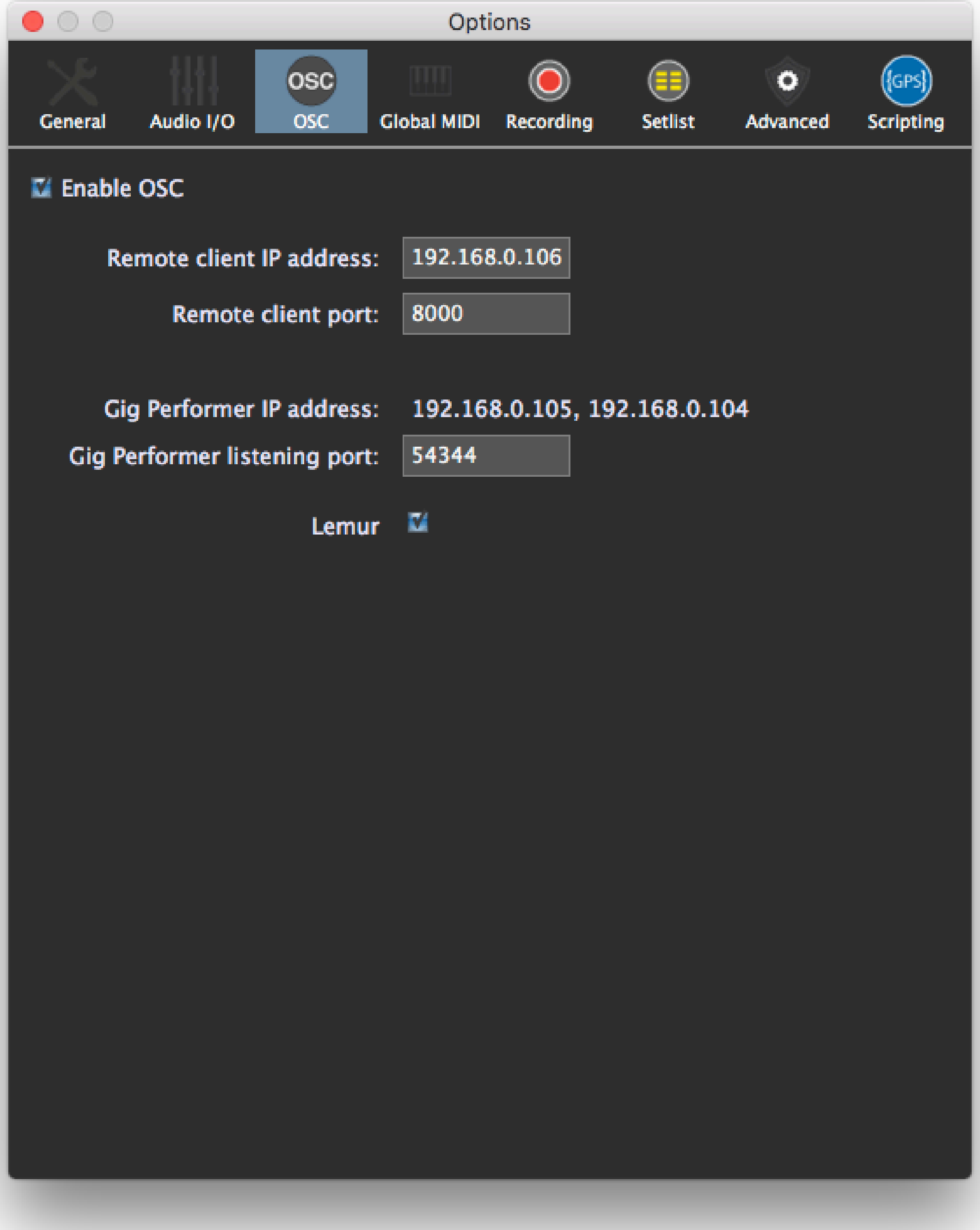
Active output channels/Active input channels–Determines how many input and output channels are to be made available to Gig Performer. If you are using an audio interface with many channels, you may not want to access them all. For example, you may have an interface with 8 analog output channels and another 16 Lightpipe (ADAT optical) channels that you're not using. So instead of displaying audio IN/OUT blocks with 24 or more channels, you can disable the ones you don't need by unchecking them in the list

Sample rate–The sample rate determines the quality of the audio signal. Typical values in the live world are 44.1kHz and 48kHz. Note that there is a tradeoff between higher sample rates and CPU utilization. To view our website article about audio latency, buffer size, and sample rate, click here.

Audio buffer size–The buffer size determines how often samples are actually processed. The smaller the buffer size (or the higher the sample rate), the lower (better) the latency but at the cost of higher CPU utilization. Depending on your plugins and your computer, you may need to find a compromise that gives you the best tradeoff between CPU utilization and latency. If you start using too many CPU cycles, you will begin to hear clicks and other noise in the output. To view our website article about audio latency, buffer size, and sample rate, click here.

Apply Settings – This button will appear if you make any changes to the parameters in this dialog. Click on it to accept the changes, or simply close the Preferences dialog to maintain the current settings.

OSC setup



☑ Enable OSC

Remote client IP address: 192.168.0.106

Remote client port: 8000

Gig Performer IP address: 192.168.0.105, 192.168.0.104

Gig Performer listening port: 54344

Lemur ☑

Enable OSC–Check to turn on OSC. Click here for more information about available OSC commands.

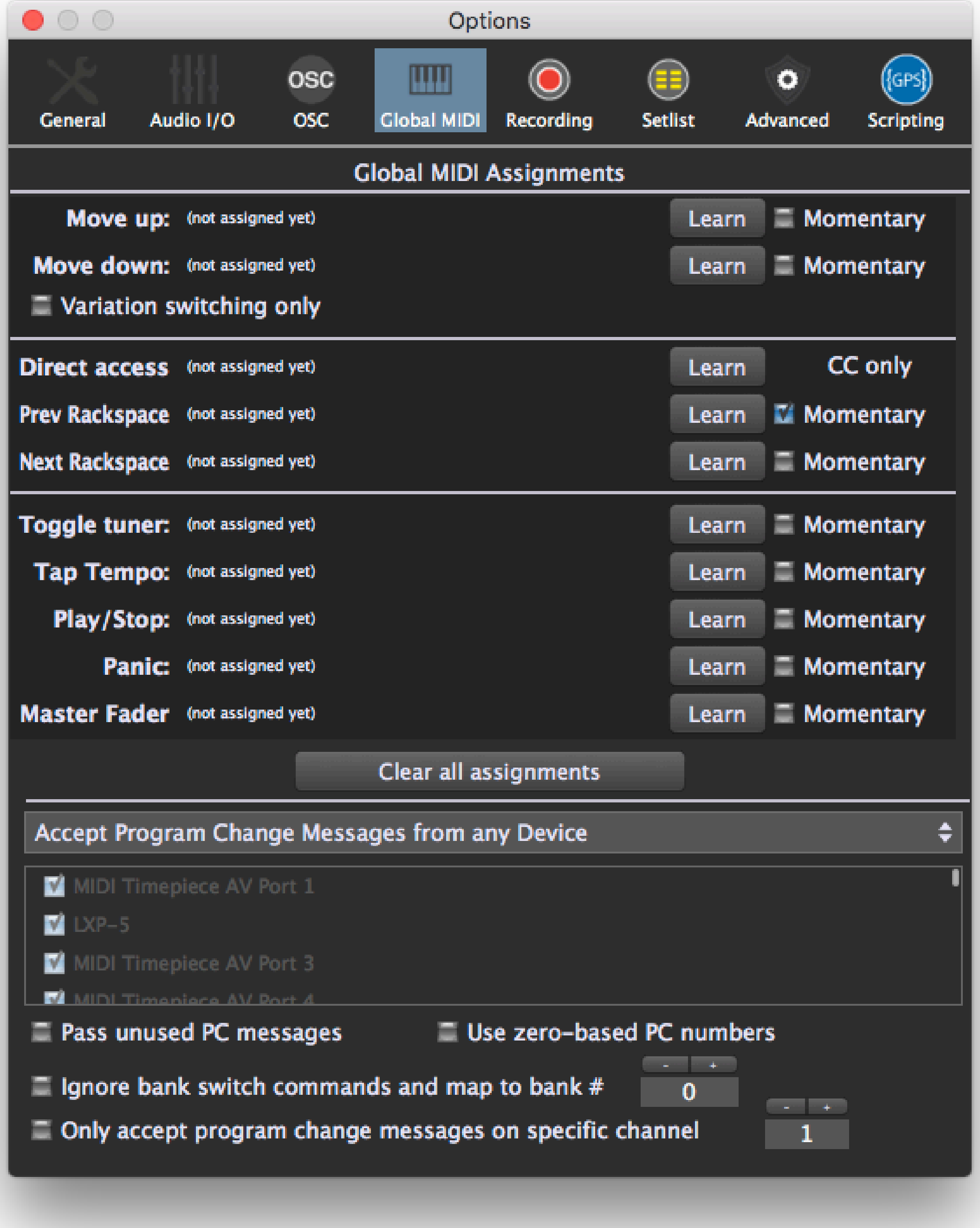
Remote client IP address–The address of the remote device that is running an OSC application.

Remote client port–The port on the remote device that is listening for OSC messages over UDP (User Datagram Protocol).

Gig Performer listening port–The port on which Gig Performer is listening for incoming OSC messages over UDP (User Datagram Protocol). **Note:** The Gig Performer listening port is permanently set to 54344.

Lemur– If you are using Lemur or another OSC application that supports scripting, enable this option to cause Gig Performer to transmit parameterized OSC messages. Disable this option if you are using TouchOSC or other applications that are not programmable. Click here for more information.

Global MIDI



Global MIDI Assignments

Move up: (not assigned yet) Learn □ Momentary

Move down: (not assigned yet) Learn □ Momentary

□ Variation switching only

Direct access (not assigned yet) Learn CC only

Prev Rackspace (not assigned yet) Learn ☑ Momentary

Next Rackspace (not assigned yet) Learn □ Momentary

Toggle tuner: (not assigned yet) Learn □ Momentary

Tap Tempo: (not assigned yet) Learn □ Momentary

Play/Stop: (not assigned yet) Learn □ Momentary

Panic: (not assigned yet) Learn □ Momentary

Master Fader (not assigned yet) Learn □ Momentary

Clear all assignments

Accept Program Change Messages from any Device

☑ MIDI Timepiece AV Port 1

☑ LXP-5

☑ MIDI Timepiece AV Port 3

☑ MIDI Timepiece AV Port 4

□ Pass unused PC messages □ Use zero-based PC numbers

□ Ignore bank switch commands and map to bank # 0

□ Only accept program change messages on specific channel 1

In this dialog, Gig Performer can learn MIDI messages, either from a continuous controller or a momentary switch (in which case you'll need to click on the accompanying "Momentary" box so that it is checked) in order to accomplish the following actions:

Move up–Move to the previous variation, or to the previous rackspace if you are at the first variation.

Move down–Move to the next variation in the current rackspace, or to the first variation of the next rackspace if you are at the last variation of the current rackspace.

Variation switching only–If this option is enabled, you will only be able to move up or down within a single rackspace.

Direct access – Direct access allows you to use a slider or knob to switch between variations in a rackspace. If you had a pedalboard with five MIDI switches, each switch would be set to the same channel and controller number, but assigned a different control value. This value would select a variation in a rackspace, so if the five pedals were assigned values of 0, 1, 2, 3, and 4, they would recall the first five variations.

Next Rackspace – Move to the next rackspace. This function is useful when you wish only to move between rackspaces, not between variations in a rackspace.

Previous Rackspace – Move to the previous rackspace. This function is useful when you wish only to move between rackspaces, not between variations in a rackspace.

Toggle tuner–Switch to or from Tuner View.

Tap Tempo–Gig Performer can receive taps from a pedal or button on a connected MIDI controller or via OSC messages and use that information to set the tempo of delay, echo, or looping pluginsso that they can be synced to the tempo in which you are playing.

To enter Tap Tempo, click on the Learn button and tap on the button or pedal you want to use for setting tempo. (You will most likely need to enable the **Momentary** option as well.) Once you start tapping, the BPM display at the top right of the main Performer View screen will display the current tempo that has been detected. All plugins that respond to global tempo will now synchronize to this tempo. (It may be necessary to enable 'sync' to external tempo on individual plugins. Please check your plugin documentation for more information.)

Note: The tempo used by connected plugins is always the global tempo set here unless it has been overridden in the Rackspace Properties dialog

Play/Stop – Start or stop playback of audio files from a third-party media player plugin or from the supplied Audio File Player plugin.

MIDI Panic – Generate"All Notes Off" messages on all MIDI channels, along with a Sustain Off message and a Center PitchBend message, thus stopping stuck notes in their tracks. This performs the same action as clicking on the MIDI panic button in the upper right-hand corner of the screen.

Master Fader –Gig Performer's master fader can be remotely controlled by a MIDI controller by clicking the Learn button and moving the physical control.

Clear all assignments–Clear all learned assignments. (**Note:** Holding the Shift key down while clicking any of the Learn buttons clears just that assignment.)

Accept Program Change Messages from any Device–Determines whether Gig Performer should respond to MIDI program change messages from all external devices, or only from specific external devices.

Pass through unused incoming Program Change messages–When enabled, if a program change is received for which there is no corresponding rackspace, the message will instead be sent to the first plugin in the current rackspace. For example, if you have created 10 rackspaces and a program change message of 11 or higher is received, it will not cause the rackspace to change but will instead affect the first plugin in the current rackspace. Most plugins respond to incoming program change messages by changing presets; however, the behavior is entirely dependent upon the particular plugin. Consult your plugin's documentation for further information.

