IR Convolution Reverb
for HW4 and HW5 Basic

Gernot Wurst

PROSPECTUM virtual organs
www.prospectum.com

The Virtual Pipe Organ Institute
www.vpoinstitute.org

Version 1.02

August 3, 2020

Contents

1 About Impulse Response (IR) convolution reverb

1.1 What is IR convolution reverb?

A well-known organ builder once said: "The most important stop of any organ is the acoustics of its church". This statement is underpinned by the facts that an organ is usually specifically designed with its future environment in mind and that its final voicing is performed on site in order to optimize the overall sound which is a result of the sound produced by the organ and the acoustical properties of its environment.

For a virtual organ, it follows that, in order to create a convincing representation of a real organ, faithfully reproducing the acoustics of its environment is an important task.
Taking a closer look at reverb, the following observations can be made:

1. It appears as a response to a sound (the original sound is referred to as *excitation*)

2. It delays sounds (reverb occurs after the excitation)

3. It changes in amplitude (reverb fades away)

4. As long as the reverberating space (referred to as *environment*) remains unchanged, its reverb remains the same, too (reverb creation does not change over time)

In technical terms, these properties are referred to as *causal* (1), *linear* (2,3) and *time invariant* (4). Since sound is a mechanical phenomenon (moving air molecules) and since in classical mechanics causality is taken for granted, all we need to note regarding this property is that reverb is a response to an excitation. The other properties allow us to classify reverb as a so-called *Linear Time Invariant System* or in short LTI system. That sounds complicated but, as you already know, it simply means that reverb is a response to an excitation such that amplitude changes and delays occur and that these amplitude changes and delays are constant characteristic acoustical properties of a specific environment.

Now that we have understood what reverb is, the next question is: How can we get a description of the acoustical properties of the environment of an organ that can be used for recreating its reverb? You see, whenever such an environment is excited by the sound of a single pipe, the resulting reverb is the response to exactly this unique pipe sound. Since pipe sound and reverb are superimposed, you can not separate the two (it is theoretically possible but VERY complicated). And even if you could isolate the reverb perfectly, it is still useless for other pipes or different lengths of the same pipe sound because the reverb is the response to exactly this pipe sound.

So, what we really need is a recording of the response of the environment to all possible excitation signals! This would make it possible to calculate the reverb brought about by all the sounds the organ can produce. Fortunately, without going into details here, all excitation signals can be described as sequences of a very large number of very short impulses. The only differences between these impulses are their amplitudes and the points of time at which they occur. This helps us a lot because (in case of a Linear Time Invariant System) it means that if we had the response to such an elementary impulse, we could recreate the reverb for any excitation signal by adding up a very large number of scaled and time shifted versions of this elementary response according to the sequence of impulses that make up the excitation signal. You might want to reread the last sentence a few times - as soon as you have fully understood its meaning, you will most likely agree that what it describes is amazing! One last new term needs to be introduced: This process of creating and adding those scaled and shifted copies of our elementary impulse response is called *convolution*.

Now we can finally define what impulse response convolution reverb, generally referred to as IR convolution reverb (or IR reverb, or convolution reverb), is: We take the response of an LTI system (such as a church) to an impulse (a very short, intense elementary excitation), referred to as its IR, and apply it to arbitrary excitations (e.g. pipe sounds) by means of convolution.

In practice, this means that we need two components for IR reverb: An IR and software that performs the convolution, referred to as convolution engine. Obtaining high quality IRs is a topic that is complex enough to fill an entire book and will thus not be discussed here. Instead, the focus will be on the convolution engine: Hauptwerk 5 Advanced comes with a built-in convolution engine. Hauptwerk 4 and Hauptwerk 5 Basic do not. **Users of HW4 and HW5 Basic therefore need to connect Hauptwerk to additional software that provides this functionality. Helping you with this is the main purpose of this document.**

1.2 How does IR reverb compare to other reverberation methods?

With regard to virtual organs, there are three reverberation methods: (1) Reverb included in recordings of individual pipe sounds (so-called *wet recordings*), (2) Artificial algorithmic reverb, and (3) IR convolution reverb. Each method has its advantages and disadvantages. However, in most cases (meaning for most organs) there are convincing arguments for considering IR convolution reverb (when done right!) the gold standard. Let's have a closer look at these three methods:

1.2.1 Wet recording

Wet recording is the oldest, most straightforward and therefore also most common way of including reverb in virtual organs: Pipe sounds are recorded and played back together with their reverb. The primary knowledge required for creating wet sample sets is how to choose and place the microphones so as to achieve a good balance between direct sound and reverb.

