Tobias Escher

Orchestral Templates with VE Pro 5

A Tutorial for Media Composers

novatlansound
music for media

novatlansound
music for media

Tobias Escher

Orchestral Templates

with Vienna Ensemble Pro 5

based on Apple Logic Pro X

A Tutorial for Media Composers

V. 2.4 (11.11.15)

http://www.orchestraltemplates.com
http://www.novatlan-sound.de

Introduction

The working template is one of a composer‘s greatest treasures.
It defines our sound, it saves time and it is incredibly impressive when people come to our studio. Few things make a more lasting impression than hundreds of tracks distributed over several physical machines and then seamlessly switching between a soft flute and a full marcato string section at full volume.
Oh, and by the way: You can actually use it for work!
So you will be well advised to create a template for yourself, too.
If you have arrived at this stage, you should be pretty familiar with the supposed content of such a template. If not, you should first get your homework done and plan ahead a bit. If you are stuck, I'll be happy to assist you in planning your template-to-be - just send me a message. For some general information about templates, look at the section about planning the template.

This tutorial will show you step by step how to setup the necessary technology for your template and how to set it up in Logic Pro 9. I chose this particular sequencer because I use it myself, know how it works and because of its place in the film music community. I firmly believe that you will benefit more from specific instructions on such a complex topic as setting up a template than from unspecific and vague "steps to reproduce". Furthermore, Logic has a few oddities, which make setting up such a template a bit harder than it is on other platforms. I suppose no one on earth really knows why Logic does what it does sometimes, but there are workarounds and it _is_ possible to have a big template in Logic, even though some say otherwise.

If you are well versed in all technology stuff, please nevertheless read this text! There may be things you do not yet know and learning new things is always great. Please also read the tutorial if you are not using Logic. Most things apply to other sequencers as well.
Furthermore: If you would be interested in helping me "port" this tutorial to other sequencers, drop me a line!

2.0 Update Notes

Version 2 of this tutorial updates the whole text to reflect Vienna Ensemble Pro 5 with its new features and slightly redesigned GUI. There now also is a section about controlling instruments within VE Pro from Logic via MIDI CCs; as well as some information on working around the MIDI Port limit for AU plugins. Once again I would like to thank the composers with whom I had the joy to work on their templates; the insights gained there are part of this tutorial.

2.1 Update Notes

This new version as usual corrects a few quips, but also adds a whole new section about using VSL MIR Pro / MIR Pro 24 for reverberation and positioning of instruments. There's also a section about load optimization and of course quite a lot of the text has been updated to reflect new/additional software. And just as before I can only ask you to give me feedback so I can continue to improve this tutorial!

2.2 Update Notes

The release of Logic Pro X in regards to template building has mainly brought UI changes to the software, but these are pretty substantial. Some other minor changes also influence VE Pro indirectly. This new version of the tutorial has been updated for Logic Pro X, with completely redone screenshots and some content changes here and there.

2.3 Update Notes

This update mainly brings the tutorial in line with current technology (mainly SSDs and newer CPU options, etc...) and fleshes out some paragraphs a bit more.

2.4 Update Notes

2.4 adds and clarified a few things here and there and again brings a few tech updates.

Planning the template

Before dealing with setting up the software, I want to briefly touch up on what a template should do, what it shouldn't do and why you cannot create a template in one hour, but instead need to spend some time with planning.

It is likely that you already have some kind of plan as concerning what is going to be in your template. Creating a template always means working on the fine line between too much and too less. It is of no use to have a huge template that will never, ever run on your system without crackles, pops or even crashes. Conversely, a tiny template that doesn't even cover the bread and butter stuff is equally useless. For determining what you may want to have in there, the following checklist may help:

1. *Which instruments do I use in almost all of my projects?*
   It is useful to have all the instruments in your template that you come back to constantly. That means that they will always be ready while at the same time not cluttering your screen with tons of tracks you rarely use. Broadly speaking, this category will include a basic orchestral palette, possibly mixed from various libraries in custom-built patches/multis. If you have difficulties finding those "sweet combos", feel free to drop me a line; with my experience of most of the common libraries as well as from setup sessions with composers I will gladly assist you.
2. *Are there special single patches from libraries that I use very often, but that are hard to find?*
   Often, there are single patches in a library that are very useful. For me this is for example the Steinway D from the NI Kontakt 2 Library. I rarely use that library anymore save for this special patch. You should look for those gems and have them in the template, otherwise you will forget those patches!
3. *Which AUs/VSTs do I regularly use that are resource hungry?*
   Besides the obvious advantage of having important instruments pre-loaded, you can also use a template (especially with VE Pro) for moving resource-hungry instruments out of your sequencer's memory space (especially on a 32bit sequencer). So for very memory intensive instruments like Spectrasonics Trillian, you might want to put that instrument outside your DAW.
4. *Do some AUs that I want to have in the template involve constant heavy parameter tweaking that cannot be done via MIDI?*
   Instruments within VE Pro do not appear as plugins in Logic, so to access their GUI, you need to open VE Pro. If you can do the tweaking with CCs via MIDI, this is not an issue, but it is if you need to constantly access the GUI. Furthermore, the more parameters chance from project to project, the more your external template files change, which makes projects not directly compatible with each other! The same instrument may have totally different settings in various projects. I would therefore try to put most of the "heavy editing" plugins inside the DAW if possible and use the template for the non-changing stuff. VE Pro 5 eased this a fair bit by providing full support for all automation parameters, but still if you need to access the GUI a lot, leave this instrument in Logic.
5. *Do I want to keep using some 32bit plugins while exclusively working 64bit in my sequencer?*
   With more and more instruments going 64bit chances are high that you will only have very few 32bit plugins left in daily use. Even though you can continue using them in Logic by using the 32bit bridge, it is very convenient to create a 32bit instance of VE Pro where all of these plugins reside. Apart from not having to deal with the bridge

anymore, you will also see some performance improvement. If at all possible, try to find alternatives to old 32bit plugins. Chances are high that any software that has not yet been updated to 64bit will be rendered nonfunctional by OS changes, forcing you to switch.

If you write in different musical genres, you should contemplate having more than one template. Build a template for each genre, tailored to the specific needs of the style you are writing in. For big projects it can even be useful to have project-specific templates. This is a lot of work, but for long-running TV shows or things like that it may be the most sensible thing to do. Also once the basic foundation of the template is laid adjusted to your workflow, creating „forks“ is relatively easy.
Another road to take are single instances for different instruments. This works especially well with combinations of sample libraries focused on a special task. You could for example create an instance just for ethnic instruments. In there put all your favourite ethnic patches. Only actually load the instance when you need those instruments. You can even add or remove instances right in the middle of a project. Combine this with pre-built Logic sessions containing all tracks pre-mapped and you can load in additional instruments in seconds. This allows you to use the convenience of templates (no setup required after the initial setup), but not be limited to the selection of instruments in your main template.

One last way to do stuff is having instances full of custom-programmed multis which you load when you need them. Combine patches playable from one single MIDI channel from different libraries and tweak them to your hearts content. Whenever you need those custom multis, just load in the instance!

Carefully plan the number of instances to use so you make the most out of your resources. Try to balance the workload so that on a typical project, each machine has about the same amount of work to do. Don't forget to reserve a bit of power for the DAW, effect plugins, video and all the other things running on your main computer.

Preparing the technical side

Computer hardware

So now comes the technical aspect of creating this template. After you have created a list of everything you want to have in it, you need to assess what you have in terms of processing power and what you might need.

We will deal with all necessary software later, but first I want to touch on the subject of hardware. In general, you can either host your whole template on one machine, or you can use multiple machines. Some people also prefer to dedicate the computer their sequencer (DAW) runs on solely to running the application and maybe video if scoring for film. In this case you need at least one extra computer for the template, possibly more than one depending on the size.

If you think about going the two or more computers route, before you buy anything or even start programming the template, a careful evaluation of how you want to spread the sounds is necessary. You need to plan which samples will be loaded on which machine so that for most of your daily work, the load will be balanced in a way that not one machine works its' butt off while the other is sound asleep. This requires you to know what instruments and articulations you generally use. I recommend working on a few projects and taking notes. Then you will have a pretty good idea what needs to be in there.
There are indeed a few big libraries out there that (if you can afford it) require a machine of their own. The most important ones that come to mind are East West's Hollywood Strings and, to a lesser extent, Hollywood Brass. Also with Orchestral Tools‘ Berlin series (full disclaimer: the author of this tutorial works for OrchestralTools) and Spitfire‘s British Modular Series in the process of completion, you have at least two major modular libraries that would make sense to be hosted on their own machine. If you want to rely on those libraries to form a major part of your sonic palette for that particular instrument group, you will need to have a considerable part of them loaded all the time. To avoid performance bottlenecks, in this case give them their own machine. Of course you can also host these VIs together with other plugins, but this may give you performance issues. Hollywood Brass works together pretty well with other instruments, but Hollywood Strings really shines when it has its dedicated machine to run on, especially when using multiple microphone positions.

Here is a real-world, mid-sized sample setup for Logic 9 or X that should be within budget range of most people serious about doing this stuff. If you are using non-OSX-exclusive sequencers, a PC is a better alternative, so the specs below are just for Logic.

- DAW

  For using Logic, any 2008-2013 MacPro will do. If you do not host your libraries on the Mac, an iMac will work fine, but if a lot of work needs to be done on the DAW, a MacPro is better. Keep in mind that the 2013 MacPro will require external Thunderbolt hardware.

- Main Slave for doing the hard work:

  Intel i7 3930K or 4930K, 6*3.2 Ghz
  64 GB Ram
  Small SSD system drive

2 or 3 SSD drives
1 to 2 7.200 RMP drives

- Possible smaller slave, capable of 64GB Ram
  Intel i7 4820, 4*3.7 Ghz
  32 or more GB Ram
  10.000 RPM system drive
  1 to 4 7.200 RMP or SSD drives, depending on the number of libraries

Let me give you a short explanation on why I chose this configuration:

- You need a Mac for Logic Pro. The MacPro (with perfect reason) is called the "world‘s largest dongle". So for Logic on a Mac, a MacPro it is. Amen. A powerful machine like this with a nice amount of RAM (ignore the 32bit thing for the moment if you know what I am talking about) allows you to host at least a portion of your template directly on your DAW (saving money for more slaves), while still allowing you to load in instruments at your leisure as well as having video running and possible even other software, for example Harrison Mixbus for summing.
- Up to version 9.1.2, Logic did not fully utilise all cores on the most recent Mac Pros; for the technically savvy among you: It did not deal particularly well with virtual cores through Hyperthreading. This has changed in 9.1.2, so if you have the money it would be wise to invest in the most recent model. The "old" 2008 Mac Pros, however, hold their ground very well! I use one myself and see no reason to part with it.
- There is absolutely no need to use a Mac as a slave. If you absolutely want to use OSX as an operating system, get another MacPro if you can‘t help it. For all the others, a decent and well-configured running Windows 7/8.1/10 slave will host a huge template easily for a lot less money. You set it up once and then forget. For the case of emergency, you can make a mirror image of your hard drive and just write it back from time to time to get a fresh system without all the hassle of installation. In fact, I recommend doing that, because it saves you a lot of maintenance! You will have no problem using a PC even if you have had no previous experiences with Windows whatsoever. The basics are easy and you do not need to delve deep into the system because most manual work will still be done on the Mac. This is the reason why I advise you to use a fairly powerful DAW, so once correctly configured, there is no need to touch the Slave again, because everything else you might need, your main DAW will do.
- In fact, in my experience, having a system "from the other side" is pretty useful. You can run plugins that do not work under OSX or work better on Windows. Native Instruments Kontakt is one of these in my experience. I have found it far more powerful and good on resources when running it on Windows. If you have Kontakt 3.5 and want to save the money to upgrade to K4 or 5, or have some libraries that have bugs in the newer versions, on Windows you have the added bonus that K3.5 is already 64bit. In my experience, Kontakt 4.1 dramatically improves performance, so at least for new templates and when you are sure that there are no bugs, you should switch to K4. Kontakt 4.2 and 5 as well have made things even better with greatly reduced loading times.
- Why so many hard drives? First, I am a big advocate of SSDs or at least 10.000 RPM drives for the operating system and DAW. One single reason: Speed. If your system needs to load anything from the system drive while doing something (i.e. loading QuickTime for displaying a video or loading some graphics because your DAW software just decided to go all flashy images and animations on you), every millisecond counts. Even if you only get a small speed boost, it means that there is one less bottleneck. As an added bonus it saves startup and shutdown time. It makes sense to have some demanding sample libraries on an SSD, but having one as a system drive is not a must

unless you want ultra fast boot times. Some libraries, like Hollywood Strings greatly benefit from an SSD - for this library in particular, having one is a must.

- You should try to spread samples that are used at the same time on different physical drives. Physical drives, _not_ just different partitions on the same drive! This makes loading times a lot faster and is the reason why the configuration I recommended above has such a high number of hard disks. By the way: Partitioning a drive in OSX does not hurt, but there is no reason to do it. It doesn't make things faster, it only clutters the finder. For sample drives I would not do it in Windows, either. In fact, partitioning should most be a thing of the past because apart from maybe a slightly nicer look, there are no advantages to it anymore.

Help! How many computers do I need?

You wouldn't believe how many times I get emails saying something like "Help! Someone told me that no rig under X computers and Y dollars will enable me to make serious music. Is this correct?".

No, it isn't. Sort of. A sheet of paper (doesn't even need to have lines on it) and a pencil (preferably also an eraser) will enable you to compose serious stuff. Of course you need some technology to actually record it. But for the basic things like doing a middle-sized orchestral piece, a fairly recent Mac (any Mac, even a MacBook Pro) or PC will do the trick.
For a larger template like the one we are talking about here, you should probably have something like a MacPro, just because those huge orchestral libraries need lots of power. But a MacPro _will_ be able to run huge sessions without problem. So don't believe the folks telling you that you need 10 computers. Hans Zimmer does. But Hans Zimmer happens to be Hans Zimmer, while you and I do not (Hans, if you are reading this, I would be profoundly honoured if you'd drop me a line). Also Hans uses a different system with his template and has far more demanding instruments. As far as I know he is currently using 20 slave PCs and running libraries with tons of microphone positions. No commercially available libraries come even close to that.

Audio interface and getting audio across

Just a short section about getting sound from your Slave to the Main DAW. You may want to use the Slave to have another sequencer application running. I have a ProTools M-Powered system on it for film and post-production work, which is pretty nice :)
Strictly speaking, if you only want to use your slave for hosting samples, you do not need any audio interface at all, because there are better ways of getting audio to your Main DAW than through a sound card. I even recommend deactivating the onboard sound chip if you don't need it. You can do this in the BIOS, see your PCs operation manual for details. The same by the way goes for fancy 3d graphics cards - even an onboard chip will be sufficient. Having a graphics card with passive cooling (i.e. without a fan) also helps to reduce noise. Also look out for 3d party drivers installed by the manufacturers for 3d gaming and things like that. You don't need them, so save the memory their control panels take up for more sensible things. Make sure to not allow Windows Update to install these driver applications when new versions come around. They most of the time load a background application residing in your notification area and consuming precious resources.

If the "second DAW-approach" is to your liking, you may want to invest in a second audio interface to harness the power of a second DAW. As of Pro Tools 9, you can use any audio interface you like, you do not need to have an M-Audio or Digidesign interface anymore!

There is a pretty easy way of getting audio from the Slave to your Main DAW: Make sure both audio interfaces on the DAW as well as the Slave have SPDIF. Then just connect the SPDIF Out on the Slave with the In on the Main DAW and all audio will go to your main audio interface with just one cable! Why would you want to do that, you ask? Because if everything goes this way, you can use your big studio monitors without having to invest in a monitor controller! And by using SPDIF, there is no quality loss. Great, isn‘t it? Of course, with a dedicated audio interface you have even more reasons for deactivating any onboard sound chip; especially on Windows those onboard chips tend to misbehave. When shopping, make sure you get an audio interface that can work without its host switched on. Some interfaces are operational without the machine they are connected to needing to stay on. This allows you to use the big speakers with Pro Tools even if the DAW is switched off.

Controlling Slaves

You will of course need some means to control your slave machine(s). You could hook up a screen, mouse and keyboard as usual, but chances are high that you will not need to access your slave computers too often. If you set the power button of the machine to switch the computer off when pressed and set your template to startup automatically as soon as you power on your machine, you can operate it without using a mouse and keyboard at all!
In case you _do_ need access, you can setup a Remote Desktop Connection. If your slave runs Windows and your Main DAW OSX, then you can for example use the excellent application "Cord" (http://cord.sourceforge.net/) to do this. This Remote Desktop Connection software will forward your input peripherals to the slave machine and show the screen on your main machine.
Try it out, it works really well!
Be advised however, that i.e. Sibelius has some graphic anomalies when used through Remote Desktop, but this is nothing really serious you need to worry about. Just accept it.

Tip:
If you use Pro Tools 9 on your slave and have an Avid Artist control surface, you can use it to control Logic as well as PT. The Artist series used to only work on OSX (but there was a workaround to make it work with Windows), but with the recent versions, it has native Windows support.

Vienna Ensemble Pro, i.e. "The Software"

Now comes the part you've probably been looking forward to: How on earth do you actually get the slave to know which instruments to compute. And how do you get the audio back to the machine?!?
In the past the only way was to use lots of pretty complicated solutions like AudioOverLAN, FXteleport and many more. Some involved huge arrays of dedicated audio interfaces for routing back audio channels via MADI. Lots of people still do it this way and this is fine. It actually works very well once you have set it up and managed the considerable expenses and there are still enough situations where such a system is desirable.
For users of the Vienna Symphonic Library, there has been a very user-friendly solution out there for some time: Vienna Ensemble. This allows you to transmit MIDI data to the slave and audio back via a simple network cable. No audio interfaces, no complicated wiring, just a single ethernet connection. Not so long ago, with version 4, Vienna Ensemble Pro was finally able to do this not only with VSL's own instruments, but with almost every single third-party plugin.
Even better: It can do this with AudioUnits as well as VSTs!
Better yet: It can do it for 32bit and 64bit plugins even with 32bit hosts.
And best of all: It can do all this cross-platform!
This means, that with VE Pro you can host your plugins on other computers or outside the RAM space of your sequencer and all you have to do is buy a license for VE Pro and make an ethernet connection between your computers.
If that's not like totally freakin' awesome, what is?

Tip of the day:
I usually always am a big fan of "Don‘t fix it if it ain‘t broke". But in the case of VE Pro, I have found all their versions to really be far better than the previous one, even the betas. So if you have a niggling issue, I can only encourage you to try their newest version, even if it is a beta. It certainly worked for me. If you are experiencing weird behaviour, first make sure you have the latest version. There may be host-specific bugs that are fixed in new versions. Please also note that VSL most of the time updates all their software packages at once. So if there is a new version of VE Pro, chances are high that there will also be one for VI Pro and MIR Pro.

Why VE Pro on a single computer?

Now comes the 32bit vs. 64bit part! As you probably know, any application that is 32bit can only access about 4 GB of memory. As this number refers to virtual memory as well, the actual amount of physical RAM you can address is lower, much lower. Usually it moves around 1.6 GB. But: Even if you don't come too near to this limit, the applications get pretty slow and unstable.
With 64bit applications, you can in practice use as much RAM as you like. But: The plugins (in the case of Logic that would be Audio Units) you want to use have to be 64bit, too, to be able to use that much RAM. By now (Spring 2014) all major plugin vendors have introduced 64bit versions of their plugins, so you will rarely encounter 32bit plugins anymore.

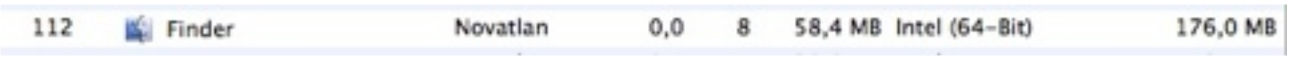
But what if a plugin you rely on is still 32bit, with no chance of getting updated? Or maybe the manufacturer charges a lot of an update with no added benefits for you than 64bit compatibility? VE Pro to the rescue! The 32bit server will allow to use 32bit plugins, but still use a 64bit host. No need for any kind of „bridge“, as it was present in Logic Pro 9. Logic Pro X has removed the bridge anyway, so VE Pro is your chance to still be able to use your treasured 32bit plugins.