Use zero-based program change numbers – Causes Gig Performer to display program change numbers in a range of 0-127 instead of 1-128.

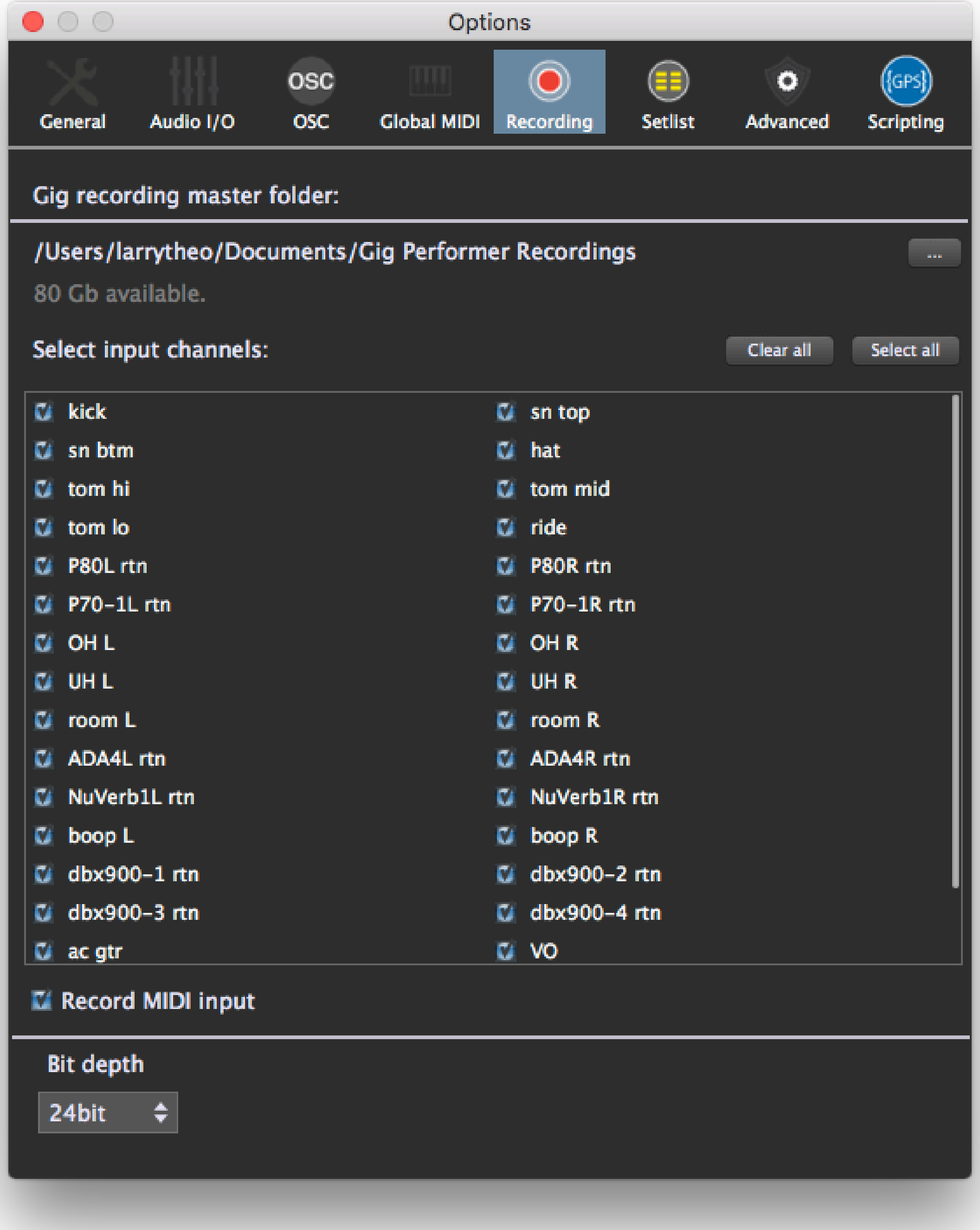
Ignore bank switch controls and map to bank # - Selecting this causes Gig Performer to ignore incoming MIDI program bank switch messages and stay on the specified bank.

Only accept program change messages on specific channel – Gig Performer can change variations, rackspaces, and even instances in response toincoming MIDI program change messages. Checking this box limits Gig Performer to responding only to program change messages on the channel specified in the box.

Note: Mappings to physical controllers made in the Widget Properties inspector override Global MIDI Assignments.

Recording

Details on this dialog are in the Audio Input Recorder section.



Gig recording master folder:

/Users/larrytheo/Documents/Gig Performer Recordings

80 Gb available.

Select input channels:

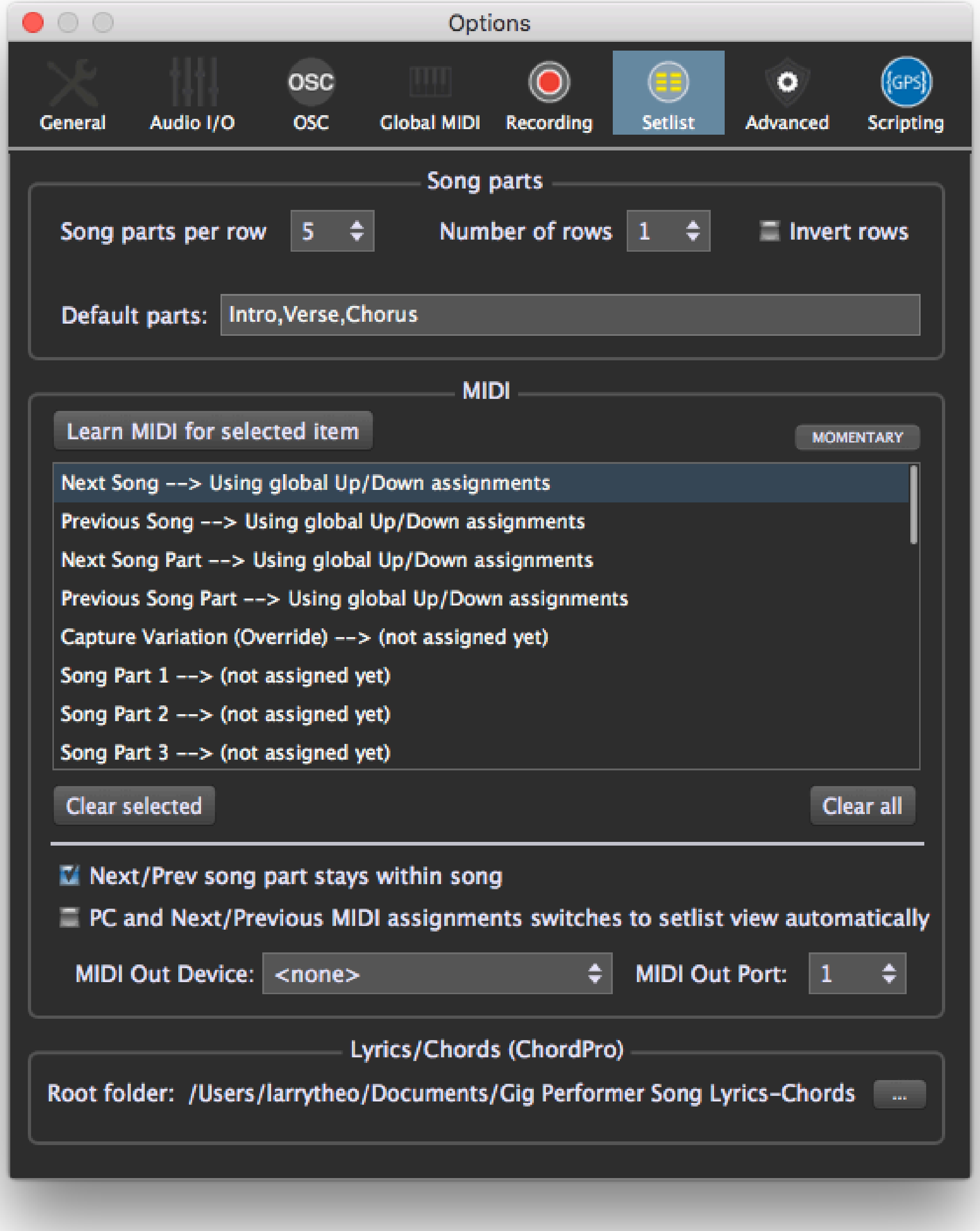
☑ kick ☑ sn top
☑ sn btm ☑ hat
☑ tom hi ☑ tom mid
☑ tom lo ☑ ride
☑ P80L rtn ☑ P80R rtn
☑ P70-1L rtn ☑ P70-1R rtn
☑ OH L ☑ OH R
☑ UH L ☑ UH R
☑ room L ☑ room R
☑ ADA4L rtn ☑ ADA4R rtn
☑ NuVerb1L rtn ☑ NuVerb1R rtn
☑ boop L ☑ boop R
☑ dbx900-1 rtn ☑ dbx900-2 rtn
☑ dbx900-3 rtn ☑ dbx900-4 rtn
☑ ac gtr ☑ VO

☑ Record MIDI input

Bit depth

24bit

Setlist/Song Settings



Song parts

Song parts per row 5

Number of rows 1

□ Invert rows

Default parts: Intro,Verse,Chorus

MIDI

Next Song --> Using global Up/Down assignments

Previous Song --> Using global Up/Down assignments

Next Song Part --> Using global Up/Down assignments

Previous Song Part --> Using global Up/Down assignments

Capture Variation (Override) --> (not assigned yet)

Song Part 1 --> (not assigned yet)

Song Part 2 --> (not assigned yet)

Song Part 3 --> (not assigned yet)

☑ Next/Prev song part stays within song

□ PC and Next/Previous MIDI assignments switches to setlist view automatically

MIDI Out Device: <none>

MIDI Out Port: 1

Lyrics/Chords (ChordPro)

Root folder: /Users/larrytheo/Documents/Gig Performer Song Lyrics-Chords

Song Parts

Song parts per row – Choose a value from the drop-down menu to set the number of tiles per row in the song part bar.

Number of rows – Choose a value from the drop-down menu to set the number of rows of tiles in the song part bar.

Invert rows – Flips the vertical order of the rows of tiles in the song part bar. Instead of 1-2-3, when this box is checked the bar shows the rows 3-2-1.

Default parts – A comma-separated list of song part names that are added automatically when a new song is created.

MIDI

Learn MIDI for selected item button– Setlist operations in the MIDI list can be mapped to MIDI controllers. Select the item in the list, click the button, and use the MIDI controller you want mapped to it.

Momentary button – If the physical control you are mapping is a momentary switch (rather than a latching/toggle switch), click this button.

MIDI controls list – The list shows setlist operations that can be externally controlled and the MIDI events they are mapped to. To map a control, select it in the list and click the **Learn MIDI for selected item button.**

Clear All/Clear Selected buttons– These buttons remove the mapping from all or just a selected set of MIDI controls.

Next/Prev song part stays within song–With this box checked, the MIDI controller assigned to Next and Previous actions will never move to a different song. When the first song part in a song is reached, the Previous command has no further effect; when the last part is reached, the Next command has no further effect.

Note: This checkbox has no effect when using arrow keys on the computer keyboard to move between song parts; they will still move to the next or previous song.

PC and Next/Previous MIDI assignments switches to setlist view automatically– When this box is checked, any incoming MIDI program change (PC) messages or

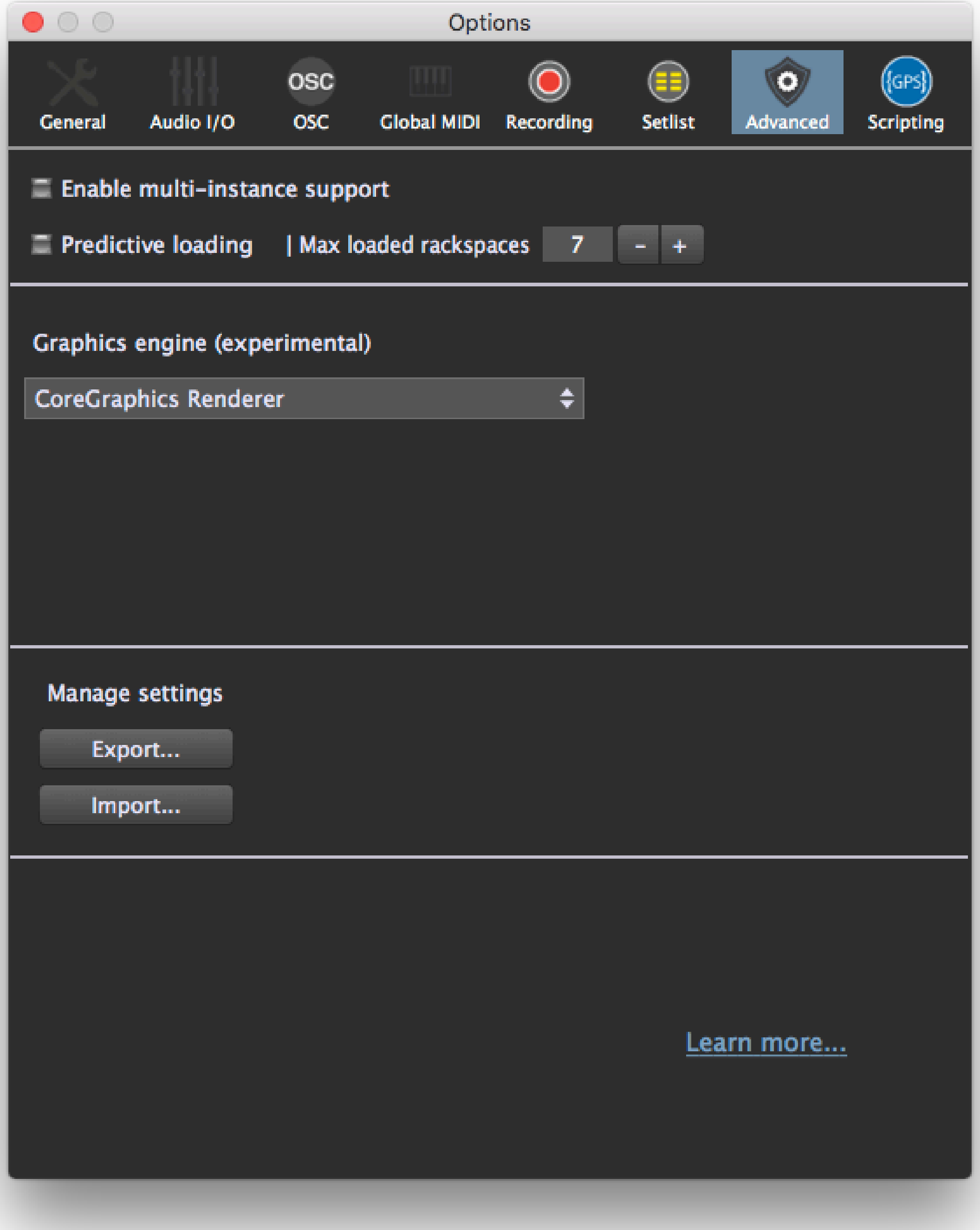
Next/Previous Song Part or Song commands cause Gig Performer to switch to Setlist view immediately.