However, three problems are associated with this approach: (1) Only responses to specific excitation signals (pipe sounds, action noises etc.) are recorded together with the excitation signals, (2) artifacts are created by the violation of causality when playing these virtual organs and (3) the signal-to-noise ratio deteriorates as the reverb fades away while the background noise remains constant. Several techniques have been developed in order to address these problems. Unfortunately, they are usually workarounds and/or themselves sources of further problems.

For instance, multiple reverb samples per pipe can be used to reduce the artifacts that can be heard whenever the lengths of a recorded tone and its played back version are differing significantly. Usually, recordings of short, medium and long tones are used in order to approximate the natural responses to tones of all lengths by a so-called next neighbor interpolation, which can lead to a significant increase in memory requirements. While indeed reducing the artifacts, the switching from one release sample to another (when playing short, medium and long notes) can be heard and can still produce unnatural results.

Another big problem is that modeled tremulants, meaning that the tremulant effect is applied to the pipe sound in real time, sound unconvincing whenever the tremulant effect is applied to the wet sample. In reality, reverb is created in response to the modulated pipe sound, which is a much more complex process and produces a much more complex sound. Unfortunately, the straightforward solution of simply recording the original tremulant and using these samples directly is not a good solution either because it again introduces a variety of new problems (not to be discussed here).

A third big problem is that noise reduction is required to prevent the noise of multiple superimposed release samples from adding up to annoying levels. Usually, these denoising methods are based on statistics, which means that they will overlook some of the background noise which can lead to artifacts and also remove some of the desirable parts of the pipe sound that contribute to the character and texture of that sound. PROSPECTUM solved this problem by developing a knowledge-supported and context sensitive algorithm referred to as multi layer denoising that was used for the Anloo, Zöblitz and Weissenau sample sets as well as a few sample sets of other creators. The main goal was to preserve the details and texture of the pipe sounds.

In conclusion, wet recording has substantial drawbacks and should thus be considered a legacy technology to be used whenever it is not possible to record dry samples and IRs.

Overview of pros and cons:

Producer	User
+ Ideal for commercial mass production	+ Large number of virtual organs available
+ Only elementary knowledge required	- Potentially diminished sound quality (denoising)
- Multiple reverb samples per pipe required	- Diminished quality of modulated sounds
- Denoising is crucial	- Large installation packages
	- High polyphony requirements
	- High memory requirements

1.2.2 Artificial reverb

Here, artificial or algorithmic reverb is defined as any method of applying algorithmically calculated reverb to pipe samples that is not based on a description of the real acoustics of the location of an organ (meaning it does not use an IR). Using *dry* (unreverberated) samples and an additional reverberation step, the typical problems of wet sample sets can be avoided or reduced. Algorithmic reverb can sound great and be easy to use, especially in the form of (usually expensive) hardware reverberation devices. However, since it can be expected to differ from the real reverb the original organ was created for, it is here not considered a viable alternative.

Overview of pros and cons:

Producer	User
+ No need to create IRs	+ Avoids or reduces typical problems of 'wet' organs
+ No need for multiple reverb samples per pipe	+ Substantial reduction in system requirements
+ Denoising is much less crucial	+ High flexibility regarding reverberation
+ Reverberation becomes the user's responsibility	- Algorithmic reverb is not the original reverb
- Not widely used with virtual organs	- High quality algorithmic reverb can be expensive

1.2.3 IR convolution reverb

As already described in 1.1, IR convolution reverb uses a universal description of the acoustical properties of the original environment of an organ, which is the IR, and recreates the original response of this environment to arbitrary sounds. Since this approach recreates the natural process of reverb creation in such away that its causality is respected, it solves those problems of the wet recording approach that are caused by violations of causality. In addition, much less denoising is required, and system requirements for the use of the virtual organ (memory, polyphony) are substantially reduced. In short: Spend less money on hardware and get a better sound!

Although this approach has the potential to improve virtual organs a lot, it is all but new. In fact, PROSPECTUM has pointed out the problems of the wet recording approach and championed the use of IR reverb for virtual organs as early as (roughly) 2005. When the Anloo organ was recorded in 2008, it was planned as a dry+IR sample set from the beginning. The assumption was: The advantages of this technology are so obvious that it should be safe to assume that Hauptwerk will support it soon. And indeed, there have been plans to include IR reverb in an earlier Hauptwerk version.