Applications are most stable when not too much is loaded into them. The same should probably apply to you: You are much more stable with zero alcoholic beverages in your veins than when you successfully managed to replace the larger amount of your blood with alcohol. That means that from this point of view it is prudent to try to have as much of your big huge sample libraries outside of the memory space of your sequencer, even if it is 64bit and even if your plugins are, too. It cuts loading times a fair bit and if you chance an instrument in the template, it gets changed in all projects, too (useful for library updates).
As an additional bonus, when changing projects, this way you can just keep everything loaded because your sequencer does not need to load the samples when loading any project - it merely connects to the samples hosted elsewhere.
It should be obvious that it would not be wise to have a huge number of instruments loaded outside you rarely need.

Therefore, as a rule of thumb we can say: Try to load as many of the instruments you need all the time outside of your sequencer, not just for memory reasons, but also for ease of switching projects. If you need special instruments that very likely will only be used in this particular project, load them inside your sequencer.

64bit system boot vs. 64bit applications

There sometimes is much confusion about how to actually enable all this fabled 64bit stuff. This supposedly comes from the Windows world, where you need a special 64bit version of Windows if you want to get the advantages 64bit computing offers. As of 2014, even that situation has somewhat cleared itself, because pretty much all PCs now use a 64bit Windows by default.
This, however, does not apply to OSX! You only have one single version which is capable of running 32bit and 64bit applications side by side. OSX starts a 32bit kernel by default on older machines, but it can run 64bit applications simultaneously! You can check this with the Finder.app. Just open Activity Monitor (in the Applications:Utilities folder). There you will see "64-bit" after the application name provided you have a mac that supports it (which would be anything with a Core2Duo processor or higher; 1st generation MacBooks for example have to stay 32bit).



```
112   Finder            Novatlan     0,0    8    58,4 MB  Intel (64-Bit)          176,0 MB
```

So the answer to the question what you need to do to be able to run 64bit applications is: Nothing. You do not need to do anything. As long as you have at least a Core2Duo processor inside your Mac, you are good to go. Just start 64bit applications like VE Pro and OSX will do the rest.

With the advent of Logic Pro X, you need not worry - LPX only runs in 64bit anyway!

Why VE Pro on multiple computers?

Well, that should be obvious. In addition to all the great advantages detailed above, you can give your Main DAW a break - it does not have to compute all the stuff, as large parts of your template are computed on a different machine.
For most normal composers, one or two slave machines are absolutely sufficient. You can extend the number of slaves at any time by simply activating (or buying) a new license of VE Pro. There is no limit of the number of slaves you can have. In fact, by using different network cards you can establish different subnets which means your PCs do not need to share a single gigabit connection in terms of bandwidth. But so far even with four slaves I have not run into bandwidth issues.

Contrary to several other methods of "audio-slaving", VE Pro does not require more setup work if you use it on several computers as opposed to one. In fact, there is no difference at all, other than that you need an additional eLicenser (that nice dongle thingy) per computer and that you have to master the murderously complicated task of plugging in an ethernet cable. Now comes a tip for all those of you who have just ordered a bunch of Vienna Keys to use with their slaves: You can also use the Steinberg eLicenser. This dongle is usually used for Cubase and is technically identical to the Vienna Key, but is a bit more affordable! It doesn't look as nice, though. If you buy VE Pro, you get three licenses for the server to run. That means that you can use your Main DAW for hosting samples and two slaves with the only additional cost being 40 bucks for the eLicensers and 5 bucks for two cheap ethernet cables. One question I get a lot is whether the ethernet cables used make any difference. I would very cautiously say: No, not as far as I can tell. All the cables I tried worked the same. Where you will notice a difference is with very long cables, so if your Slaves reside in a machine room, you should consider getting very good cables. Likewise try to not have too many other cables interfere with your ethernet connection. Rule of thumb: CAT6 cables, double shielded should do fine.

So, to make a long story even longer: If your Main DAW is struggling a bit with all the stuff loaded on it, and you have a bit of change left, get a decent PC with Windows and let it do the slave work. You can even do this in the middle of a project by transferring VE Pro instances from one machine to another. I use Intel CPUs - but you may also use AMD, it is just that so far I have not seen a single AMD system in use for audio. If you want detailed info, email me!

Clarification:
I got some mails asking whether or not you need a PC from Dell or any established manufacturer (because I mentioned a Dell machine in an earlier version of this tutorial). I even got a mail accusing me of getting money from Dell.

Short answer: No, you don't need a Dell and no, I don't get any money from them.

Long answer: Every decent PC should do the trick. You can also build (or have someone build) a PC from scratch. That is what I did for my new DAW as well as my new slave. Oh, yes, the long answer to the second question still is NO, I don't get any money from Dell. But Dell, if you want to give me some, I'd be the last person to say no.

The only thing I would recommend is to spend some time looking at the ethernet card. How well VE Pro runs in terms of speed largely depends on how good your ethernet card is, so make sure your PC of desire has good gigabit ethernet (I like Intel chips)!

Setting up the instruments

So, here we go. Less text, more nice pictures.
I will describe step by step how to setup VE Pro with Logic in the way that works best for me. You will (especially when it comes to the Logic part) see that I do some things completely differently than in the Logic manual. This because I think this is the better way. Your mileage may vary and I encourage you to try everything out and see if it works for you.

Setting up your VE Pro instances

Before even thinking about actually programming your template, you should have a solid plan in mind of what it is going to contain in terms of articulations (see introduction). You should also be familiar with the plugins you are going to use. On both counts, if you have difficulties, contact me. I'll be happy to help.
You need to determine how many instances you intend to have and how you want to fill them. There is one problem which comes with the Audio Unit standard Logic is using: You can only have 16 unique MIDI Channels per instance of any Audio Unit. The VST3 versions of plugins does not have this limitation, but if you use Logic then you have to live with it (the same applies to Digital Performer, by the way). This limitation used to mean that any instance of VE Pro could only control 16 different instruments at the same time. With VE Pro 5, this has changed: A new plugin that created additional virtual MIDI ports allows you to have more than just 16 channels per instance, so even though you need to use an additional plugin compared to the folks who are able to use the VST3 version, you basically get the same capabilities now, which is absolutely great.

One slight drawback: I have found some AUs acting weird when using the first channel. They sometimes react on all channels or have random hanging notes. This may actually be a Logic issue, but I never really found out what is going on there. That‘s why I recommend only using channels 2 to 16. I also recommend leaving at least one channel per instance empty so you have room to add things.
You also need to think about how many channels of audio you want to route back to Logic. It makes sense to have instruments that can easily be routed back together in the same VE Pro instance, because the number of audio channels you can route back is limited by your ethernet connection. This is less an issue when you are only using a single machine, this not needing to get audio back through ethernet.
I have found that with East West Play, having their instruments inside VE Pro makes Logic far more stable. The Play engine still needs a lot of resources, so I tend to load things as needed. But having the most important Keyswitch patches from EWQLSO inside your VE Pro template is really neat :)

The VE Pro manual at great length describes how to setup plugins within VE Pro, but I‘ll give you a brief rundown:

- You should start the VE Pro server first without Logic running. Choose that server that fits your bit architecture, i.e. if you want to use 64bit plugins like Omnisphere, use the 64bit version. Otherwise use the 32bit version. This is _not_ dependent on your sequencer! Even if you are using a 32bit sequencer, you can use the 64bit VE Pro server! You can use both servers at the same time without problems.

- If you do not own any instruments of the VSL or don‘t want to use any in this instance of VE Pro, delete the one Vienna Instrument that is automatically created. Just click on it in the overview on the left and hit delete on your computer keyboard.
- Then add instances of the plugins you want to have. You can add as many instances as you wish, with up to 32 MIDI ports (even if using an AU host, more about that later). Then fill the plugins with sounds and assign ascending MIDI channels starting from 2 to 16 for each port (or 1 to 16 if you want to use the first channel).

  IMPORTANT:
  If you are using Native Instruments Kontakt, you need to setup multiple outputs inside the Kontakt mixer. Kontakt does not do this on its own. DO THIS FIRST BEFORE LOADING ANY INSTRUMENTS INTO KONTAKT! You can setup this once and save it as a multi for future use.
- Add additional "Inputs" in VE Pro to route your audio out to the sequencer. There is no need to have a stereo pair for every single instrument. If you are using VE Pro on a slave machine, having so many channels would choke the ethernet connection very quickly, so maybe use one stereo output per section. Do not mix instruments of different sections in one output. You‘ll know why later. Or just think of your own intelligent system for reducing track count. I will show you how to control the volume of all your instruments from Logic later, so it is absolutely fine if a number of instruments share the same stereo output.
- If your instrument patches are mono, you should also use mono channels in VE Pro. I usually have instruments with ambience stereo and very dry ones mono, but that depends on how well they fit into the mix. If you work with lots of custom panning (remember, some instruments such as EWQLSO come pre-panned!), mono is better.
- Only for 32bit: Kontakt's memory server grabs all memory. So if you plan to use Play and Kontakt at the same time, instantiate Play first, then Kontakt. Or deactivate the Memory Server. Or just update to Kontakt 4.1 or higher, which is 64bit on Win and Mac. Play 3 is 64bit on all systems, so updating to Play 3 solves this.
- The VE Pro manual is pretty good in explaining how the whole setup part works. Read it :)
- You can create as many instances as you like, but be sensible! As a rule of thumb, every instance uses one core of your machine. You should leave a bit of room for your system and your sequencer. So on a MacPro with 8 cores, I‘d use 6 or 7 instances at the most. If you need more, get a slave (get one anyway!). On very CPU-friendly plugins like some Kontakt instruments, you can do more, but better be safe than sorry.
- Once all your instances are ready, save the whole thing as a metaframe.

Depending on the number of cores in your machine you need to set VE Pro as well as the instruments hosted within to the correct number of cores. As a rule of thumb, you need to have at least one core for your sequencer (if it is on the same machine as VE Pro). There is however no need to set aside one core for the operating system! So if you are on a machine with four cores (hyperthreading cores on Intel i7 count as well!) and have three instances of VE Pro and, say, Cubase, you should set VE Pro to one core per instance.

As for the settings of the plugins: Find out. Kontakt 3 to 5 usually do fine with multicore support enabled, but on other plugins performance could improve if you only let them use one core. Read more about that later in the Load Optimisation section.

I get lots of questions on why I use External MIDI tracks and so on. Let me quote a post by me from the VSL board (slightly edited):

If you use any multi instruments (checking the multi-instrument box when creating new tracks) the faders in Logic control the whole VI instead of only the one channel, so all your instruments on all channels will change volume with CC7. This is why I insert a transformer in the environment which makes the sliders effectively send not CC7 but another CC for volume. As you cannot (easily) add a transformer to the "regular" multi instruments, I chose to use external midi tracks, which also allows me to only activate the MIDI channels I really need. Behind the scenes, the Logic multi instrument activates all channels. But if I only want 6 channels, with external MIDI I can only activate those 6 and leave the others switched off. Minimises interference.
Furthermore, with the regular multi instruments, you have the first channel as well as the first output on the instrument channel, but for all further midi channels you need extra tracks and auxes. It works, but the first channel looks different than the others. I want everything to be as easy and coherent as possible which is why I don't use MIDI channel 1 (which is on the track the VI is instantiated on), but start with 2 and use external midi tracks. Those allow me to cable everything to my liking in the environment and so far I could not find any disadvantages. Because channel 1 is not even activated, there is no need to have the VI slot visible, which is why I hide it. It is out of the way and I only see my MIDI tracks. The hidden instrument tracks actually prevent clutter. I mean, why would you want the plugins to appear on the arrange page? They are VE Pro plugins, which you never need to touch after they are set up. I felt these tracks are visual clutter, so I am glad to get rid of them.
There is one last reason: Those multi instruments you use only came with Logic 8. My method works with earlier versions, too. This wasn't a big decision to use it as I use Logic 9, but I had a lot of people following the tutorial who still use Logic 7.

Neat trick: If you use the transformer to route the fader to a different CC, you can easily make the fader control a parameter just by changing the CC. So if you want to use the fader for, say, breath control, remap it to CC2 in the transformer. This only works with external midi tracks, if I am not mistaken.

IMPORTANT:
Some readers pointed me to the fact that you cannot use Logic's bounce in place with External MIDI tracks. This is correct. I never use bounce in place, so I totally forgot to mention it.

Instrument Plugins in VE Pro

Quite often, people are mixing up VE and VE Pro. Users of the VSL libraries are used to Vienna Ensemble (without Pro), which allows them to load their instruments outside of their sequencer. VE has been around for quite a while. VE Pro then added network support and, most notably, support for VST and AU plugins. That's the biggest difference. VE can only load Vienna Instruments while VE Pro can load everything.

Now when you load a VST or AU instrument in VE Pro, it automatically loads it multi-out capable. What does that mean? Well, it means that the instrument can send audio out on different channels. In some sequencers, including Logic, you need to choose whether you want a stereo version or a multi-out version. Sometimes there are lots of choices (like with Kontakt, for example). VE Pro does all of this automatically! So just load the AU or VST.

Plugins work exactly like in any sequencer for practical use, with one exception. Where you used to use host automation to automate stuff, you now need to use MIDI automation, because Logic will send to VE Pro via MIDI. This mostly concerns Kontakt. Other than that, you can do everything in exactly the same way as before. Basically Logic sends to VE Pro which sends to the AU/VST plugin and gives everything back to Logic. With VE Pro 5 you can now fully use parameter automation just as you can directly in Logic, so usability-wise there is no difference between plugins inside Logic and plugins inside VE Pro. With Logic it makes sense to assign CCs to automatable parameters in VE Pro, because it saves you having to use the Event Input plugin to take care of automation data. This is why in this tutorial, CC asignments are used for automation.

Please also note the difference between multi-timbral (more than one independent MIDI channel) and multi-out (more than one output stereo pair)!

Setting up everything with Logic

Finally we come to the whole reason for this looooooong tutorial: Creating a template in Logic. There are a few things that make doing this in Logic a bit harder than it could be. Some of those we can do something about, others we can‘t.

We will have to deal with the following "issues":
- Logic does not have track folders, i.e. you cannot group your tracks; all strings together, all woodwinds together or something along these lines. We can‘t do anything in this case, but we can make everything a bit more easy to grasp.
- Logic with its factory settings wastes acres of space on your screen. As Logic is about as customizable as it gets, we sure can find a remedy for that.
- Logic‘s implementation of multi-timbral plugins (plugins that react on more than one MIDI channel) is not really that great. By delving into the infamous environment we can do something about this, too.
- Logic has a bunch of nice bugs which affect us here. We cannot do anything in this case, but I can at least tell you and hope that Apple will fix them. And you can send them a message via their feedback form. They _do_ read it.

Despite all these issues, using a properly configured template in Logic is just great, so don‘t back away because of a few gripes.

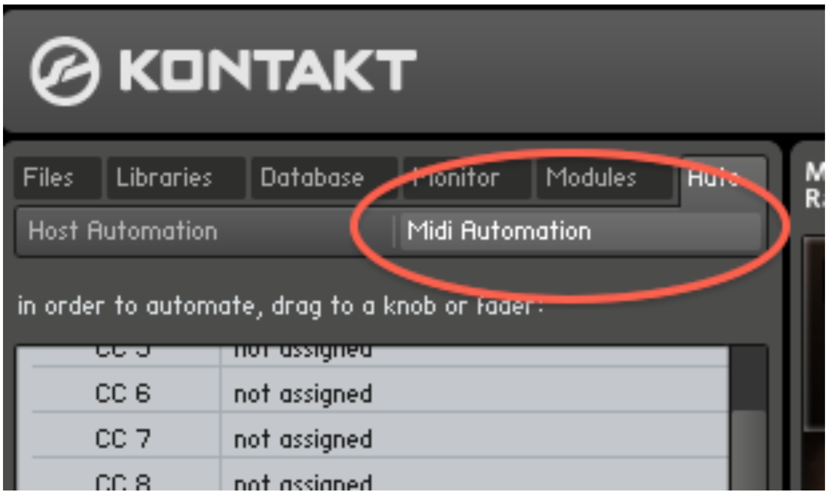
Making your instances "Logic-friendly"

Before going into Logic and creating all the tracks, we will do a thing inside VE Pro that makes working with Logic far easier:
Logic somehow, when using multi-timbral instruments, assigns CC7 (Volume) to the volume of all MIDI channels the instrument track (or rather its channel strip, which comes from the old days of early multi-timbral synths). That means than whenever you try to adjust the volume of one MIDI channel, all the others in the same AU instrument will jump to the same volume. With VE Pro this is even more severe: You want to control the volume of the instruments _inside_ VE Pro, not the volume of the software-instrument track within Logic.
But fortunately, the solution is dead easy: We just reroute CC7 to any other CC that is not used.

Most times, you can use CC11 (Expression), which is pre-mapped in nearly all plugins and works great. CC11 technically is a subset of CC7 and should always be used for dynamics when possible. Unfortunately, there are tons of ways to control "the volume", CC11 is but one of them, so study the manual of your library to find out how it works. Some libraries even use the modwheel for this, which I personally really like. For the libraries that don't you can reroute CC1 (which is modhweel) to volume, by the way.

But in Kontakt I advise you to use yet another CC because the mapping of CCs is not consistent with all Kontakt instruments. Some react to CC11, some do not. If they do, great, and you should use it in this case. By using a CC that is not used at all otherwise, you always know that you are getting the intended behaviour.
For ease of use, we will setup our instrument to control the overall patch volume with a new CC. This is useful for big volume changes and for roughly dialing in the general volume level for recording.
Because I love the number 7 and because CC17 is used by some synths, I use CC27. But you can use any CC that is not used any other way.
We will go through the steps of making Logic change the CC later, now we will do it in VE Pro with Kontakt as an example.

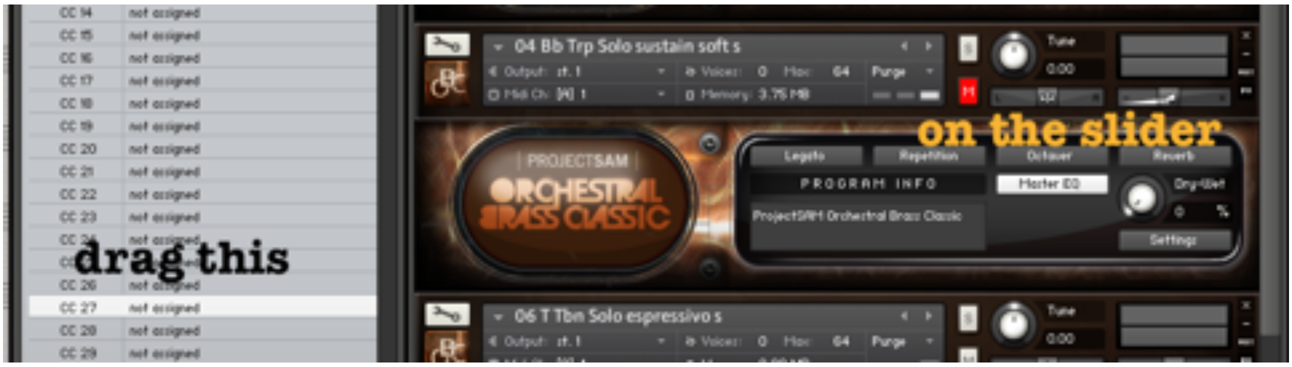
- In the Kontakt instance, select the "Auto" tab on the left of the window.