MIDI Out Device and Port – These settings determine where program change messages sent when songs are selected. Choose a MIDI device from the drop-down menu and a MIDI Out Port from the drop-down menu in that field.

Lyrics/Chords (ChordPro) – This field holds the path to the default ChordPro file folder (root directory) for Gig Performer. The Chords/Lyrics editor can only open ChordPro files located in this folder.

"..." – Opens a dialog for locating the folder you want to designate as the root directory.

Advanced



□ Enable multi-instance support

□ Predictive loading | Max loaded rackspaces 7 - +

Graphics engine (experimental)

CoreGraphics Renderer

Manage settings

Enable multi-instance support–Checking this option gives you the ability to launch Gig Performer multiple times in the same computer, with each such “instance”operating independently.

This has many different practical applications. For example, each bandmember can have the ability to change the rackspace affecting their particular instrument without changing the rackspaces affecting different instruments; they can even use separate MIDI controllers or pedalboards to do so. Instances also make it far more efficient to process vocals, since in many circumstances only a couple of rackspaces are required to manage all vocal sounds. If those rackspaces live in a separate instance, you no longer have to duplicate them in every single rackspace you create.

Keyboardists that want to dedicate one of their controllers (such as an 88-note weighted keyboard) to one particular sound—say a grand piano—will also find instances useful. Simply create an instance with just a single rackspace containing the grand piano plugin and the appropriate MIDI In block. Now you no longer have to include that controller and grand piano plugin in every single rackspace you create, something that can help simplify your overall design and be a real timesaver. There are also many advanced usages: For example, you can use a MIDI In plugin to send program change messages from one instance to another so that one instance can be used to change the rackspace in another instance.

The number of instances you will be able to successfully have open at any one time is dependent upon your system’s RAM and CPUresources. Each instance can access entirely different plugins and MIDI In blocks; however, their Audio In/Out blocks must be consistent. (In other words, separate instances cannot access different audio interfaces.)

Note that when multi-instance support isenabled,three new options are added to the **File** menu, allowing you to create new instances, and to open or delete existing instances.

A demonstration of how to implement instances is provided in the Configuring Audio Plugins tutorial in this user guide.

Predictive loading– One of the biggest problems for live performers is how much memory and CPU cycles are required to keep all the desired plugins available. In particular, plugins that use a significant number of samples can use many gigabytes of RAM to hold the required data.

In normal operation, Gig Performer loads all plugins for all rackspaces and does some optimization to reduce the impact of hosting a large number of plugins simultaneously while still allowing you to switch quickly from one rackspace to another. However, this comes at the cost of RAM and CPU usage, which may create issues in systems with limited resources and/or significant audio hosting requirements. The alternative—loading just one rackspace (i.e., one set of plugins) at a time—would certainly reduce RAM and CPU requirements considerably, but at the cost of having to wait while each new rackspace is loaded.

When using setlists, predictive prioritizes what gets loaded. The current song is always fully loaded, after that are loaded, in order: the next song, previous song, and any further songs until the limit imposed by the Max Loaded Rackspaces setting in the Advanced panel of the Options dialog is reached.

Enabling the "predictive loading" option provides a creative solution: Load only those rackspaces that will be needed in the immediate future (e.g. while playing the current song). Other rackspaces are not loaded and will therefore not consume any memory. As an added benefit, startup time is significantly shortened since not all plugins need to be loaded.You may also see some reduced overall CPU utilization.

Predictive loading provides glitch-free instant switching among those rackspaces that are currently loaded, but with the tradeoff that you might not always be able to jump to an arbitrary rackspace without a short delay, depending on the particular plugin requirements. As a result, when using this option, you should configure your rackspaces in such a way that they are in rough order of when they will be needed in a performance; i.e., song intro -> first verse -> first chorus, etc. In that way, you can best ensure that you will never have to "jump" to another rackspace that is farther away that the specified number of loaded rackspaces (see Max loaded rackspaces below). Note that rackspaces can easily be reordered by dragging them up and down using your mouse or trackpad; holding down the Shift key while doing so results in the dragged rackspace not being loaded, making it more efficient to do setlist reordering without having to wait for each rackspace to load.

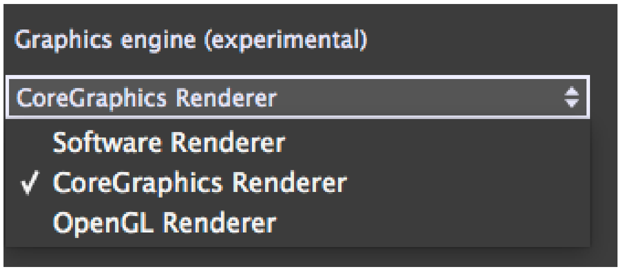
More details about predictive loading can be found on the Gig Performer website: Click here.

Important note: While we have tested predictive loading extensively, it is impossible to test it with all of the thousands of plugins out there. We therefore recommend that you test this feature ahead of time before using predictive loading in a live situation to ensure that the particular plugins you want to use are stable enough to withstand repeated initialization in a short period of time.

Max loaded rackspaces–Specifies the maximum number of rackspaces that will be pre-loaded around the currently selected rackspace. The default value is 5, where the current rackspace as well as the next two and previous two rackspaces are also loaded and instantly accessible. Optimal use of this feature assumes you will be moving from your current rackspace to rackspaces that are close to it (within two, if the default value of 5 is being used). As you move to a new rackspace, Gig Performer will then load and unload other rackspaces so that there are always two available before and two after where you are. This happens in the background and does not affect your audio.

Graphics engine (experimental)

This field selects the graphics library that Gig Performer uses for its displays. Mac OS X users will likely never need to change this setting from CoreGraphics Renderer, as shown below. Depending on the graphics card in their machine, Windows users may find that OpenGL Renderer delivers better quality than Software Renderer.



Graphics engine (experimental)

CoreGraphics Renderer

Software Renderer
✓ CoreGraphics Renderer
OpenGL Renderer

Software Renderer – This choice means that Gig Performer uses your computer's CPU to render the display without the help of any graphics library optimizations. Windows users without high-performance graphics in their machines will likely get the best results with this setting.

CoreGraphics Renderer – Mac OS X uses its CoreGraphics engine to render the display. Most Mac users will use this setting.

OpenGL Renderer – On systems with high-performance graphics capabilities, OpenGL may provide better visual quality, as high-quality graphics cards are typically optimized to run OpenGL.

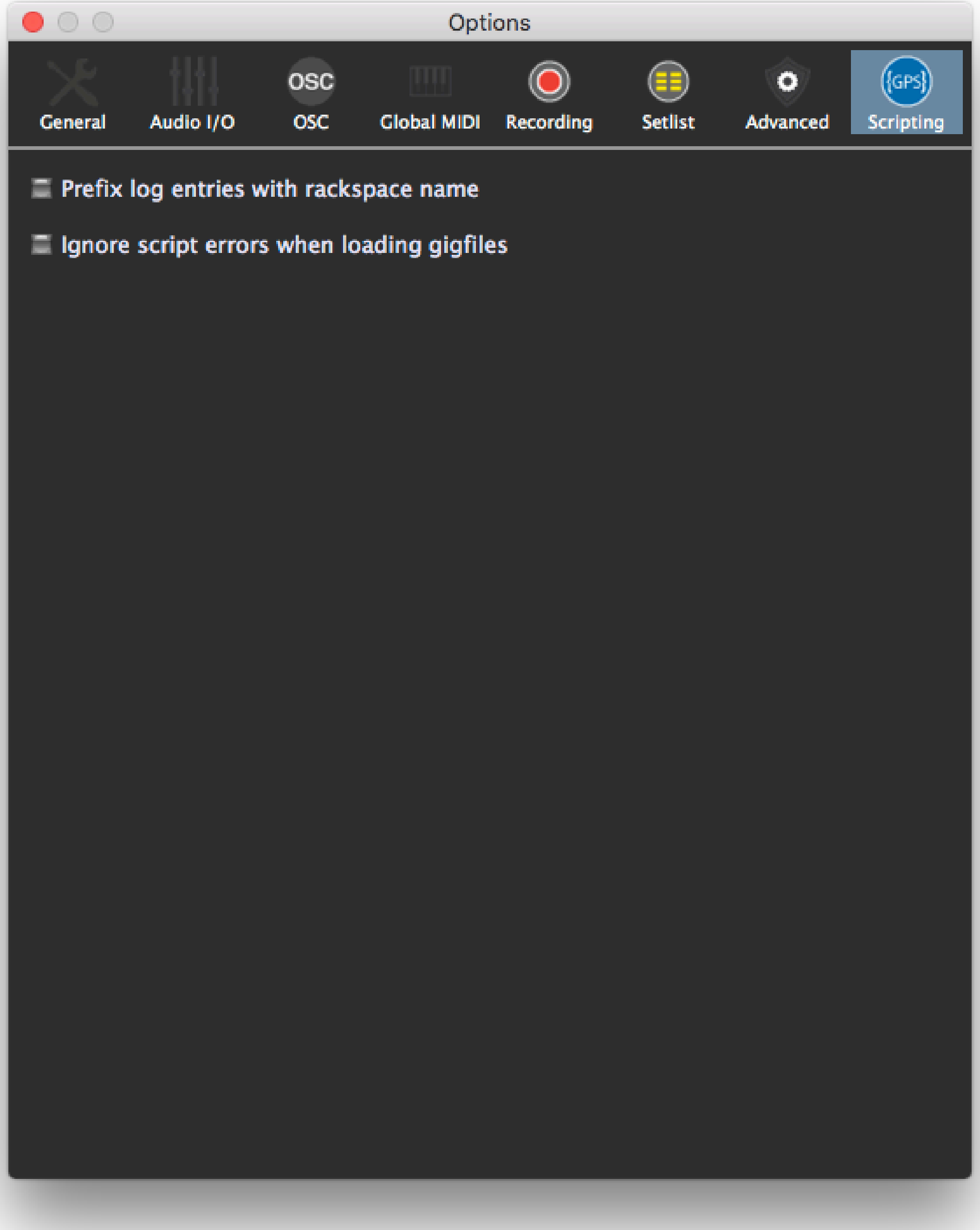
Manage settings…

Export –This command exports all global settings to a .gpsettings file, including global MIDI assignments, all settings in the Options screens, and the entire Rig Manager setup.

Import – This command imports saved global settings in a .gpsettings file.

Learn more – A link to a Gig Performer website article about the Advanced options described above.

Scripting



□ Prefix log entries with rackspace name

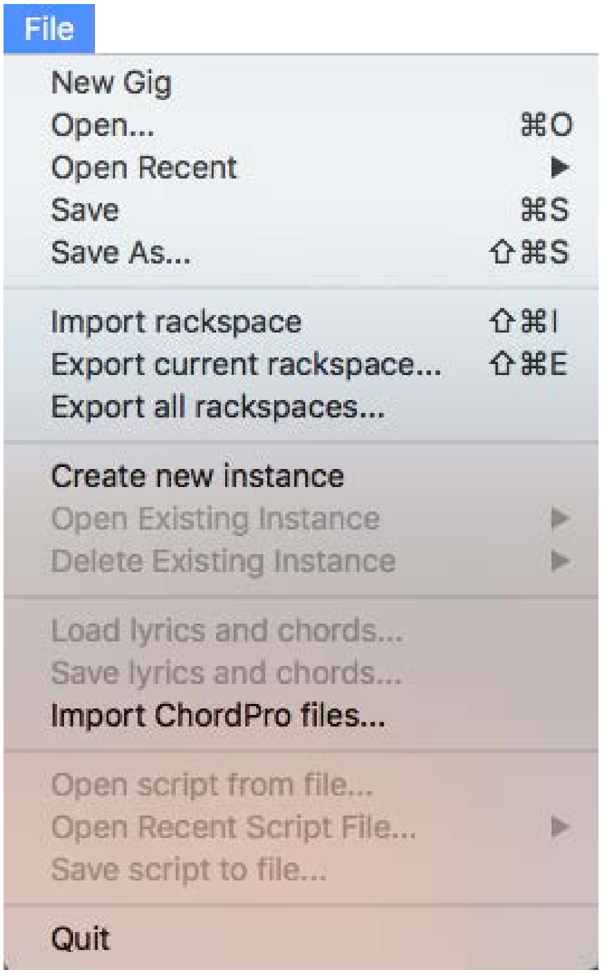
□ Ignore script errors when loading gigfiles

Prefix log entries with rackspace name – Checking this box causes messages in the Script Log window to be identified by the rackspace that generated them. Since there is only one log window for all rackspace scripts, troubleshooting is greatly aided when the log is formatted this way.