So, why have we kept on creating and selling (or buying) virtual organs made using the wet recording technology that could have been justifiably called a suboptimal legacy technology more than a decade ago? Apart from the fact that wet sample sets have been a great commercial success (arguably for lack of a better alternative), there are considerable inconveniences for producers: The process of making high quality dry recordings and creating the corresponding high quality IRs is much more complicated and requires much more knowledge and commitment than conventional wet recording. While, as demonstrated by the demo IRs that come with Hauptwerk 5, it is easy to add some nice sounding but unspecific IR reverb to any virtual organ, convincingly recreating the real reverb of the environment of a real organ is challenging in most cases.

Difficulties begin with the fact that in reality each individual sound source is a three dimensional object and usually resides in a very complex environment (just take a look inside an organ!). So, ideally there

should be an IR for each individual sound source. Next, each IR is a superposition of a large number of partial IRs that represent individual components of the environment of the sound source and the interaction between these components (by the way: This is how bats navigate in the dark: They can extract spatial information from IR-like environment responses). Next, the creation of the ideal impulse that is used to excite the environment in the right shape and at the right location is all but trivial. And finally, when recording IRs, microphone placement is very important, and very complex post processing is required in order to recreate a convincing stereo image for the virtual organ. Keep in mind that the number of IRs that can be used in a real time convolution engine is limited. So you need to reduce the sound of hundreds or thousands of individual IRs to a small number of IRs without losing the complex spatiality of the sound.

To give you an idea of the required effort: For the Ansbach organ (not yet released), IR recording alone took three weeks, and five individual programs had to be written to faithfully recreate the spatiality of the sound of this magnificent organ.

For now, the takeaway messages of this section are that the dry+IR approach has the potential to make virtual organs sound better than conventional wet recording and that for you as a user it is worth giving it a proper try.

Overview of pros and cons:

Producer	User
+ Eliminates the well-known problems of wet recording	+ Better, more convincing sound (if done right)
+ No need for multiple reverb samples per pipe	+ Most likely a future-proof technology
+ Denoising is much less crucial	+ Reduces some system requirements
+ Most likely a future-proof technology	
- High effort required for high quality reverberation	

2 How to make it work: IR convolution reverb for Hauptwerk 4 and 5 Basic

IR convolution reverb is a standard feature of virtual organ platforms such as GrandOrgue (free) or Great Organ. In the Hauptwerk world however, currently only Hauptwerk 5 Advanced comes with a built-in convolution engine. Hauptwerk 4 Advanced users can use the Hauptwerk VST plugin, which is again an Advanced feature, in order to connect Hauptwerk to a so-called VST host application (explanation below). Hauptwerk 4/5 Basic users can add IR convolution reverb using the loopback method (explained below).

Please note that this guide is not a replacement for the user manuals of Hauptwerk, third party software or your audio hardware. Please also refer to these documents as needed.

2.1 Hauptwerk 4 Advanced

Hauptwerk 4 Advanced comes with a *VST plugin*. VST stands for *Virtual Studio Technology* which is a plugin standard in *Digital Audio Workstations*. VST was developed by the well-known Steinberg company. It became popular in the late 90s and is probably the most important standard in its area.

Usually, virtual instruments are released as plugins (called *VST instruments*). Such virtual instruments can be easily and conveniently integrated into any compatible application, referred to as a *VST host*.

At least one earlier Hauptwerk version was available as a standard VST instrument but nowadays (versions 4 and 5) Hauptwerk is only available as a standalone application. The VST instrument version has been replaced by a very simple plugin that does no longer have any Hauptwerk functionality itself but is just an interface that allows MIDI and audio data transfer between Hauptwerk and a VST host. This is most likely achieved using another Steinberg technology called *VST System Link*. In Hauptwerk it is referred to as *Hauptwerk VST Link*.