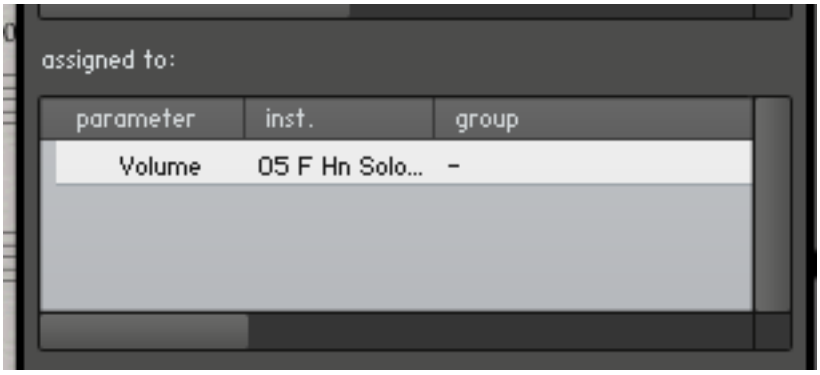
in order to automate, drag to a knob or fader:

CC 5	not assigned
CC 6	not assigned
CC 7	not assigned
CC 8	not assigned

- Switch to "MIDI Automation" as Kontakt will receive its instructions via MIDI (Host Automation in this case would be VE Pro directly).
- Look for your desired CC (i.e. 27). Grab it and drag it to the volume slider of all the individual instruments. The assignment will then be visible on the bottom left.



CC 14	not assigned
CC 15	not assigned
CC 16	not assigned
CC 17	not assigned
CC 18	not assigned
CC 19	not assigned
CC 20	not assigned
CC 21	not assigned
CC 22	not assigned
CC 23	not assigned
CC 24	not assigned
[illegible]	[illegible]
CC 26	not assigned
CC 27	not assigned
CC 28	not assigned
CC 29	not assigned

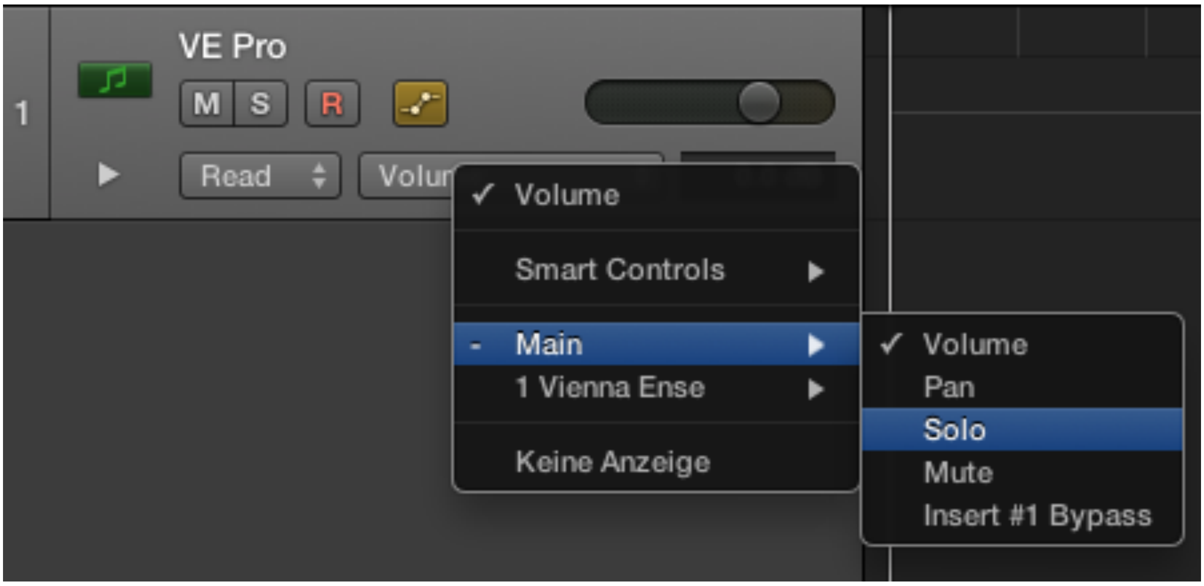


- Do not forget to save the metaframe!

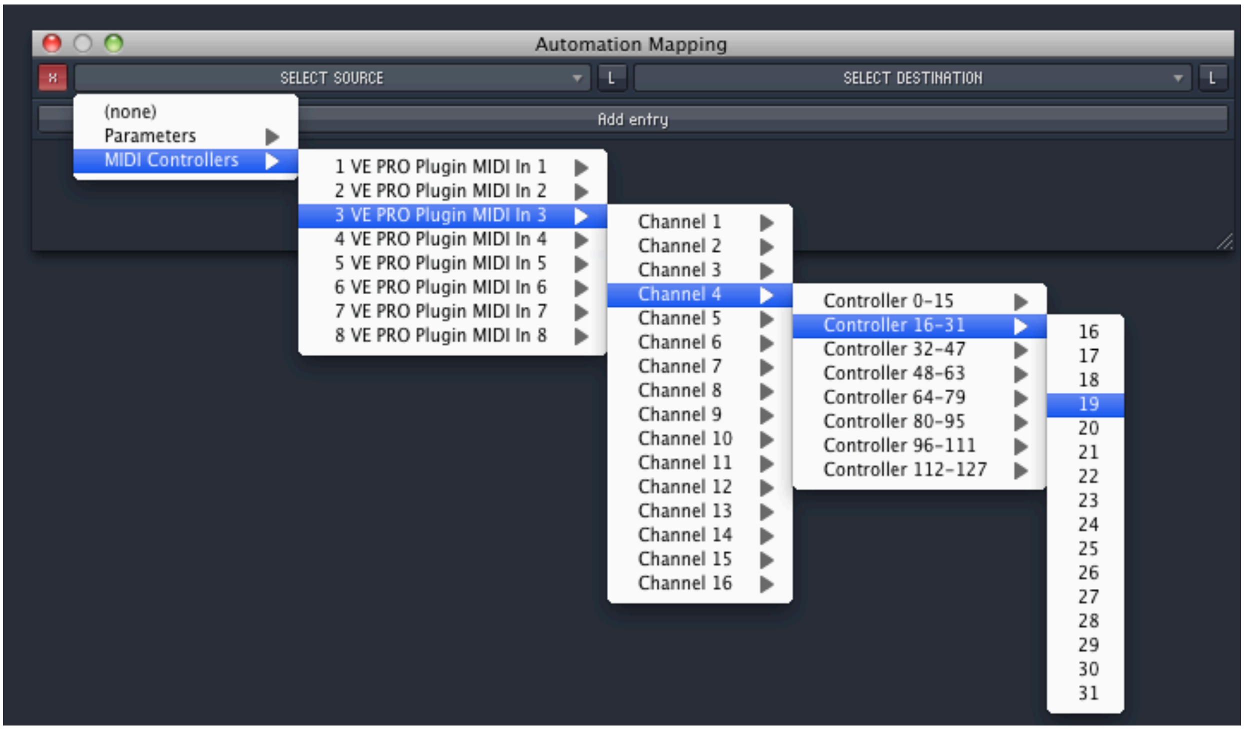
Naturally you can automate every other parameter in exactly the same way. If you have lots of CCs to deal with, think about getting a dedicated hardware controller with lots of faders and bells and whistles.
That‘s all there is to do inside VE Pro. Keep everything running and fire up Logic. Don't forget to save!

Parameter Automation with "Internal Parameters"

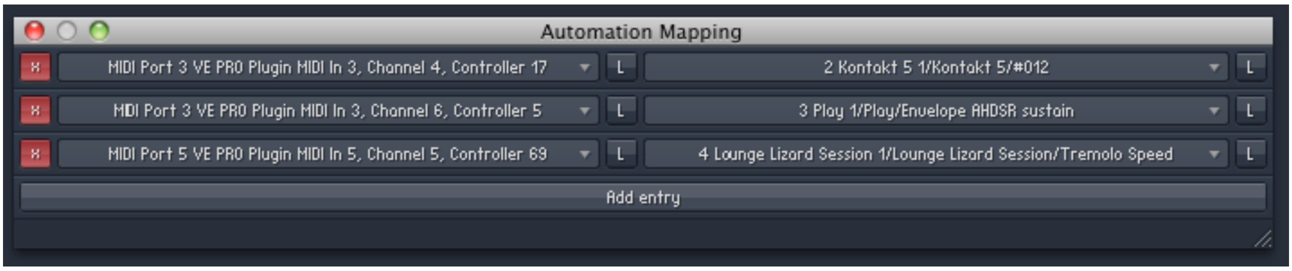
While it is extremely easy to assign CC-controlled automation, as we have just seen, only now with VE Pro 5 there is a way to control parameters not assignable to CCs. There are some plugins, which do not support controlling their parameters via MIDI CCs, but instead only via direct track automation. While these parameters can be controlled via automation if the plugin is inserted into a Software Instrument slot by selecting the parameter from Logic's automation lane (see screenshot below), it is not possible to directly assign a CC to control them like for example Kontakt does.



VE Pro 5 has a feature, though, that remaps incoming CCs to those internal parameters. You can access is through VE Pro's window menu, where you then can configure everything to your liking.



In the following screenshot you can see that this way it is possible to control EW Play's ADSR envelope controls inside VE Pro, which was not possible before and is probably the number one question I receive via email all the time.



Automation Mapping

MIDI Port 3 VE PRO Plugin MIDI In 3, Channel 4, Controller 17 → 2 Kontakt 5 1/Kontakt 5/#012

MIDI Port 3 VE PRO Plugin MIDI In 3, Channel 6, Controller 5 → 3 Play 1/Play/Envelope AHDSR sustain

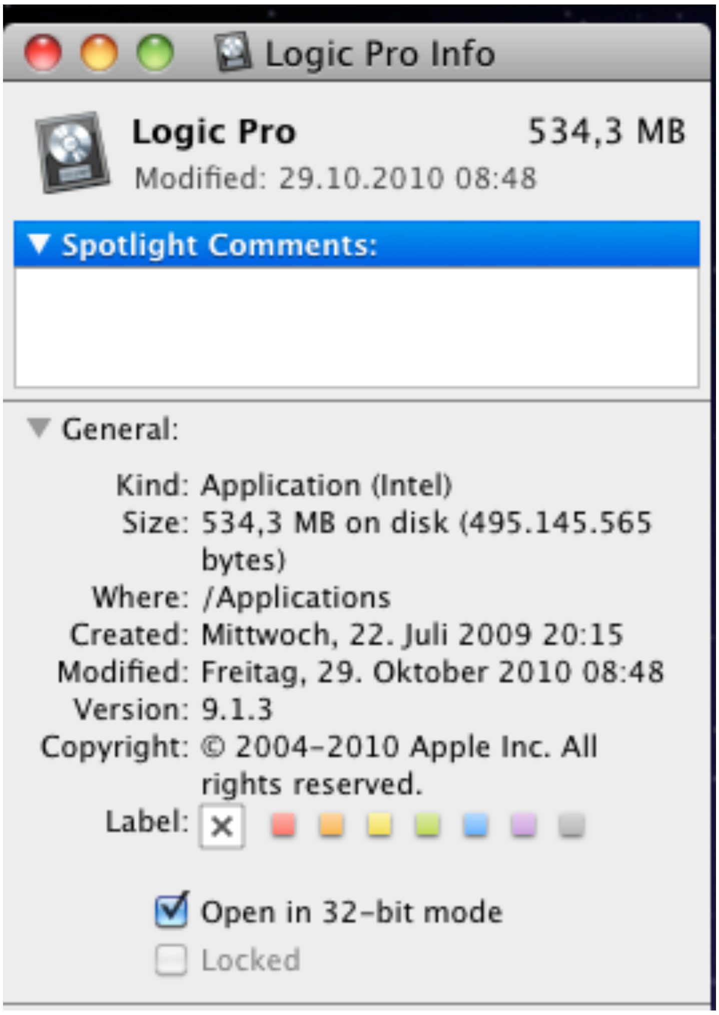
MIDI Port 5 VE PRO Plugin MIDI In 5, Channel 5, Controller 69 → 4 Lounge Lizard Session 1/Lounge Lizard Session/Tremolo Speed

Once you have set these assignments up, you can use physical faders and knobs on your control surface(s) to draw automation in Logic's automation lane. Select the CC you wish to draw automation for and record/pencil in the automation data as usual.

Starting Logic Pro 9 in 64bit

Logic Pro X does not have a 32bit mode anymore, so if you are using LPX, you can ignore the following section in italics :)

If you have Logic Pro 9, you should start it in 64bit mode unless you plan to use lots of 32bit plugins within Logic. If a large part of your sessions will run from VE Pro, you can gain a lot of speed when using the 64bit mode.
Don't know how to switch modes in Logic?
Go to your applications folder and find Logic Pro.app. Select it and press Cmd-I ("Info"). A window will appear where you can choose to start the application in 32bit mode. If the box is checked, uncheck it. And yes, I am perfectly aware that this screenshot is ancient, but even on 10.9 with Logic 9.1.8, it still looks the same :)



Logic Pro Info

Logic Pro 534,3 MB
Modified: 29.10.2010 08:48

▼ Spotlight Comments:

▼ General:

Kind: Application (Intel)
Size: 534,3 MB on disk (495.145.565 bytes)
Where: /Applications
Created: Mittwoch, 22. Juli 2009 20:15
Modified: Freitag, 29. Oktober 2010 08:48
Version: 9.1.3
Copyright: © 2004-2010 Apple Inc. All rights reserved.
Label: ☒

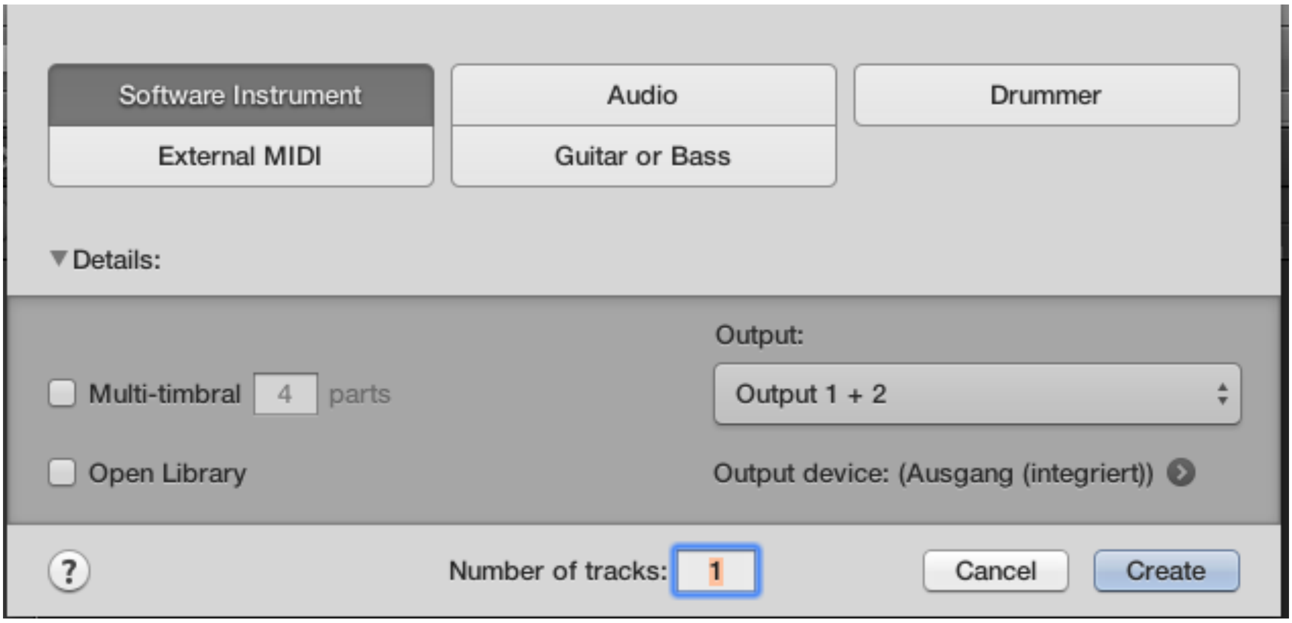
☑ Open in 32-bit mode
□ Locked

In 10.6.3, Apple introduced some performance enhancements for Logic 9.1.1, which makes the 64bit mode a lot better than 32bit when you are mainly using 64bit plugins. As you may have guessed, VE Pro is a 64bit plugin inside Logic.

Creating tracks

Now we will get our first VE Pro instance into Logic. I will describe everything step by step so you should have no difficulties following what to do:

Open a new project within Logic. When the dialog asks you to create new tracks, create one (!) Software-Instrument track. Do _not_ check multi-timbral! What we are doing here is creating a single track that will host the VE Pro plugin. This track will react to MIDI channel 1, which we agreed to not use. That means that you can safely tuck this track out of the way later on. For every patch assigned to a MIDI channel inside your VE Pro instance we will create special MIDI tracks. This is very similar to other DAWs such as Digital Performer, where you need to create one track hosting the instrument and then a number of MIDI tracks that are used to play and record.



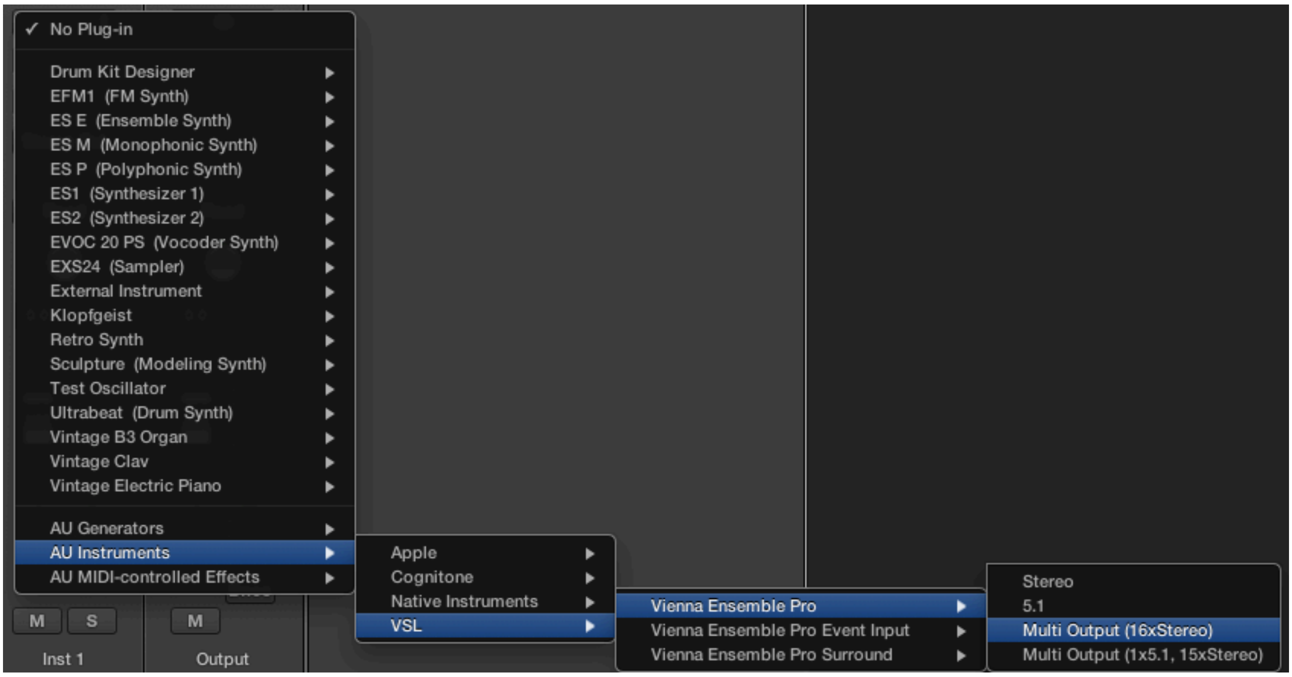
□ Multi-timbral 4 parts

□ Open Library

Output: Output 1 + 2

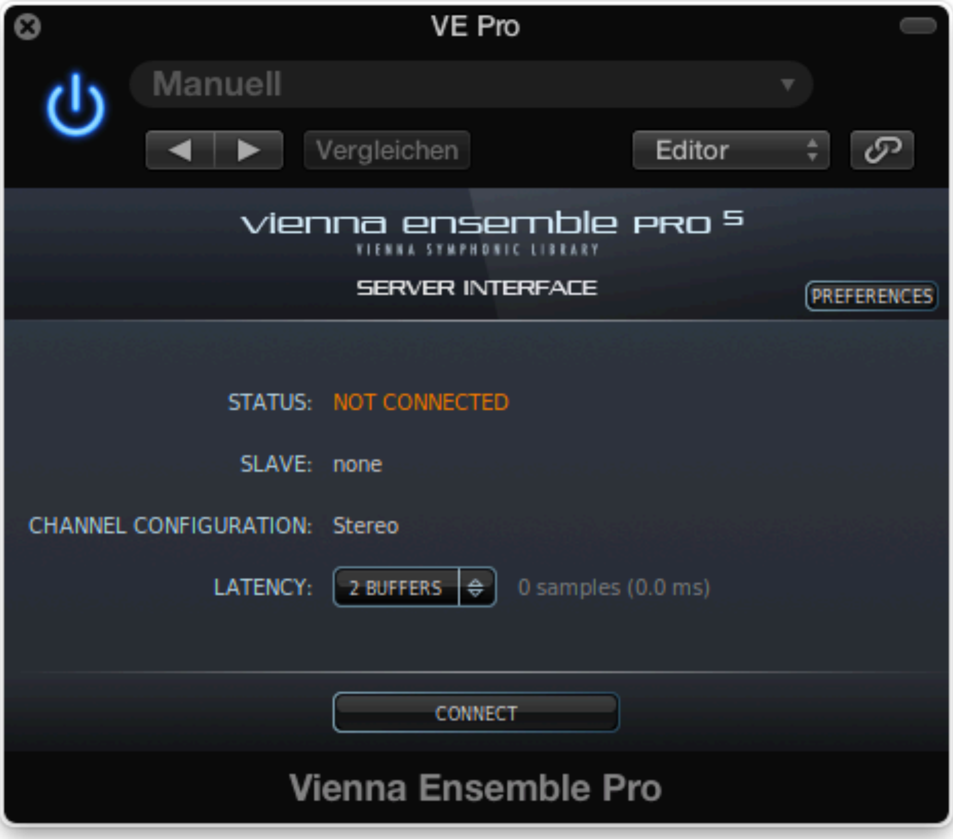
Output device: (Ausgang (integriert))

Number of tracks: 1



Instantiate VE Pro 64bit multi-out into this track.
Even if your VE Pro instances are 32bit, you can still use the 64bit plugin to access them. This is one of the advantages of VE Pro as it allows you to run a 64bit DAW while still using 32bit plugins without resorting to the AU bridge.

