How to run a Audio Modeling SWAM VSTi on Linux

By Nicolas Abello

Version 1.1

March 4, 2017

1. Table of contents

1. Table of contents....1
2. Foreword....1
3. System used for the demonstration....1
4. System preparation....2
5. Install the SWAM VSTi....3
6. Preparing PlayOnLinux....3
7. Running VSThost....11
8. Installation of 64-bit VSTi's....14
9. Troubleshooting....15
10. Version history....16

2. Foreword

Thanks to the Windows compatibility later Wine, many programs written for Microsoft Windows may run (more or less) well under Linux. The bad news is: SWAM VSTis do not run out of the box using Wine at the time of writing. But the good new is: with a few tweaks, you can make it work easily!

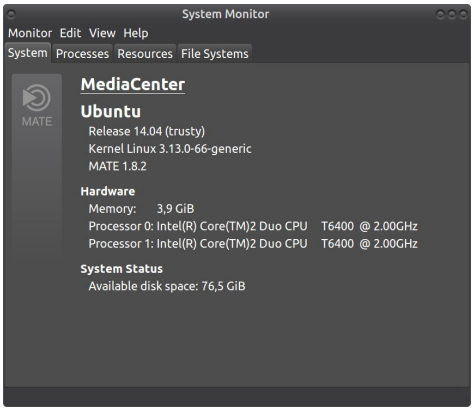
I want to thank here Emanuele Parravicini from SWAM Engine for giving me the opportunity to evaluate whether SWAM VSTi's could run under Linux by providing me with temporary evaluation license keys.

3. System used for the demonstration

I initially managed to run the SWAM VSTi on a Dell XPS13 laptop running Ubuntu MATE 16.04.2 64-bit. After I succeeded, and since I had made multiple changes on my system during hours of trial-and-error quest, I checked whether I had well understood what is needed to make the VSTi work by reproducing the workflow on a nameless assembled laptop I use as a Media Center, running Ubuntu MATE 14.04 32-bit. The screenshots presented in this walkthrough are taken from this second PC (Figure 1).

The 32-bit versions of SWAM VSTi "The Saxophones 2.6.1" (and later SWAM-S VSTi "The Cello 1.2.1") were used for this walkthrough. The 64-bit versions could be installed as well with pretty much the same procedure, except for some specific steps described in Section 8.

Figure 1. System used for the demonstration

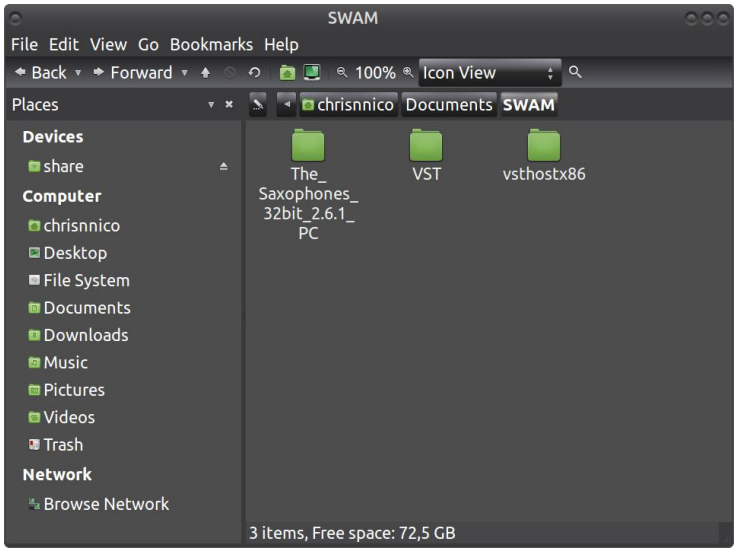


4. System preparation

Download the 32-bit PC version of The Saxophones (link received by email together with your license key). Download the 32-bit version of VSTHost (vsthostx86.zip) at http://www.hermannseib.com/english/vsthost.htm

Extract the zip files. In the example shown in Figure 2, I also created, for the sake of the demonstration, a 'VST' folder where to install the SWAM VSTis.

Figure 2. System preparation



If you do not have it already, install the Wine graphical front-end 'PlayOnLinux' using your favorite package manager. On a Debian/Ubuntu system, you may do so by typing the following command line in a terminal:

sudo apt-get install playonlinux

5. Install the SWAM VSTi

If you did not have Wine on your system, it should be present now that you have installed PlayOnLinux. Open 'SWAM The Saxophones_32bit_2.6.1_PC.exe' with Wine (Figure 3). Follow the installation process accepting the default choice, except for the choice of the VST folder where you want the files to be installed (Figure 4).