Ignore script errors when loading gigfiles - Normally, attempting to load a rackspace that has an invalid script triggers a warning dialog. Checking this option removes this warning. **CAUTION**: This is an expert function. Checking this box is not recommended unless you know exactly what you’re doing, as a rackspace with a broken script may behave in unexpected ways.

Main Menus

File



New Gig –Clears all memory and opens a new rackspace with all default values.

Open... –Opens a saved .gig file.

Open Recent –Provides a popup menu listing the five most recent .gig files.

Save –Saves your work in a .gig file.

Save As... –Saves your work in a .gig file but allows you to enter a new name first (thus preventing previously saved .gig files from being overwritten).

Import rackspace –Allows you to import a single rackspace from a list of previously exported rackspaces. The newly imported rackspace is simply added to your currently created rackspaces—it does not overwrite or delete any of your work. Note that you can also drag and drop one or more rackspaces from a folder directly into Gig Performer, making it fast and easy to create a complete new setlist from existing rackspaces.

Export current rackspace... –Allows you to export a single rackspace for use in another Gig Performer project.

Export all rackspaces... –Allows you to export all rackspaces for use in another Gig Performer project.

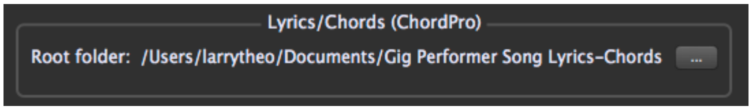
(Note: The following three menu items appear only when the "Enable multi-instance support" option is checked in the Advanced tab of the Preferences dialog.)

Create new instance –Allows you to create a new instance. (For more information click here.)

Open Existing Instance –A popup menu allows you to open a previously created and saved instance. (For more information click here.)

Delete Existing Instance –A popup menu allows you to delete a previously created and saved instance. (For more information click here.)

Load lyrics and chords –When the song lyrics/chords editor window is open, this command is active, and allows opening ChordPro files from the root directory specified in the Options>Setlist tab of the Options dialog.



Lyrics/Chords (ChordPro)

Root folder: /Users/larrytheo/Documents/Gig Performer Song Lyrics-Chords

Save lyrics and chords–When the song lyrics/chords editor window is open, this command is active, and saves the contents of the Song Lyrics/Chords Editor as a .gpchord file. The default directory for these documents isthe root directory specified in the Setlist/Song Settings tab of the Options dialog

Import ChordPro files–This command copies a selected ChordPro file to the ChordPro documents root directory specified in theSetlist/Song Settings tab of the Options dialog and then opens it in the Song Lyrics/Chords editor. It is only active when Gig Performer is showing the setlist view.

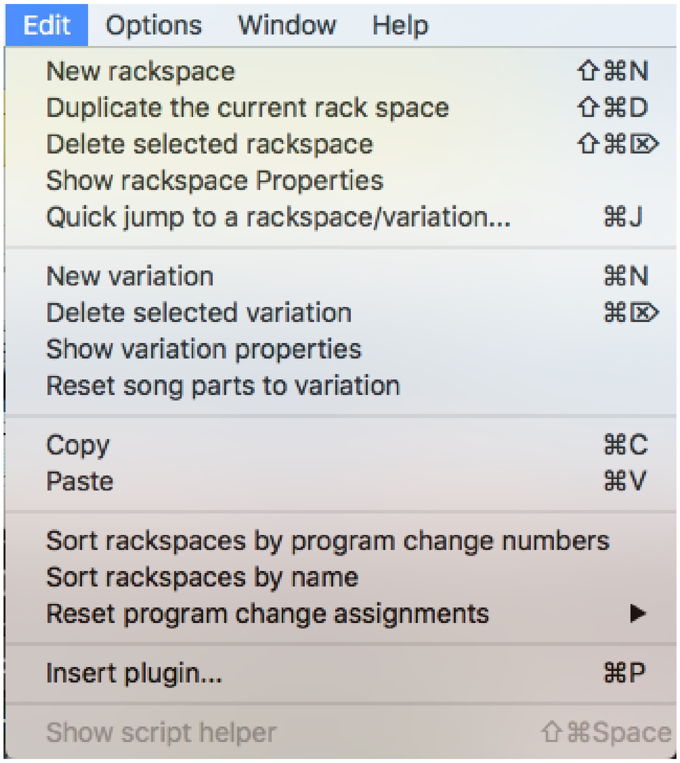
Open script from file – Choosing the command brings up a standard Open File dialog for you to navigate to and select a script to open. Click for more information on GP Script.

Open Recent Script File – A popup submenu lists all GP script files that have been worked on recently. Select one to open. Click for more information on GP Script.

Save script to file– Saves a GP script you have been working on to a new file. Click for more information on GP Script.

Quit –Quit Gig Performer. If you have any unsaved work, a dialog will ask you if you want to save your data first.

Edit – Performer and Connections views


Edit
Options
Window
Help
New rackspace ⇧⌘N
Duplicate the current rack space ⇧⌘D
Delete selected rackspace ⇧⌘⌦
Show rackspace Properties
Quick jump to a rackspace/variation... ⌘J
New variation ⌘N
Delete selected variation ⌘⌦
Show variation properties
Reset song parts to variation
Copy ⌘C
Paste ⌘V
Sort rackspaces by program change numbers
Sort rackspaces by name
Reset program change assignments ▶
Insert plugin... ⌘P
Show script helper ⇧⌘Space

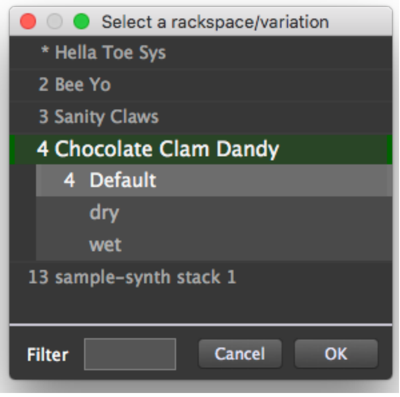
New rackspace –Creates a new rackspace.

Duplicate the current rack space –Duplicates the current rackspace.

Delete selected rackspace –Deletes the currently selected rackspace.

Show rackspace properties… –Opens the Rackspace Properties dialog.

Quick jump to a rackspace variation…- Opens the rackspace/variation quick finder. Search terms can be entered in the **Filter** field to help find the variation you want. Click the variation and then click the OK button.



New variation –Creates a new variation in the current rackspace.

Delete selected variation –Deletes the currently selected variation.

Show variation properties… –Opens the Variation Properties dialog.

Reset song parts to variation – Copies changes made to a variation everywhere the original variation was used, so that all uses of that variation are in sync.

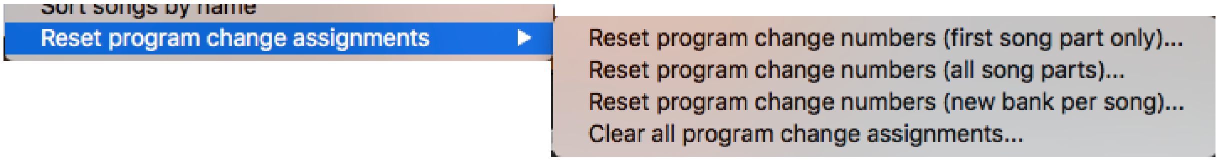
Copy –Copies the current selection to the Clipboard.

Paste –Pastes the data from the Clipboard.

Sort rackspaces by program change numbers –An easy way to sort your rackspaces by the program change numbers associated with each. (For more information click here.)

Sort rackspaces by name –Sorts your rackspaces alphabetically.

Reset program change assignments… –Resets all the program change numbers associated with all rackspaces and variations so that the first variation in the first rackspace is assigned Program Change 0 and the first variation in the second rackspace is assigned Program Change 1, etc. (For more information click here.)


Sort songs by name
Reset program change assignments
Reset program change numbers (first song part only)...
Reset program change numbers (all song parts)...
Reset program change numbers (new bank per song)...
Clear all program change assignments...

Reset program change numbers (first song part only)… -Specify a starting value for the numbering and then this command resets the program change numbers of only the first part in each song. The program change number is derived by adding the specified start number to the number of the song in the setlist.So, if you specify starting with “2,” the first part of the third song in the list will be numbered “4.”

Reset program change numbers (all song parts)… - All parts in all songs in the setlist are assigned program change numbers sequentially, from the top of the setlist to the bottom. You specify a starting value for the numbering.

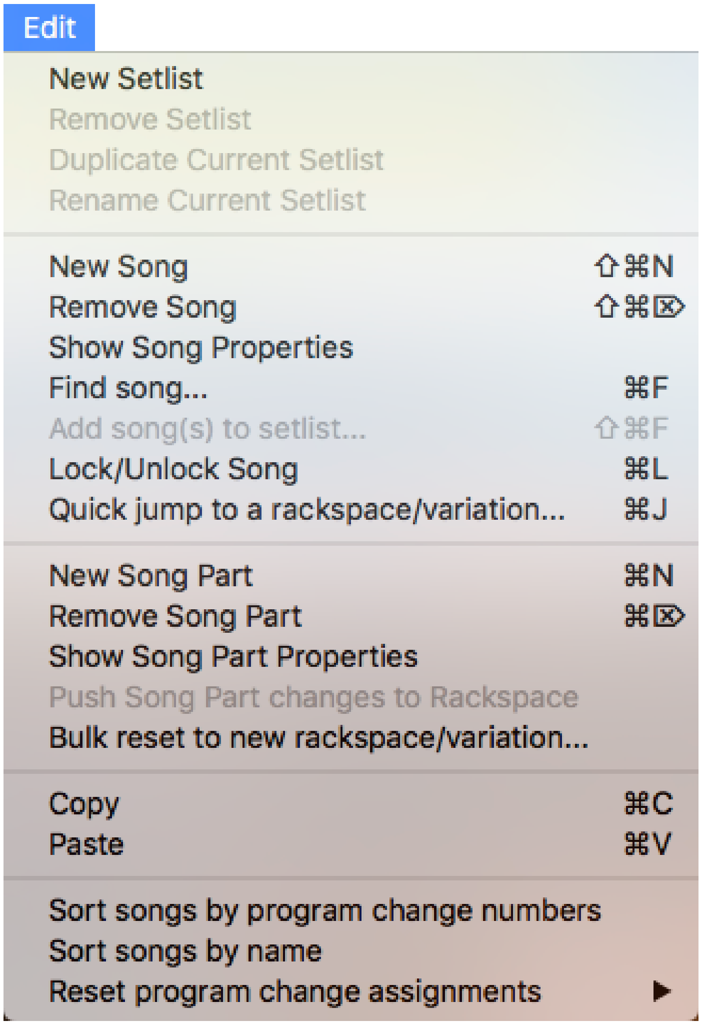
Reset program change numbers (new bank per song)… - This command resets program change messages for all of the variations in a setlist by setting all parts in a song to the same program change bank, and numbering them sequentially. So, bank 2, program change 7 would call the seventh part of the second song in the setlist.

Clear all program change assignments… - Removes all program changes for all parts of all songs in the setlist.

Insert plugin... –Opens the Quick plugin finder. (Gig Performer will switch to Connections View automatically if you are currently in Performer View.)

Edit – Setlist/Song view

The Edit menu is different in Setlist/Song view than in Performer and Connections views.


Edit
New Setlist
Remove Setlist
Duplicate Current Setlist
Rename Current Setlist
New Song ⇧⌘N
Remove Song ⇧⌘⌦
Show Song Properties
Find song... ⌘F
Add song(s) to setlist... ⇧⌘F
Lock/Unlock Song ⌘L
Quick jump to a rackspace/variation... ⌘J
New Song Part ⌘N
Remove Song Part ⌘⌦
Show Song Part Properties
Push Song Part changes to Rackspace
Bulk reset to new rackspace/variation...
Copy ⌘C
Paste ⌘V
Sort songs by program change numbers
Sort songs by name
Reset program change assignments ▶

New Setlist – Lets you name and create a new setlist.

Remove Setlist – After a confirmation dialog, removes the active setlist.

Duplicate Current Setlist – Lets you create and name a duplicate of the active setlist.

Rename Current Setlist – Lets you rename the active setlist.

New Song – Lets you create and name a new song that will have the default set of song parts as defined in the Song Properties dialog.

Remove Song - After a confirmation dialog, removes the active song.

Show Song Properties – Opens the Song Properties dialog.

Find Song… - Opens the Quick Song Finder window.