In the VE Pro window that opens, connect to your desired instance.



Manuell

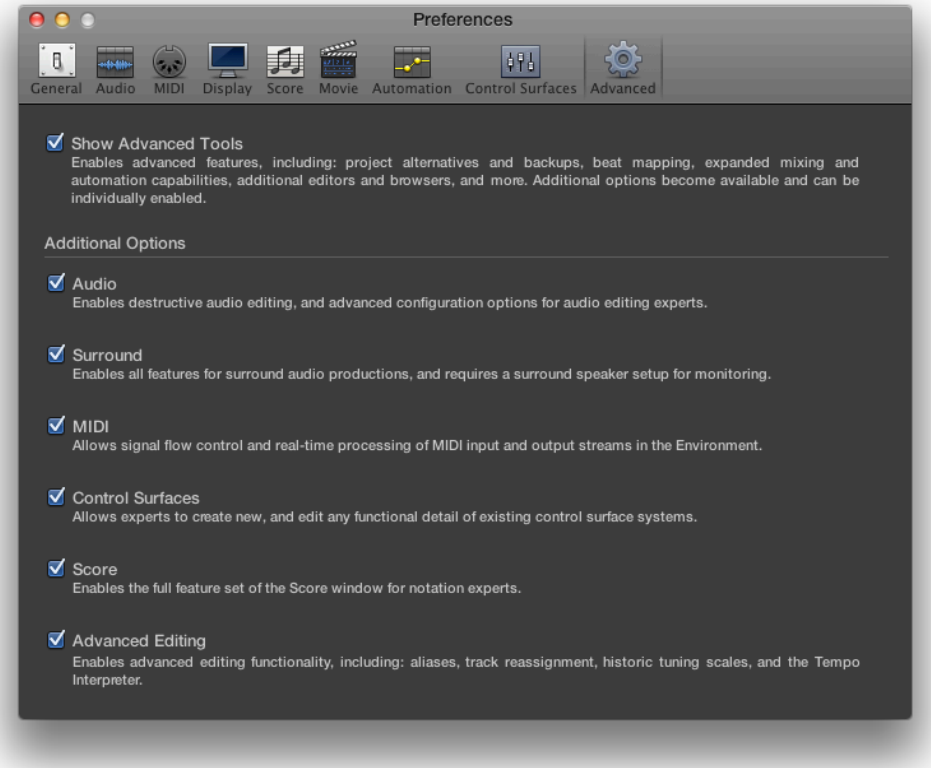
STATUS: NOT CONNECTED

SLAVE: none

CHANNEL CONFIGURATION: Stereo

LATENCY: 2 BUFFERS 0 samples (0.0 ms)

Now go to the environment and make sure that "Mixer" is selected on the top left.
Note: Make sure you have all the advanced options checked in Logic X‘s preferences. Otherwise you won‘t be able to see the environment.



☑ Show Advanced Tools
Enables advanced features, including: project alternatives and backups, beat mapping, expanded mixing and automation capabilities, additional editors and browsers, and more. Additional options become available and can be individually enabled.

Additional Options

☑ Audio
Enables destructive audio editing, and advanced configuration options for audio editing experts.

☑ Surround
Enables all features for surround audio productions, and requires a surround speaker setup for monitoring.

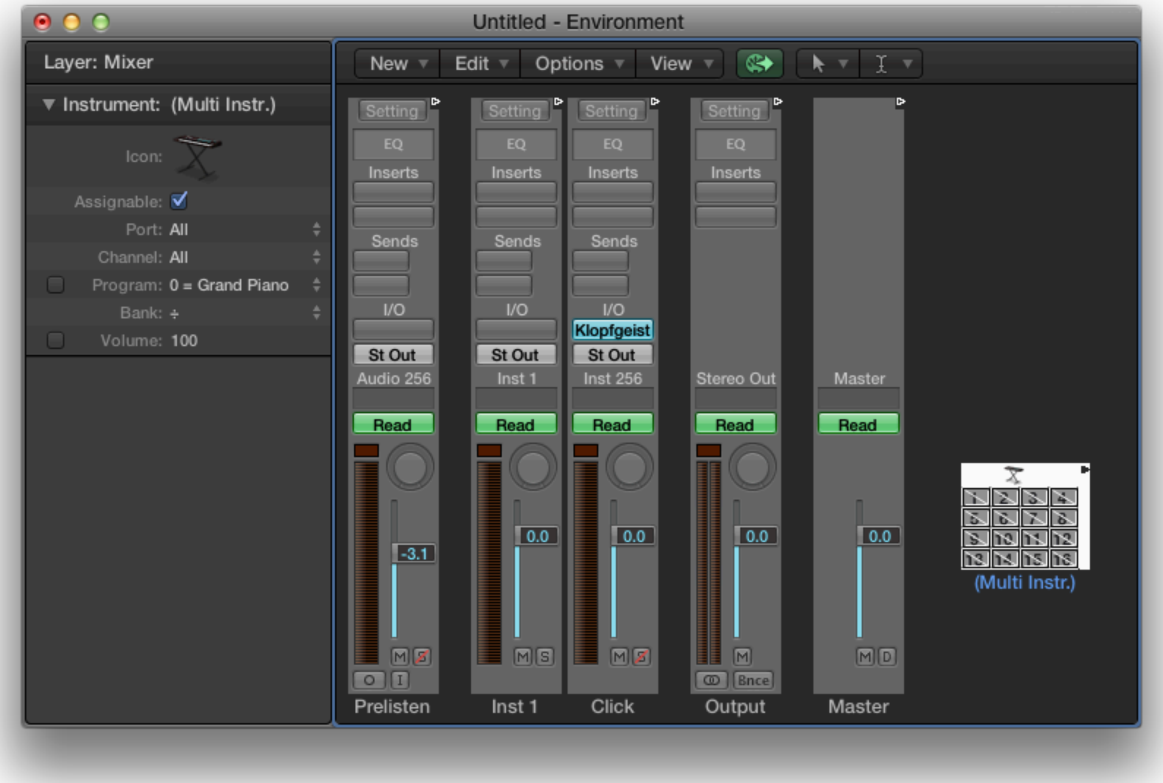
☑ MIDI
Allows signal flow control and real-time processing of MIDI input and output streams in the Environment.

☑ Control Surfaces
Allows experts to create new, and edit any functional detail of existing control surface systems.

☑ Score
Enables the full feature set of the Score window for notation experts.

☑ Advanced Editing
Enables advanced editing functionality, including: aliases, track reassignment, historic tuning scales, and the Tempo Interpreter.

Create one new multi-instrument from the "New"-menu. Give it a sensible name that describes the content of this particular VE Pro instance, i.e. "EWSO Brass" or "SYM2 Clusters".



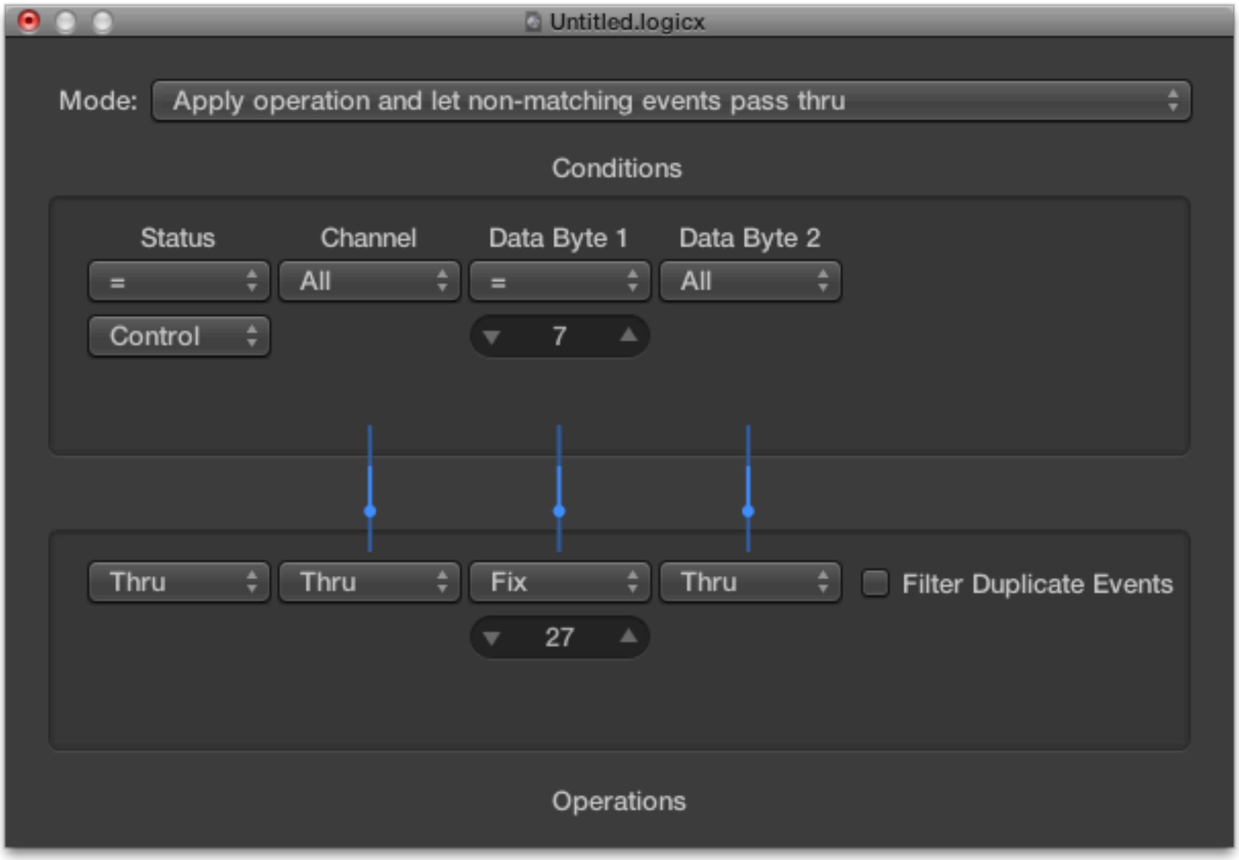
Layer: Mixer
▼ Instrument: (Multi Instr.)
Icon:
Assignable: ☑
Port: All
Channel: All
□ Program: 0 = Grand Piano
Bank: ÷
□ Volume: 100

Now check any number of MIDI-Channels you require. You can activate all channels, but I recommend activating only those you will really need.

Important:
Click all the channels in order, i.e. 2,3,4,5,6,7.... Otherwise Logic will mess up things when creating the MIDI tracks later! If you decide to follow my advice not to use channel 1, don't even activate it.

Create a Transformer object from the "New"-menu. Double-click it to open it. This is used for making Logic reroute the volume CC from CC7 to whichever CC you specified in your plugins within VE Pro.

Choose the following settings:



Untitled.logicx

Mode: Apply operation and let non-matching events pass thru

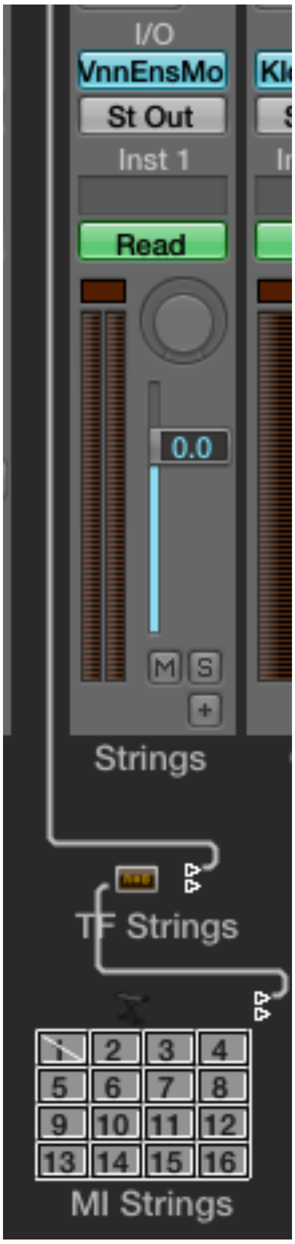
Conditions

Status	Channel	Data Byte 1	Data Byte 2
=	All	=	All
Control		7	

Thru	Thru	Fix	Thru	□ Filter Duplicate Events
		27		

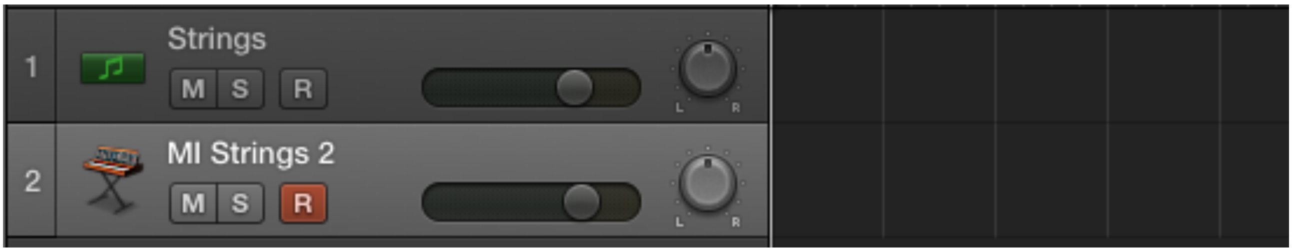
Operations

Now cable the multi to the transformer and the transformer to your VE Pro instrument. Click "Remove" when the dialog box comes up.

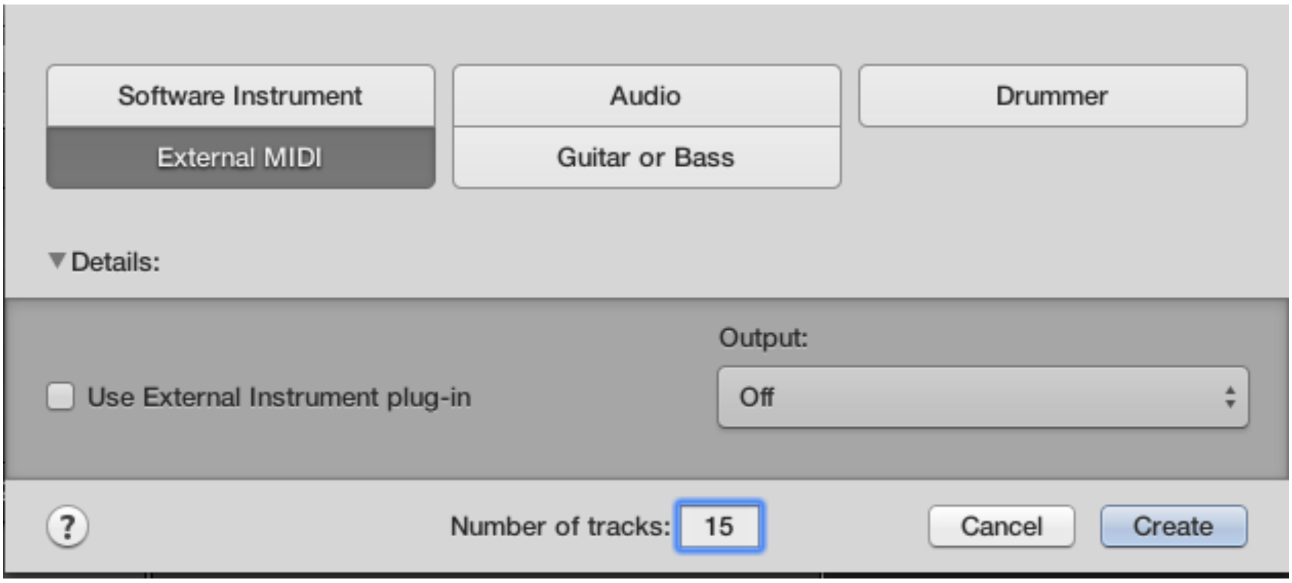


You can now close the environment.

Back in the arrange window, we will now create all the tracks for your instruments.



Click on "Track"-"New" and choose External MIDI. Create as many tracks as you need for the particular multi instrument (15, usually).

