
Loomer String User Manual

End User Licence Agreement	4
Software Licence Agreement for String	4
Credits	6
Development Credits	6
Requirements	7
Mac OS X requirements	7
Windows requirements	7
Linux requirements	7
String Overview	8
What is String?	8
Plug-in vs Standalone	9
String FX	9
String formats	9
Supported channel configurations	10
Installation and Registration	11
Mac OS X installation	11
Windows installation	11
Linux installation	12
Product registration	12
Audio and MIDI Configuration	14
Setup Audio and MIDI	14
Preset Management	16
Parameters and Presets	16
Navigation Controls	16
Opening Presets	16
Preset Hot-Swap	17
Preset Meta Data	17
Searching for Presets	17
Categories	18
Saving Presets	18
XML / FXP / FXB support	18
Copying & Moving Presets	18
MIDI Program Change Lists	18
Create a new Preset	19
Compare a changed Preset to the original Preset	19
Restoring Deleted Factory Presets	19
User folder management	19
Undo / Redo	19
MIDI Controller Mapping	20
MIDI controllers	20
Display current MIDI controller mappings	20
Assign MIDI controllers to controls	20

Remove MIDI controller mappings from controls	21
Using String	22
Instrument Interface	22
Layers	23
Select Modulation / Effect Tab	24
Modulation	25
Effects	26
Note Splits	27
Output	27
FX Interface	28
Control types	29
Status Display	30
CPU Usage monitor	30
MIDI input monitor	30
Active Voices monitor	30
Tempo control	31
Contact Details	32
Loomer Online	32
Email	32

End User Licence Agreement

Software Licence Agreement for String

1. GRANT OF LICENCE: Subject to the terms below, Loomer hereby grants you a non-exclusive, non-transferable licence to install and to use String ('Software'). Under this licence, you may: (i) install and use the Software for your personal, internal use (ii) copy the Software for back-up or archival purposes; (iii) use an unregistered or evaluation version of the Software for evaluation purposes only. (iv) You may not distribute the software to others without first obtaining the required licences, where applicable. Whether you are licensing the Software as an individual or on behalf of an entity, you may not: (i) reverse engineer, decompile, or disassemble the Software or attempt to discover the source code; (ii) modify, or create derivative works based upon, the Software in whole or in part without the express written consent of Loomer; (iii) distribute copies of the Software; (iv) remove any proprietary notices or labels on the Software; (v) resell, lease, rent, transfer, sublicense, or otherwise transfer rights to the Software; By installing, copying, or using the Software, you (either on behalf of yourself as an individual or on behalf of an entity as its authorized representative) agree to all the terms of this End User Licence Agreement ('Agreement') regarding your use of the software. If you do not agree with all the terms of this Agreement, you may not use the Software.

2. String: You acknowledge that no title to the intellectual property in the Software is transferred to you. Title, ownership, rights, and intellectual property rights in and to the Software shall remain that of Loomer . The Software is protected by copyright laws and international treaty provisions.

3. DISCLAIMER OF WARRANTY: You agree that Loomer has made no express warranties, oral or written, to you regarding the product and that the products are being provided to you 'as is' without warranty of any kind. Loomer disclaims any and all other warranties, whether expressed, implied, or statutory. Loomer shall not be liable for indirect, incident, special, cover, reliance, or consequential damages resulting from the user of this product.

4. LIMITATION OF LIABILITY: You use this program solely at your own risk. In no event shall Loomer be liable to you for any damage, including but not limited to any loss, or other incidental, indirect or consequential damages of any kind arising out of the use of the software, even if Loomer has been advised of the possibility of such damages. In no event will Loomer be liable for any claim, whether in contract, tort, or any other theory of liability, exceed the cost of the software. This limitation shall apply to claims of personal injury to the extent permitted by law.

5. TERMINATION: This Agreement shall terminate automatically if you fail to comply with the limitations described in this Agreement. No notice shall be required to effectuate such termination. Upon termination, you must remove and destroy all copies of the Software.

6. MISCELLANEOUS: In the event of invalidity of any provision of this Agreement, the parties agree that such invalidity shall not affect the validity of the remaining portions of this Agreement. You agree that this is the entire agreement between you and Loomer, which supersedes any prior agreement, whether written or oral, and all other communications between Loomer and you relating to the subject matter of this Agreement.

Reservation of rights.

'VST' is a trademark of Steinberg Soft - und Hardware GmbH. All other trademarks are the property of their respective owners.

All rights not expressly granted in this Agreement are reserved by Loomer.

Development Credits

String © 2009 - 2014 Loomer

Developed by Loomer
Interface by Kevin Barry - <http://www.inquer.co.uk>

Testers Clifton Cameron
kuniklo
Richard Michael
Scott Simons (rexlapiin)
taoyoyo

VST PlugIn Technology by Steinberg Media Technologies GmbH. This software is based in part on libpng, zlib, and JUCE.

Requirements

Mac OS X requirements

- Requires Mac OS X v. 10.5 or later
- x86 based processor 1.0Ghz, or above, with SSE
- 512 MB RAM

Windows requirements

- Windows XP or Windows Vista, or above
- x86 based processor 1.0Ghz, or above, with SSE
- 512 MB RAM

Linux requirements

- x86 32-bit or or 64-bit based processor 1.0Ghz, or above, with SSE
- 512 MB RAM

The following shared object libraries are required:

- libfreetype
- libasound
- libXinerama

String Overview

What is String?

String is a virtual instrument that emulates the beautiful sound of the polyphonic string synthesizer.

String's authentic tone comes from its accurate emulation of the divide-down oscillator technology behind the lush ensemble synths of the 70s.

String has two individual sound generating Layers, each housing three phase-locked oscillator stops at 16', 8' and 4' ranges for Layer 1, and 8', 4', and 2' ranges for Layer 2. The relative volumes of each stop can be finely tuned to produce an enormous range of timbres. Both Layers have controllable note split points. Layers can be placed so that each is controlled by a specific range on the keyboard, or even overlapped and stacked together.

Each Layer has its own Voicing Mode from a choice of Monophonic with Last, Low, and High Note Priorities, Paraphonic, and Polyphonic. In Polyphonic mode each voice has its own Attack/Release amplitude modulation envelope. In Paraphonic mode all voices within a Layer share a single monophonic Attack/Release amplitude envelope. The Monophonic Modes produce a single voice, which can be either the Lowest, Highest, or Last note pressed.

Layers can be further shaped using the monophonic Filter section. This section has several Filter types; a Highpass / Lowpass pair of filters in series, an analogue resonant filter in either Lowpass, Bandpass, or Highpass modes with various modulation options (including Channel Aftertouch), and a 9-Band EQ. The Filter section can also be switched off altogether.

Both Layers can be individually routed into the Effects section. Layer 1 can also optionally be routed directly into Layer 2's Filter to produce serial Filter tones.

String's Chorus section is modeled on the bucket-brigade device that characterized the sound of the classic ensemble synths. Select from two different Chorus algorithms for either an authentic Ensemble timbre (Type 1), or a more modern, denser sound. (Type 2). A built-in stereo Phaser can be used to add movement, depth, and interest to String timbres. The Effects section is rounded off with two mono bucket-brigade device delay units with independent time and feedback settings, which can be tempo synced, and linked together into a single stereo delay.

