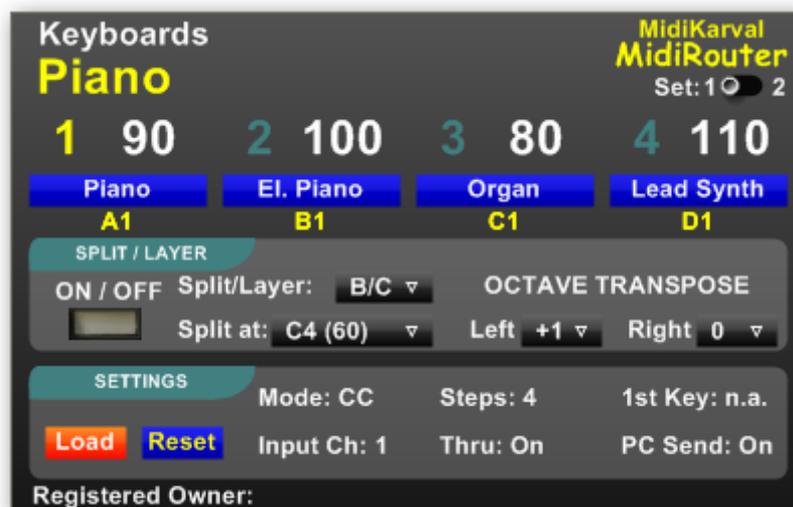# MidiRouter

MidiRouter is a VST midi processor made for all live performing keyboardists whose gear is "simply" a midi keyboard and a computer stored with VST instruments.

Let's suppose that the VST instruments you play are an acoustic piano, an electric piano, an organ and a synth, or a multitimbral VSTi, and that during your performance you need to change instrument very often or to split (or layer) the keyboard in order to play two instruments at the same time; moreover you want to change the volume of each instrument in real time, including the volume of instruments which will be played later during the same performance.

Of course, this is possible in various different ways, but it's not so easy to find the ideal solution, i.e. the simplest, the most efficient and the one that allows you to have the whole view of your instruments' setting (including which instrument is associated to which midi channel and which volume level it has) in any moment. **MidiRouter** makes it possible.


MidiRouter stands in your VST host between its midi in and your VST instruments (up to eight instruments, each one assigned to a different midi channel) and allows you <u>to change the transmitting midi channel and, as a consequence, the playing instrument and the relevant volume level</u> in different ways:

a) from the <u>MidiRouter's interface</u> using the mouse (directly pushing on the relevant blue labels to choose the instrument to be played and clicking and vertically dragging on the relevant white number to increase or decrease the volume level);

b) sending <u>Control Change messages</u> from your midi keyboard, using its buttons, knobs or sliders ("<u>CC mode</u>");

c) using some (4 or 8) <u>keys of your midi keyboard</u> to choose the instrument to be played ("<u>Keys mode</u>"), whilst you have to use the a) or b) method above in order to change the volume level.

By doing this you will see the midi channel assigned to the chosen instrument (the number on the left above the relevant blue label) turning from grey to yellow (midi channel no. 1, in the picture) and the name of such instrument shown in the upper left side of MidiRouter ("Piano" in the picture).

MidiRouter has two "Sets", each one showing its own name (user-definable) above the instrument's name in MidiRouter's interface ("Keyboards" in the previous picture, "Orchestra" in the following one.
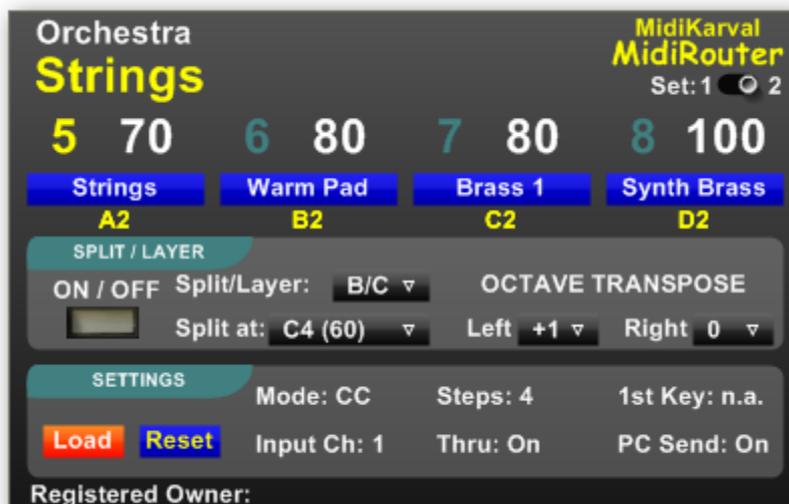
You can change the current Set using the switch in the MidiRouter's interface or sending a CC message from your midi keyboard. If you use the "Keys mode" with 8 keys designated to change the midi channel, the switching from a set to the other one is automatic and depending on the key you push.