Lock/Unlock selected song– Choosing this commandblocks changes to rackspace assignments in songs. This protects the songs from accidental modification. Clicking on a rackspace in the browser when the song is locked produces an error message.

Quick jump to a rackspace/variation– This command opens the Quick Rackspace/Variation Finder window, allowing you to immediately switch to any rackspace and variation.

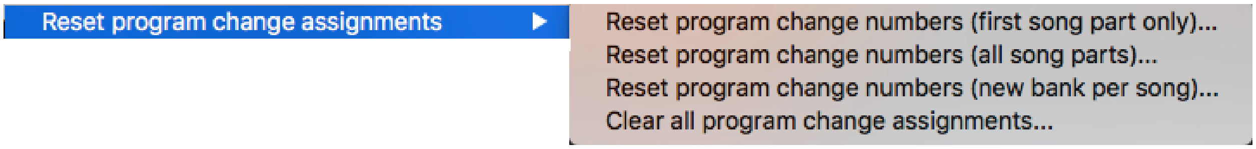
Copy –Copies the current selection to the Clipboard.

Paste –Pastes the data from the Clipboard.

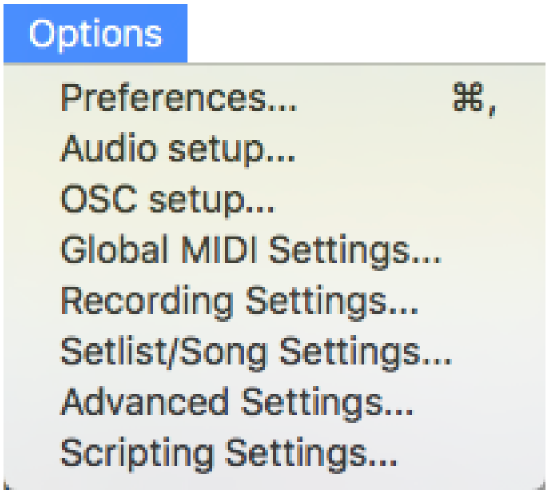
Sort rackspaces by program change numbers –An easy way to sort your rackspaces by the program change numbers associated with each. (For more information click here.)

Sort rackspaces by name –Sorts your rackspaces alphabetically.

Reset program change assignments –For details of these options, see Reset Program Change Assignments.


Reset program change assignments
Reset program change numbers (first song part only)...
Reset program change numbers (all song parts)...
Reset program change numbers (new bank per song)...
Clear all program change assignments...

Options



Preferences... – Opens the General tab of the Preferences dialog.

Audio setup...–Opens the Audio I/O tab of the Preferences dialog.

OSC setup...–Opens the OSC tab of the Preferences dialog.

Global MIDI Settings setup...–Opens the Global MIDI tab of the Preferences dialog.

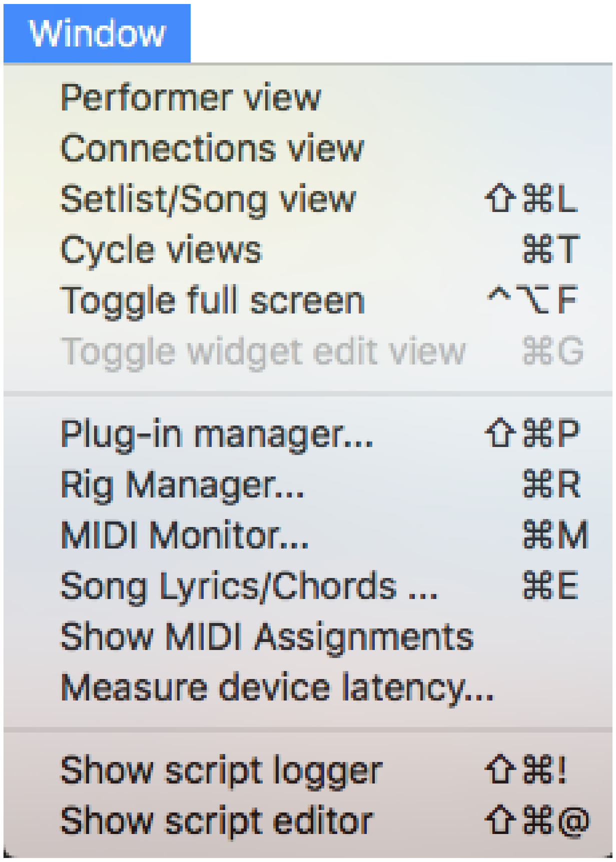
Recording Settings setup- opens the Recording tab of the Preferences dialog. For complete details see the Audio Input Recorder section.

Setlist/song settings– Opens the Setlist/Song Settings tab of the Preferences dialog.

Advanced Settings setup...–Opens the Advanced tab of the Preferences dialog.

Scripting Settings setup…– Opens the Scripting Options tab of the Preferences dialog

Window



Performer view– Switch to Performer View.

Connections view– Switch to ConnectionsView.

Setlist/Song view – Switch toSetlist/Songs View.

Cycle views – Toggles between Performer and Connections views

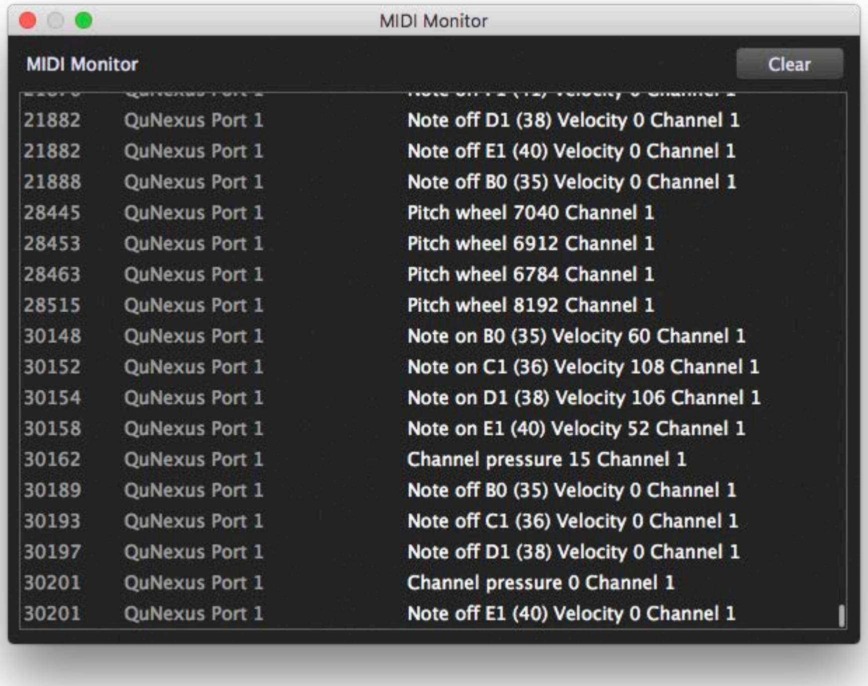
Toggle full screen – Switches the display between full screen and a window.

Toggle widget edit view –Takes Gig Performer in and out of Edit mode.

Plug-in manager...– Opens the Plugin Manager

Rig manager… – Opens the Rig Manager

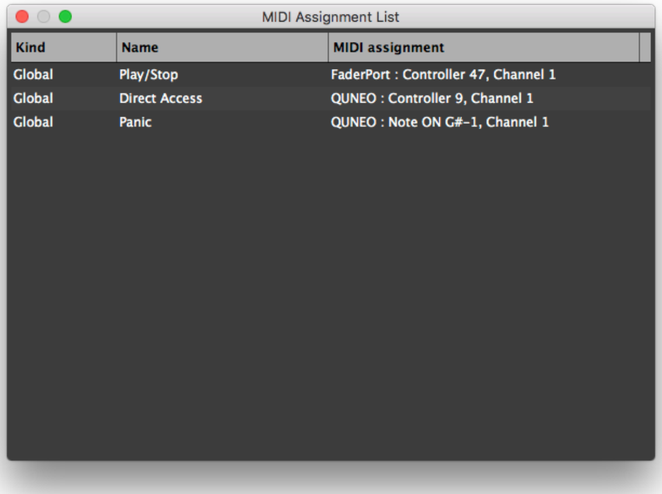
MIDI Monitor– Opens the MIDI Monitor window (shown below). This is similar to the supplied MIDI Monitor plugin, except that it shows all MIDI messages being received into Gig Performer itself except MIDI clock and system exclusive (SYSEX) messages.



```
21882   QuNexus Port 1   Note off D1 (38) Velocity 0 Channel 1
21882   QuNexus Port 1   Note off E1 (40) Velocity 0 Channel 1
21888   QuNexus Port 1   Note off B0 (35) Velocity 0 Channel 1
28445   QuNexus Port 1   Pitch wheel 7040 Channel 1
28453   QuNexus Port 1   Pitch wheel 6912 Channel 1
28463   QuNexus Port 1   Pitch wheel 6784 Channel 1
28515   QuNexus Port 1   Pitch wheel 8192 Channel 1
30148   QuNexus Port 1   Note on B0 (35) Velocity 60 Channel 1
30152   QuNexus Port 1   Note on C1 (36) Velocity 108 Channel 1
30154   QuNexus Port 1   Note on D1 (38) Velocity 106 Channel 1
30158   QuNexus Port 1   Note on E1 (40) Velocity 52 Channel 1
30162   QuNexus Port 1   Channel pressure 15 Channel 1
30189   QuNexus Port 1   Note off B0 (35) Velocity 0 Channel 1
30193   QuNexus Port 1   Note off C1 (36) Velocity 0 Channel 1
30197   QuNexus Port 1   Note off D1 (38) Velocity 0 Channel 1
30201   QuNexus Port 1   Channel pressure 0 Channel 1
30201   QuNexus Port 1   Note off E1 (40) Velocity 0 Channel 1
```

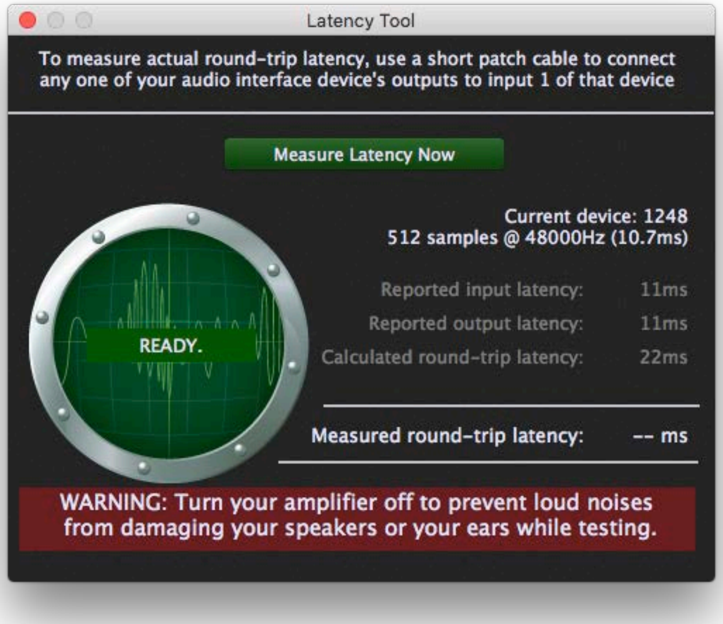
Song Lyrics/Chords… - This command opens the Song Lyrics/Chords Editor.

Show MIDI Assignments – This command opens the MIDI Assignment List window. This window displays the information from the Global MIDI Assignments section of the Global MIDI preferences.



Kind	Name	MIDI assignment
Global	Play/Stop	FaderPort : Controller 47, Channel 1
Global	Direct Access	QUNEO : Controller 9, Channel 1
Global	Panic	QUNEO : Note ON G#-1, Channel 1

Measure device latency – This command allows you to measure the latency through your computer system and interface.



To measure actual round-trip latency, use a short patch cable to connect any one of your audio interface device's outputs to input 1 of that device

Current device: 1248
512 samples @ 48000Hz (10.7ms)

Reported input latency: 11ms
Reported output latency: 11ms
Calculated round-trip latency: 22ms

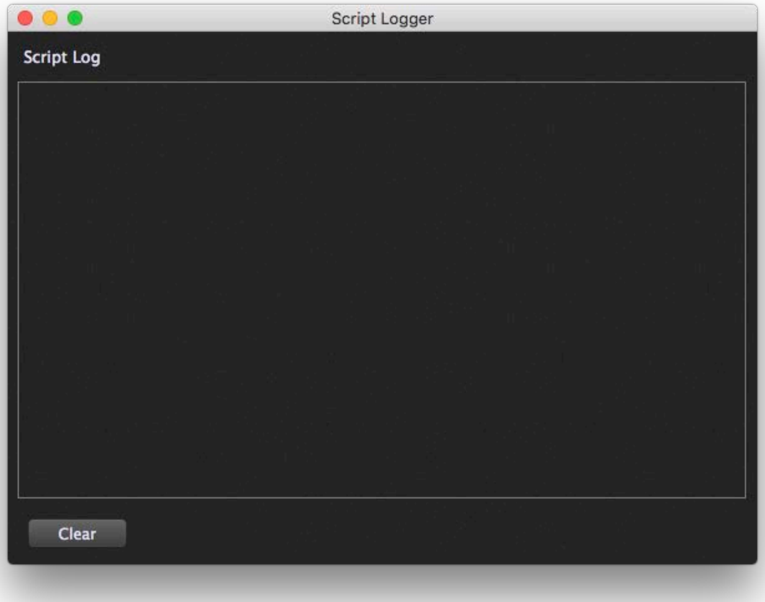
Measured round-trip latency: -- ms

WARNING: Turn your amplifier off to prevent loud noises from damaging your speakers or your ears while testing.