All controls can be easily mapped to a hardware controller using the simplified MIDI Learn feature.

Plug-in vs Standalone

Depending upon your individual studio setup, String can either be used as a plug-in component of a host application, or as a standalone application requiring no other dependencies. Generally, by using the host application's routing flexibility, running as a plug-in allows easier integration with existing plug-in effects and instruments. If this integration is not required, the standalone version, without the extra layer of complexity added by the host consuming resources, performs slightly better.

Presets and MIDI Mapping assignments are identical in both standalone and plug-in versions. This means any sounds created in one format can be opened in the other.

String FX

External instruments can also be processed using String's powerful Effects Section with the StringFX plug-in.

String formats

For the Mac, the following formats are provided:

- Standalone Application
- Audio Unit (AU) Plug-In (32-bit and 64-bit)
- VST Plug-in (32-bit and 64-bit)
- RTAS Plug-In
- AAX Plug-In (64-bit)

These formats are available for Windows PC:

- Standalone Application
- VST Plug-in (32-bit and 64-bit)
- RTAS Plug-in
- AAX Plug-In (64-bit)

These formats are available natively for Linux:

- Standalone Application (32-bit and 64-bit)
- VST Plug-in (32-bit and 64-bit)

Supported channel configurations

For maximum flexibility, String, as an instrument, can be used in several possible output channel configurations. The following configurations are possible:

- mono output
- stereo output

String, as an effect plug-in, can be used in the following channel configurations:

- mono to mono (1 input, 1 output)
- mono to stereo (1 input, 2 outputs)
- stereo to stereo (2 inputs, 2 outputs)

Note that some host applications may not support one or more of these configurations; consult your host's documentation for further details.

Installation and Registration

Mac OS X installation

1. Start the installation process by double-clicking the String installer application.
2. Read the introduction screen dialog text. Press **Continue** when you are ready to proceed.
3. Read the Licence Agreement. Click **Continue**, then **Agree** to confirm you have read and agree to the licence terms.
4. Select a destination volume into which to install String, and then click **Continue**.
5. From the list of installable components, uncheck any formats that you do not wish to install. Click **Continue** to proceed.
6. To complete the installation, press **Install**.

The components are, by default, installed into the following directories:

- VST plug-in into */Library/Audio/Plug-Ins/VST*
- Audio Unit plug-in into */Library/Audio/Plug-Ins/Components*
- RTAS plug-in into */Library/Application Support/Digidesign/Plug-Ins*
- AAX plug-in into */Library/Application Support/Avid/Audio/Plug-Ins*
- Standalone application into */Applications*
- Documentation into */Library/Documentation/Loomer*

Windows installation

1. Start the installation process by double-clicking the String installer application, Setup.exe.
2. Read the welcome screen dialog text. Press **Next** when you are ready to continue.
3. Read the Licence Agreement. Click **I Agree** to confirm that you have read and agree to it.
4. From the list of installable components, uncheck any formats that you do not wish to install. Click **Next** to continue.
5. If you have chosen to include the VST plug-in format, select the directory into which it will install. Click **Next** to continue.
6. Select a directory into which the common files will install. The user documentation and the Standalone application (if selected in the the list of components), will be placed in this location. To complete the installation, press **Install**.

Unless any other directories are selected during the installation, the components are installed into the following directories:

- VST plug-in into the VST plug-ins folder, which is, unless configured otherwise, *C:\Program Files\Steinberg\VSTPlugins*
- RTAS plug-in into *C:\Program Files\Common Files\Digidesign\DAE\Plug-Ins*
- AAX plug-in into *C:\Program Files\Common Files\Avid\Audio\Plug-Ins*
- Standalone application, documentation and default Bank into *C:\Program Files\Loomer\String*. On a 64-bit Windows platform, the Standalone application will install into the *Program Files (x86)* folder.

Linux installation

String for Linux is distributed as a standard Linux tarball. To install, extract the package using any unarchiving utility; the following command-line invocation will suffice:

```
tar xzvf String.tar.gz
```

If you have a Linux VST host and have configured your VST Path, the VST plug-in can be moved into the required directory with the following commands:

```
cd String
mv StringVST.so $VST_PATH
```

Without the VST Path configured, you will need to ensure that your host uses the String directory when searching for VSTs; consult your host's documentation for information on how to do this.

String requires the following shared object libraries to be installed:

- libfreetype
- libasound
- libXinerama

Without these libraries installed, the application will not launch. Install the libraries using your distribution's particular package management system, such as apt-get; consult your Linux distribution documentation for more information.

Product registration

Having installed String, it will run in evaluation mode with the following limitations:

- User created Presets can be saved, but not loaded.
- Audio output will stop after 30 minutes of continuous use. The only way to continue using String is to restart the application.
- Output will be periodically interrupted, approximately every 30 seconds, by a short moment of silence.

These limitations can be removed by purchasing a licence and registering your licence information into the product. Note that registering String on a particular machine will remove the evaluation limitations for all String plug-in formats installed on this machine; you don't need to individually register all plug-in formats. Licence information will be in the form of a user name and licence key.

To purchase a String licence:

1. Launch String. This can be either the standalone application, or as a plug-in.
2. From the **Options** menu, choose **Purchase String Licence**. This will launch the shop website, from where a String Licence can be purchased by clicking **Add To Cart**, followed by **Checkout**. Follow the on-screen prompts to enter your payment details and confirm your purchase.
3. Once you have purchased String, you will receive your licence key information via email. It is important to keep this information safe because you will require the licence key if you ever need to reinstall and re-register String.

To register your copy of String:

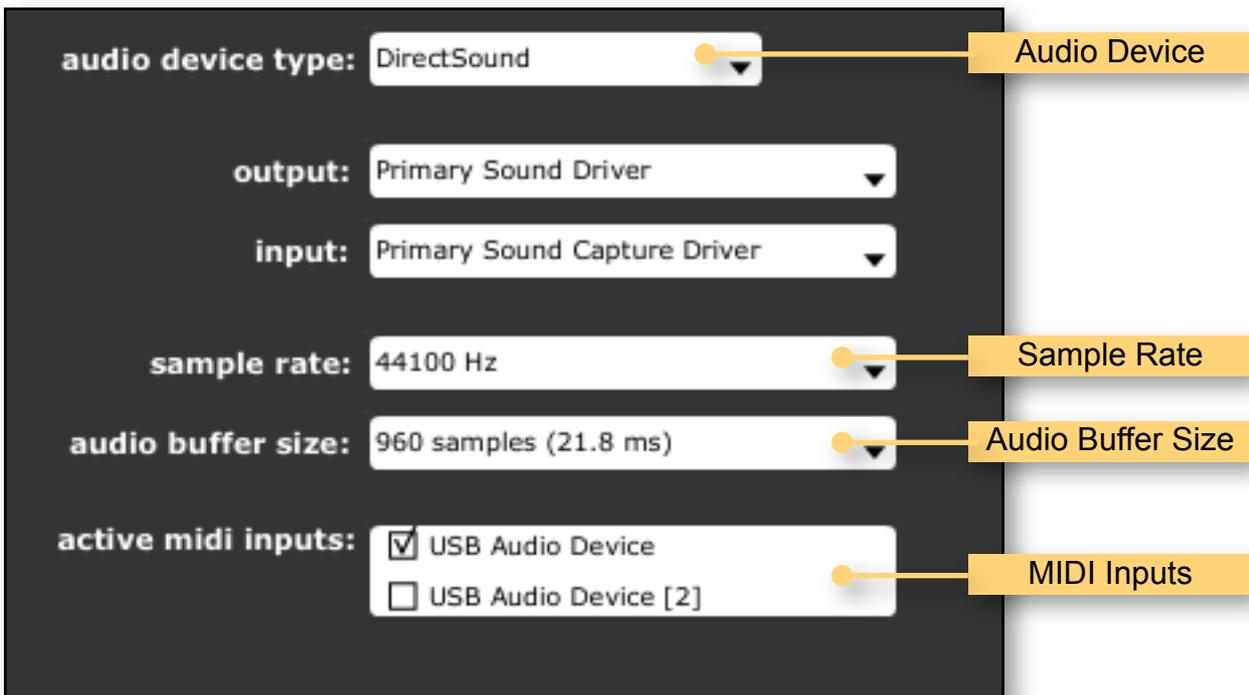
1. Choose **Enter String Licence** from the **Options** menu.
2. Enter your name in the **Name** field. This must match exactly the name that was used to purchase your String licence, including capitalization of letters.
3. Enter your licence key in the **Licence Key** field. This must match exactly the licence that you received when you purchased String, including any hyphens. We recommend using Copy and Paste to ensure that the Licence Key is entered exactly as specified in your registration email.
4. Press **OK** to confirm that your name and corresponding licence key have been entered. You will be informed that the application must be restarted in order for registration changes to take effect. If you are currently running the standalone application, close and relaunch it. If String is currently running as a plug-in, remove it from the host and add a new instance of it.
5. String should now be registered. If the label on the bottom of the interface still reads Unregistered Evaluation, you must have entered the licence information incorrectly. Go back to step 1, paying careful attention that the name and licence key entered match exactly the licence information in the registration email.

If you are having problems registering your installation of String, please consult the detailed step-by-step instructions at <http://www.loomer.co.uk/support.htm> .

Audio and MIDI Configuration

Setup Audio and MIDI

When running String as a Standalone application, you will need to setup your audio and MIDI interfaces. Note that when running as a plug-in, audio and MIDI routing is provided by the hosting application; consult the host's documentation for details. To open the Audio & MIDI Configuration dialog, select **Audio & MIDI Configuration** under the **Options** menu. The following dialog will be displayed:



From this dialog, the following details of your Audio and MIDI setup can be configured:

- **Audio Device.** Select the Audio Device you want to use from the list of available devices. Certain devices tend to perform better than others, so if multiple devices are available, you should favour: on Mac, using Core Audio; on Windows, using ASIO; on Linux, using JACK.
- **Sample Rate.** The Sample Rate dictates the overall quality of the audio output. Using a lower Sample Rate will reduce the presence of high-frequency components. Higher Sample Rates should therefore be preferred. However, be aware that the CPU Usage is directly proportional to the Sample Rate. A good compromise between quality and CPU Usage is 44100Hz, which is the Sample Rate of CD audio. Select the Sample Rate you require from the available list.

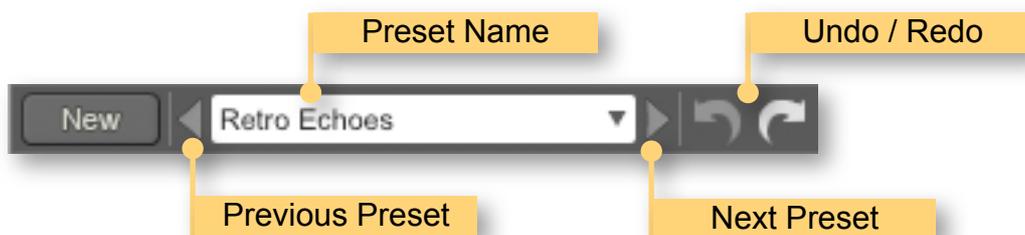
- **Audio Buffer Size.** The Audio Buffer Size governs latency; a smaller buffer means that String will respond more quickly to MIDI messages and Parameter changes. However, a smaller Audio Buffer Size will require more CPU Usage. Select the required Audio Buffer Size from the available list. Using too small of an Audio Buffer can overload your computer to the extent that audible clicks are heard. In this case, increase the Audio Buffer Size until clicks are no longer present.
- **MIDI Inputs.** Check any MIDI Input devices that you want to use with String.

Preset Management

Parameters and Presets

You can modify the audio output that String produces by setting the value of *Parameters*. Each Parameter, such as Volume, Filter Type, or Cutoff, will affect a specific element of the sound. The state of all Parameters are collectively called a *Preset*. Presets are given a suitable name, displayed in the Menu Bar, to describe the sound they produce.

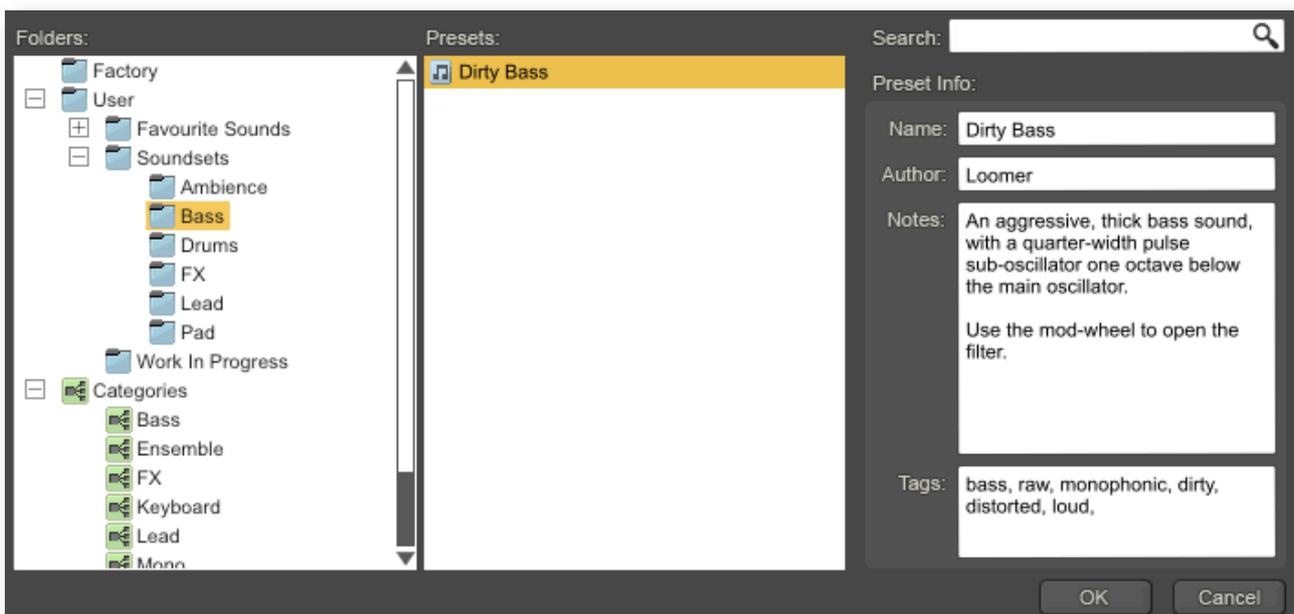
Navigation Controls



If you wish to browse quickly through the Presets, you can do so using the the Previous and Next controls. These controls are context sensitive, only displaying Presets that are relevant to you. For example, if you search for “analogue, bass”, only Presets that match this will be shown in the preset browser. The mouse-wheel can also be used to quickly browse through presets.