Figure 3. Start the VSTi installation process

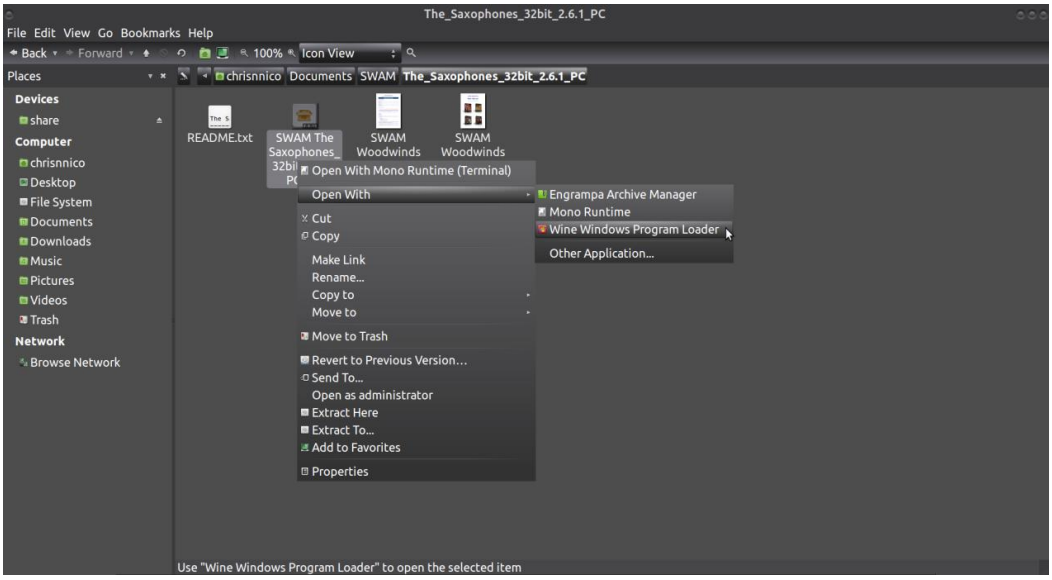
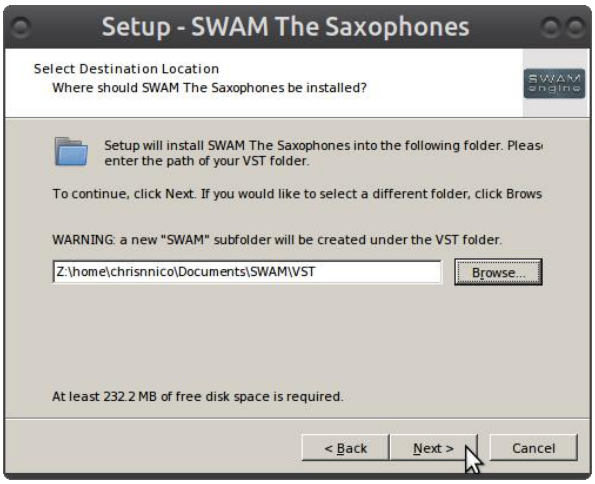


Figure 4. Choose where to install the VSTi.

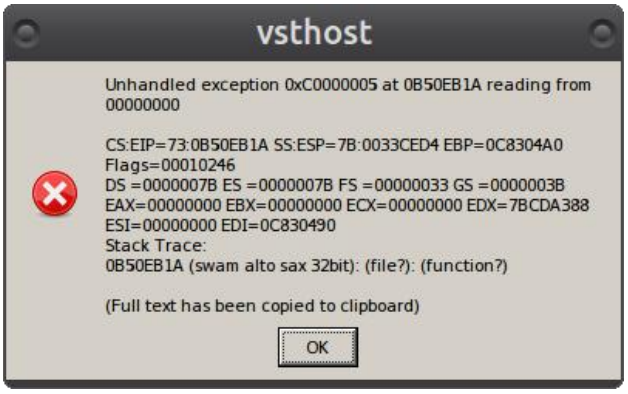


6. Preparing PlayOnLinux

We want to run PlayOnLinux to address the major reason why SWAM VSTi's do not work with Wine out of the box. With the standard Wine version (1.8) on my Ubuntu 16.04 64-bit machine, I somehow managed to get the VSTi kind of running, but there were some issues with encrypted connection, resulting in the impossibility to activate the license on line. With my Ubuntu 14.04 32-bit machine (running Wine version 1.6.2), the result is even worse: trying to get the VSTi GUI directly results in a fatal error and the

crash of VSTHost (Figure 5).

Figure 5. Fatal error when loading the VSTi GUI with Wine 1.6.2



PlayonLinux allows to run an isolated container (Wineprefix), containing a tailor-made version of Wine that will remain stable even when your default system version of Wine gets updated. In the PlayOnLinux main window, click first on 'configure' (Figure 6), then create a new Wineprefix (Figure 7). When asked which Wine version you want to use, do not mind at this stage, just take the default 'System' version (Figure 8).