To measure:

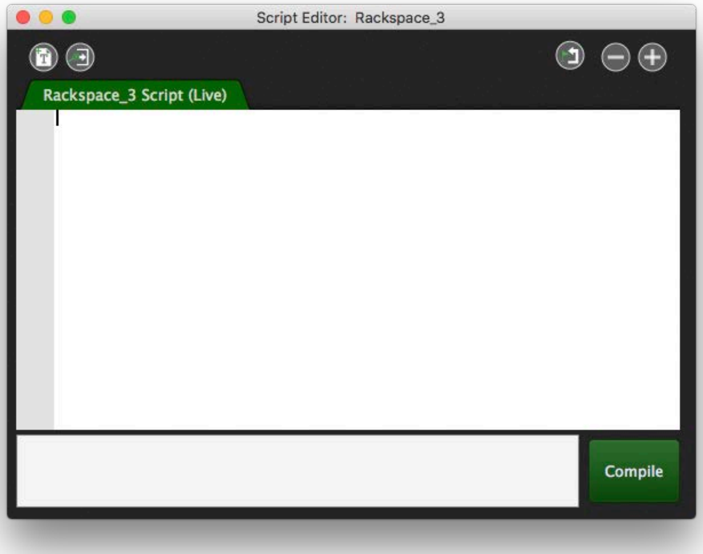
1. Mute your speakers or turn them all the way down.
2. Connect a physical patch cable from any output from your audio interface to input 1 of the interface.
3. Click the **Measure Latency Now button**.
4. The result will appear in the **Measured round-trip latency display**.

Show Script Logger – This command opens the Script Logger window for help in developing GP Scripts. For more information on GP Script, see Going Above and Beyond: GP Script.

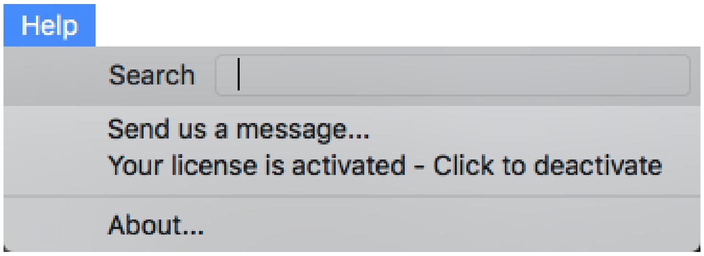


Show Script Editor

This command opens the Script Editor window for use in developing GP Scripts. For more information on GP Script, see Going Above and Beyond: GP Script.



Help



Search– Opens a Search dialog in which you can type your Search subject.

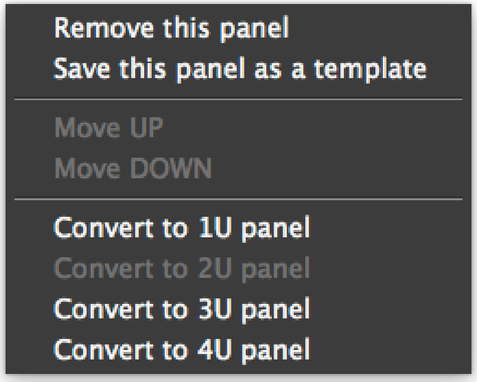
Send us a message...– Opens a Feedback Form dialog in which you can email us a detailed description of your problem, suggestion, bug report, or question. If you do experience a problem with Gig Performer, we strongly urge you to let us know about it so that we can fix it for the benefit of all users.

License activation/deactivation– Opens a dialog that allows you to activate your license or, if it is already activated, deactivate your license. Warning: Do not deactivate your license unless you are sure that you want to run Gig Performer on another system.

About...– Opens a dialog that shows the current version of Gig Performer, along with a link to our website.

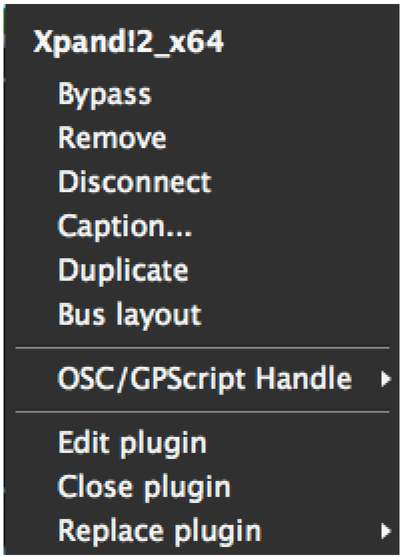
Right-Click/Contextual menus

Rack Panel



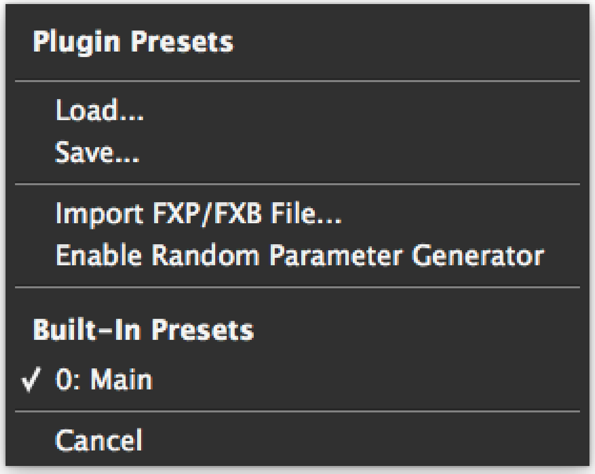
This menu is discussed in the Rack panel Right-click menu section.

Plugin



This menu is discussed in the Plugin Right-click menu section.

Plugin Presets contextual menu



Load… -Loads a Gig Performer preset file (gpp.audiounit or gpp.vst).

Save… -Saves a Gig Performer gpp.audiounit or gpp.vst preset file. The preset is saved in a folder named for the plugin, which is in the folder ~/Documents/GPPluginPresets.

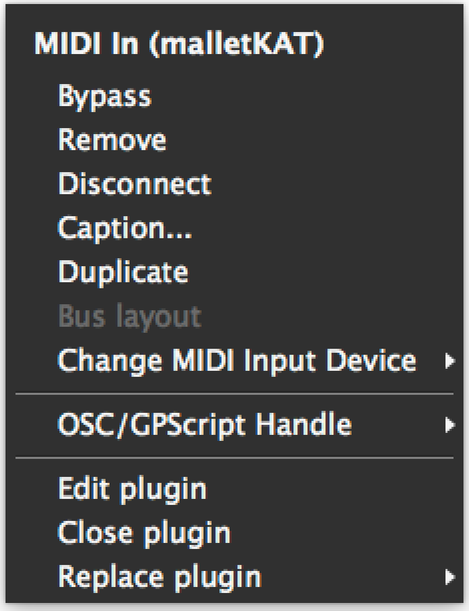
Import FXP/FXB File… - Import a VST preset file. FXP is an individual preset, FXB is a bank of presets. Not all VST plugins are compatible with the FXP/FXB format. Most of those that are compatible are audio plugins.

Enable Random Parameter Generator (experimental)– A bit of fun included for play. Randomizes widget values to generate random patches.

Built-In Presets –Some plugins include presets, most commonly factory presets. If a plugin has any such built-in presets, then will show up in this drop-down menu.

Cancel –This command simply closes the menu.

MIDI Controller



The MIDI Controller plugin right-click menu is the same as other plugins, but with the addition of:

Change MIDI Input Device – Reassigns the MIDI In or MIDI Out block to the MIDI controller chosen from the popup menu.

Manage Rackspaces contextual menu

See (...) "More" Menu (Manage Rackspaces/Variations)

Manage Setlists contextual menu

See"More..." Menu (Manage Setlists)

Keystroke shortcuts

File management

Mac OS X	Windows	Function
Cmd-O	Ctrl-O	Open a .gig file
Cmd-S	Ctrl-S	Save to the current .gig file
Shift-Cmd-S	Ctrl-Shift-S	Save the current work to a .gig file with a new name
Shift-Cmd-I	Ctrl-Shift-I	Import a single rackspace

		from a file
Shift-Cmd-E	Ctrl-Shift-E	Export the current rackspace to a file

Screen management

Mac OS X	Windows	Function
Cmd-T	Ctrl-T	Cycle between Performer View and Connections View
Cmd-G	Ctrl-G	Toggle widget edit view

Rackspace management

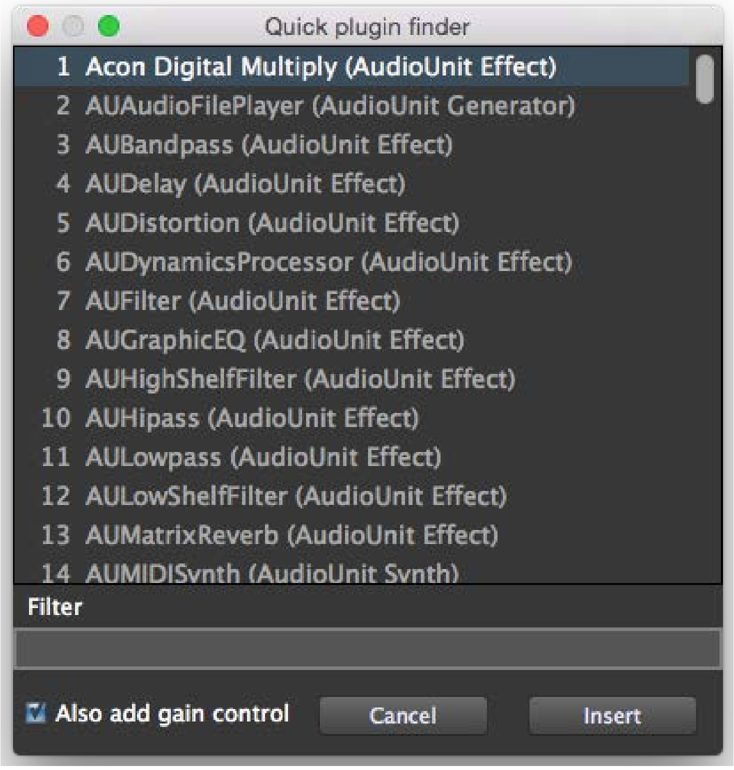
Mac OS X	Windows	Function
Shift-Cmd-N	Ctrl-Shift-N	Add a new rackspace
Shift-Cmd-D	Ctrl-Shift-D	Duplicate the current rackspace
Shift-Cmd-BACKSPACE	Ctrl-Shift-BACKSPACE	Delete the current rackspace
Cmd-N	Ctrl-N	Add a new variation to the current rackspace
Cmd-BACKSPACE	Ctrl-BACKSPACE	Delete the current variation
Cmd-,	Ctrl-,	Open the General preferences dialog

Connections management

Mac OS X	Windows	Function
Cmd-P	Ctrl-P	Insert a new plugin (Gig Performer will switch to Connections View automatically if you are currently in Performer View.)

Quick Plugin Finder

Note: The Cmd-P /Ctrl-P keyboard shortcut provides a more efficient way to create a plugin than right-clicking since it opens the Quick plugin finder dialog (shown below), which allows you to type a partial plugin name, with the resultant list restricted to those that match. Checking the "Also add gain control" box in this dialog causes a Gain and Balance plugin to be inserted immediately after the selected plugin.



1 Acon Digital Multiply (AudioUnit Effect)
2 AUAudioFilePlayer (AudioUnit Generator)
3 AUBandpass (AudioUnit Effect)
4 AUDelay (AudioUnit Effect)
5 AUDistortion (AudioUnit Effect)
6 AUDynamicsProcessor (AudioUnit Effect)
7 AUFilter (AudioUnit Effect)
8 AUGraphicEQ (AudioUnit Effect)
9 AUHighShelfFilter (AudioUnit Effect)
10 AUHipass (AudioUnit Effect)
11 AULowpass (AudioUnit Effect)
12 AULowShelfFilter (AudioUnit Effect)
13 AUMatrixReverb (AudioUnit Effect)
14 AUMIDISynth (AudioUnit Synth)

Quick rackspace/variation finder

A fast way to access any variation. See Quick Rackspace/Variation Finder**.**

Additional key shortcuts

Startup

Hold the Shift key down while Gig Performer is starting and it will not reload the previous Gig, even if you have the preference "Load last GIG file on startup" checked.

Hold down the Option/Alt key during startup to block Gig Performer from autoloading and executing a startup script.

Dragging rackspaces

Holding the Shift key while dragging rackspaces prevents the dragged rackspace from being activated. (This can be very useful for adjusting your setlist order when Predictive Loading is enabled).

Quick stereo connection

To make a quick stereo connection, hold the Shift key down while dragging a plugin until its output pins touch the input pins of another plugin, then release (unclick) the mouse. If there is already a connection, this operation will cause it to be removed instead.

Fine control

Hold down the Cmd (Mac OS X) or Ctrl (Windows) key for fine control while adjusting a widget.

MIDI In plugin channel toggling

In the MIDI In (OMNI) and MIDI In (OSC) plugin editors, if you hold the Shift key down while enabling or disabling individual MIDI channels by clicking on their respective checkboxes, the other checkboxes will be set to their opposite states. This is a convenient way to quickly enable a single channel while blocking all others.

Preferences>Global MIDI LEARN buttons

Holding the Shift key while clicking any of the LEARN buttons in the Preferences>Global MIDI dialog clears that assignment.