MidiRouter is **fully customizable**, as its heart is a file (whose extension is ".mrf"). You can create, as we'll see later, the "*.mrf" files with any text editor in order to set: *each instrument's (i) name, (ii) midi channel, (iii) volume; (iv) the CC message numbers to be used to change the playing instrument and each instrument's volume, (v) the keys to be used to change the playing instrument (in "Keys mode") and all the other settings we'll explain later*.

By default, when it is loaded in a Vst host, MidiRouter take its initial settings from the "midirouter.mrf" file, that has to be located in the same folder where "MidiRouter.dll".

INSTALLATION

Extract the content of MidiRouter.zip (including the "midirouter.mrf" file) to your VSTplugins folder.

As MidiRouter is a VST plug-in, you have to open it in a VST host application (we recommend EnergyXT or Hermann Seib's VSTHost), connecting the host midi input to the MidiRouter input and its output to the VST instruments input.
The following diagram shows the connection example using *EnergyXT*:

Please, be sure that the *only midi flow* reaching the VST instruments you want to control from MidiRouter comes from MidiRouter itself, otherwise the results could be unpredictable.
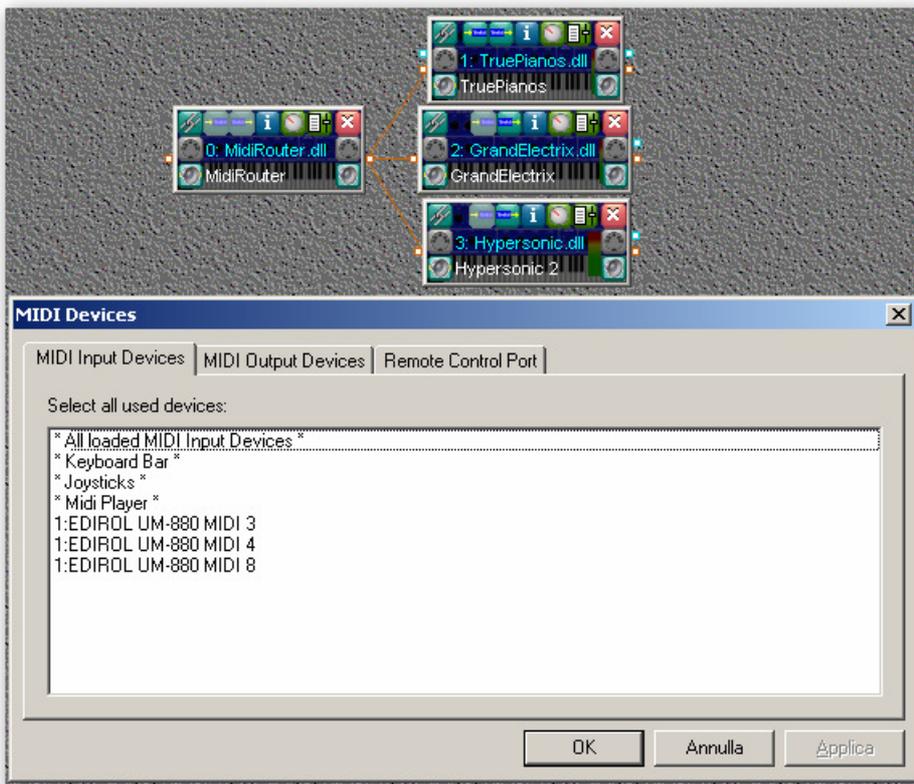
For example, if you use *VSTHost*, only the MidiRouter should be connected to the midi input devices, whilst all the other VST instrument's midi input devices should be disabled (left-click on the enabled ones while pressing the <ctrl> key of you PC keyboard), as shown in the picture on the left. (i.e. No blue bar showing across any midi device).