Using Hauptwerk VST Link and the VST plugin, Hauptwerk 4 Advanced users can send the audio output of Hauptwerk to a VST host and process it with an IR convolution reverb plugin there. The result is then sent from the VST host to your hardware audio outputs. This process is visualized in fig. 1.
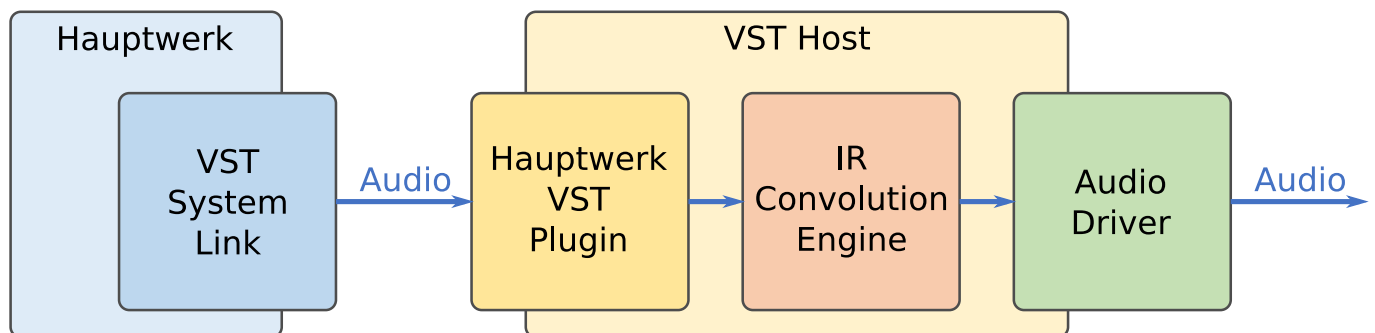

Figure 1: Audio signal flow from Hauptwerk through a VST host application (based on `https://sites.google.com/site/meltonesoundssa/a-vtpo-blog/july2019`)

Again, here is what you will need:

1. Hauptwerk 4 Advanced

2. The Hauptwerk VST plugin

3. A VST host application

4. A plugin that can calculate IR convolution reverb

In the following it is assumed that you are using Hauptwerk 4 Advanced (and thus also have the plugin). There is a large number of VST host applications available, like the well-known DAWs Cubase, Reaper, Ableton Live etc.. Most of them are too complex and too expensive for the sole purpose of adding reverb to a Hauptwerk organ. Fortunately, a selection of simple, free VST hosts is available:

- Cantabile `http://www.cantabilesoftware.com`

- NanoHost `http://www.tone2.com`

- SAVIHost `https://www.hermannseib.com/savihost.htm`

- Many others, see e.g. `https://bedroomproducersblog.com/2011/05/16/bpb-freeware-studio-best-free-`

There is also a growing number of free IR convolution plugins. Some are mentioned at `https://sites.google.com/site/meltonesoundssa/a-vtpo-blog/july2019`:

- Convology XT : `https://impulserecord.com/project/convology-xt-plugin/`

- HybridReverb2 and Mverb v.1.0 : `http://www.vstplanet.com/Effects/Effects5.htm`

- SIR1 : `https://www.siraudiotools.com/sir1.php`

- Reverberate LE : `https://www.liquidsonics.com/software/reverberate-le/`

- Halls Of Fame 3 Free : `https://www.bestservice.com/halls_of_fame_3_free.html`

- Freeverb3 Impulser2 and Hibiki : `http://freeverb3vst.osdn.jp/`

The VST hosts and plugins mentioned above are examples. This guide does not aim to be exhaustive. Your options my vary depending on factors such as your operating system, your budget etc..

The following examplary setup adds stereo reverb to Anloo 2 using Cantabile Lite as the VST host and Convology XT as the IR convolution plugin on a Windows 7 Professional 64bit system. Both IRs that come with Anloo 2 are used serially. Since VST is a well-established standard, the procedure should be similar for other systems. Here is what you have to do:

1. Download and install Cantabile (available at `http://www.cantabilesoftware.com`)

2. Download and install Convology XT (available at `https://impulserecord.com/convology-xt/`)

3. Make sure Cantabile knows where the Hauptwerk VST plugin and Convology XT are by adding their folders to the *VST Plugin Folders* list.
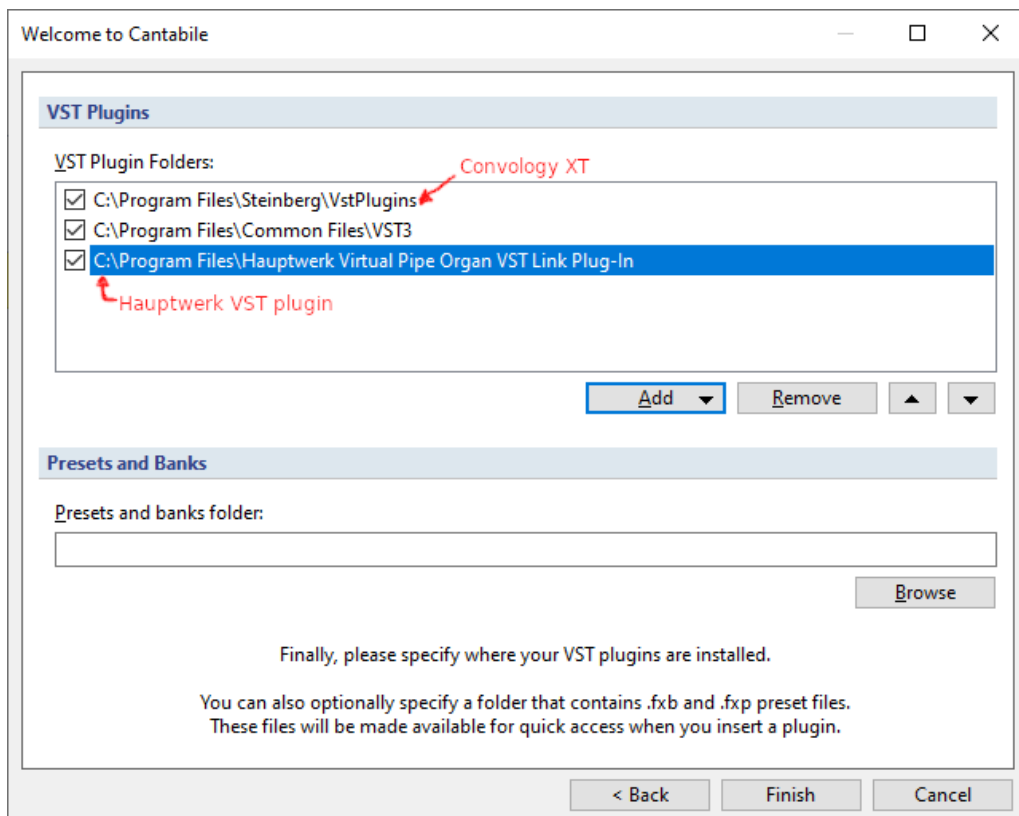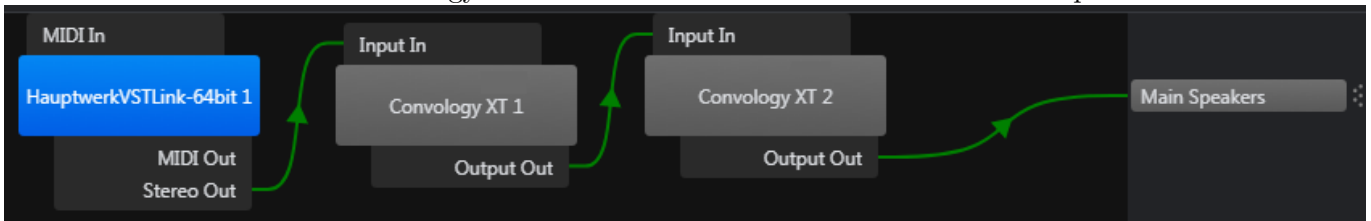
4. Configure Hauptwerk to send its audio output to the Hauptwerk VST plugin
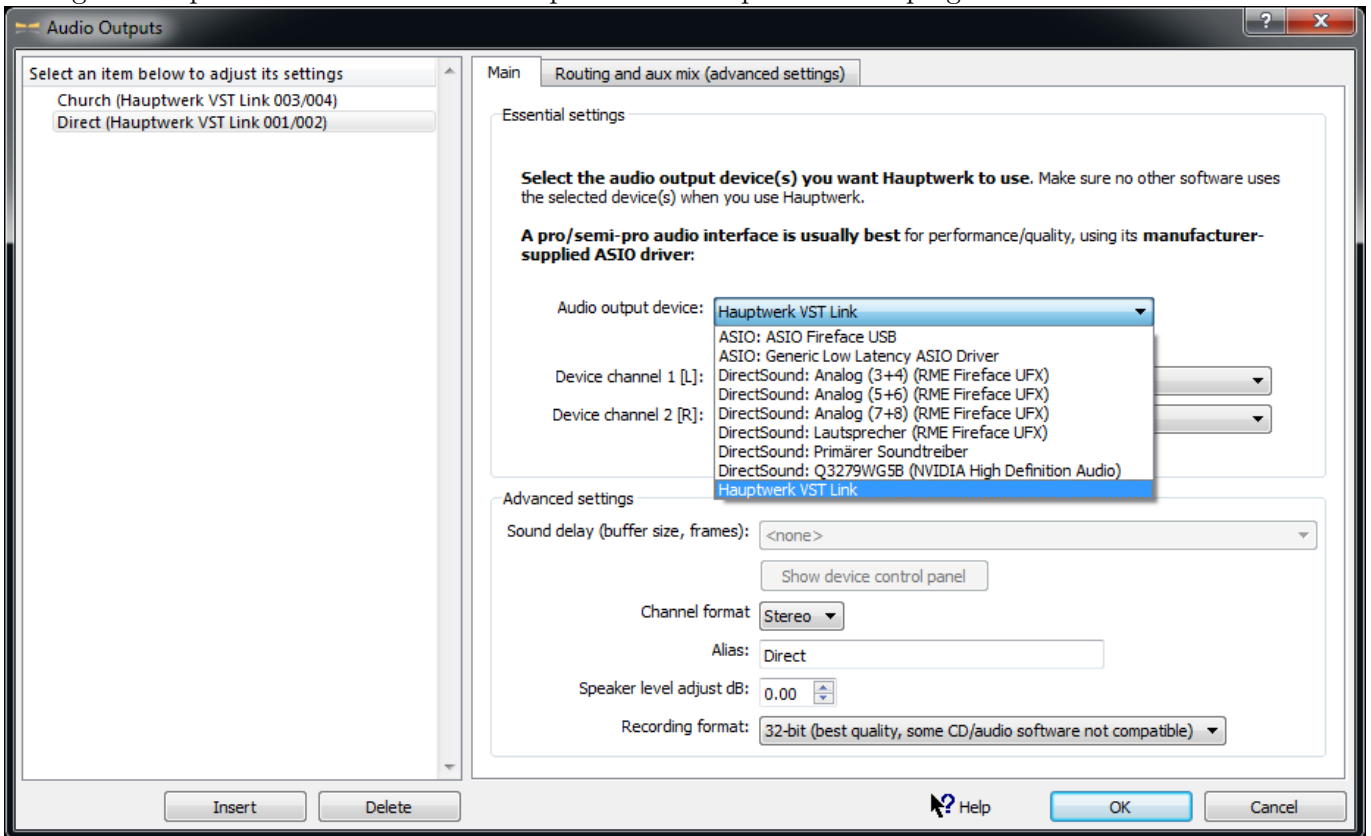
5. In Cantabile, load the Hauptwerk VST plugin