Figure 6: PlayOnLinux - step 1

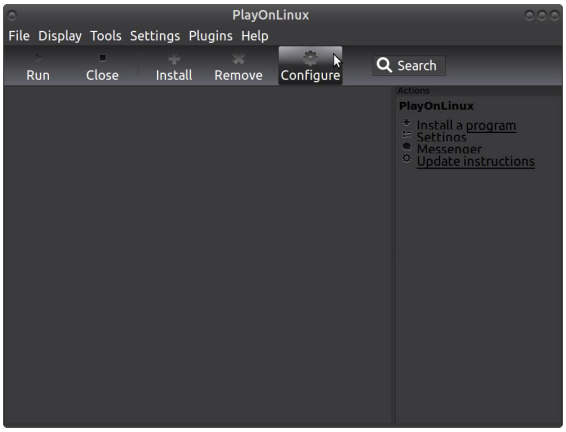


Figure 7: PlayOnLinux - step 2

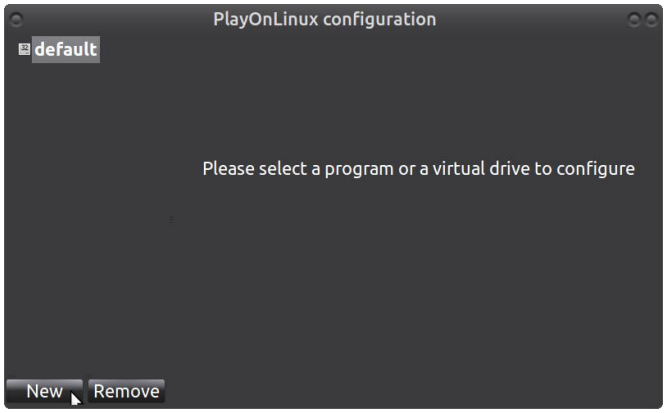
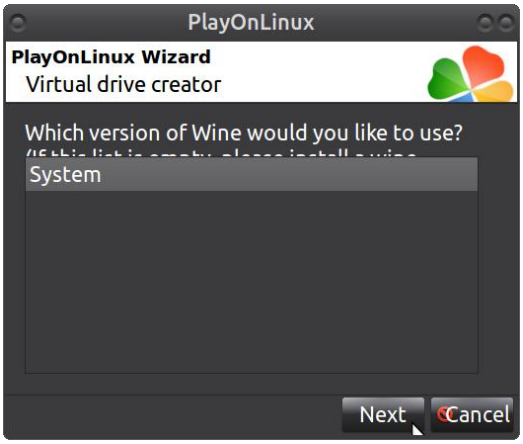


Figure 8: PlayOnLinux - step 3



Name the Wineprefix the way you want, e.g. 'SWAM' (Figure 9). Subsequently, in the tab 'General' of the newly created Wineprefix, click on the '+' symbol to add a new Wine version (Figure 10). Select and install version 2.1 (Figure 11).

Figure 9: PlayOnLinux - step 4

Figure 10: PlayOnLinux - step 5

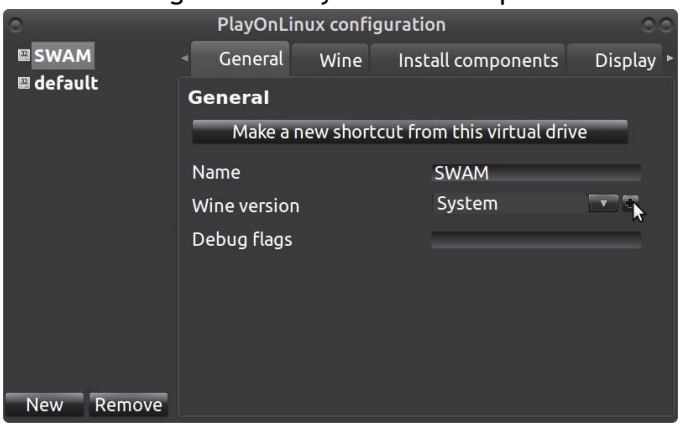


Figure 11: PlayOnLinux - step 6

Once version 2.1 is installed, you can select it for your Wineprefix (Figure 12).
The next thing you need to do is to deactivate a specific Wine library called 'd2d1'. Don't ask me why, but I found in various fora that d2d1 is a source of trouble for the GUI of Windows VSTi's running under Linux. In our case, part of the text of the pop-up appearing to activate the software license is not displayed properly, and maybe other problems may appear (I did not experiment further). To do so, click on the 'Wine' tab and select 'Configure Wine' (Figure 13). You will be asked if you want to Mono and Gecko packages. Not sure it is needed, but to be safe, just say yes (Figure 14).

Figure 12: PlayOnLinux - step 7

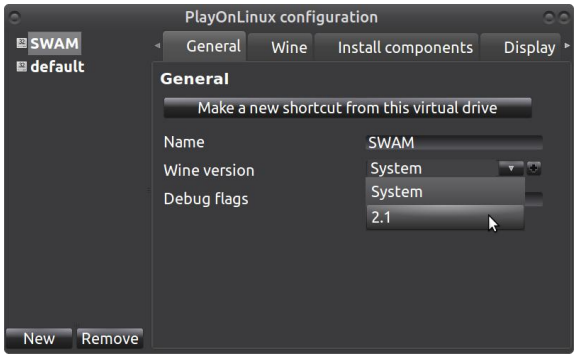
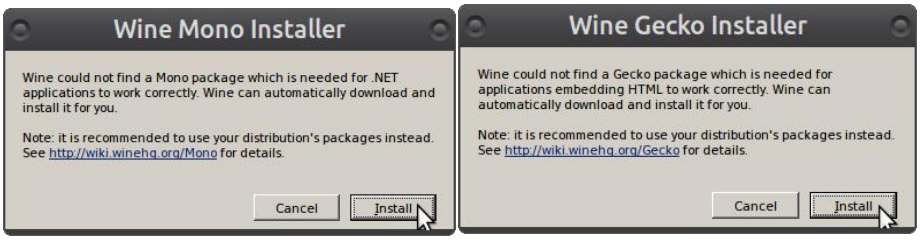