Opening Presets

To select a preset from within your preset library, click on the Preset Name in the menu bar to open the Preset Browser.



Presets can be divided into two groups; **Factory**, and **User**. Factory Presets are a collection of example Presets built into String. Factory presets can't be overwritten, although you can, of course, edit and save your own presets derived from Factory Presets.

User Presets are stored as individual files on your computer's hard drive. User Presets are found in the following location:

Mac OS X: *~/Library/Application Support/Loomer/String/Presets/*

Linux: *~/Loomer/String/Presets/*

Windows: *My Documents/Loomer/String/Presets/ or Documents/Loomer/String/Presets/*

Only registered users can open User Presets. Unregistered versions of String are limited to the Factory Presets. Unregistered versions of String can, however, still save Presets. Any Presets created in the unregistered evaluation of String can be opened by purchasing a licence for String.

Preset Hot-Swap

Selecting a Preset in the browser will automatically load the preset: this functionality is called Preset Hot-Swap. Preset Hot-Swap allows you to preview Presets in the context of the current project. If you decide to use the selected Preset, close the browser by pressing *return* or by clicking **OK**. To return to your original Preset, close the browser by pressing *escape* or by clicking **Cancel**.

Preset Meta Data

Presets can have meta data attached to them. Meta data doesn't influence the sound of the Preset, but is used in categorizing, searching, or annotating them. The following meta data is available:

Author: The name of the Preset creator.

Notes: Any additional notes about the Preset, such as influences, playing tips, description, etc.

Tags: Tags are short words that describe the Preset. Examples are: "bass", "distorted", "monophonic", "trance", "quiet", etc. Tags are used to help quickly search for a specific sound.

Searching for Presets

Type keywords into the search field to locate all Presets that match this description. The search will match the keywords in the Preset Name, Author, and Tags fields, and parent folder name.

Categories

Categories allow you to quickly access a subset of your Presets. Categories are listed under the Categories node in the Preset Browser. A Category is simply a shortcut to searching for the Category name; a Category called “Bass” will list all Presets that match the search criteria “Bass”. You are not limited to create Categories based on the type of sound: you can, for example, create Categories for Preset Authors, or indeed for any other search term. Provided you tag your own Presets, they will automatically be added to the correct Category (or Categories: Presets can belong to multiple Categories; a Preset could be both a “Bass” and “Lead” sound, for example.)

Saving Presets

Click **Save** to save the current Preset. Choose the location in which to save the Preset in the Folders window. Press **OK** to actually save the Preset.

XML / FXP / FXB support

Presets in the User Presets folder need to be in either native .xml format, or VST .fxp or .fxb format. Standalone, Audio Unit and RTAS versions can also open .fxp or .fxb files within the browser. Please ensure any files in the User Presets folder have the correct file extension.

Copying & Moving Presets

Presets can be moved or copied to another folder by dragging and dropping a Preset onto the destination folder. By default, Presets will be moved. You can instead copy the Preset by holding *shift* whilst dragging. Note that Presets inside Banks (either .xml or .fxb Banks), can only be copied, not moved. Multiple Presets can be moved or copied at once by selecting more than one Preset.

Default Preset

The Default Preset is automatically opened when a new instance of String is created. To specify the Default Preset, right-click on a Preset and choose **Set as Default**.

MIDI Program Change Lists

You can change Presets by using MIDI Program Change commands. This requires a MIDI keyboard or a control surface with the ability to send MIDI Program Change commands; consult your controller’s documentation for details. If using String as a plug-in, this functionality is dependent upon the host correctly forwarding MIDI messages to the plug-in; check the documentation provided with your host for details.

To select which Presets are loaded in response to MIDI Program Change commands, you will need to create a MIDI Program Change List. Select **Edit MIDI Program Change List...** from under the **MIDI** menu to display the MIDI Program Change List editor.

To add Presets to the MIDI Program Change List, either double-click on a Preset, or select a Preset and click the >> button. Use the **View** button to toggle between viewing the MIDI Program Change List and information on the Selected Preset.

Any Presets in the MIDI Program Change Lists are automatically loaded and cached in memory. This means that Preset changes can be instant. There may be a slight pause after editing the MIDI Program Change List whilst the Presets within it are loaded. When a Preset is saved, any entries in the MIDI Program Change List are saved alongside it. Note that setting a Preset with a MIDI Program Change List as the Default Preset will cause all Presets within the list to be loaded when a new instance of String is created.

Create a new Preset

Select **New** to create a new Preset. This Preset will be in the default initialized state.

Compare a changed Preset to the original Preset

Often you'll want to compare a Preset that you have amended to see if it is actually an improvement on the original. Pressing the **A | B** button temporarily reverts a changed Preset back to its original state. Pressing the **A | B** button a second time returns to the changed Preset. This method allows you to easily contrast the Preset changes you have made to the original Preset.

Restoring Deleted Factory Presets

To reinstate any deleted Factory Presets, right-click on the Factory branch and choose **Resore Factory Presets**.

User folder management

Folder management is performed by right-clicking on a User folder, and selecting an option from the pop-up menu. From here, you can create a **New Folder**, **Rename Folder**, or **Delete Folder**. On Windows or Mac OS X, you can also quickly locate this folder in Finder or Explorer by choosing **Show In Finder / Explorer**.

Undo / Redo

By pressing **Undo**, you can take back the most recent change made to a preset. **Redo** will re-apply the last Undo changes. Note that only changes made via the Sequent interface can be undone; changes made externally (such as from a Host application, or from a MIDI controller) are not eligible for Undo.

MIDI Controller Mapping

MIDI controllers

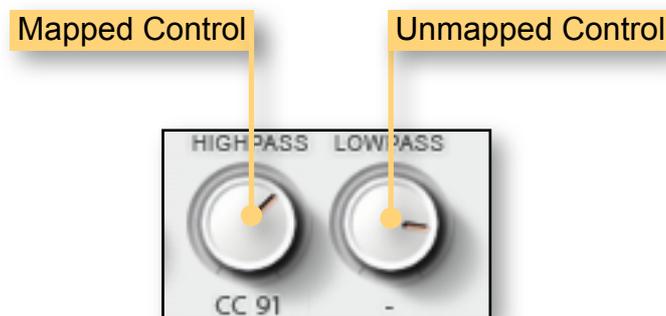
Instead of configuring a Program using a mouse, you can change any Parameter values using a MIDI keyboard or a control surface. This is done by mapping each Parameter to a *MIDI Continuous Controller (CC)*. A Parameter can only be mapped to a single MIDI controller; the same MIDI control can, however, be mapped to several different Parameters.