6. Also load two instances of Convology CT and connect them in series and to the output

7. Configure both instances of Convology XT: Load Anloo IR 1 into the first instance and Anloo IR 2 into the second instance. The only other change made here is that the wet signal (meaning the reveb) is attenuated by -18db in the first instance and by -24db in the second instance. Feel free to adjust these settings to your liking. You can ignore the other settings.

2.2 Hauptwerk 4/5 Basic

Hauptwerk 4 Basic and 5 Basic do not come with the VST plugin that connects Hauptwerk to the VST host. However, assuming that your organ setup includes a professional multi channel audio interface (such as the interfaces made by RME Audio https://www.rme-audio.de), you should be able to send the output of Hauptwerk back into the VST host (see above) of your choice by means of a mechanism called *loopback*. If your audio interface does not support *internal loopback*, you can still implement *external loopback* by connecting the respective hardware audio outputs of your interface to adequate inputs. Again, please refer to the user manual of your audio interface for details.

For obvious reasons, make sure you do not create a feedback loop (DO NOT send any output back into your system to a part of your signal flow where it has already been)!

In the following, it is assumed that you are familiar with the terms introduced in 2.1 and that your audio interface supports internal loopback. The creation of a signal flow that allows for a VST host to be used together with Hauptwerk is demonstrated using an RME audio interface but, apart from the differences with regard to how to use the loopback function of your specific audio interface, should work the same way with your audio interface.

What we are trying to achieve here is create a substitute for the missing Hauptwerk VST plugin. All the other steps are identical to those described in 2.1.

1. Download and install Cantabile (see 2.1)

2. Download and install Convology XT (see 2.1)

3. Make sure Cantabile knows where Convology XT can be found by specifying its location in the *VST Plugin Folders* list
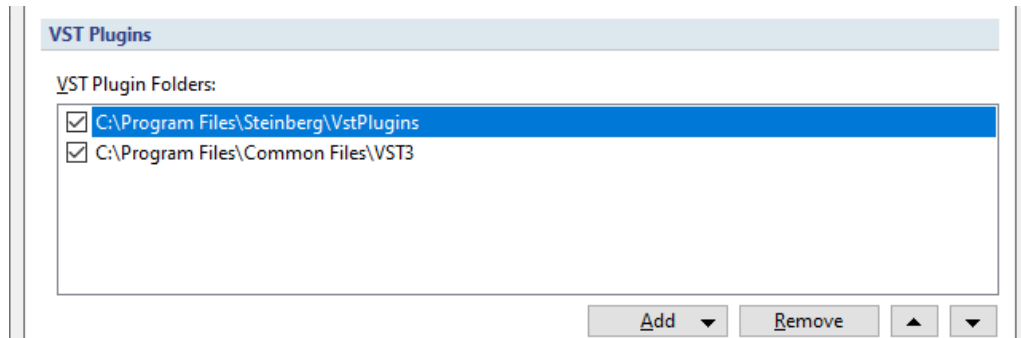