Figure 13: PlayOnLinux - step 8

Figure 14: PlayOnLinux - step 9



In the tab 'library' of the new window, add an exception for library 'd2d1' (Figure 15), click on 'Edit' and disable it (Figure 16). You can now close the Wine configuration window.

Figure 15: PlayOnLinux - step 10

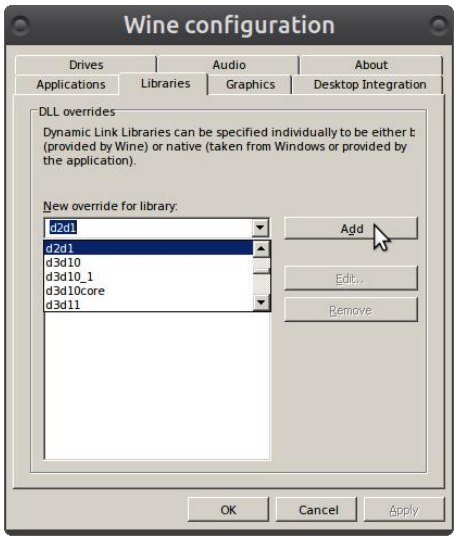


Figure 16: PlayOnLinux - step 11

You will now create a shortcut to vsthost.exe (through Wine 2.1). To do so, go the tab 'Miscellaneous', and select 'Open virtual drive's directory' (Figure 17). In the newly open window, go inside 'drive_c' (Figure 18) and copy the 'vsthostx86' directory into it (Figure 19).

Figure 17: PlayOnLinux - step 12

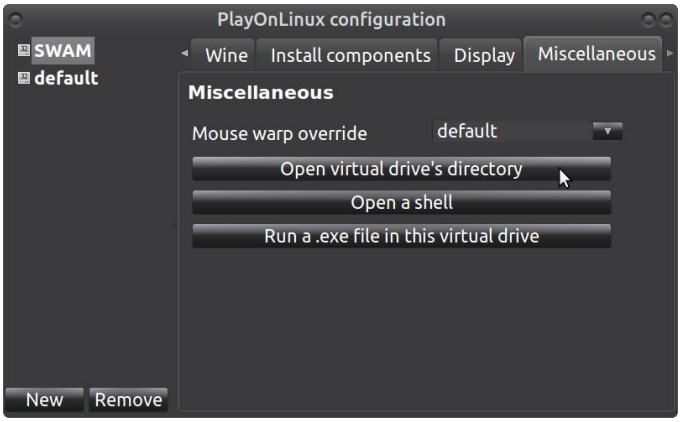


Figure 18: PlayOnLinux - step 13

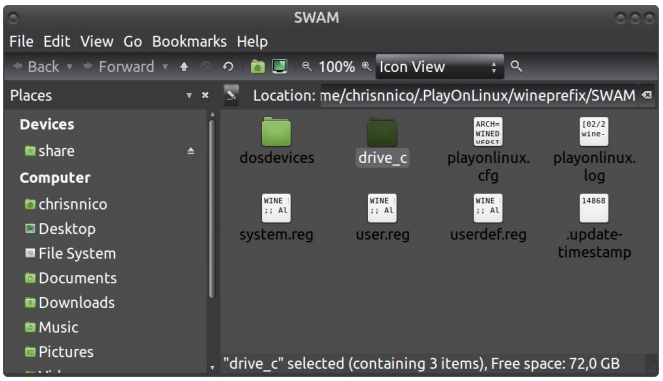
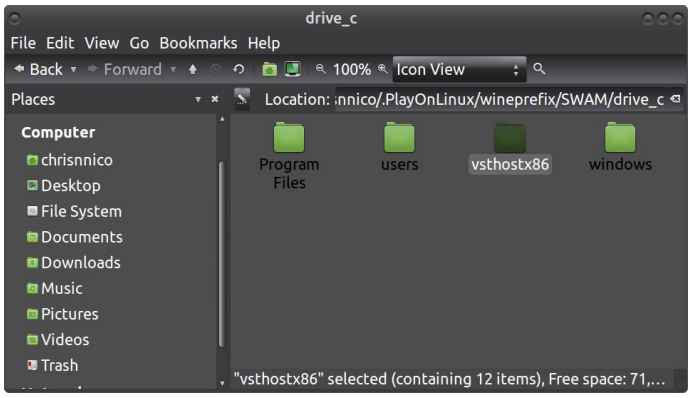


Figure 19: PlayOnLinux - step 14



In the tab 'General', you can then click on 'Make a new shortcut for this virtual drive' (Figure 20). Select 'vsthost.exe' (Figure 21) and name your shortcut (Figure 22). You now have a shortcut in PlayOnLinux and another one on your desktop (which you can use without starting PlayOnLinux first). Then indicate you do not want to make a new shortcut when asked (Figure 23).