MIDI Controller mappings are global; once defined, the same mapping assignments will be used by all String Programs. This means that you only need define mappings once for your particular controller.

Display current MIDI controller mappings

To view the current MIDI controller mappings:

1. Enter MIDI Learn Mode by choosing **Start MIDI Learn** under the **Options** menu. Each Parameter's mapped MIDI controller will now be displayed beside it in yellow text. Parameters without any mapping will be labelled -.
2. When you have finished viewing the MIDI controller mappings, choose **Stop MIDI Learn** under the **Options** menu.



Assign MIDI controllers to controls

To map a MIDI controller to a Parameter:

1. Enter MIDI Learn Mode by choosing **Start MIDI Learn** under the **Options** menu.
2. Click on the Parameter you wish to map to a MIDI controller.
3. Turn your MIDI controller. The Parameter should now be labelled with the MIDI CC number of this controller.
4. You can now either:
 - Map another Parameter by repeating these steps from Step 2.
 - Finish mapping Parameters by choosing **Stop MIDI Learn** under the **Options** menu.

Remove MIDI controller mappings from controls

To remove a mapped MIDI controller from a Parameter:

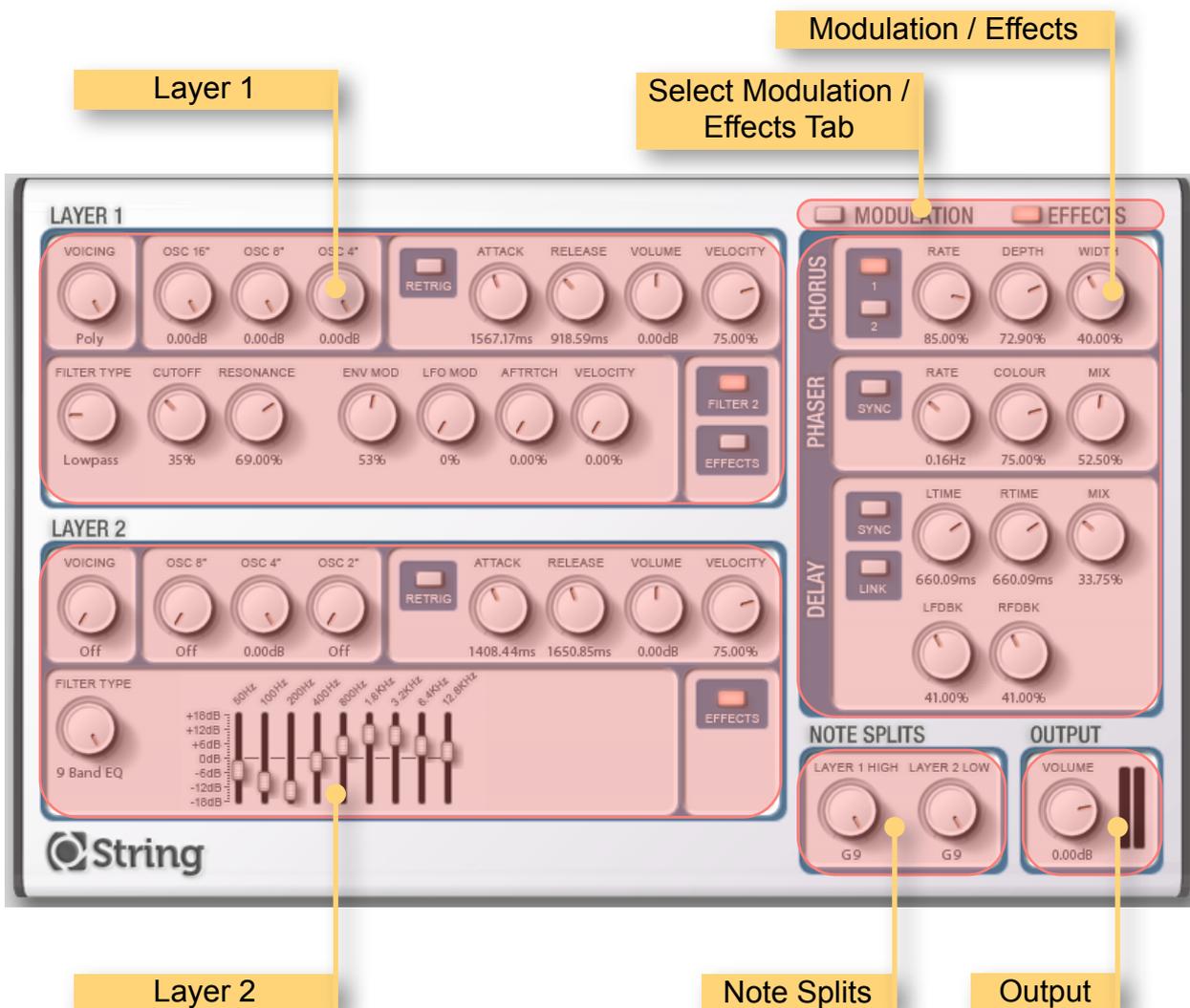
1. **Enter MIDI Learn Mode** by choosing Start MIDI Learn under the **Options** menu.
2. Click on the Parameter from which you wish to remove the mapping. This Parameter will become unmapped, and the CC number label on the Parameter will disappear to reflect this.
3. You can now either:
 - Remove the mapping from another Parameter by repeating these steps from Step 2.
 - Finish removing Parameter mappings by choosing **Stop MIDI Learn** under the **Options** menu.

Using String

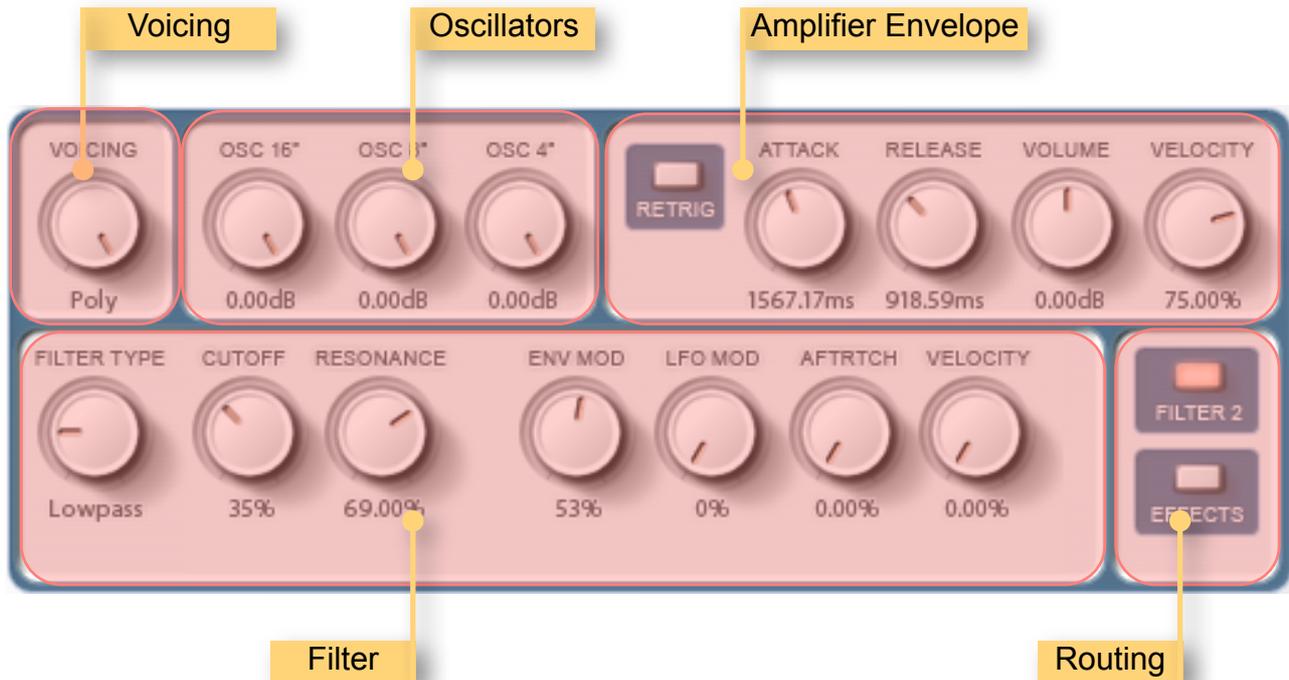
String can either be used a Standalone application, or as a plug-in. When used as a plug-in effect, String is placed directly in a host instrument channel. Consult your host's documentation to find out how to add a plug-in instrument.