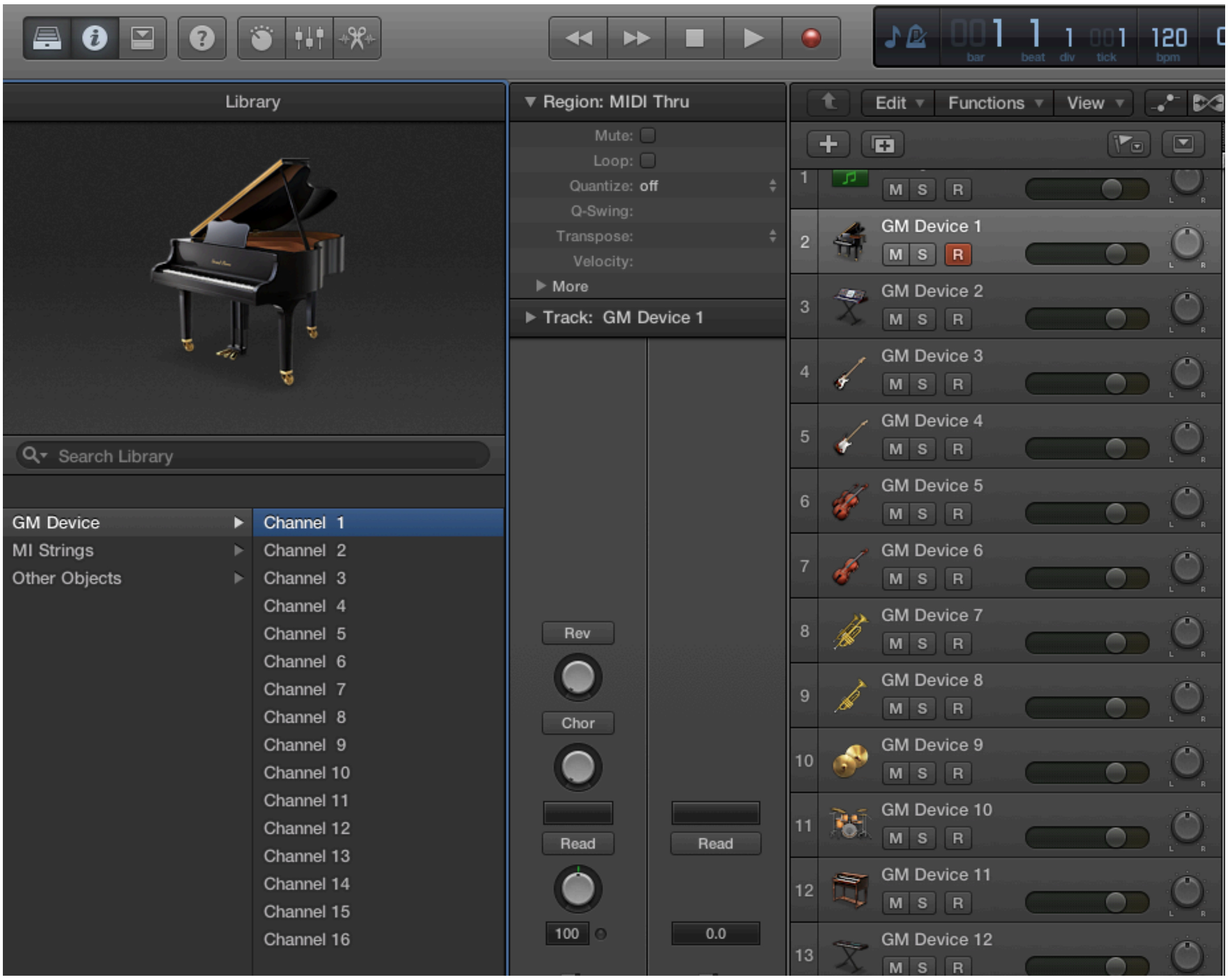


□ Use External Instrument plug-in

Output: Off

Number of tracks: 15

You now have a number of External MIDI tracks in your project. They are assigned to the GM Device right now and you need to remap them one by one to your Multi Instrument. It is possible to create and assign one track, then create the others with the „Create new track with next MIDI channel“ command.



Region: MIDI Thru

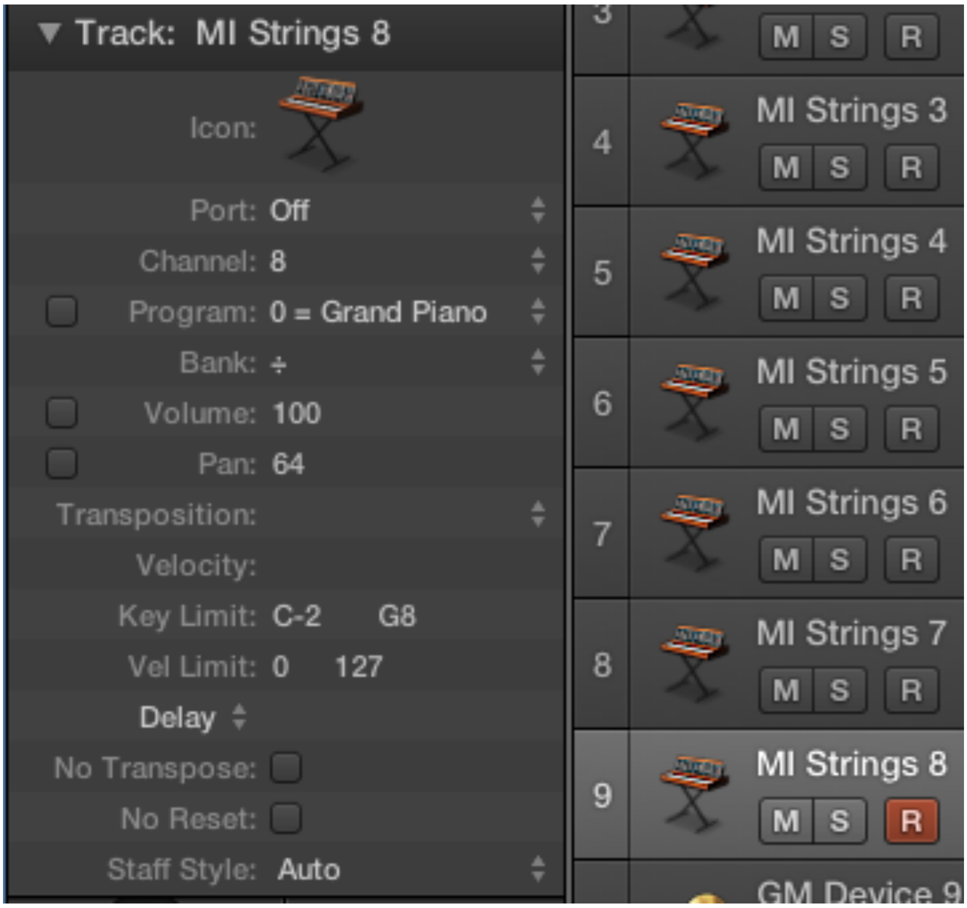
Mute: □
Loop: □
Quantize: off
Q-Swing:
Transpose:
Velocity:
More

Track: GM Device 1

When you are finished, you need to name the track as per the instruments they contain.

You should now be able to play all your instruments when the corresponding track is selected and you can use the volume slider to lower or raise the volume inside VE Pro!
But if you save the project and reopen it, you will see that the instruments in VE Pro do not jump to the correct volume if it differs from the volume in VE Pro. Try it out: Save the project and then change any volume slider (i.e. in Kontakt) within VE Pro. When you now reload the Logic project, the volume slider inside Kontakt will only jump to the correct value when you move it. The reason for this is that Logic does not send MIDI volume when the project loads.

But we can make it to.
Open the parameter-box of the first track and look for the "Volume" checkbox. Check it.
Do this with _all_ MIDI tracks. On a template with 200 tracks you need to do this 200 times.



▼ Track: MI Strings 8

Icon:
Port: Off
Channel: 8
□ Program: 0 = Grand Piano
Bank: ÷
□ Volume: 100
□ Pan: 64
Transposition:
Velocity:
Key Limit: C-2 G8
Vel Limit: 0 127
Delay
No Transpose: □
No Reset: □
Staff Style: Auto

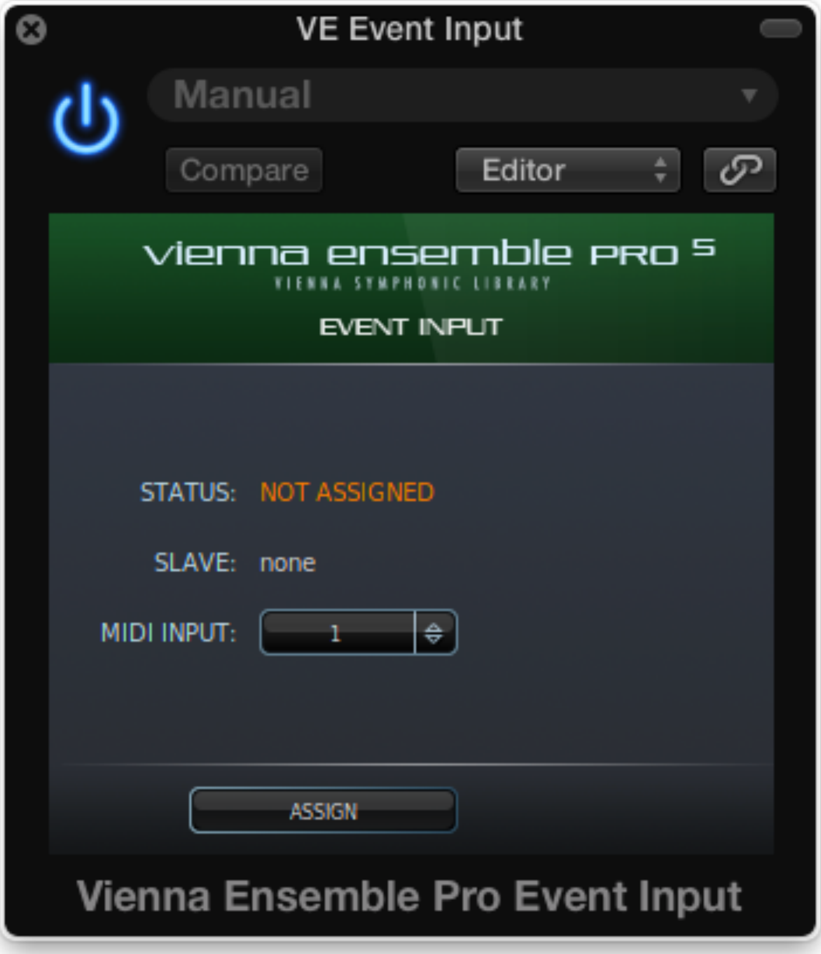
Caution!
Logic, as of 9.1.7 has a bug that makes the value of this checkbox _not_ change when moving a fader on a control surface. Only when you move the fader with the mouse, the value changes and is saved. Therefore you should ALWAYS after using a control surface click every track fader with the mouse before you save. When reloading the project, the value from the checkbox is loaded! Hopefully this bug will be fixed by Apple.
Actually, it is no big deal as you can use automation just fine and you should not need to touch the faders anyway on a balanced template (we‘ll come to this!).

Your instance of VE Pro should now work. Repeat the steps above for every other instance. Do never forget to give everything a sensible name! In three years time, "Multi Instrument 1-2678" will mean nothing to you, but "MI EW Strings" will.

Multiple MIDI ports with the AU plugin

As mentioned before, until VE Pro 5 you were only able to use a single MIDI port with 16 channels when using VE Pro as an Audio Unit plugin (which is the only way to use it in Logic). VE Pro 5 has a workaround for this problem: The Event Input plugin allows you to send MIDI information to ports 2 to 7 of any instance.

When you run out of channels on the first port (which is addressed through the VE Pro AU plugin), you need to create a new software instrument track, but instead of adding a new VE Pro AU plugin and connect it to a new instance of VE Pro, you add the VE Pro Event Input plugin.

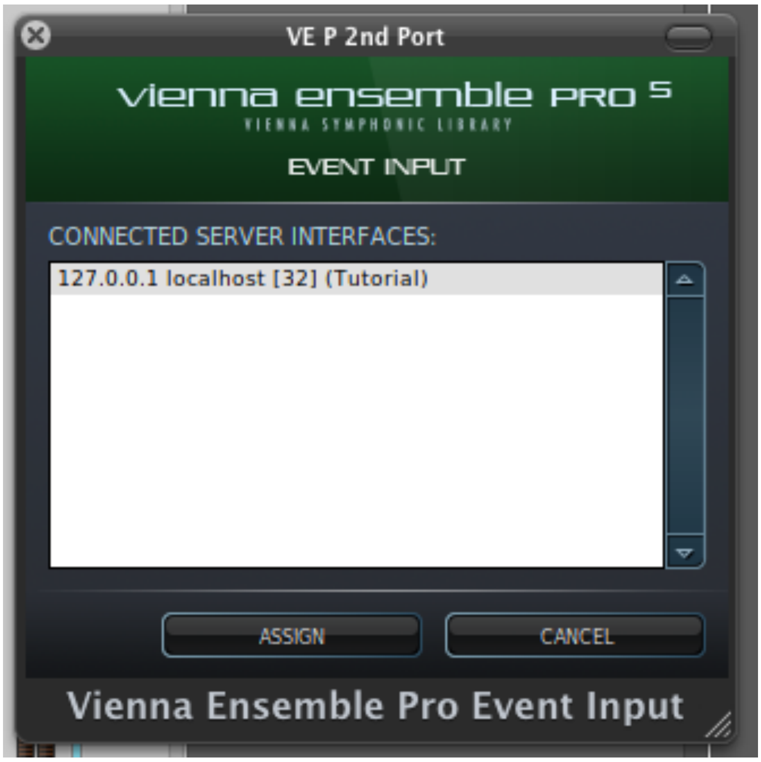


STATUS: NOT ASSIGNED

SLAVE: none

MIDI INPUT: 1

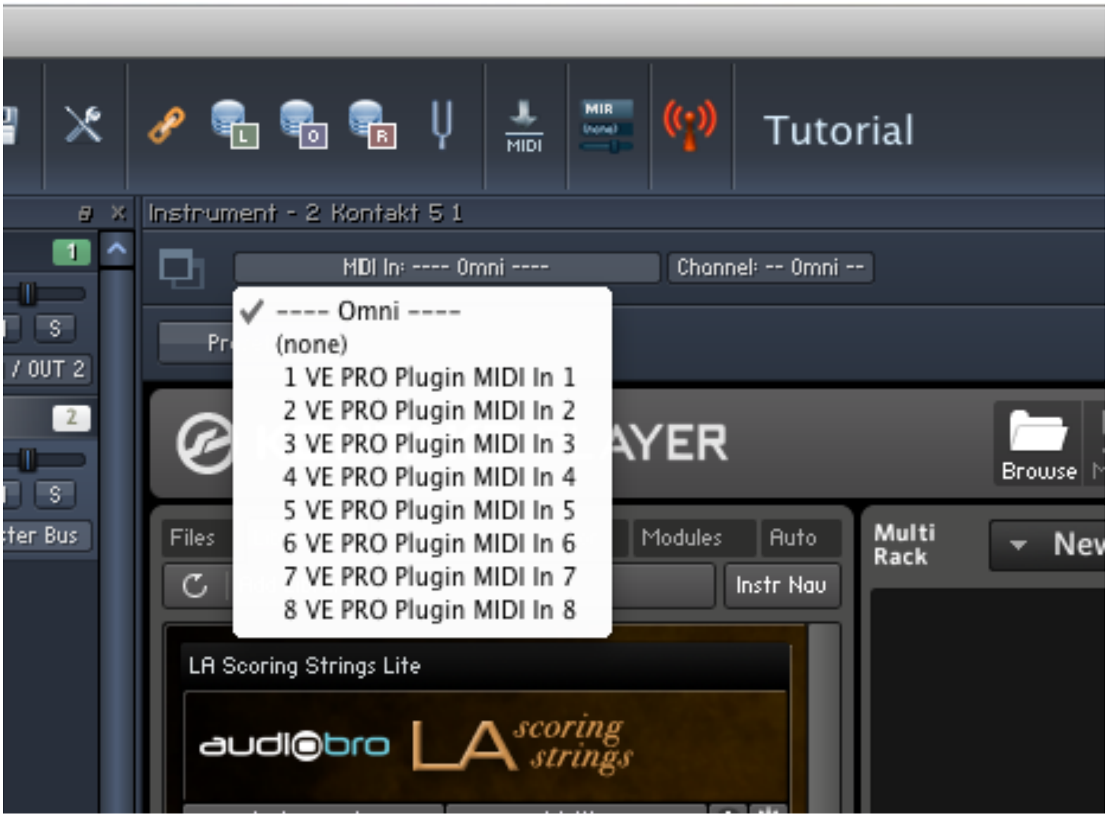
Set the MIDI port (MIDI INPUT) and assign the instance you want to use this port with. Just as with the first MIDI port used through the VE Pro Plugin, you can use all eight ports in every instance of VE Pro now!



CONNECTED SERVER INTERFACES:

127.0.0.1 localhost [32] (Tutorial)

In your VE Pro instances you can select the MIDI port you want a specific plugin to listen to at the top of the window or at the top of the corresponding channel strip in the mixer. One you have selected the port in both VE Pro and the Event Input plugin in Logic, everything else works just as with Port 1 (setting up tracks, naming....).



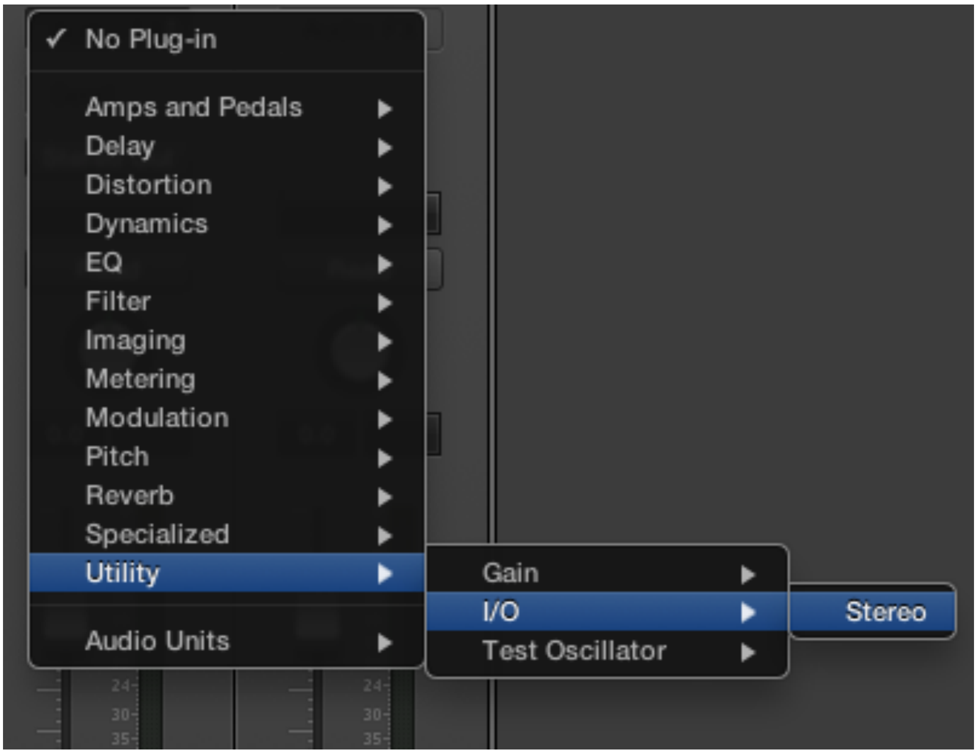
Instrument - 2 Kontakt 5 1

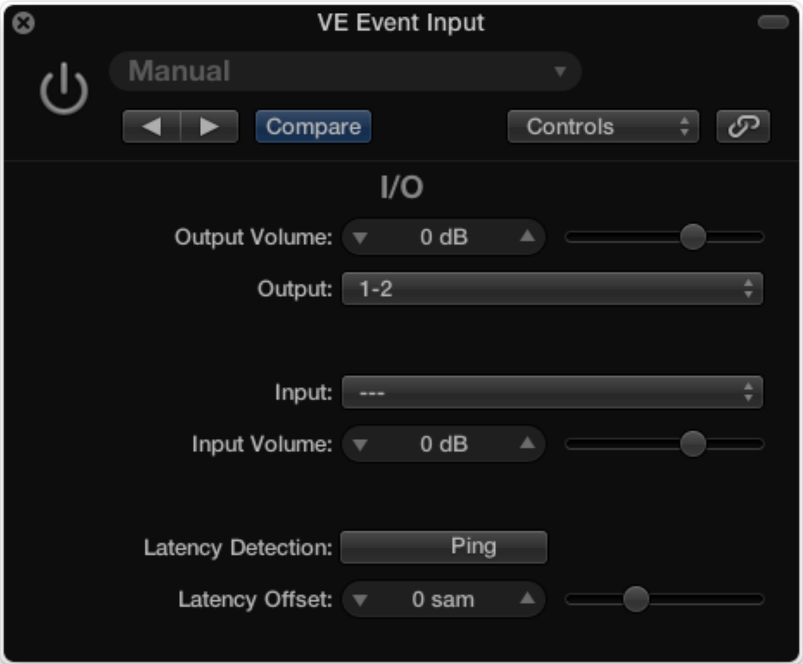
MIDI In: ---- Omni ----

Channel: -- Omni --

MIDI
MIR
Tutorial
/ OUT 2
ter Bus
✓ ---- Omni ----
(none)
1 VE PRO Plugin MIDI In 1
2 VE PRO Plugin MIDI In 2
3 VE PRO Plugin MIDI In 3
4 VE PRO Plugin MIDI In 4
5 VE PRO Plugin MIDI In 5
6 VE PRO Plugin MIDI In 6
7 VE PRO Plugin MIDI In 7
8 VE PRO Plugin MIDI In 8
AYER
Browse
Files
Modules
Auto
Instr Nav
Multi Rack
New
LA Scoring Strings Lite
audiobro LA scoring strings

There is, however, one caveat: To avoid very high latency during playback or recording of tracks when using the Event Input plugin, you need to make sure that Logic puts all VI Pro software instruments into Live Mode.
This is very easy: In **every** Software Instrument channel strip, where you have a VE Pro AU plugin or a VE Pro Event Input Plugin, you need to insert a copy of the I/O plugin from the Utility folder as an insert effect. Set the Output to Logic's main out (usually 1-2) and bypass it with the button on the top left.
Do this with **every** VE Pro / VE Pro Event Input channel strip!