Figure 20: PlayOnLinux - step 15

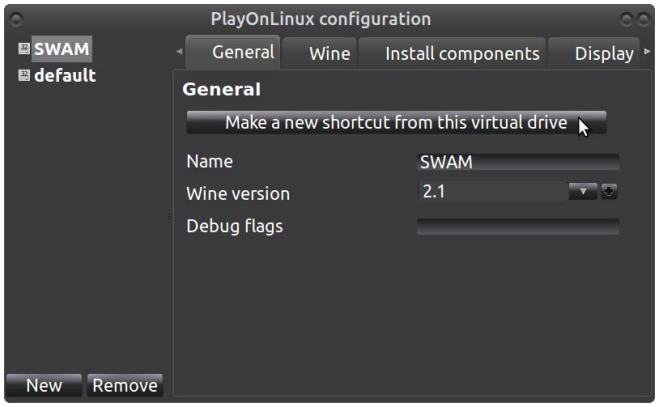


Figure 21: PlayOnLinux - step 16

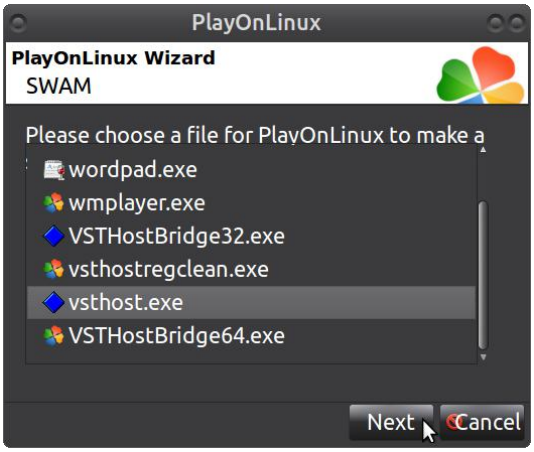


Figure 22: PlayOnLinux - step 17

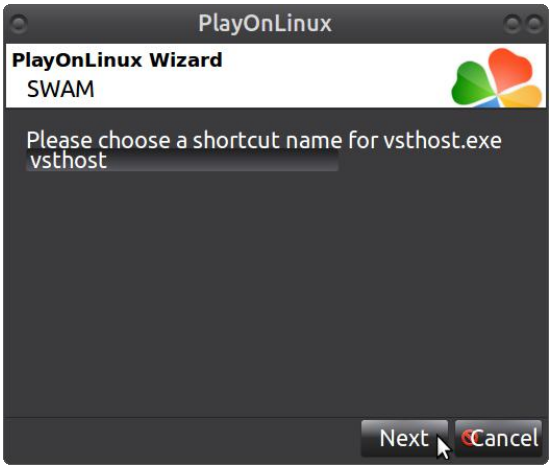
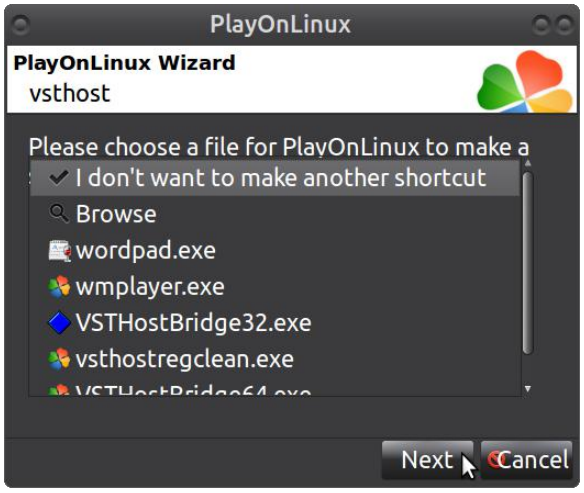


Figure 23: PlayOnLinux - step 18



If you have some reasons to want vsthost.exe outside the virtual drive, in order to make a shortcut you need first to enable shortcuts to point outside the drive. Go to the PlayOnlinux console in the 'Tool' menu (Figure 24) and type, as shown in Figure 25:
POL_Config_Write NO_FSCHECK TRUE