String comes as two separate plug-ins: String, and StringFX. The FX versions are shipped as separate plugins, and identified with the _FX prefix. Add an instance of String to use the product as a powerful ensemble synthesizer. Add an instance of StringFX to use the product as an effects processor of other external instruments.

Instrument Interface



Layers



The **Voicing** Parameter selects the number of voices that the Layer will produce, and which note has priority. When only a single Layer is required, selecting **Off** turns the unused Layer off completely. In **Mono** mode, only one note - the last note played - is produced. **MonoLow** generates the lowest note played in the Layer's Note Split range. **MonoHigh** also generates a single note, that being the highest note played. In **Paraphonic** mode, all polyphonic voices within the Layer share a single monophonic Amplifier Envelope. In **Polyphonic** mode, each polyphonic voice has its own Amplifier Envelope.

Sound is generated by the divide-down Oscillators. Each Layer has three Oscillators tuned an octave apart. Layer 1 has **Osc 16"**, **Osc 8"**, **Osc 4"**. Layer 2 is tuned an octave higher, having **Osc 8"**, **Osc 4"**, **Osc 2"**. The relative volume levels of each Oscillator can be controlled to produce a huge array of different timbres. Oscillators can be turned off entirely by setting them to **Off**.

The Amplifier Envelope consists of a simple Attack / Release curve that controls the volume of the sound. The **Attack** Parameter governs how long it takes for the sound to fade in when a note is played. The **Release** Parameter controls how long it takes for the sound to fade out after a note is released.

By turning on **Retrig**, the curve will always start again from the beginning of the Attack stage, giving a swelling feel to the sound.

The **Volume** control is used to give a boost or cut to the loudness level on the entire layer. This control allows for easier relative volume balancing of Layers 1 and 2. At 0.0dB, no boost or cut is applied to the Layer.

The **Velocity** Parameters specifies how much the overall volume of the Amplifier is controlled by playing dynamics. At 0%, notes will always play at the same volume, At 100%, the Amplifier volume is entirely controlled by the MIDI Note Velocity and the full loudness range, from near silence to very loud, is possible by playing articulations.

Each Layer has a monophonic **Filter Section** for further sound shaping. The **Filter Type** Parameters specifies the type of Filter.

Off bypasses the Filter altogether, allowing the raw Oscillator sound through.

Lowpass, **Bandpass**, and **Highpass** are analogue resonant filters with a variety of modulation options. Lowpass allows only frequencies below the Cutoff through. Bandpass allows only frequencies around the Cutoff through. Highpass allows only frequencies above the Cutoff through. The **Cutoff** frequency specifies the point at which frequencies begin to attenuate. **Resonance** adds emphasis at the Cutoff point. **Env Mod** control the amount of Cutoff modulation from the Modulation Envelope. Likewise, **LFO Mod** controls the amount of Cutoff modulation from the Modulation LFO. **Aftertouch** allows monophonic channel Aftertouch to modulate the Cutoff frequency. **Velocity** controls the amount that playing dynamics affect the Cutoff frequency.

HP / LP Pair are Highpass and Lowpass filters in series. The **Highpass** Parameter controls the point at which low frequencies begin to attenuate. **Lowpass** controls the point at which high frequencies attenuate. Together, these act as a very flexible Bandpass filter.

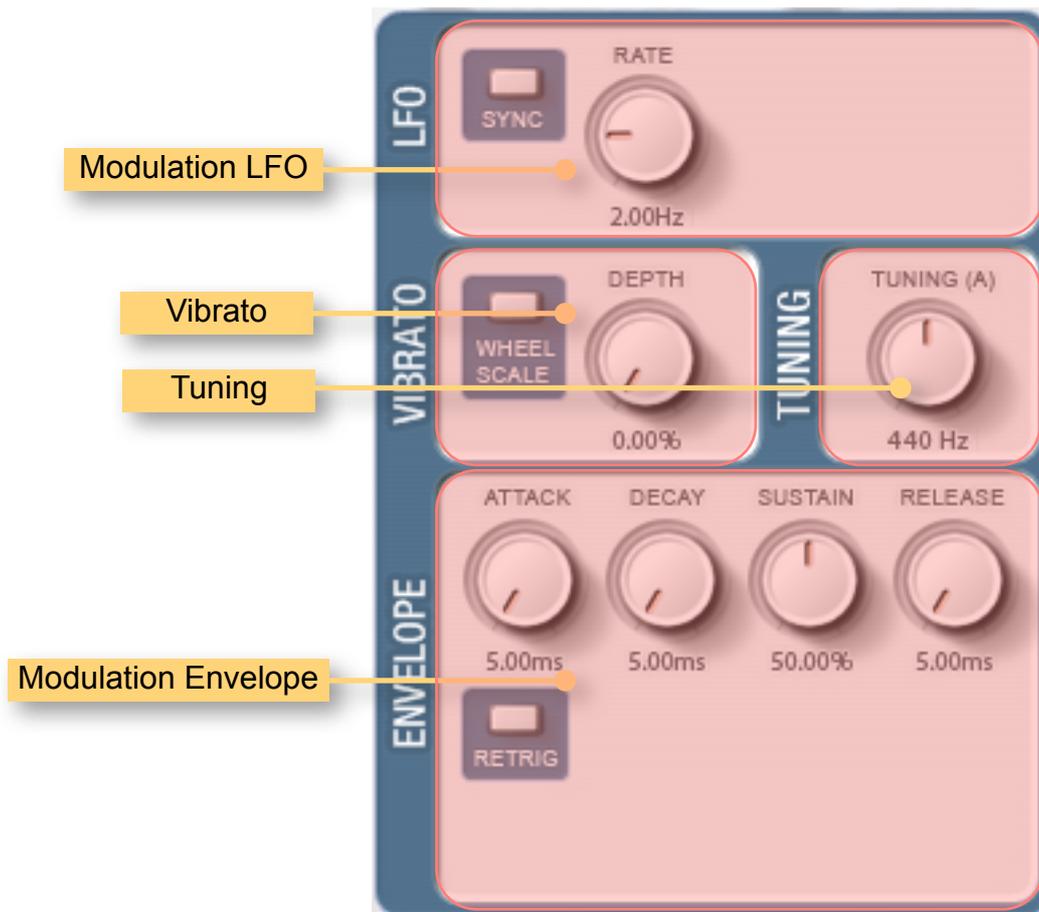
9 Band EQ is a graphic equalizer type filter, with 9 bands at **50Hz**, **100Hz**, **200Hz**, **400Hz**, **800Hz**, **1.6KHz**, **3.2KHz**, **6.4KHz**, and **12.8KHz**. Each frequency band can be amplified for up to 18dB of gain, or -18dB of attenuation.