VE Event Input

Manual

I/O

Output Volume: 0 dB

Output: 1-2

Input: ---

Input Volume: 0 dB

Latency Detection: Ping

Latency Offset: 0 sam

This puts all the channels into live mode and makes sure you can play them without high latency. Unfortunately, it also means that all the tracks are processed on a single CPU core by Logic.

This is why there is a second way to deal with this, namely using the Multiport layer, which VSL recommends using on large templates.

Using the Multiport Layer

Using the Event Input plugin has the disadvantage that all tracks are processed by Logic on the same CPU core. This is not as bad as it sounds because VE Pro does most of the actual processing (because it produces the sound), but still with extremely large projects you may run into issues.

VSL have devised a way to create multiple MIDI ports in Logic's environment. Download the Multiport Layer from your User Area on the VSL website and open it. Save it under a new name - this will become your template project.

Now you can connect VE Pro in each software instrument track, and the pre-assigned tracks should all play the correct sounds.
Done.

Event Input, Multiport Layer or Single Instances?

The Event Input plugin clearly is not the best solution in Logic because of the constraints of Live Mode as noted earlier. VSL recommends the multiport layer, now in its second major version. This layer all in all works relatively well and has the least amount of setup involved. However I have found that it sometimes has issues when project use a lot of CC automation. Dropped notes or stuck notes may occur and it simply does not feel very stable.

Also using the multiport layers means that lots of sounds will share the same instance and therefore the same VE Pro plugin inside Logic. Due to the nature of Logic‘s engine this is actually detrimental to performance.

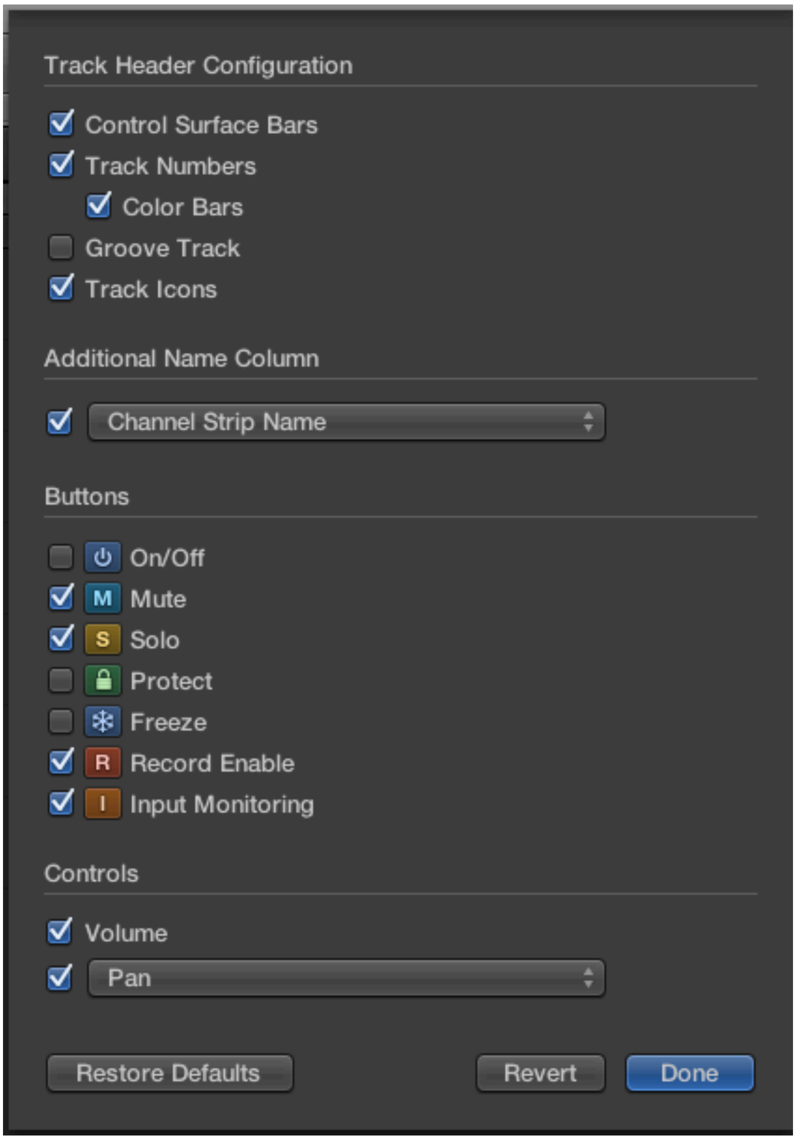
To me the best way to set up templates in Logic is by having single instances with 16 MIDI channels each, so only using their first port. Despite resulting in lots of instances, this nevertheless has shown the best performance. Dozens of instances are absolutely no problem in my experience and the load will be very evenly distributed.
Please note that this is ONLY valid for Logic and does not apply to other sequencers!

This whole instance setup is probably the hardest part about setting up VE Pro. If you are stuck and neither this tutorial not the manuals can help you, feel free to drop me a line via my website. I have planned and set up countless templates since VE Pro came out and I will gladly assist you!

Naming your tracks

I got a lot of questions about how to name your tracks and which setting to use.

Firstly, I recommend using "Channel Strip Name" as a secondary track name in your arrange. This allows you to see which channel on which multi instrument this particular track is assigned to..



Track Header Configuration

☑ Control Surface Bars
☑ Track Numbers
  ☑ Color Bars
□ Groove Track
☑ Track Icons

Additional Name Column

☑ Channel Strip Name

Buttons

□ On/Off
☑ M Mute
☑ S Solo
□ Protect
□ Freeze
☑ R Record Enable
☑ I Input Monitoring

Controls

☑ Volume
☑ Pan

Secondly, you should think of a naming convention. This heavily depends on what libraries you use in your template. If you only have one single library by, say, East West, just writing "EW Violin I" should be ok. If you have lots of libraries by the same company, you should differentiate.
Examples would be "EWgyp Vln F Mod" for "East West Gypsy Violin Fast Modwheel". Keep your abbreviations consistent at all times so you always find what you need. Have a special eye on plurals and numbers: One library may talk of "Violins I", while the other uses "Violin 1". For the sake of clarity, I would use the plural for sections and roman numerals for the Violins I and II. The former allows you to differentiate between solo

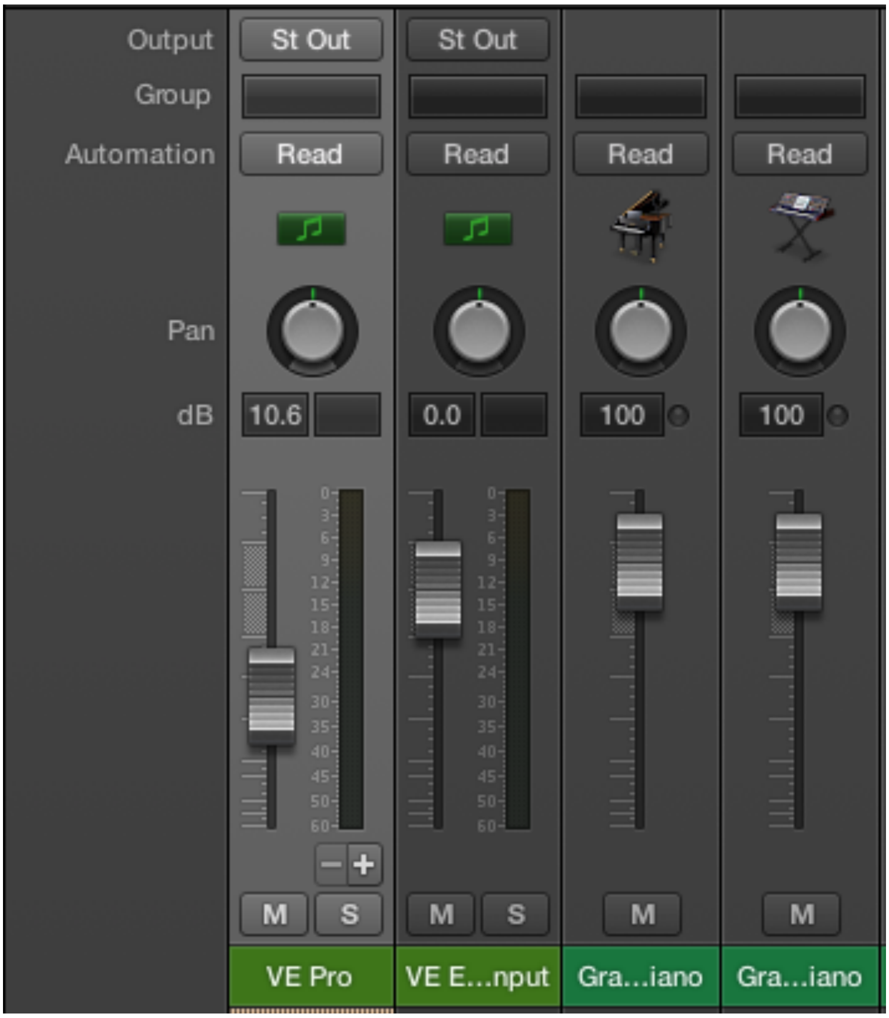
instruments and sections, the latter is useful for establishing a hierarchy: You may want to have an articulation for a certain library as your principal source of this particular articulation. I, for example, am quite fond of the Kontakt 2 Steinway piano, but also have a number of other pianos in my template. The Steinway is the principal piano, so it has a "1" in the name.

Expression control should be included in the title, as well as articulation selection. If the expression is controlled via modwheel, put a "Mod" in the name. Why not just an "M", you ask? Well, I prefer to reserve the "M" for marking patches using a mute. Keyswitch patches get a "KS", Modswitch patches (hello VSL!) get a "MS". And so on...

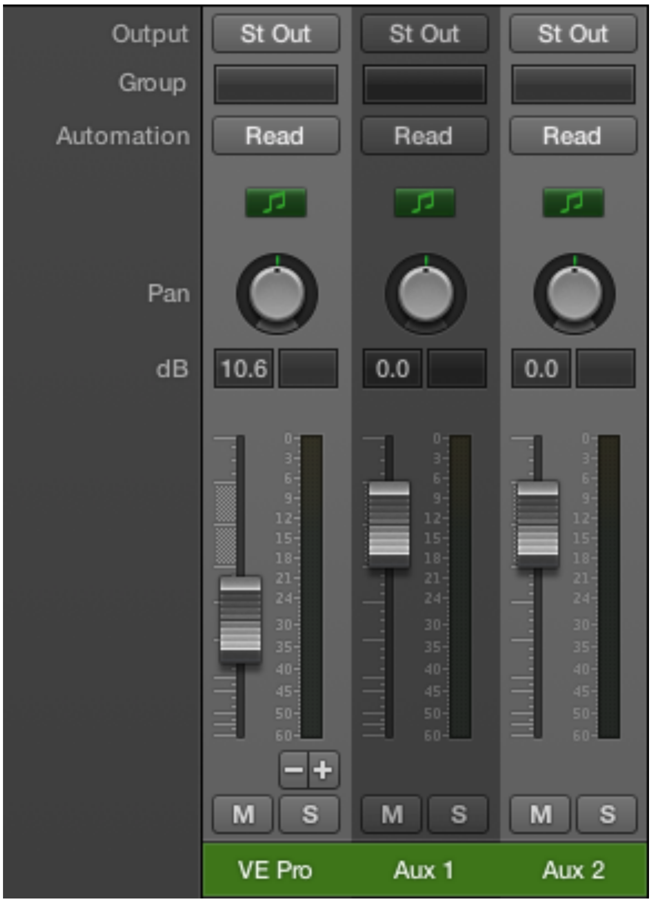
If you want to include further information (keyswitch mappings...), you can do this in the track notes (Logic 9 only).

Creating auxes

If you have assigned multiple outputs for an instrument plugin in VE Pro, you will need to create the corresponding auxes (auxiliary channels) in Logic. For this you go to the mixer where you will find a little + sign on your instrument track.



Click it to create auxes. They will automatically be assigned with ascending stereo outputs.



Name the auxes! It is pretty important to name them, otherwise you will have problems later when creating busses. Create as many auxes as you specified in VE Pro. The first stereo pair will come through the VI track (called VE Pro in my screenshot).

Ready for Prime Time

You are nearly there!

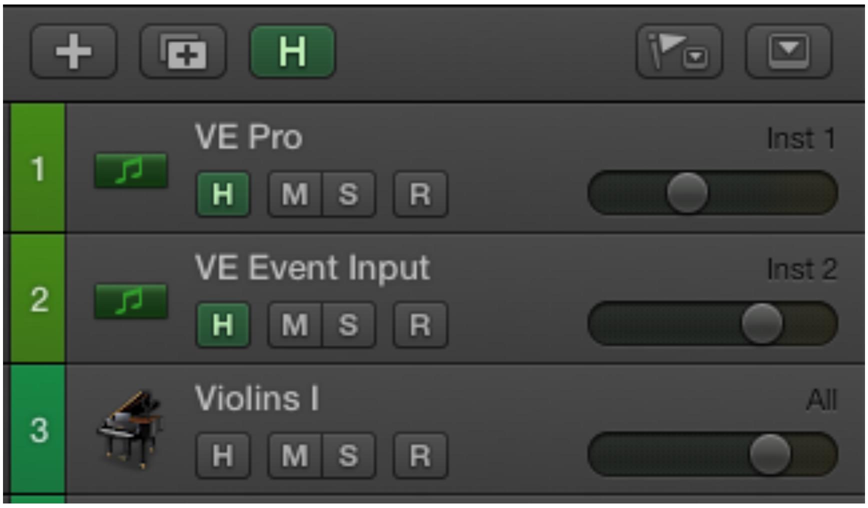
Now everything should play fine and reach your machine on the correct aux. The steps work the same when you are connecting to instances on a slave. In fact, you can even switch instances between computers. VE Pro only looks for the name, so just name the instance the same and VE Pro will find it.

There is one important thing left to do: Balancing tracks. You need to set all tracks to a volume level where lots of them can play together without clipping. Some instruments, especially from the VSL have their expression controlled by modwheel, so you need to set these very low! Just try it out. And remember the "MIDI controller not setting the volume checkbox" bug we discussed before! Make absolutely sure the volume settings are saved!!!

This is mostly just for trying out, you will rebalance things very often!
Though not strictly necessary, I would advise you to follow "analog" conventions: Try to balance all tracks within their VIs inside VE Pro, not with the faders in Logic. This has two advantages:

1) You can have all faders in Logic at 0db and only need to change things at the mixing stage, where you can easily see all changes: Everything that isn't 0db was changed :)
2) You need to touch the faders a lot less, making this an easy workaround for the bug detailed above.

One last convenience thing: Since all your instruments are controlled with MIDI tracks, you can hide the VI tracks.
Activate hiding tracks from the toolbar directly above the tracks and check all the tracks that you want hidden. Press H to hide, H again to show them.



Creating busses

You are very likely to have several sections in your template: Winds, Brass, Strings, Percussion, etc...
You are also very likely to have differently recorded libraries there. The VSL, for example is recorded very dry (without ambience of the recording stage) whereas the East West Symphonic Orchestra has very much ambience.

To make these different approaches to sampling blend well together, you need to put a bit of reverb on the dryer samples. Because each section of the orchestra needs a different reverb, but also because you can use one reverb for several instruments within one section, you should use busses. That means that you route any audio from specific audio auxes (that‘s why I told you to only use auxes for one section!) and apply a reverb (or other effects) to it before feeding it to your main outputs.

If we assume that you need one bus for dry samples and one for wet samples per section and a few extra things, you may use the following bus-setup:

1. Woodwinds
2. Woodwinds Reverb
3. Brass
4. Brass Reverb
5. Strings
6. Strings Reverb

7. Percussion
8. Percussion Reverb
9. Keys
10.Other
11.Electronic

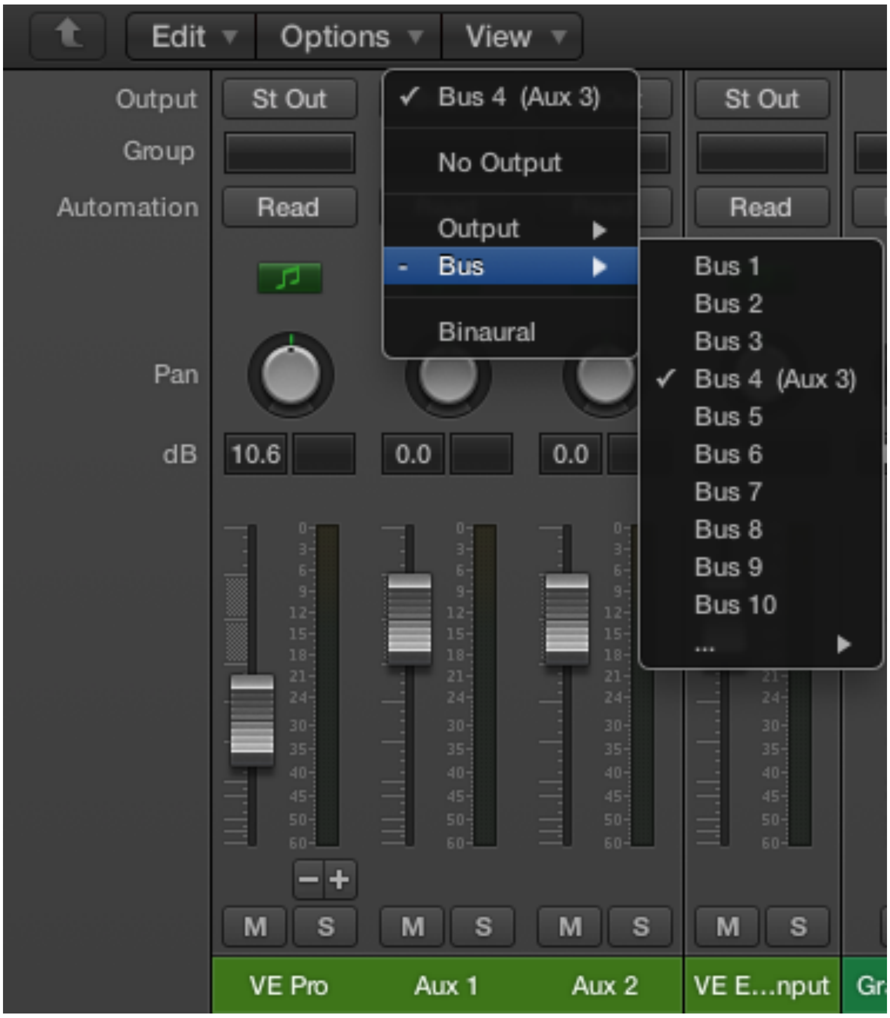
Busses 1 to 8 may then go into
12. Basic Orchestra

9, 10 and 12 then go into
13. Full Orchestra

13 (i.e. everything other than the electronics) and 11 (electronic) then go directly to your audio out.