The midi symbol on the right in the small instruments' GUI is the button used to switch on and off incoming midi messages.

The top left-hand button in the instruments' GUI is the button used to access the link to MidiRouter.

Please note that the "Split/Layer", "Split at:" and "Left and Right Transpose" labels (drop lists) present on MidiRouter can be also assigned to midi messages using your host "parameters assignment" function.

3

HOW MIDIROUTER WORKS

First of all, you may wish to **customize MidiRouter** on the basis of your needs. This can be made modifying the "midirouter.mrf" file, i.e. the one loaded by default, and/or creating other ".mrf" files, with different names, that you can load into MidiRouter, using the `Load` button, each one reflecting the setting you prefer for any occasion (even for each song you are going to play), provided that you respect the same structure of the "midirouter.mrf".

You may do so with any text editor (e.g., Windows Notepad) paying attention to change only the part of each line following the symbol "=", without leaving any empty line between the parameters' lines.

If you have loaded another "*.mrf" file or simply modified the settings on the MidiRouter's interface and want to come back to the original "midirouter.mrf" settings, you can simply push the `Reset` button.

Now let's see the structure of a ".mrf" file, for example the "midirouter.mrf" file you have received with MidiRouter. The title of each line is self-explanatory.

```
Name1=Keyboards
Instr_A1=Piano
Channel_A1=1
ProgCh_A1=0
Vol_A1=90
Instr_B1=El. Piano
Channel_B1=2
ProgCh_B1=4
Vol_B1=100
Instr_C1=Organ
Channel_C1=3
ProgCh_C1=16
Vol_C1=80
Instr_D1=Lead Synth
Channel_D1=4
ProgCh_D1=80
Vol_D1=110
-------------------------------------
```

"**Name1=**" and "**Name2=**" host the name of each set.

Then there are eight groups (four for each set), each one consisting of four lines, where you should write respectively:
- the **instrument name** (up to 12 characters),
- its **midi channel number** (1 to 16),
- the **program change message** to be transmitted to it (-1=no program change, 0-127=program changes nn. 1-128) and
- its **volume level** (0 to 127).

In the line description (the part before "=") the letter stands for the instrument position in each set and the number stands for the relevant set: so "Vol_C1=" stands for volume of instrument C in Set 1; "Channel_B2" stands for the midi channel of instrument B in Set 2.

The **Channel Numbers** within each set (*Channel_A1* to *Channel_D1* and *Channel_A2* to *Channel_D2*) must be unique, i.e. without any duplication, whilst you can use the same channel number in both the sets if you want a certain instrument to be present in both of them.

```
-------------------------------------
Volume_A=101
Volume_B=102
Volume_C=103
Volume_D=104
-------------------------------------
```

the **Volume CC Messages**: the numbers after "*Volume_A*" to "*Volume_D*" lines show the CC message numbers to be sent with your keyboard's knobs/sliders in order to increase/decrease the volumes of each of the four instruments in each set. You can change them as you prefer provided that they are in a sequential order (e.g.: "90 - 91 - 93 - 94" is wrong).

```
-------------------------------------
Switch_Mode=CC
SwitchCC#=83
SwitchKey=C2
SwitchSteps#=4
SetCC#=120
Split_On_CC#=85
ProgCh_Send=Yes
Input Ch.=1
Thru=On
```

In the "**Switch_Mode**" line you can choose if you want to use, to change the transmitting midi channel, a CC message (Switch_Mode=CC) or your keyboards' keys (Switch_Mode=Keys)