4. Configure Hauptwerk to send its audio output to an output of your audio interface
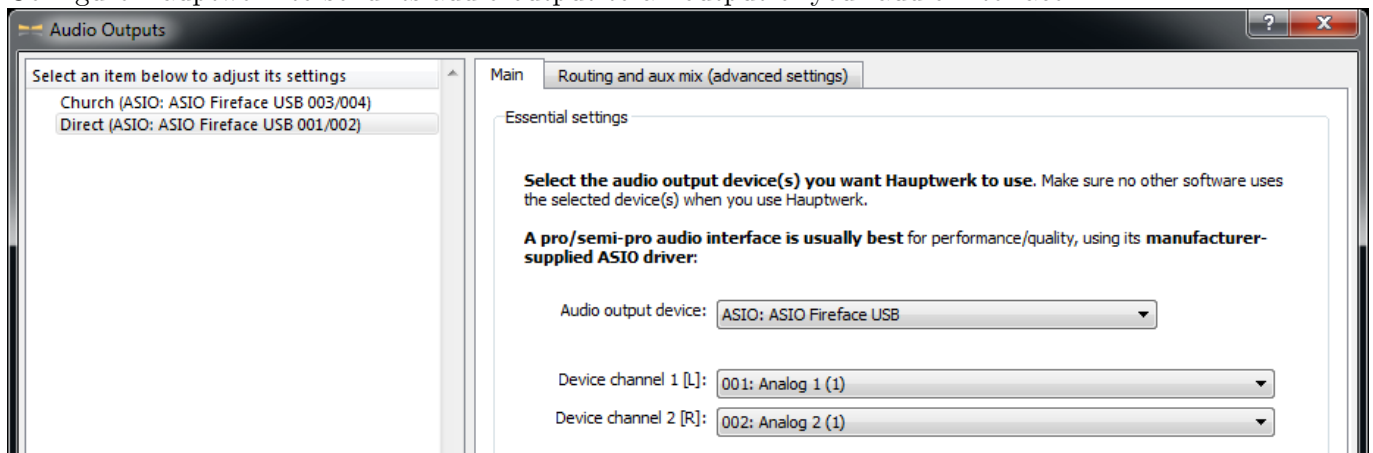
5. Configure your audio interface so that the Hauptwerk audio outputs are looped back to audio interface inputs. Here, Hauptwerk outputs 1 and 2 (software playback AN 1/2) are sent to the unused ADAT outputs 1 and 2 of the interface, for which the loopback function is activated so that their signal is copied to ADAT inputs 1 and 2. So, in short, the signal flow is HW outs AN 1/2 → interface outs ADAT 1/2 → interface ins ADAT 1/2 (this is the loopback) → Cantabile → Cantabile outs AN 3/4 → interface outs 3/4.