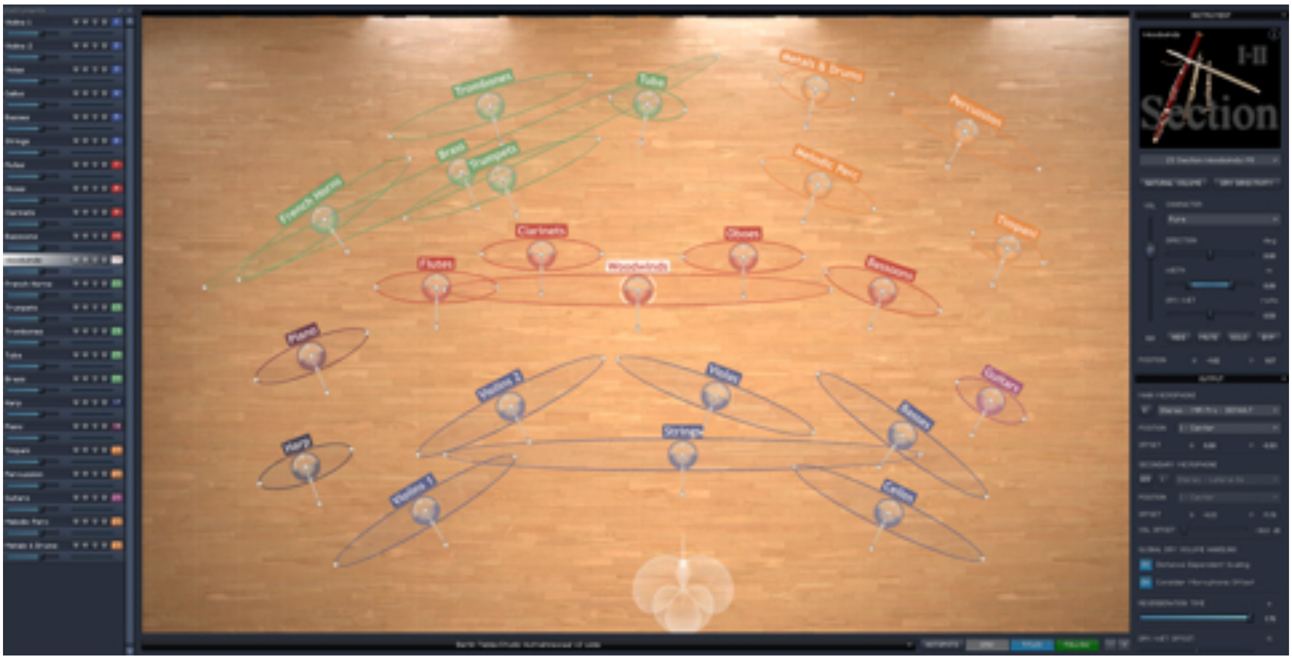
This may seem a lot of work (it is), but you are very, very flexible in terms of adding effects then! You can of course add as many busses as you want and will probably need to do so.

Creating busses is dead easy. Just right-click on the "Stereo Out" part of the aux or VI track you want to assign to a bus. Then in the menu select the bus you want. If it does not already exist, Logic will create it for you. Don‘t forget to name it then!



Using MIR Pro / MIR Pro 24

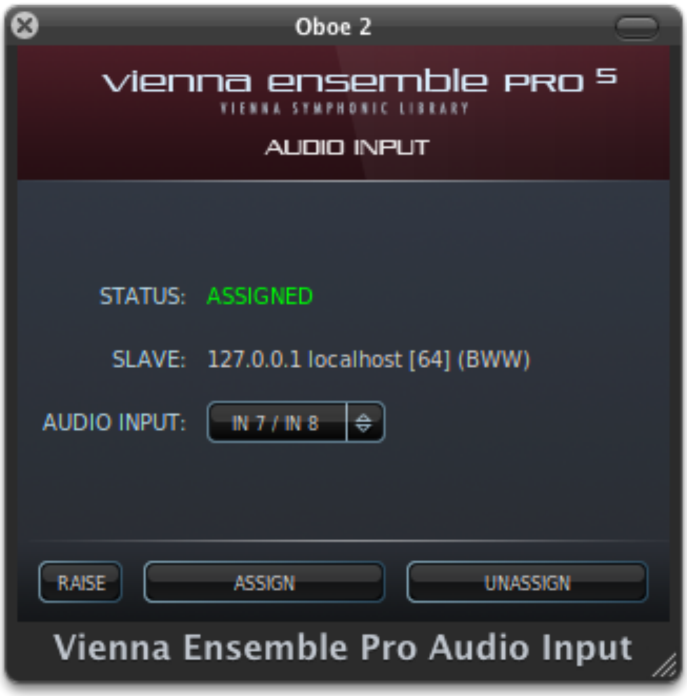
VSL MIR has shaken the world of sampling quite bit - after all it promises to bring your mixes into some of the world's greatest recording venues with an interface accessible not only to audio engineers, but also to regular composers without an MA in audio production. What MIR essentially does is to reproduce the way individual instruments (up to 24 in MIR 24) sonically behave wherever you put them on the stage. Instruments are represented with an icon each which you can drag around. For a large(ish) orchestra, this can look like this:



MIR works either within VE Pro, or as a standalone plugin which can be loaded into your sequencer. Both methods work equally well. The decision whether to use the plugin inside your sequencer or use MIR within VE Pro mainly depends on whether or not your main machine has the system resources to run MIR effectively in addition to the sequencer. You get one license for MIR with your purchase, which means that contrary to VE Pro, you can only use it on one machine at a time. If you do not have slaves, you can run MIR directly inside the VIframes. Just select it as an effect in VE Pro and it will automatically load.

It gets a bit more complicated when you are using slaves: Unless you buy additional licenses, you won't be able to use MIR directly within the instances on the slave(s). This leaves you two options:

1) With the help of the Audio Input plugin you can send audio from your sequencer to any instance of VE Pro. The audio goes from VE Pro on the slave (within your template) into the sequencer as usual, but instead of being processed within Logic, it is first sent to MIR (inside VE Pro) and goes back to Logic and ultimately your monitors. This also enables you to use a dedicated machine for MIR, if you use a lot of individual instruments.



STATUS: ASSIGNED

SLAVE: 127.0.0.1 localhost [64] (BWW)

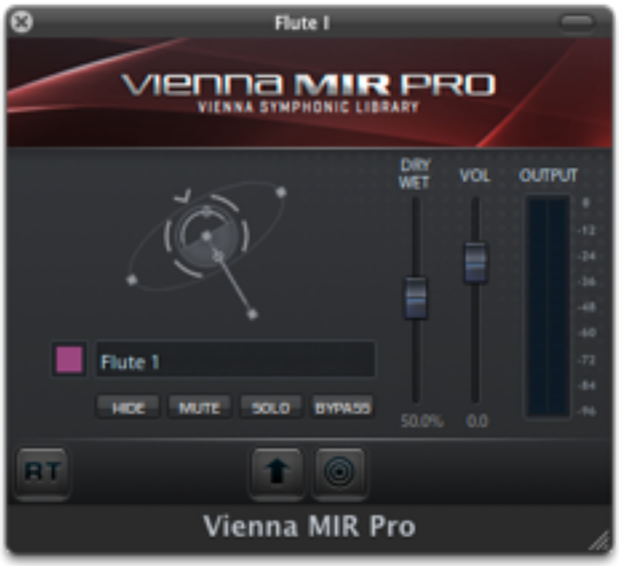
AUDIO INPUT: IN 7 / IN 8

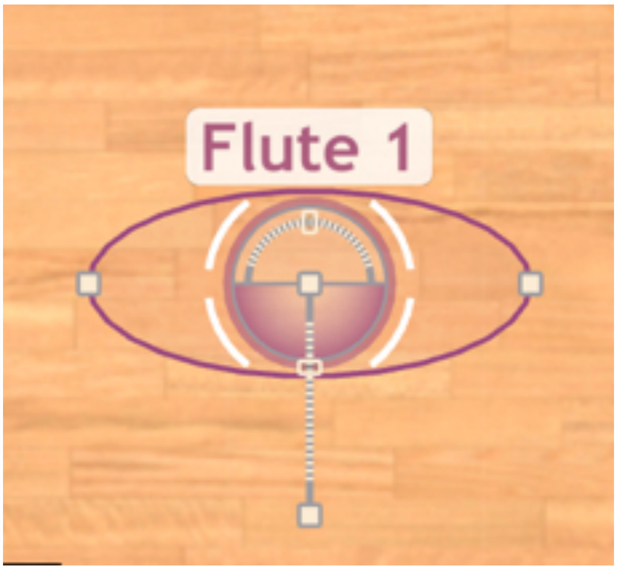
What you are doing here is using one instance of VE Pro not for hosting instruments, but instead as a reverb plugin. Channel routing is the same as with VST or AU instruments, so you should feel right at home.

2) The plugin, on the other hand, works like any other effect plugin. You insert it into any insert of track (or, better, in the insert of a send used solely for reverberation). It can be found inside the VSL folder for plugins:

You need to insert one plugin per instrument. Each plugin will be represented by one control icon inside the MIR Pro control app. This application will automatically launch when you insert the first MIR Pro plugin and serves as the backbone for the whole project. Its settings are saved within your sequencer, so you do not need to worry about it at all.

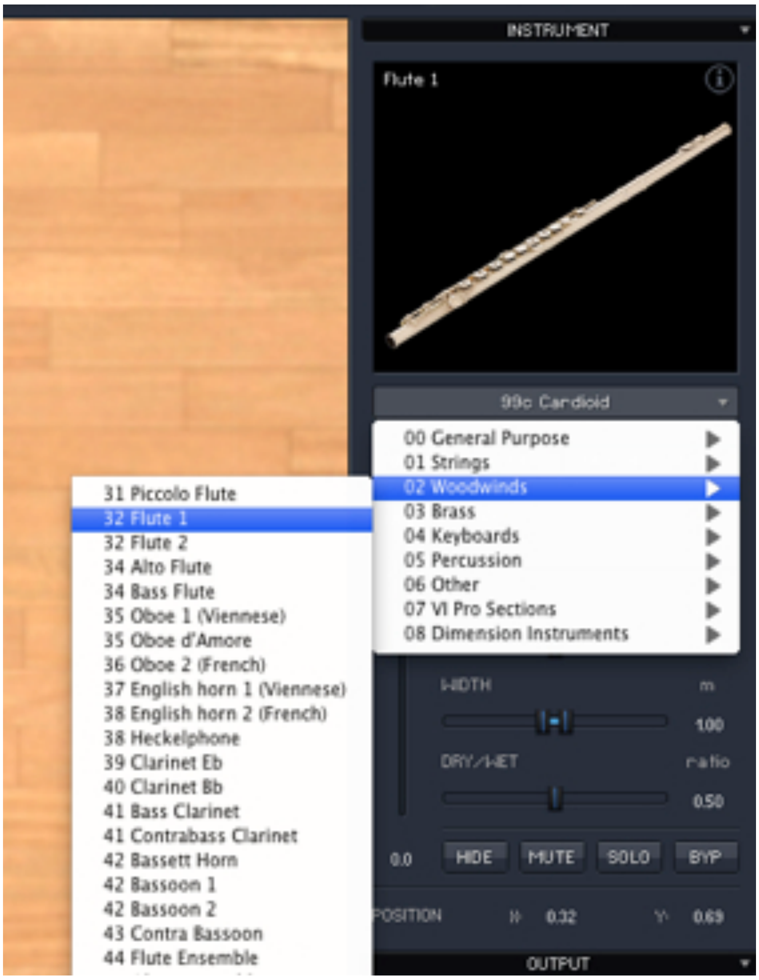
Here is how the plugin window and its corresponding control icon look like:





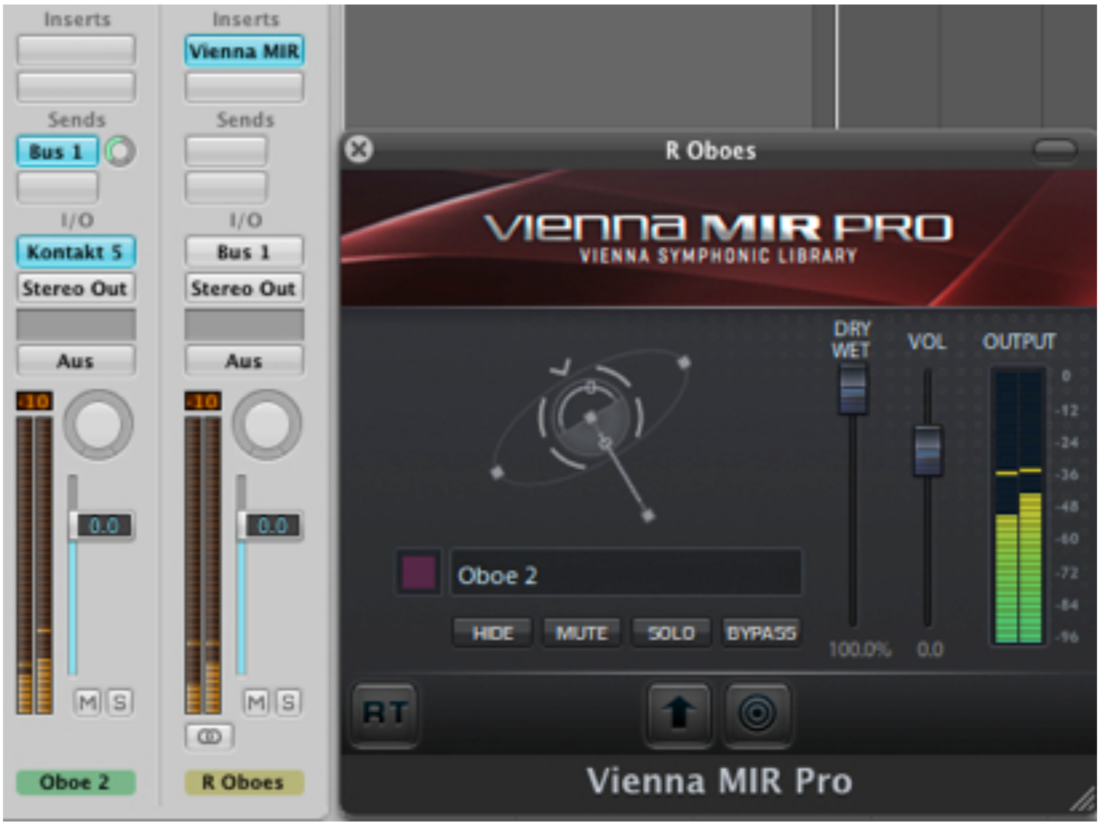
The control app pops up, which is identical in functionality and GUI to the version built into VE Pro. There is one difference, though: While inside VE Pro you can have multiple dedicated instances of MIR Pro, enabling you to have more than one stage loaded at the same time, this is not yet possible in the plugin version.

MIR works extremely well with VSL's own sample collections and in fact has special instrument profiles built in for all of them. After selecting a venue, click on an instrument icon to see its options in the sidebar. Here you can select an instrument profile.



The general interface and features are described in detail in the manual. For this tutorial, our main point is how to organise the reverb channels. If you use MIR inside VE Pro by inserting it into your actual template VIframes, everything works the same way as before. The only difference is that the signals will come back into Logic "wet" with no additional reverb needed (except for maybe one final reverb to glue everything together). If you use the Audio Input plugin to get the Audio into another dedicated instance of VE Pro, you should set the dry/wet slider inside MIR to 100% wet, because the dry part will already be inside Logic and all you need is the wet part coming from MIR.

If you use the plugin, treat it like any regular reverb: Use a send to get your instrument's audio to a reverb bus and insert the MIR plugin on this bus. Again just let MIR give back the wet part of the sound, like shown below:



Alternatively, you can also route the output of the tracks coming from your template to those reverb busses and adjust the dry/wet ratio inside MIR.

For percussion it often makes sense to have more of a dry sound. Most cinematic music relies on punchy drums, which should have less reverb than the rest of the orchestra. Just reduce the amount of "wet" in MIR.

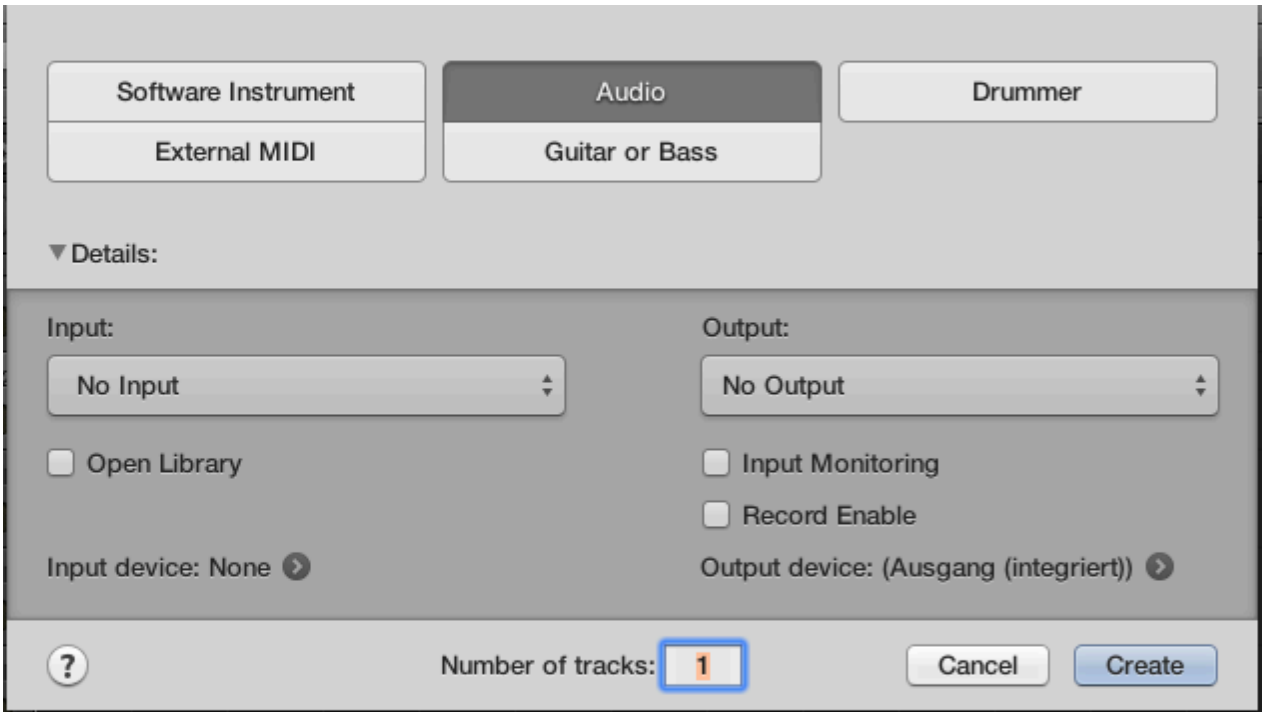
Sorting Tracks

If you have ever looked at an orchestral score, you will undoubtedly have noticed that there seems to be an inner force that organises which instrument is shown where. What this means in plain words is that there is a specific order of instruments. For creating written scores and because its just standard, you should also organise your template in this way.

That means that you should start with the Flute, go on with the rest of the woodwinds, then comes the brass and so on...

With one exception:
You (especially in Logic) may want to have one single audio track right at the top. Don‘t give it an audio input. Don't give it an output, either. You ask why?
Well, Logic always "arms" the currently selected track. That means it is ready for receiving MIDI information. But when you are just playing back, this is an unnecessary CPU strain.
So whenever not recording, you should select an audio track. But in MIDI projects like this you are likely to not have any audio tracks. That's why we create this Dummy Audio track.
Because no input it set, Logic will not use any CPU power to listen for things to record on it. And because there is no input, we need no output.