4

The "**SwitchCC#**" number shows the CC message number used to switch from an instrument to another one within each set (see below), if "Switch_Mode" is set on "CC". The CC message designated in the **SwitchCC#** line triggers the change of the instruments within each set on the basis of the following values:

Instr.	Value
A	0
B	42
C	84
D	126

So, if you use a <u>knob/slider</u> sending, for example, the CC message # 83, turning/moving it will cause the change of the instrument (i.e., of the transmitting midi channel) when it will send one of the above-mentioned values, i.e. at the beginning (A), at 1/3 (B), at 2/3 (C) and at the end (D) of its range. If you use <u>buttons</u>, you have to set four buttons on CC message # 83 (in this example), the first sending a value of "0", the second a value of "42" and so on.

In the "**SwitchKey**" line you can choose the first of the 4 or 8 keys of your midi keyboard (the other 3 or 7 are those immediately following) pressing which you can change the transmitting midi channel if "Switch_Mode" is set on "Keys": the format is the note's name followed by the relevant octave's number (e.g., C2, F#3, Bb2).

In the "**SwitchSteps#**" you can indicate how many keys (4 or 8) you want to use in "Switch_Mode=Keys": if you use 8 keys they are enough to select the midi channels of both Set 1 and Set 2; if you use 4 keys, in order to change Set you need to use the switch on MidiRouter's interface or, better, a button of your keyboard transmitting the CC message indicated in the "SetCC#" line (see below).

The "**SetCC#**" number shows the CC message number that can be used for switching between Set 1 and Set 2, in the "*CC*" Mode (or in "Keys" Mode if you have chosen to use only 4 keys), without having to use the switch in the MidiRouter's interface: a value of "0" switches to Set 1, an higher value switches to Set 2.

The "**Split_On_CC#**" number shows the CC message number used to switch on and off the "*Split/Layer*" function (see below): a value of "0" switch it off, a value of "127" switch it on.

In the "**ProgCh_Send**" line you can choose if you want that MidiRouter transmits the program changes messages chosen for each instrument (value="Yes") or not (value="No"); combining "Yes" with the "-1" value in the "ProgCh" line of some instruments you can choose to transmit the program changes only on certain midi channels.

The "**Input Ch.**" number allows to select the midi channel number, whose midi events are elaborated by MidiRouter: a value of "0" means that <u>all midi channels are received and elaborated by MidiRouter</u>, a different value means that <u>only the events transmitted on such midi channel are elaborated by MidiRouter</u>, whilst the destination of all the other events depend on the "Thru" line value.

If the "**Thru**" line value is "On" the "excluded" midi channels' events pass through MidiRouter to be used for other Vst instruments or with another instance of MidiRouter; if such value is "Off" the "excluded" midi channels' events are locked by MidiRouter.

All the options chosen in the loaded file are shown on the MidiRouter's interface:
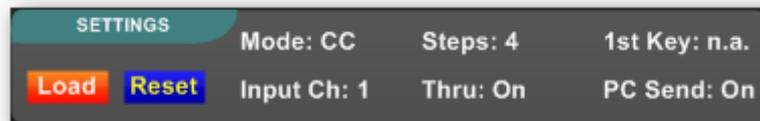
In the middle of the MidiRouter's interface you can find the "**Split/Layer**" section.

Through the "*Split/Layer*" drop-list you can choose <u>which instruments of the current set are assigned respectively to the left and to the right side of your keyboard</u>: all possible combinations are provided. The split point is definable through the "*Split at*" drop-list.

If you set the "*Split at*" drop-list on the "C-1 (0)" position <u>the chosen instruments are layered in the whole range of your keyboard</u>.

Moreover, you can choose to transpose the left or right (or both) splitted parts up to 3 octaves above or below.

Also the "Split/Layer" section can be pre-programmed using the ".mrf" file as in the following example:

You can split or layer your keyboard using the "ON/OFF" button on the MidiRouter's interface or sending the CC message indicated in the "Split_On_CC#" of the loaded ".mrf" file.

The demo version of MidiRouter is fully functional for a 15 minutes period of time. It then expires and you have to remove it from your host and re-load it to use it again.

---

www.midikarval.net