Figure 24: PlayOnLinux

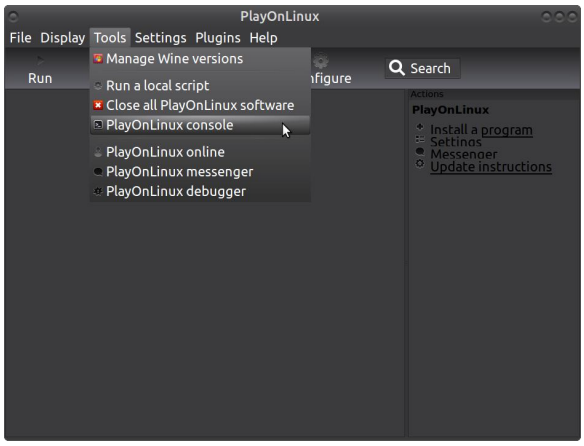
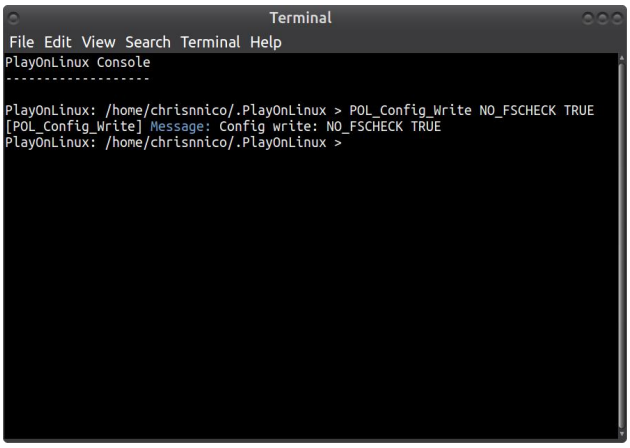


Figure 25: PlayOnLinux



7. Running VSThost

Launch vsthost.exe using the shortcut on your desktop or the one in PlayOnLinux (Figure 26). Note you can also start vsthost.exe manually by going, in the wineprefix window, to the tab 'Miscellaneous', selecting 'run an .exe file in this virtual drive' (Figure 27), and selecting the 'vsthost.exe' file were you placed it.

Figure 26: Launching VSThost with the PlayOnLinux shortcut

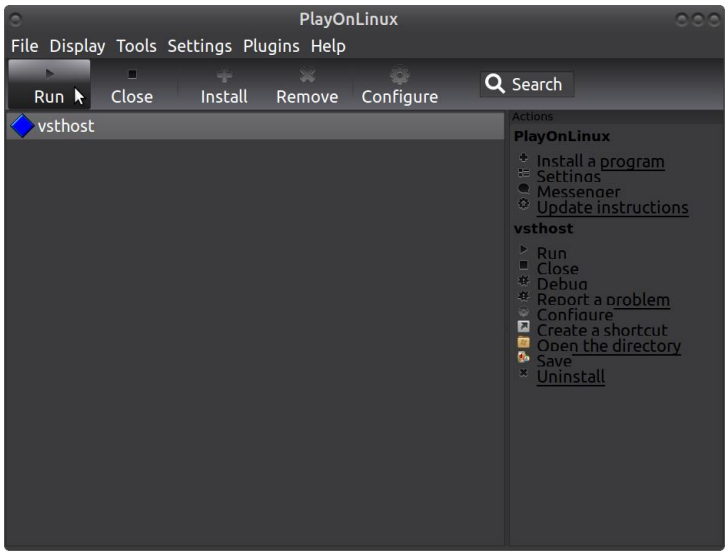
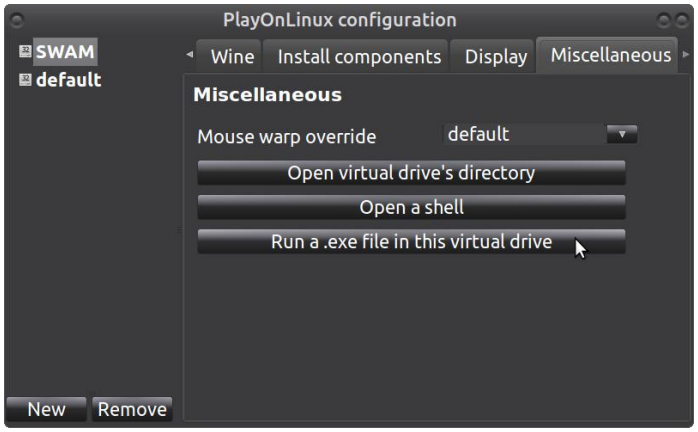


Figure 27: Launching VSThost manually



In VSTHost, click on the 'File' menu and go to 'New Plugin' (Figure 28). Find the VST folder where you installed the VSTi earlier and select the .dll of for instance, the Alto Sax (Figure 29). Repeat the operation for the other instruments.

Figure 28: VSTHost.exe - step 2

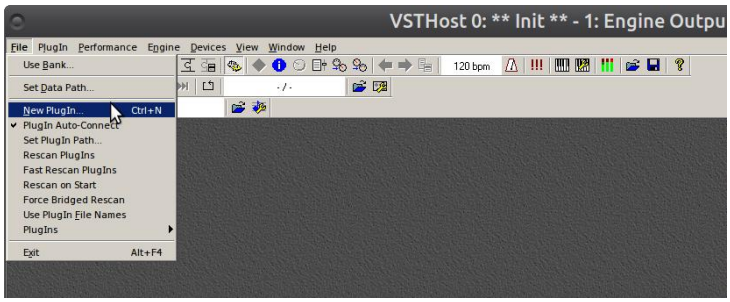
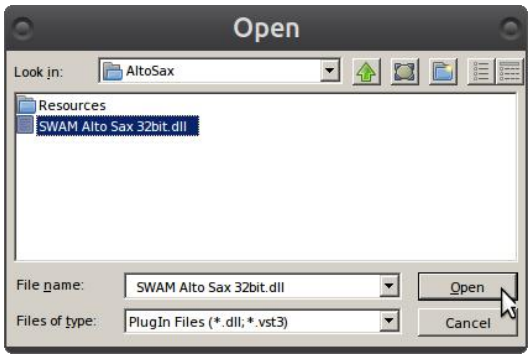


Figure 29: VSTHost.exe - step 3



Open then the GUI of one of the sax VSTi's (Figure 30) and click where you see 'click here to authorize your product (Figure 31).