Here‘s a screenshot of the settings you should choose:



▼Details:

Input: No Input

Output: No Output

□ Open Library

□ Input Monitoring

□ Record Enable

Input device: None

Output device: (Ausgang (integriert))

Number of tracks: 1

Usability

There are a few things you can do to make up for the absence of track folders in Logic:

- Colour-code your tracks: Give every section of your template another track colour. If you use a slightly darker or lighter colour for the auxes, you can identify these easily, too. Unless you want to automate the auxes, there is however no need to have them in the arrange window. You cannot do anything against the VE tracks, so just hide them as described above.
- Use the track notes: It will be quite hard for you to remember which instrument is controlled by what. Do they use key velocity for volume? Or the modwheel? Which keyswitches do you have? All these things you can write in the track comments.
- Use project notes: I recommend using the project notes for a changelog. This way, at a glance you can see which "version" of the template a given song is using.
- This is especially useful if you changed patches in your template, i.e. when updating a sample library. The new patch might react slightly different to the old one. If this happens and you think that it may be caused by changes to your template, try to start VE Pro empty and load the song. Logic should then make VE Pro load everything needed (only works if the instances are coupled). If the samples are still where they used to be, you're lucky and everything should work as intended.

About relationsships: Coupled or Decloupled?

Many people are confused by the whole coupled/uncoupled thing. Actually, it is quite straightforward:
- A coupled instance is saved within your project file
- A decoupled instance is not saved inside the project file

There are two big practical differences this imposes:
1) Coupled instances can be loaded from within the Logic project file. All you need to do is to start VE Pro, but not load anything in it and then load the project in Logic. VE Pro will then load everything from the project file.
   This is useful when you want to access i.e. an old version of your template, which is used by that particular project. Your proper template may have many changes from the version that was used back then - but the old version still is in the project file.
   Uncoupled instances are not saved in the project, so they rely on you loading the corresponding instances (or metaframe/s) before opening the project. To be sure you should always have at least one coupled version, just to be safe. You should also always have an external version as a viframe or mframe, just to be even safer. Logic projects can get corrupted, as can VE Pro metaframes or viframes. But chances are slim that both get broken at the same time.
2) Saving a project with VE Pro coupled takes ages. The more instances you have and the bigger the project gets, the more ages it takes. Take this into account!

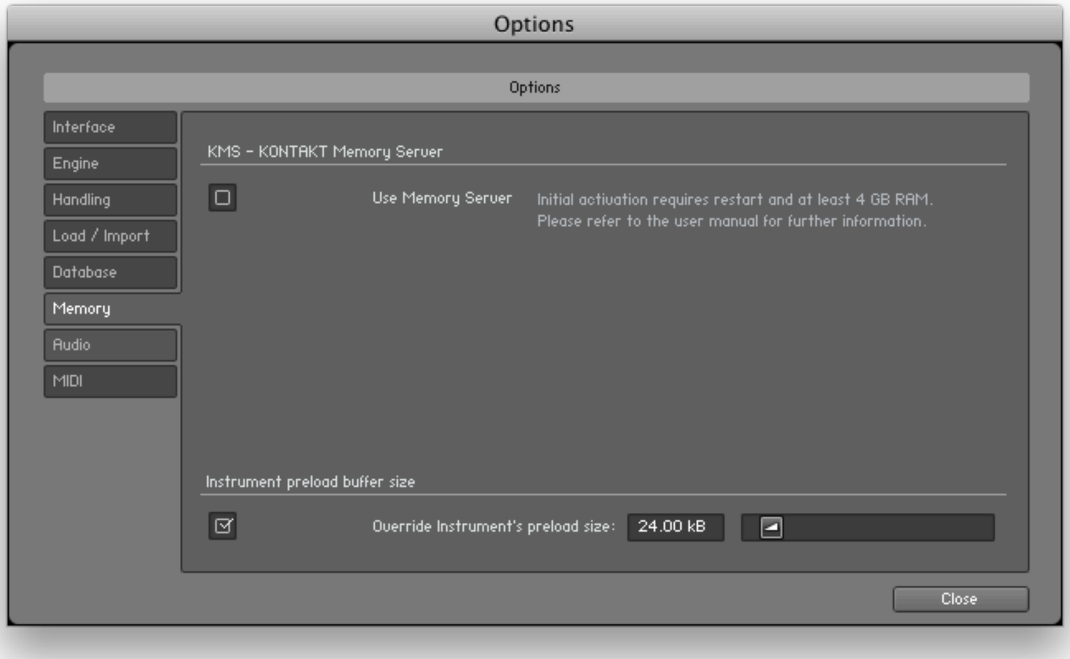
So what does that mean for daily use?
While it makes perfect sense to run the instances uncoupled for most of the time to speed up project saving time, you should after important edits save a coupled file under a different name.
Not only does this help when our metaframes get accidentally corrupted, but this way you can also access earlier versions of your template. This is important when you change instruments as you perfect your template. On new projects, a new string patch might

sound noticably better, but on old project you will want to have access to the original sound! After all, you most likely spent a considerable amount of time programming the line and mixing!

Load Optimization

VE Pro will make sure that the load is distributed well across all the CPU cores, but still there is some room for manual tweaking. Sometimes when you run into CPU spiking issues, fiddling with these settings can also help.

Most sampled instruments only load a bit of the sample material into the system memory and stream the rest from disk. The amount of what is loaded into memory is determined by the preload buffer setting. Lowering this setting means that your loaded samples consume less memory at the expense of your sampler needing to read much more directly from disk. Toy around with this setting and see what works for you. If you get crackles, pops, lost notes and other issues, raise it a bit.
In Kontakt 5, you will find the preload buffer setting in the options inside the Memory tab. By default it is set to 60kb; on regular 7.200 rmp drives you should be able to pull it back to 40 - 45 or sometimes even lower on pieces not requiring every single instrument to play at the same time.



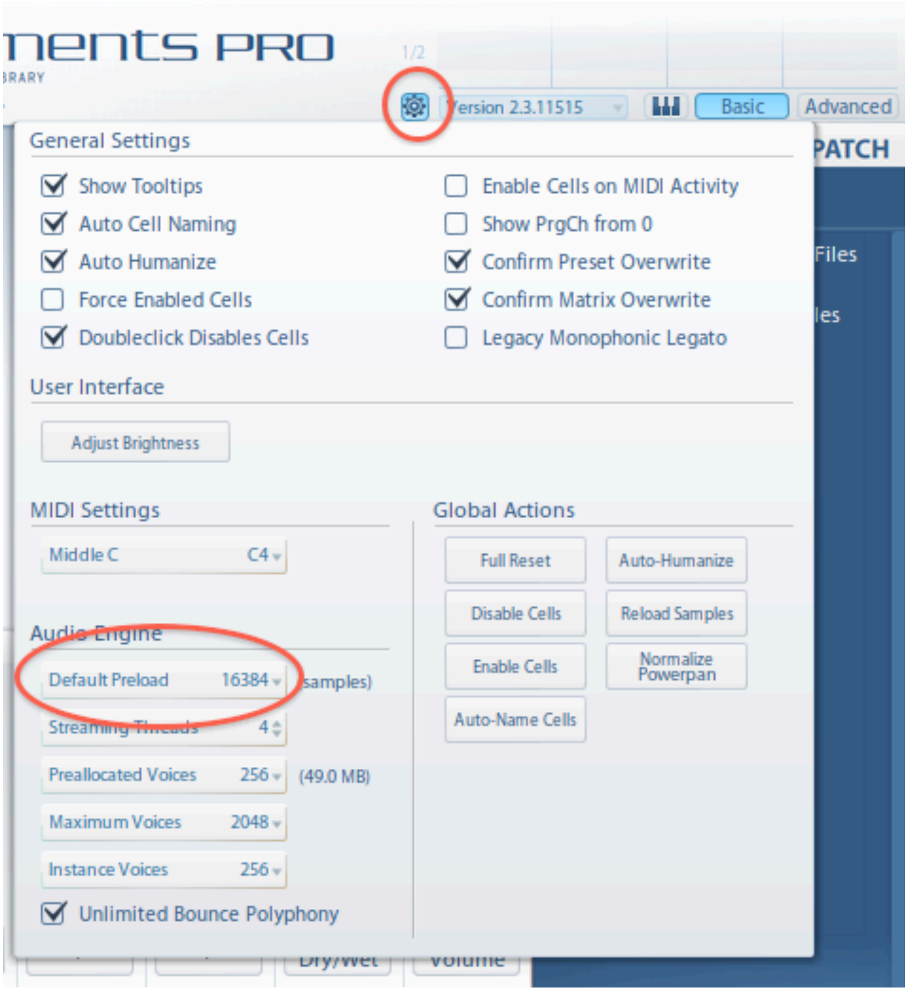
KMS - KONTAKT Memory Server

☐ Use Memory Server — Initial activation requires restart and at least 4 GB RAM. Please refer to the user manual for further information.

Instrument preload buffer size

☑ Override Instrument's preload size: 24.00 kB

Vienna Instruments (Pro) has a similar setting in its options, accessible by clicking the little gearwheel icon:



General Settings

☑ Show Tooltips
☑ Auto Cell Naming
☑ Auto Humanize
□ Force Enabled Cells
☑ Doubleclick Disables Cells
□ Enable Cells on MIDI Activity
□ Show PrgCh from 0
☑ Confirm Preset Overwrite
☑ Confirm Matrix Overwrite
□ Legacy Monophonic Legato

User Interface

MIDI Settings

Middle C C4

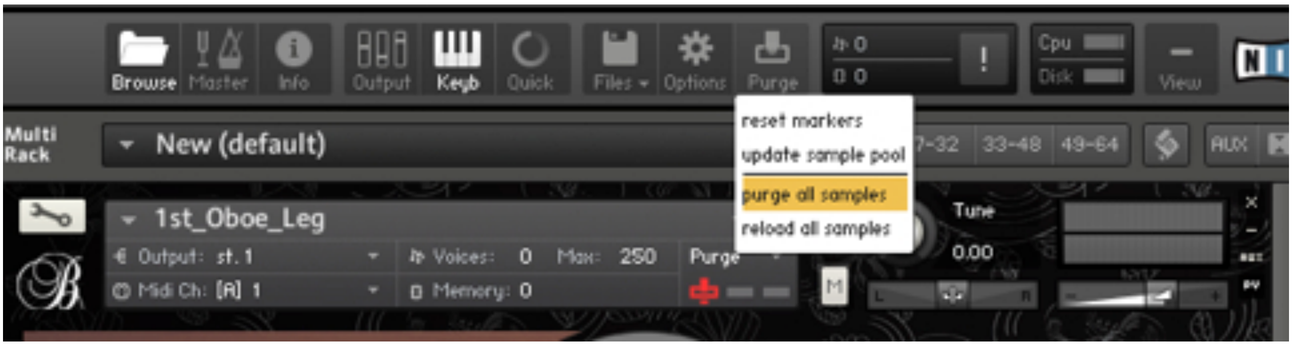
Audio Engine

Default Preload 16384 (samples)
Streaming Threads 4
Preallocated Voices 256 (49.0 MB)
Maximum Voices 2048
Instance Voices 256
☑ Unlimited Bounce Polyphony

Global Actions

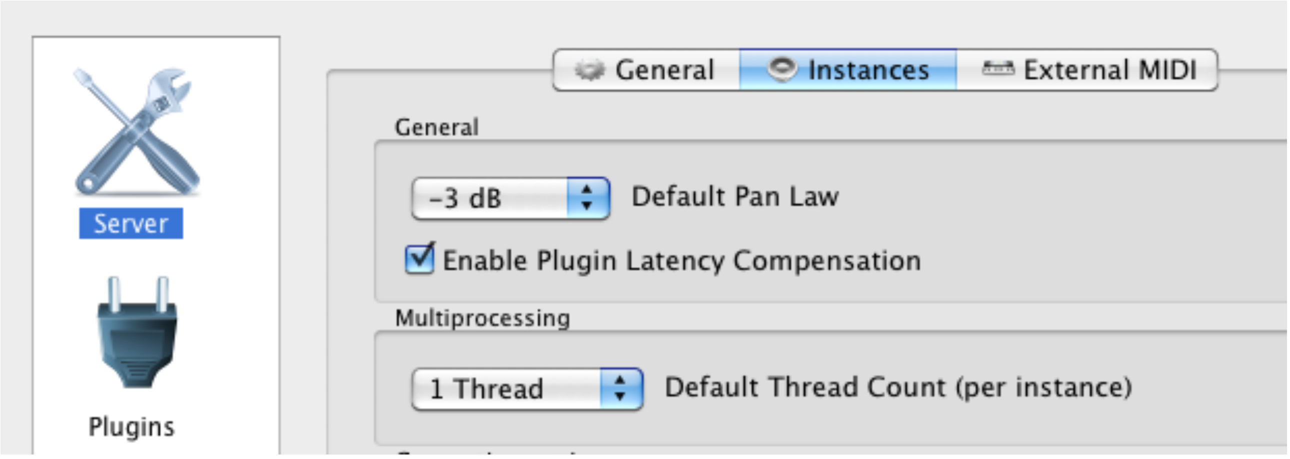
Again experiment with the settings to see what works for you.

Many sample libraries allow you to unload unused articulation directly within the plugin GUI. If that does not work, use the "Purge all“ feature if such a feature is present in the sampler you are using. Kontakt, for example, has this ability: Click "Purge all samples“ in the purge menu and Kontakt will unload all samples. Samples will then be reloaded while you use the instruments on a "need to“ basis.



Most hosting applications and samplers today have multicore support. That means that they will intelligently balance their workload over all available cores of your machine. Some also allow you to specify a number of cores for the application to use. While this in theory is all nice and well, the cold harsh reality quickly shows that these multicore balancing

features tend to get in each others way. One practical example: If you host Kontakt 5 within VE Pro 5, by default, both applications have multicore support enabled. It is my experience that CPU consumption will drop noticeably when you either switch off multiprocessing in Kontakt, or alternatively in VE Pro.
You can find the relevant settings in VE Pro's and Kontakt's Options:



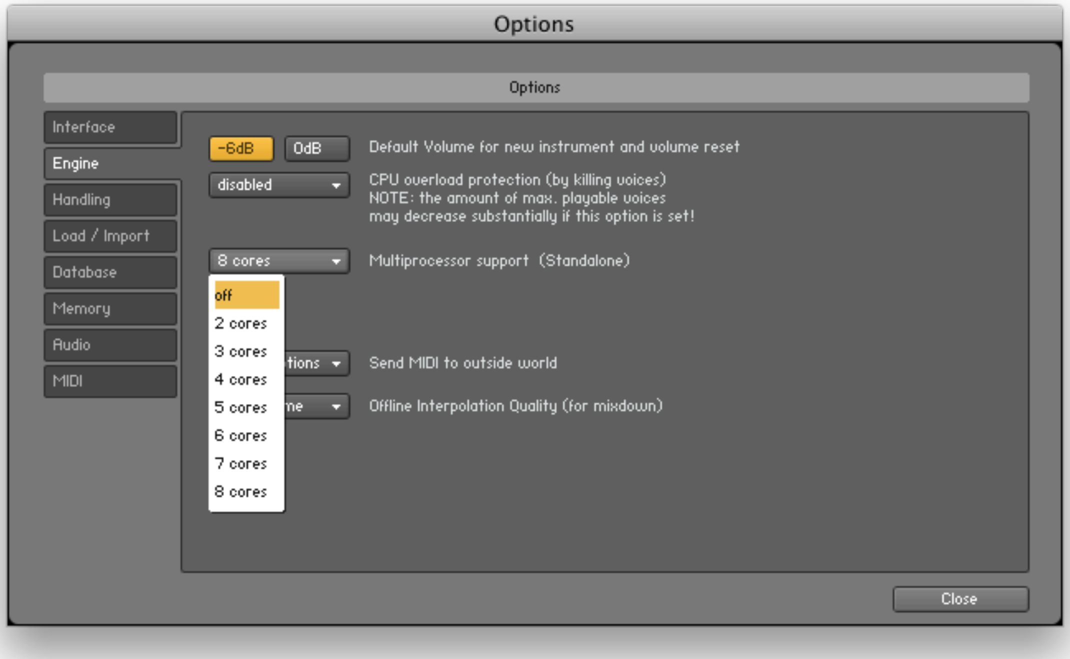
General

-3 dB Default Pan Law

☑ Enable Plugin Latency Compensation

Multiprocessing

1 Thread Default Thread Count (per instance)



-6dB 0dB Default Volume for new instrument and volume reset

disabled CPU overload protection (by killing voices)
NOTE: the amount of max. playable voices
may decrease substantially if this option is set!

8 cores Multiprocessor support (Standalone)

tions Send MIDI to outside world

ne Offline Interpolation Quality (for mixdown)

Especially on OSX, if your VE Pro instances consume 10-20% CPU per instance when doing absolutely nothing, this will most likely solve the issue! I have so far not seen any performance impact by switching off multicore support - on the contrary. If you are running parts of your template in the DAW, make sure to experiment a lot with different multicore settings: Having both VE Pro (or any other host) as well as your sequencer running, both of them probably offering support for threads, it might even make sense to deactivate multiprocessing support completely in the sequencer. I have made very good experiences doing that with Cubase, especially on the Mac.

A slightly enlarged version of this section is available at http://ow.ly/fWLWW

Troubleshooting

In my work with VE Pro and from emails and calls from readers of this tutorial, I have observed a few things that tend to go wrong sometimes. I will update this section when I hear of new issues.

Crackles and pops with only very few audio channels

Make sure you are using an ethernet connection. Should your machines have WiFi, switch it off to make sure! If you use the most recent version of VE Pro, the software should not even list WiFi-based connections, so please update.

VE Pro doesn't find the license, but the dongle works

Did you just perform a VE Pro Update? Download and install the newest eLicenser drivers. Sometimes new versions of VE Pro require new drivers. Also installation of iLok-protected software can require a driver reinstall.

The plugin finds no instances on other machines

I only had this once but heard of it occasionally. It seems to be a problem related to OSX mixing up ethernet ports. If you have a MacPro, try using another ethernet port (it has two). If you encounter this issue, please contact me, as I want to try to reproduce it.

VE Pro instances do not load from sequencer project after the first start

This is sometimes caused by an IP address mixup. Use manual IP addresses in your network, you can set this up in System Preferences : Network.

VE Pro refuses to connect to instances

Make sure that you have the same version of VE Pro installed on all machines. Make also sure your VSTs and AUs are up-to-date.

When I add an Aux in Logic, my instruments are silent

This seems to be a new behaviour in Logic Pro 9.1.7. When adding an Aux, the SI slot defaults to Channel 1 instead of Omni. Simply set the MIDI channel back to Omni.

Conclusion

I hope this tutorial was valuable for setting up your orchestral template.
Now with 2.0 ready for VE Pro 5 we have reached another milestone! And this means it is your turn again with suggestions. So please help me improve the tutorial by sending your comments and questions to me! Just visit www.orchestraltemplates.com/contact and send me a message. I really need your help to improve this!

As mentioned before: I would very much like to "port" this tutorial to other sequencers. So if you want to help by providing screenshots or tips on how to deal with your favourite DAW, please contact me.

I invest a lot of time into writing this tutorial, testing with different setups and constantly expanding the content. So if you find this tutorial useful and if it helps you get your template going, please consider making a donation at www.orchestraltemplates.com. I solemnly swear I will never, ever use your precious money to buy booze, but only to help with keeping track of new developments and making this tutorial better! Many people suggested releasing this tutorial commercially, but I decided against it. See it as "honor-ware". If you like it, please make a donation.

The Big Speech

This is the place where it is customary to thank all kinds of people. So here we go:

- First and foremost, a big thanks to all my readers. Your comments tremendously helped shaping this tutorial and I am sure this will continue.
- Kudos to the folks at the Vienna Symphonic Library for putting this up on their website.
- I have had the pleasure to aid an ever growing number of people in setting up their rig with VE Pro. I learned a lot in the process, which benefits all users by being incorporated into new versions of the tutorial.
- Maybe I should also thank myself for getting around to actually write this :) But maybe I shouldn't. Too much work. No, I definitely will not thank myself.

Copyright

novatlansound
music for media