6. The last step to complete the signal flow configuration is to configure Cantabile to use ADAT in 1/2 as inputs and AN 3/4 as outputs:
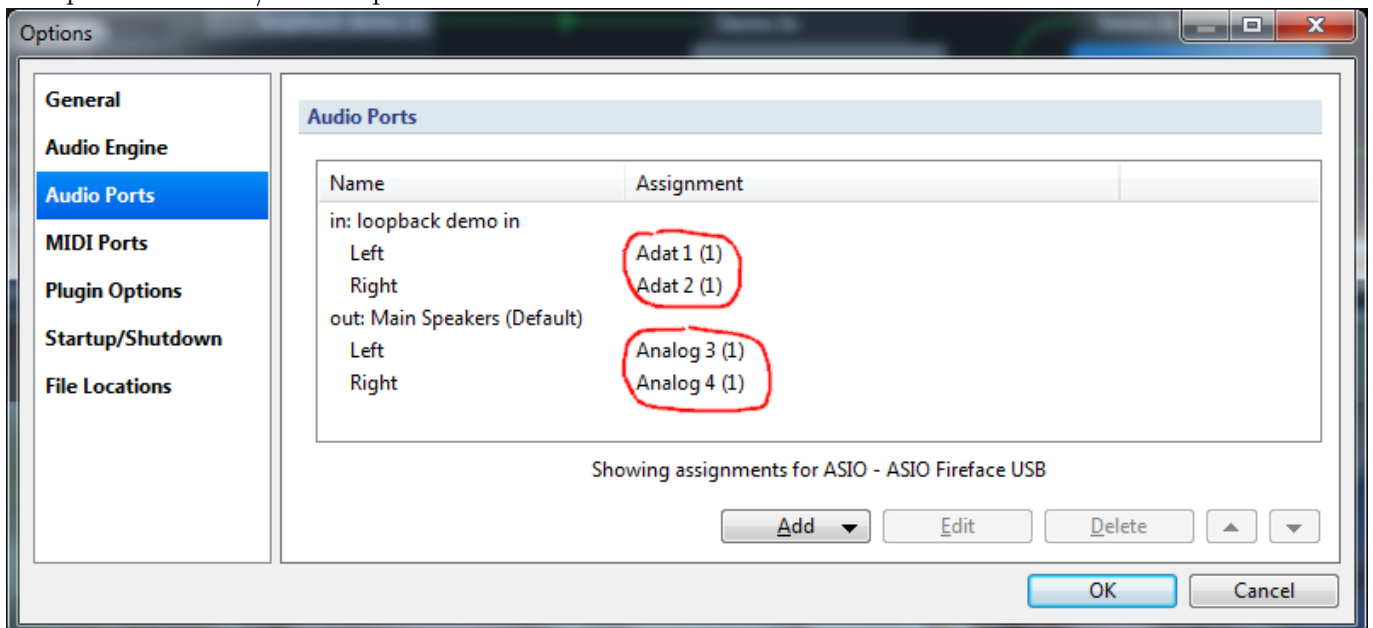
7. Load two instances of Convology XT like in 2.1. Now however, connect inputs and outputs as shown below:

8. Configure both instances of Convology XT for Anloo 2 (see 2.1)

It is obviously impossible to provide exact instructions for each combination of VST host, IR convolution plugin and audio hardware in this guide. However, as long as your audio interface supports internal loopback, you should be able to transfer the technique demonstrated above to your system.

If your audio interface does not support internal loopback, you can create external loopback by connecting hardware outputs and inputs. This is usually done by jack cables. Since this is basically just a an analog replacement of step 5, creating the external loopback is not explicitly demonstrated here. If you are an Anloo 2 user and need specific instructions, please contact PROSPECTUM via contact@prospectum.com.

3 About IR formats

In this section it is assumed that you are familiar with the terms introduced in 1.1.

In reality, the response of a reverberating space to an excitation is a four dimensional phenomenon (3D + time). Any real/physical sound source is three dimensional. The response evolves over time and space and is different at each spatial position.

When recording IRs, usually a loudspeaker is used to create the excitation signal (which translates to the universal impulse described in 1.1), and the choice and placement of microphones defines the spatial positions and directions at which the response is recorded. The usual assumption is that the loudspeaker as well as the microphones can be represented by points in space. In this case, the excitation signal is a simple mono signal located at the source position, and the acquired IR consists of as many individual mono signals as there are microphones.

A common scenario is the recording of a stereo IR, which means that there are two excitation signals (left and right channel), and two microphones are used to make the recordings. Hauptwerk 5 exclusively supports stereo IRs, so for the purpose of this document it should be sufficient to consider this case.

Such a stereo system with two excitation sources and two recorded channels per excitation source will lead to IRs that are represented by four individual channels: Left source to left mic (LL), left source to right mic (LR), right source to left mic (RL) and right source to right mic (RR). An IR that is available in this four-channel IR format is referred to as a *true stereo IR*. Sometimes, the LR and RL are omitted, for example in demo versions of commercial IR convolution plugins (example: `https://www.liquidsonics.com/software/reverberate-le/`). Since real true stereo sounds better and is a generally used standard, reduced formats will be ignored here.

The standard format for true stereo IRs is the four channel wav file where the channels are stored in the following order: LL, LR, RL, RR, see e.g. `https://steinberg.help/cubase_plugin_reference/v9/en/_shared/topics/plug_ref/reverence/reverence_true_stereo_r.html` or `https://www.siraudiotools.com/downloads/SIR2_Manual.pdf`. This is the format in which the Anloo IRs are provided. They are found in the archive HW4_and_HW5basic_IRs.zip. Make sure the IR convolution plugin of your choice can read and use this format.

If you are an Anloo 2 user and have a compatibility issue related to the format of the IRs and the IR convolution plugin you want to use, please contact us at contact@prospectum.com.