Filter 2 routing allows Layer 1's post Filter output to be routed into Layer 2's Filter section. Effects allow the post Filter output to be routed to the Effects section. Note that when **Filter 2** routing is **On**, Layer 1's Effects button is bypassed.

Select Modulation / Effect Tab

Only one of these tabs can be visible at a time. Select which tab is visible using the **Modulation** and **Effects** button.

Modulation



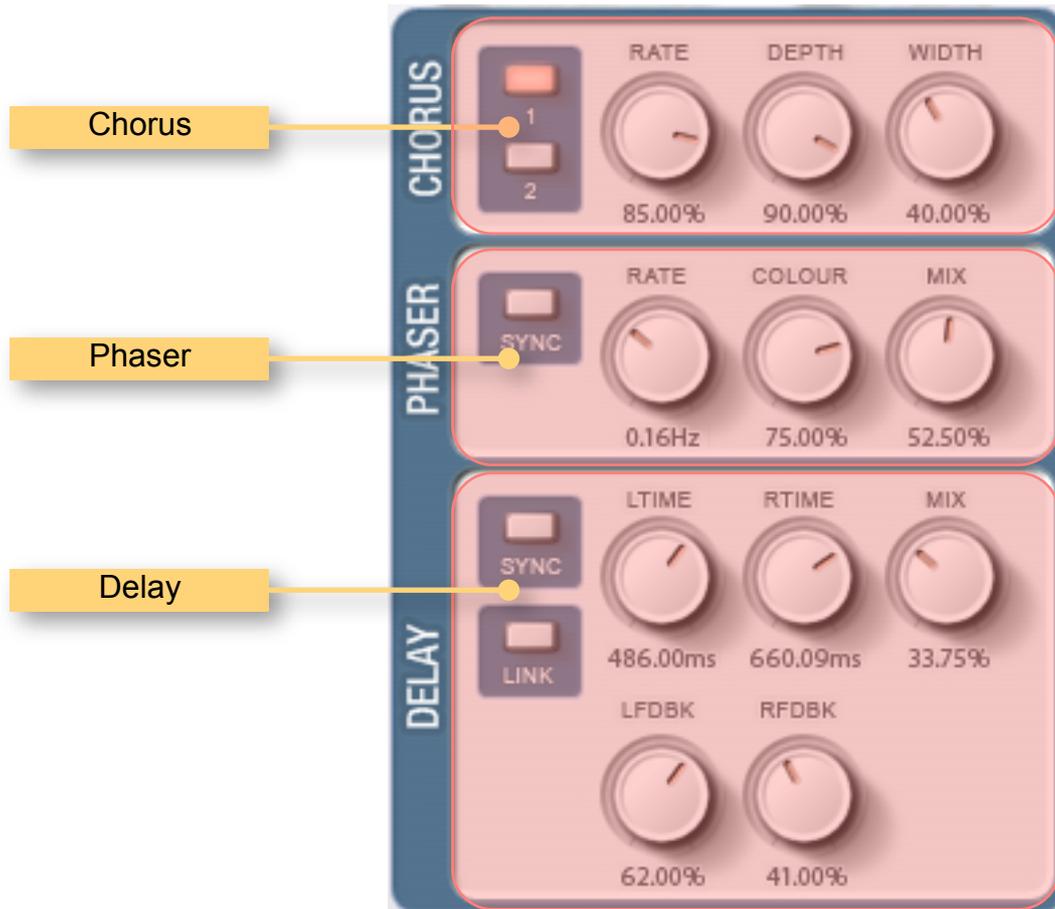
The **Modulation LFO** can be used to modulate the Oscillator pitch via the Vibrato section, or the Lowpass, Bandpass, or Highpass Cutoff via the Filter Env Mod. **Rate** controls the speed the LFO. When **Sync** is enabled, the Rate is specified in musical subdivisions and synced to the host tempo. With Sync off, the Rate is specified in Hertz.

In the **Vibrato** section, the **Depth** Parameters controls the amount that the Oscillator Pitches are modulated by the Modulation LFO. When **Wheel Scale** is enabled, the vibrato depth is controlled by the **Modulation Wheel**, up to the maximum specified by the Depth.

The **Tuning** parameter sets the overall tuning. This value is the frequency of the A note above middle C, in Hertz.

The **Modulation Envelope** is a monophonic modulation source that can be used to change the Cutoff frequency of the Lowpass, Bandpass, or Highpass filters. The amount of change is specified by the filter's **Env Mod** Parameter. The **Attack** time governs the time the envelope takes to reach the initial maximum. It then falls to the steady-state **Sustain** level over the time specified by the **Decay**. Finally, when all notes have been released, the envelope decays to zero over the **Release** time. By turning on **Retrig**, the curve will always start again from the beginning of the Attack stage when a note is played.

Effects



The **Chorus** effect produces the thick ensemble sound of a string synth. String's Chorus section is modeled on the bucket-brigade device that characterized the sound of the classic ensemble synths. Select from two different Chorus algorithms for either an authentic ensemble timbre (**Type 1**), or a more modern, denser sound. (**Type 2**). The speed of the Chorus's pitch modulation is controlled with the **Rate** Parameter. The amount of pitch modulation is controlled with the **Depth**. Turning Depth to 0% will bypass the Chorus. The **Width** Parameter controls the amount of Chorus stereo spread.

The **Phaser** effect is a sweeping filter that adds moving peaks and troughs in the signal spectrum. The speed of the movement is controlled with the **Rate** Parameter, expressed in Hertz when **Sync** is off, and musical subdivisions when it is on. The **Colour** Parameter alters the width of the peaks and troughs, as well as the amount of movement, by introducing feedback into the Phaser signal. **Mix** is used to balance the original 'dry' signal with the effected 'wet' one. At 0%, only the dry signal is heard; at 100%, only the wet signal.

String's Delay effect produces a series of echoes. String has two independent mono bucket-brigade device delay units, which can be linked to form a single stereo delay. Turning **Link** on will connect the Left and Right delay unit controls together so that either set of controls will affect both channels. With Link off, the Left and Right channels will act independently. The **Left Time** and **Right Time** Parameters govern the time between repeats. This is expressed in ms when **Sync** is off, and in musical subdivisions when it is on. The amount of signal that is fed back into the delays is controlled with the **Left Feedback** and **Right Feedback** parameters. At high levels of feedback, the signal can repeat nearly indefinitely. **Mix** is used to balance the original 'dry' signal with the effected 'wet' one. At 0%, only the dry signal is heard; at 100%, only the wet signal.