Widget captions

Normally the widget caption will be assigned the parameter name only. However, if you hold the Shift key down while selecting a parameter, the widget caption will include the plugin name as well as the parameter name. If you also hold the ALT key down, the widget caption will include the plugin name only.

TROUBLESHOOTING

No sound

Test your audio system to make sure it is receiving signal from your connected audio interface.

Make sure one or more plugins have connections from their audio output(s) to one or more audio input(s) on your connected audio interface.

If you have multiple audio interfaces connected to your system, make sure the correct one is selected.

Gig Performer isn't finding my plugins

Normally Gig Performer scans only certain "standard" folders for plugins, such as (in Mac OS X systems) Library/Audio/Plug-ins/VST or (in Windows systems) C:\Program Files\VstPlugins. However, some plugin and DAW manufacturers install their plugins in different folders. If Gig Performer isn't finding your plugins, that's the likely cause. To instruct it to scan different folders, open the Plugin Manager (from the Window menu) and click on the Manage button in the bottom left-hand corner, then select "Set VST folders for scanning...". A dialog will appear allowing you to add or delete folders to the list of those scanned by Gig Performer.

Audio interface does not appear in Connections View

In most cases, Gig Performer will automatically detect all audio interfaces connected to your system. If it does not, check to make sure that the interface is connected and powered on, and relaunch Gig Performer. If your audio interface still does not appear, take the following steps:

- Mac OS X: Start up theAudio MIDI Setup app in the Utilities folder and make sure that your interface appears in the list of input and output devices on the left-hand side of the Audio Devices window. If it does not appear, consult the documentation for your interface.
- Windows: Go to Control Panel>Hardware and Sound>Sound and make sure that your interface appears in the list of playback and recording devices. If it does not appear, consult the documentation for your interface.

MIDI controller does not appear in Connections View

In most cases, Gig Performer will automatically detect all MIDI controllers connected to your system via USB. If it does not, check to make sure that the controller is powered on, and relaunch Gig Performer. If your MIDI controller still does not appear, take the following steps:

- Mac OS X: Go to Utilities>Audio MIDI Setup and select Show MIDI Studio from the Window menu, then make sure that your controller appears in the window and is not grayed out. If it is grayed out or does not appear, consult the documentation for your controller.
- Windows: Go to Control Panel>Hardware and Sound>Devices and Printers>Device Manager and make sure that your controller appears in the "Sound, video and game controllers" category. If it appears with a yellow question mark or exclamation point, right-click and choose the Update Driver Software option. If it does not appear, consult the documentation for your controller.

Widget not working

Double-click on the widget to open the Widget Properties inspector and make sure that it is assigned to a plugin and a plugin parameter.

If your question is not answered here, please check our website's Knowledge Base and troubleshooting tips. If your problem is still not addressed, feel free to reach out to us through our support page.

APPENDIX A: OSC

Gig Performer provides extensive support for OSC (Open Sound Control)—a networking protocol that can be used to control one application from another, with a particular focus on apps used for live performance. OSC names can be associated with plugins and widgets, allowing you to transform your tablet or smartphone intoa remote display and control surface, with virtual knobs, sliders, and buttons for full wireless control.For example, you canuse OSC to change the output levels of each plugin independently or adjust the cutoff frequency of a synth plugin. It can also be used to set Tap Tempo, turn Tuner View on and off, and select the desired rackspace and variation. In addition to well-known OSC products like Lemur and TouchOSC, it is also possible to use programming tools like Max/MSP to create very sophisticated applications that can also work interactively with Gig Performer.

For the most current version of Gig Performer's OSC implementation, see:

https://gigperformer.com/downloads/OSCMessages_3x.pdf

MIDI over OSC

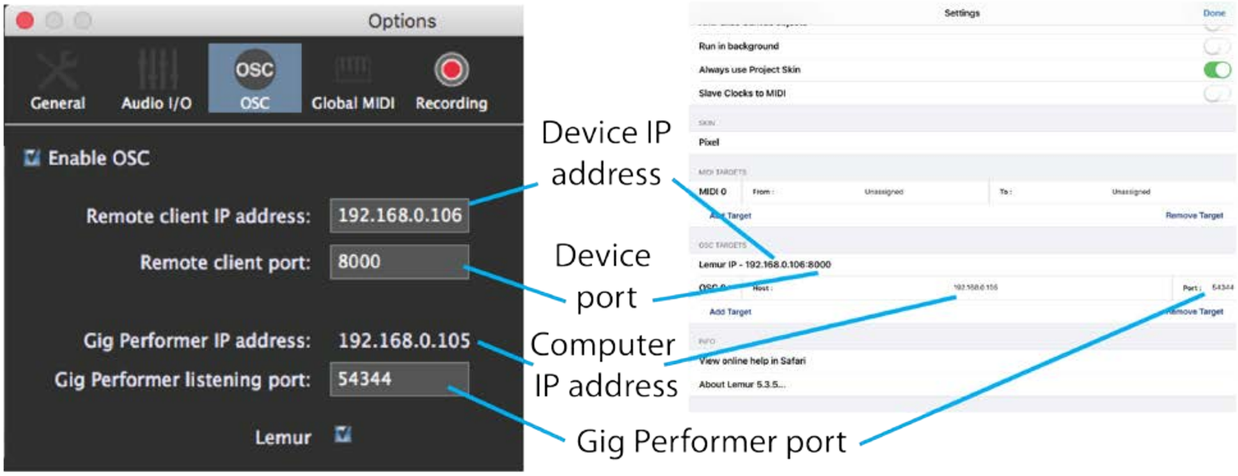
Gig Performer's supplied MIDI In (OSC) plugin is designed to receive MIDI messages from remote OSC sources. As described in the Controlling Gig Performer Using OSC tutorial in this user guide, one of the requirements of OSC is that each object in both the host computer and remote device must have matching unique case-sensitive names comprised of characters only (no spaces). These names must be unique within a single Gig Performer rackspace, although they can be reused in different rackspaces.

APPENDIX B – Communications Between Lemur and Gig Performer

Important note: Deskew Technologies cannot provide support for either the Lemur app or wireless communications in general. However, we offer the information below to try and facilitate Gig Performer users getting the two applications working together.

It is not complicated to get the Lemur app and Gig Performer to communicate, but neither is it immediately obvious. The basic idea is that the Lemur app and Gig Performer each need to have the other's contact information. Once they do, communications can be established. The contact information consists of two parts: the IP address of the device and the port number to which the application is listening for messages.

1. In Gig Performer, choose the **Options>OSC Setup command**. The OSC settings open.
2. The IP address of the computer running Gig Performer is shown in the **Gig Performer IP address** display. The port to which Gig Performer is listening for OSC messages is shown in the **Gig Performer listening port** field. The default port is 54344.
3. In the Lemur app, open the **Settings** menu and choose the **More Settings…** command.
4. In the **OSC Targets** area, enter the IP address of the computer running Gig Performer into the **Host** field, and the port in the **Port** field.
5. The Lemur IP display shows the IP address of the device running Lemur. The right-most segment, after the colon (":") is the port. Enter this into the **Remote client IP address** field in the OSC settings in Gig Performer.
6. Enter the port portion of the Lemur IP display into the **Remote client port** field.
7. Click the **Lemur** checkbox in the OSC settings. This enables Gig Performer to send more detailed OSC messages, which Lemur can handle. NOTE: Be sure this box is unchecked when using Gig Performer with TouchOSC or other more basic OSC applications.
8. For the most reliable communications, launch the Lemur app first, then launch Gig Performer.



☑ Enable OSC

Remote client IP address: 192.168.0.106

Remote client port: 8000

Gig Performer IP address: 192.168.0.105

Gig Performer listening port: 54344

Lemur ☑

Run in background

Always use Project Skin

Slave Clocks to MIDI

SKIN

Pixel

MIDI TARGETS

MIDI 0 From : Unassigned To : Unassigned

Add Target Remove Target

OSC TARGETS

Lemur IP - 192.168.0.106:8000

OSC 0 Host : 192.168.0.105 Port : 54344

Add Target Remove Target

INFO

View online help in Safari

About Lemur 5.3.5...

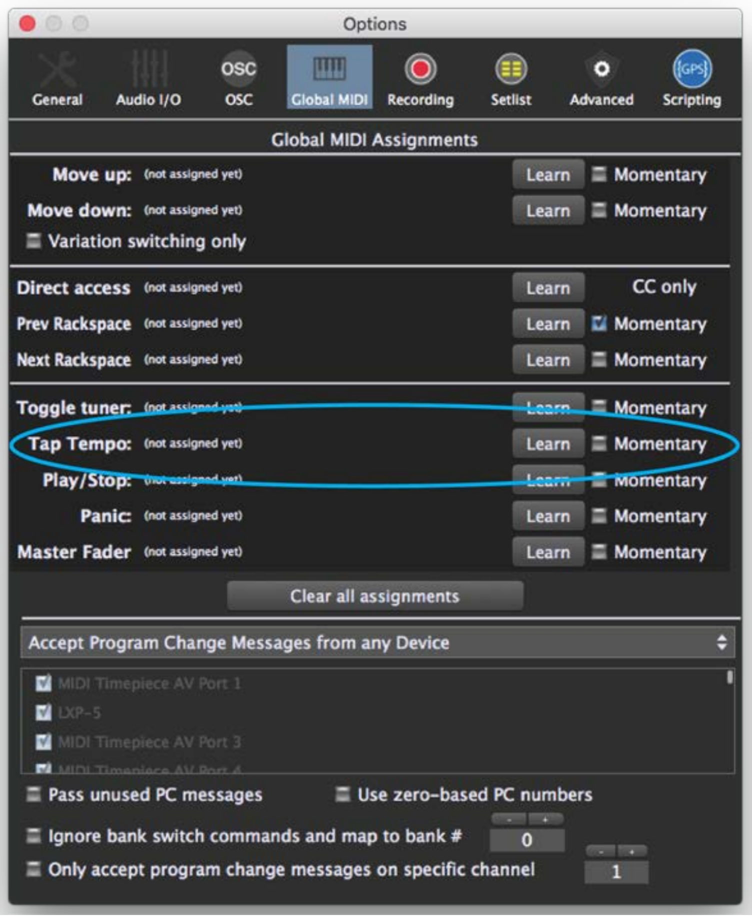
APPENDIX C – Tap and Tip

Creating a Tap Tempo button

Using a MIDI Key or Button to Tap

It is often easiest to tap in a tempo from a keyboard or other MIDI controller. To set this up in Gig Performer:

1. Choose the Options>Global MIDI Settings command, or type Cmd-, (comma) to open the Options dialog and then click the Global MIDI tab.



Global MIDI Assignments

Move up:	(not assigned yet)	Learn	□ Momentary
Move down:	(not assigned yet)	Learn	□ Momentary
□ Variation switching only			
Direct access	(not assigned yet)	Learn	CC only
Prev Rackspace	(not assigned yet)	Learn	☑ Momentary
Next Rackspace	(not assigned yet)	Learn	□ Momentary
Toggle tuner:	(not assigned yet)	Learn	□ Momentary
Tap Tempo:	(not assigned yet)	Learn	□ Momentary
Play/Stop:	(not assigned yet)	Learn	□ Momentary
Panic:	(not assigned yet)	Learn	□ Momentary
Master Fader	(not assigned yet)	Learn	□ Momentary

Clear all assignments

Accept Program Change Messages from any Device

☑ MIDI Timepiece AV Port 1
☑ LXP-5
☑ MIDI Timepiece AV Port 3
☑ MIDI Timepiece AV Port 4

□ Pass unused PC messages □ Use zero-based PC numbers

□ Ignore bank switch commands and map to bank # 0

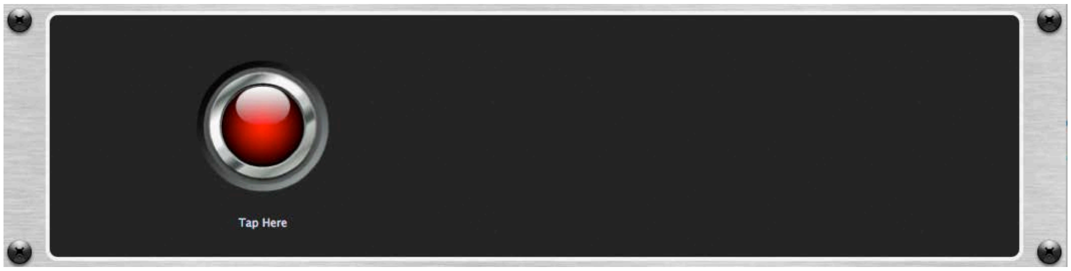
□ Only accept program change messages on specific channel 1

2. Click the Learn button to the right of Tap Tempo in the Global MIDI Assignments list.
3. Tap on your MIDI controller to complete the link.