You are then prompted to enter your license key (Figure 32). Do so to activate your product.

Subsequently, close the GUI and close VSTHost.

Reopen VSTHost through PlayOnLinux (Figure 26).

Reopen the GUI and voilà! You should now be able to play music using the SWAM VSTi on Linux (Figure 33)!

If you encounter some sound issue, see Section 9 (Troubleshooting).

Figure 30: VSTHost.exe - step 4

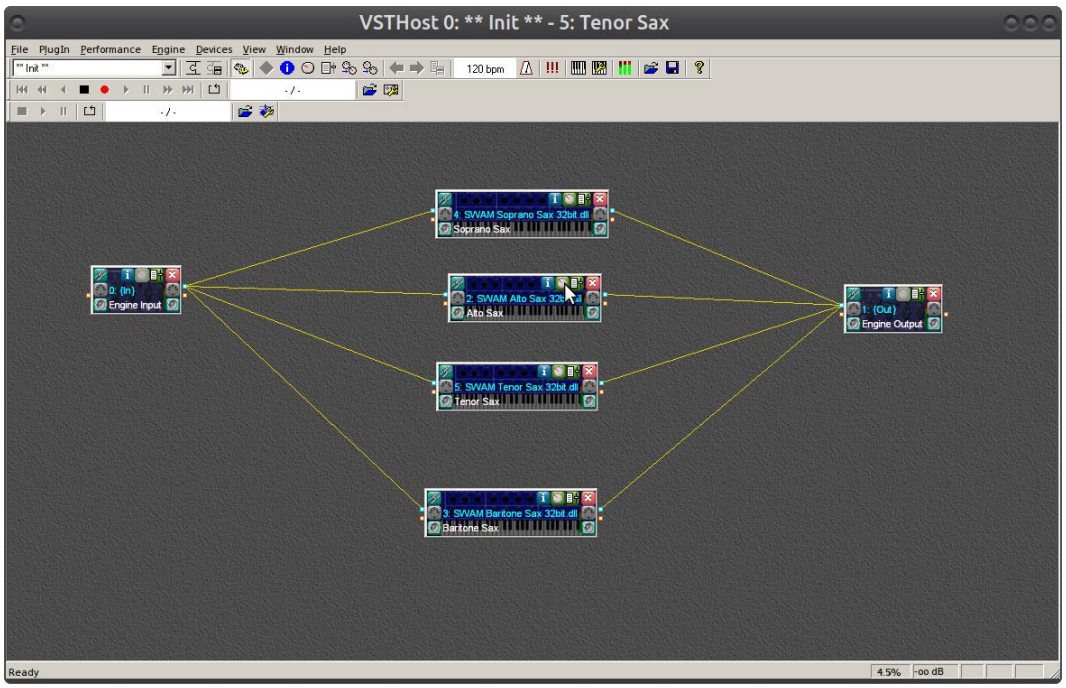


Figure 31: VSTHost.exe - step 5

Figure 32: VSTHost.exe - step 6

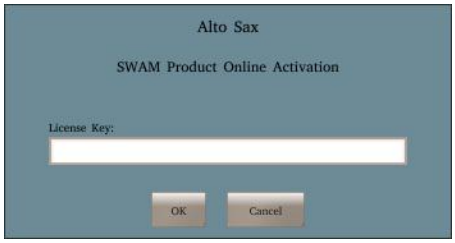
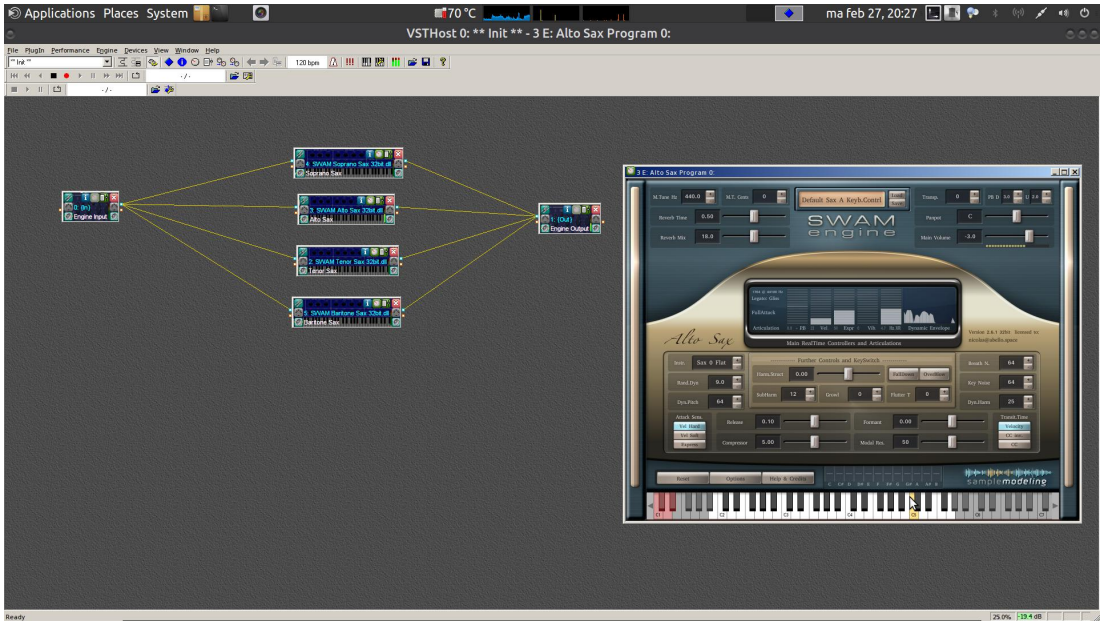