Note Splits

The **Note Splits** section controls which MIDI notes are sent to which Layer. All notes, from the bottom of the MIDI note range up to and including the **Layer 1 High** split point are sent to Layer 1. All notes, from the top of the MIDI note range down to and including the **Layer 2 Low** split point are sent to Layer 2.

Generally, one of the following setups will be used:

- To split the keyboard in half so that all notes up to a certain point go to Layer 1, and the rest to Layer 2, set **Layer 1 High** to its crossover point (eg, **B3**), and **Layer 2 Low** to the note above this (eg, **C4**).
- To produce an overlapping stack so that all notes go to both Layers, set **Layer 1 High** to its maximum of **G9**, and **Layer 2 Low** to its minimum of **C-1**.
- You may wish to divide the Layers into three so all notes up to a certain point go to Layer 1, the next set of notes go to both Layers, and the remaining notes go to Layer 2 only. To do this, set **Layer 1 High** to the Layer 1 crossover point (eg, **B3**), and **Layer 2 Low** to the Layer 2 crossover point below (eg, **C3**). In this case, all notes between C3 and B3 will go to both Layers.

Output

The Volume Parameter controls the overall loudness of String's audio output. An output signal that is too quiet can be amplified by increasing the Volume. Likewise, a signal that is too loud can be attenuated by reducing it.

The **VU Meters** can be used to provide visual cues to the current levels: the louder the signal, the higher the meter will register. Too loud of an output will produce digital clipping, a usually undesirable form of distortion. When this occurs, the meter will display a red warning indicator. Click the meter to reset the warning display.

FX Interface

When using the effects only plug-in (StringFX), only the effects section is displayed. The effect parameters function identically to the parameters in the instrument plug-in.



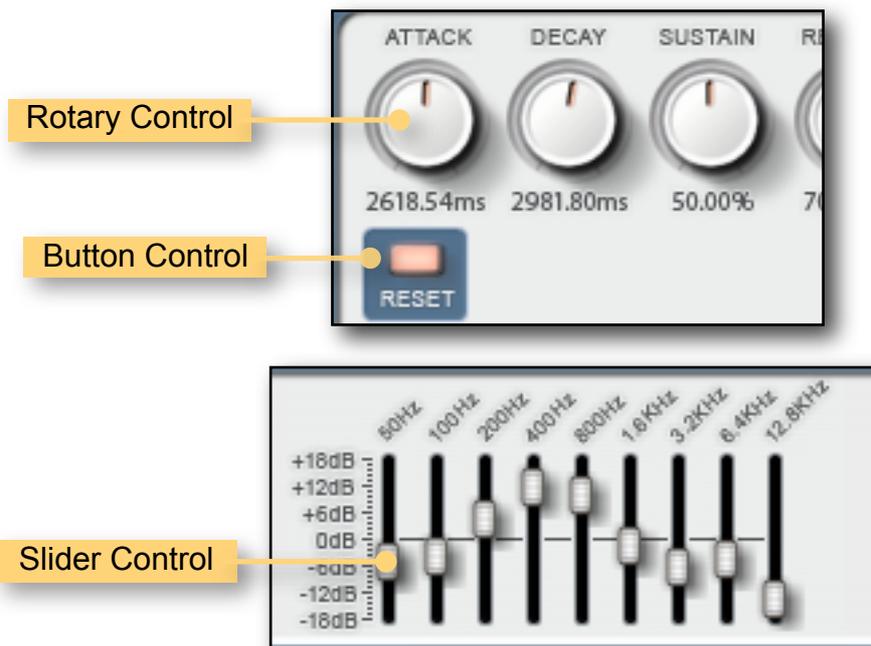
Control types

Click on a **rotary control** and drag either upwards to increase the value, or downwards to decrease it. Hold down *shift* and drag to make smaller and more precise changes. Hold down *ctrl* and click to return the control to its default value. Rotary controls can also be changed by hovering the mouse cursor over a control and scrolling the mouse-wheel up or down.

Clicking the value beneath the rotary will allow the value to be entered using your computer's keyboard, which can often be quicker than trying to fine tune a control to a specific value. When running as a plug-in, this feature requires your host to correctly redirect key presses to the plug-in: consult your host's documentation for details.

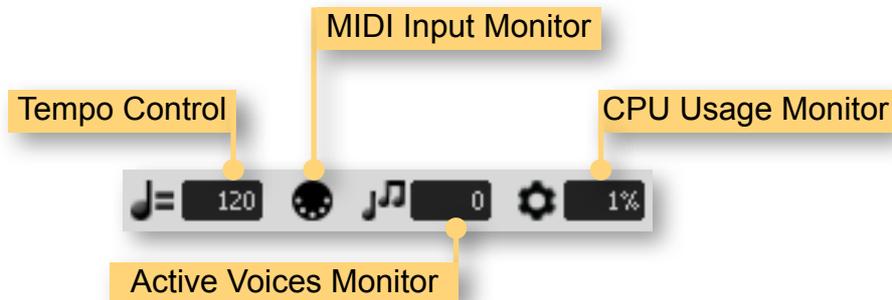
Click on a **button control** to toggle its value between On and Off. Hold down *ctrl* and click to return the control to its default value.

Click on a **slider control** and drag either upwards to increase the value, or downwards to decrease it. Hold down *shift* and drag to make smaller and more precise changes. Hold down *ctrl* and click to return the control to its default value. Slider controls can also be changed by hovering the mouse cursor over a control and scrolling the mouse-wheel up or down.



Status Display

Status Display information is displayed at the bottom right area of the String interface:



CPU Usage monitor

The CPU Usage monitor displays the percentage of processor resources dedicated to producing the audio output. The total processor resources on a computer are limited, so when String increases its CPU usage, there will be less CPU resources remaining for other plug-ins and applications. CPU Usage will differ depending on which Program is currently loaded, how that particular Program is configured, and how many voices are being played.

A value of 100% indicates that your computer is running at maximum capacity and will not have enough processing power remaining for anything else. You might notice audio output become temporarily glitchy, or that the user interface becomes sluggish. There are several methods for resolving this:

- Choose a different Program that has less CPU Usage.
- If running in Standalone mode, you can increase the latency or decrease the sample rate. See section *Setup Audio & MIDI*.
- If your host offers Freeze or Offline processing, you might be able to render the audio output in non-realtime.

MIDI input monitor

The MIDI monitor flickers to show that a MIDI Message has been received. You can use this to confirm that your MIDI controller is configured correctly for String.

Active Voices monitor

The Active Voices monitor displays the number of voices that are currently playing.

Tempo control

The Tempo control only appears when running String in Standalone mode; when running as a plug-in, the host application will provide the tempo for String. The Tempo control displays the current Tempo, in beats per minute (bpm). It can be changed by either:

- Double-clicking the tempo control, typing the required bpm and pressing return.
- Clicking on the tempo control and dragging either upwards to increase the tempo, or downwards to decrease the tempo.

Contact Details

Loomer Online

Website: <http://www.loomer.co.uk>

Support and FAQ: <http://www.loomer.co.uk/support.htm>

Email

Sales information: sales@loomer.co.uk

Technical support: support@loomer.co.uk