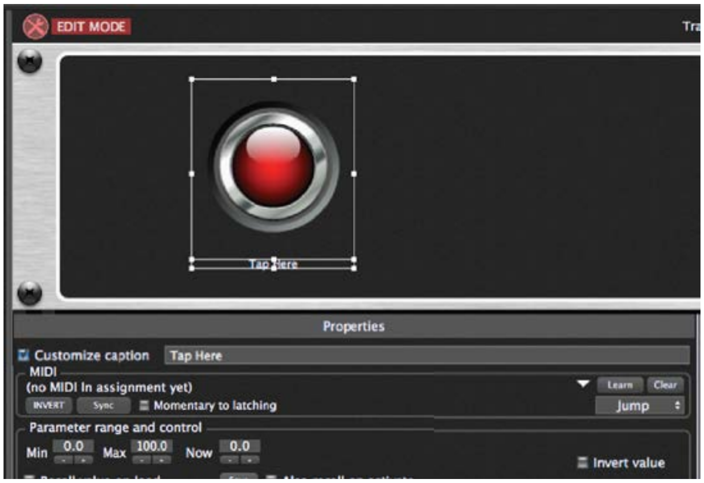
Creating an onscreen button to tap (for touch surfaces)

When controlling Gig Performer from a tablet, it may be easiest to tap a tempo in using an onscreen button, rather than a MIDI controller. GP Script can provide this functionality. Just follow these instructions exactly (or you can download this Rackspace and import it into a gig).

1. Switch to edit mode and drag a button (here, a red LED) on to a rack.



2. Select the button and,in the Name: field at the bottom of the Widget Properties area, type in "Tapper" and press return.



☑ Customize caption Tap Here

(no MIDI In assignment yet)

□ Momentary to latching

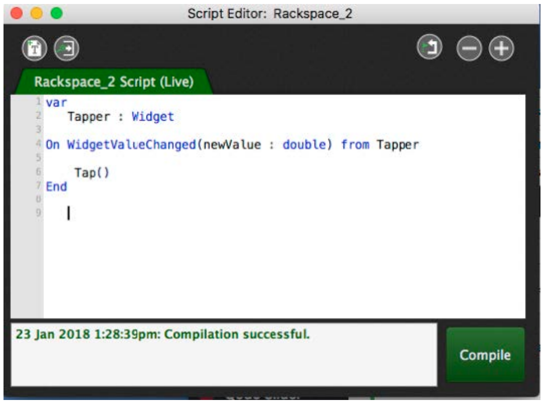
Min 0.0 Max 100.0 Now 0.0

□ Invert value

3. Open the Script Editor from the Windows menu and type in the following code exactly as shown in the image below.

```
Var

        Tapper : Widget

On WidgetValueChanged(newValue : double) from Tapper

        Tap()

End
```

4. When you've entered this code, press the Compile button and you should see a message with the current date indicating a successful compilation



```
var
   Tapper : Widget

On WidgetValueChanged(newValue : double) from Tapper

    Tap()
End
```

23 Jan 2018 1:28:39pm: Compilation successful.

5. You're done. Tap on the LED button and the tempo should change accordingly.

Tip: Creating Two Instances of Gig Performer with a Single Client ASIO Driver

Creating a second instance of Gig Performer is easy. Just select **File>Create new instance** (or open an existing one) and you're all set. This works great on OS X and on many Windows machines.

Unfortunately, some Windows ASIO drivers still allow only one application to use the audio interface at a time. These are known as single-client drivers. Multi-client audio drivers started coming out years ago, but, sadly, there still are many single-client drivers in place.

Two instances of Gig Performer count as two applications, which is a problem when running on a single-client driver.

If you find yourself in this situation, first check whether there is a multi-client update for your audio device driver available. Consider also contacting the company providing the driver; if they hear from enough people, perhaps they'll fix the issue.

If there is no update, there's still a way — just follow these instructions:

1. Install the ASIO4ALL drivers on your computer http://www.asio4all.org
2. Open a second GP instance and choose **Options>Audio setup** to open the Audio preferences tab.
3. Select the ASIO and ASIO4ALL driver matching the sample rate of your main instance.
4. Click on the "Control Panel" button once you have switched to the ASIO4ALL driver andmake sure that your real audio interface is selected within the ASIO4ALL driver.

That's it. Your second instance will remember the audio settings and will now alwaysopen using the correct driver.

Should you need to route MIDI from one Gig Performer instance to another you can install a virtual MIDI port driver (Virtual MIDI ports are built-in to OS X as IAC – InterApplication Communication - ports.)

LoopBe1 is a free MIDI loopback driver for personal use. You can download ithere http://www.nerds.de/en/loopbe1.html

There is also loopMIDI from Tobias Erichsen

https://www.tobias-erichsen.de/software/loopmidi.html

Once installed and configured, virtual MIDI ports simply show up in Gig Performer as input or output MIDI devices. Just send any MIDI data you want from one Gig Performer instance into a MIDI Out block for LoopBe1 and receive it in your other instance using a MIDI In block.

CAUTION: Do not use any MIDI In OMNI blocks if you're using this method. Otherwise you will set up a feedback loop because MIDI In OMNI blocks receive all incoming MIDI data, including data you just sent out.

APPENDIX D - LICENSE AGREEMENT

DESKEW TECHNOLOGIES, LLC - END USER LICENSE AGREEMENT FOR GIG PERFORMER

(“THESOFTWARE”) BY CLICKING THE ACCEPT BUTTON OR DOWNLOADING THE SOFTWARE, YOU AGREE TO BE BOUND BY AND BECOME A PARTY TO THIS AGREEMENT. DESKEW TECHNOLOGIES, LLC‘S ACCEPTANCE AND FULFILLMENT OF YOUR ORDER ARE EXPRESSLY CONDITIONED ON YOUR AGREEMENT TO THESE TERMS. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, YOU MAY NOT DOWNLOAD, INSTALL, EVALUATE OR USE THE SOFTWARE.CAREFULLY READ THE FOLLOWING LEGAL AGREEMENT (“AGREEMENT”) FOR THE LICENSE OF SOFTWARE YOU ARE ABOUT TO DOWNLOAD OR INSTALL, PROVIDED TO YOU UNDER THIS AGREEMENT AND THE ACCOMPANYING DOCUMENTATION (THE “DOCUMENTATION”). “DESKEW”MEANS DESKEW TECHNOLOGIES LLC. “YOU” MEANS THE INDIVIDUAL PERSON INSTALLING OR USING THE SOFTWARE ON HIS OR HER OWN BEHALF; OR, IF THE SOFTWARE IS BEING DOWNLOADED OR INSTALLED ON BEHALF OF AN ORGANIZATION, SUCH AS AN EMPLOYER,“YOU” MEANS THE ORGANIZATION FOR WHICH THE SOFTWARE IS DOWNLOADED OR INSTALLED, AND THE PERSON INSTALLING OR USING THE SOFTWARE REPRESENTS THAT HE OR SHE HAS THE AUTHORITY TO DO SOON YOUR BEHALF. YOU HEREBY EXPRESSLY CONSENT TO DESKEW’S PROCESSING OF PERSONAL DATA YOU PROVIDE TO DESKEW (WHICH MAY BE COLLECTED BY DESKEW OR ITS DISTRIBUTORS) ACCORDING TO DESKEW’S CURRENT PRIVACY POLICY WHICH IS INCORPORATED INTO THIS AGREEMENT BY REFERENCE (SEEhttp://www.deskew.com/privacy.html). IF “YOU” ARE AN ORGANIZATION, YOU WILL ENSURE THAT EACH MEMBER OF YOUR ORGANIZATION (INCLUDING EMPLOYEES AND CONTRACTORS) ABOUT WHOM PERSONAL DATA MAY BE PROVIDED TO DESKEW HAS GIVEN HIS OR HER EXPRESS CONSENT TO DESKEW’S PROCESSING OF SUCH PERSONAL DATA. PERSONAL DATA WILL BE PROCESSED BY DESKEW OR ITS DISTRIBUTORS IN THE COUNTRY WHERE IT WAS COLLECTED,OR IN THE LOCATION OF DESKEW OR ITS DISTRIBUTORS; UNITED STATES LAWS REGARDING PROCESSING OF PERSONAL DATA MAY BE LESS STRINGENT THAN THE LAWS IN YOUR JURISDICTION.

1. License Grant. Subject to the terms and conditions of this Agreement, DESKEW hereby grants you a non-exclusive, non-transferable right to reproduce and use a reasonable number of copies of the Software and the accompanying Documentation solely for your internal evaluation purposes during the term of this Agreement (as set forth in Section 3 below). You may install each authorized copy of the Software only on the computer, workstation, personal digital assistant, pager, “smart phone,” or other electronic device forwhich the Software was designed.

2. Ownership Rights. The Software is protected by United States copyright laws and international treaty provisions. DESKEW and its suppliers retain all right, title, and interest in and to the Software, including all copyrights, patents, trade secret rights, trademarks, and other intellectual property rights therein. Your possession, installation, or evaluation of the Software does not transfer to you any title to the intellectual property in the Software. All rights in the Software and Documentation not expressly granted to you in thisAgreement are reserved by DESKEW and its suppliers.

3. Restrictions. You may not rent, lease, loan, resell, or otherwise redistribute the Software or Documentation to any third party. You may not permit any third party to benefit from the use or functionalityof the Software via a timesharing, service bureau, or other arrangement. You may not transfer (by assignment, sublicense, operation of law, or otherwise) any of the rights granted to you under this Agreement; and any purported transfer will be null and void and result in immediate termination of this Agreement. You may not reverse engineer, decompile, or disassemble the Software, except to the extent (if at all) the foregoing restriction is expressly prohibited by applicable law. You may not modify, or create derivative works based upon, the Software or Documentation, in whole or in part. You may not copy the Software or Documentation, except as expressly permitted in Section 1 above. You may not remove any proprietary notices or labels on the Software or Documentation. The Software may contain a usage tracking mechanism for purposes of monitoring license compliance. You agree not to (a) disable or block the operation of this tracking mechanism and (b) delete tracking information that may be stored with the Software. DESKEW reserves the right to review such usage tracking information for the purpose ofverifying your compliance with this Agreement.

4. Warranty Disclaimer. THE SOFTWARE IS PROVIDED “AS IS.” TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, DESKEW DISCLAIMS ALL WARRANTIES, WHETHER EXPRESSOR IMPLIED OR STATUTORY, WITH RESPECT TO THE SOFTWARE AND THE DOCUMENTATION, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON INFRINGEMENT. Without limiting the generality of the foregoing, the Software is not fault-tolerant and is not designed or intended for use in hazardous environments requiring fail-safe performance, such as the operation of nuclear facilities, aircraft navigation or communicationsystems, air traffic control, weapons systems, direct life-support machines, and any other applicatio n inwhich the failure of the Software could lead directly to death, personal injury, or severe physical or property damage. You assume full responsibility for the installation of, use of, and results obtained from theSoftware. Without limiting the foregoing, DESKEW makes no warranty that the Software will be error-free.

5. Limitation of Liability and Indemnification. IN NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER IN TORT, IN CONTRACT, OR OTHERWISE, WILL DESKEW OR ITS SUPPLIERS BE LIABLE TO YOU OR ANY OTHER PERSON FOR ANY DAMAGES WHATSOEVER, INCLUDING DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY NATURE ARISING FROM OR RELATING TO THIS AGREEMENT OR THE SOFTWARE OR DOCUMENTATION, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, OR COMPUTER FAILURE OR MALFUNCTION, EVEN IF DESKEW HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THESE LIMITATIONS OF LIABILITY WILL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION.

6. United States Government. The Software and Documentation are deemed to be “commercial computer software” and “commercial computer software documentation,” respectively, pursuant to DFAR Section227.7202 and FAR Section 12.212, as applicable. Any use, modification, reproduction, release performance, display or disclosure of the Software and Documentation by the United States Governmentwill be governed solely by the terms of this Agreement and will be prohibited except to the extent expressly permitted by this Agreement.

7. Export Controls. You are advised that the Software is subject to the U.S. Export Administration Regulations. You will not directly or indirectly export, import, or transfer Software contrary to U.S. or other applicable laws, and will not cause, approve, or assist others to do so. You represent and agree that (a) neither the United States Bureau of Export Administration nor any other federal agency has suspended, revoked or denied your export privileges, and (b) you are not located in or a resident of any other country towhich the United States has embargoed goods.

8. Miscellaneous. This Agreement is governed by the laws of the United States and the State of New York,without regard to conflict of laws principles. The application of the United Nations Convention of Contracts for the International Sale of Goods is expressly excluded. Any dispute between the parties arising out of or related to this Agreement shall be settled by final, binding arbitration utilizing the dispute resolution procedures of the American Arbitration Association (AAA) in San Francisco, CA. Notwithstanding the foregoing, DESKEW may bring suit in any appropriate forum for any breach of Sections 1, 2, 5 or 10 or otherwise for infringement or misappropriation of its intellectual property rights. BY AGREEING TO ARBITRATION, THE PARTIES UNDERSTAND THAT THEY ARE WAIVING CERTAIN RIGHTS ANDPROTECTIONS WHICH MAY OTHERWISE BE AVAILABLE IN COURT, INCLUDING, WITHOUT LIMITATION, THE RIGHT TO A JURY TRIAL. DESKEW will be entitled to immediate injunctive relief (in addition to all other remedies it may have) in the event of any breach, or threatened breach, of Section 2 or Section 5 by Licensee. No right or remedy of DESKEW under this Agreement will be deemed waived unless such waiver is in writing and signed by an authorized representative of DESKEW. If any provision of this Agreement is held invalid, such provision will be deemed modified so that it is valid and enforceable to the maximum extent allowed under applicable law and the remainder of this Agreement will continue in full force and effect. The parties confirm that it is their wish that this Agreement has been written in the English language only and if this Agreement is translated into any other language, the English version will prevail.This Agreement is the entire agreement between the parties and supersedes any other written or oral communications or understandings with respect to the Software. This Agreement may not be modified except by a written document signed by authorized representatives of each party that expressly states their intention to amend this Agreement.