Figure 33: VSTHost.exe - step 7

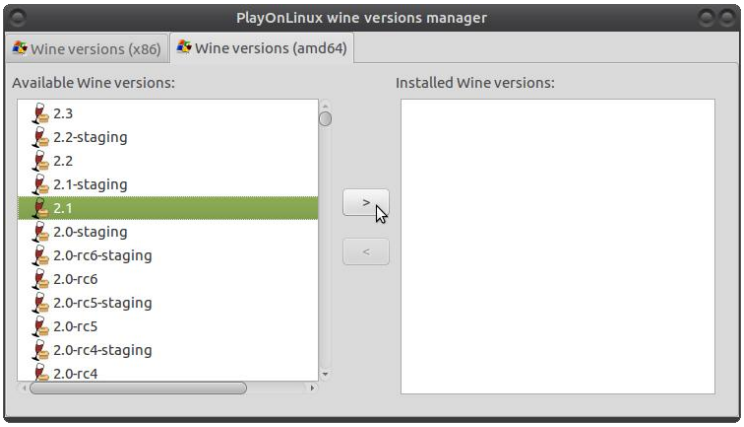


8. Installation of 64-bit VSTi's

The installation of 64-bit instruments is pretty much the same as for the 32-bit ones, except for:

- Using the 64-bit version of the VTi's obviously
- Using the 64-bit version of VSThost.exe (download 'vsthostx64.zip')
- When looking for version 2.1 of Wine of a 64-bit machine, be sure to look into the right tab, within amd64 Wine versions (Figure 34).

Figure 34: Looking for a 64-bit version of Wine 2.1



9. Troubleshooting

I personally regularly experience a choppy sound when starting playing with VSThost, on both PCs. Although it is annoying, I found an easy way to troubleshoot this. Go to your sound settings (Figure 35), and in the tab 'Hardware', select your PulseAudio output profile (in my case 'Analog Stereo Duplex'; Figure 36), turn it to 'Off' and back again to the original profile. This should refresh the profile and the sound output should now be ok.

Figure 35: Sound troubleshooting - step 1

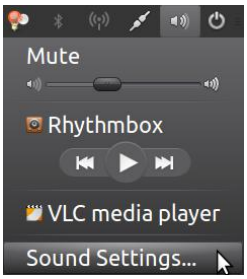
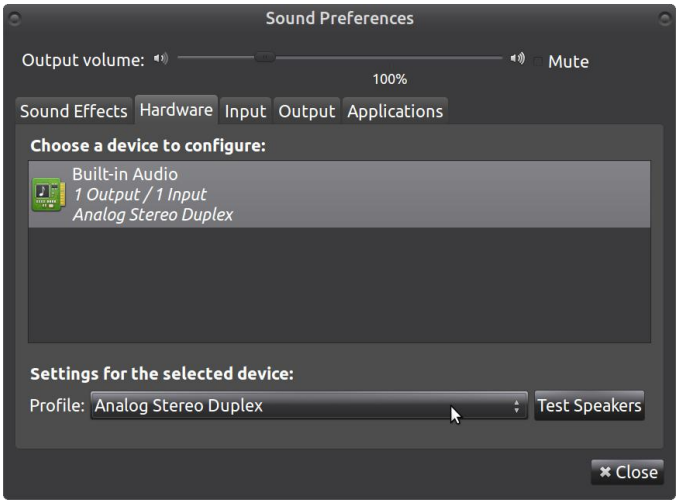


Figure 36: Sound troubleshooting - step 2



10. Version history

- 1.0 (27Feb17)
  - First version
- 1.1 (04Mar17)
  - Addition of:
    - How to create a shortcut to vsthost.exe in PlayOnLinux
    - Installation of SWAM-S 'The Cello' next to SWAM 'The Saxophones' confirmed
    - Procedure for 64-bit VSTi's added
    - Troubleshooting of sound output added