Wizard Tutorial

Download a good text editor (don't use notepad, use something like VIM) you will need one that can interpret both dos and unix text file formats.

Start with a plugin installer

-Navigate to your vfx app data directory (C:\Documents and Settings\Your user name\Application Data\vfx \vmachine) and open the "plugs" directory.

-In this folder create a new directory for the plugin. Use a good and consistent naming convention e.g. PluginVendor-PlugName

-Run the plugin installer and be sure to use its default directories. When asked for your vst directory choose the one above.

-Open the directory and make sure the vst dll file is inside. The installer may have made another folder inside ours. If so, move the directory contents back, so that our folder is the root folder.

-Run the vfx application and check that the plugin is functioning correctly.

-Close the vfx application and look inside the plugins directory where a "vfxinfo.txt" file should now exist.

-Copy this file and then open it. This is now your wizard template for this plugin.

Inside you will find a list of the plugins basic parameters eg:

```
id: 1198282345
dll_name: QuikQuak-Glass Viper/Glass Viper.dll
name: Glass Viper
is_synth: 1
parameter_count: 94
program_count: 128
has_editor: 1
editor_width: 724
editor_height: 380
```

These parameters are the essential parameters required to run the plugin and must be included in the wizard file for the current version of the PC app.

'id' is the plugins unique identifier number
'dll_name' is a relative path to the plugins dll
'name' is the plugins display name

the rest should be self-explanatory.

We now need to determine what resources the plugin requires to run. Generally, if the plugin was packaged in a zipfile (without an installer executable) all its files and/or folders will be copied next to the vst dll. For example many synthedit plugins require a .dat file which must be located next to the dll. To add a wizard instruction that imports these "relative" files and folders we can add the following field to the wizard file:

```
files_to_copy:
...
file.dat
...
```

In the same fashion we can add multiple files：

```
files_to_copy:
...
file1.dat
file2.txt
```

```
file3.foo
...
```

If we require folders to be copied, we can specify:

```
dirs_to_copy:
...
directory1
directory2
...
```

Sometimes plugins may have files residing in remote locations such as c:\program files. To specify an absolute path for file or diectory copy use:

```
files_to_copy:
...
/Program Files/Plug Vendor/Foo Plug/Foo.dat
...

dirs_to_copy:
...
/Program Files/Plug Vendor/Foo Plug/Foofolder
...
```

Note that:
1. Our wizard uses forward '/' slashes to specify paths instead of dos backslashes.
2. File paths can only be specified for the dos 'c' drive.
3. All files and folders will be copied to the plugins directory
4. Copying files from remote locations will almost certainly require the creation of
registry keys and/or symbolic links to work on v-machine (not the pc application)
5. When specifying absolute paths, be sure to use the default paths that the plugin installer specifies.

Registry Keys:

To insert registry keys for a v-machine plugin, specify by:

```
registry_keys:
...
REGEDIT4

[HKEY_LOCAL_MACHINE\Software\FabFilter\Timeless]

[HKEY_LOCAL_MACHINE\Software\FabFilter\Timeless\1.0]
"PurchaseUrl"=""

[HKEY_CURRENT_USER\Software\FabFilter\Timeless]

[HKEY_CURRENT_USER\Software\FabFilter\Timeless\1.0]
"CurrentVersion"="1.01"
"PresetFolder"="p:\\FabFilter-Timeless\\Timeless\\"
"SavePresetItemEnabled"=dword:00000000
"showInteractiveHelp"=dword:00000001
"showWelcomeHint"=dword:00000000
"ViewProgramNumbersEnabled"=dword:00000000
...
```

You can find what registry entries you will need for a plugin by searching through the windows registry. Once you have found an appropriate key export it as a "Win9x/NT4 Registration File" using the windows registry editor. Copy the text from these (1 or more) .reg files into vfxinfo.txt as shown above. NB: remember to only include the header field "REGEDIT4" once.

Editing registry paths:

When the v-machine boots up, it creates a windows drive 'p' and maps this to the plugs directory. This means a windows path is created to each plugins directory. e.g. "fooplug/foo.dll" has the wine windows path "p:\fooplug\foo.dll" (notice that registry key entries use backslashes)

Symlinks:

Symlinks are useful for relocating files or folders elswhere in the filesystem without copying them. This may be useful when a plugin has a file or folder that must always be located at the same windows path. Effectively you can "place" a file from the plugins directory anywhere in the wine windows filesystem.

e.g:

```
symlinks:
...
Program Files/Common Files/FabFilter/Shared Content/
...
```

Locates the folders in c:/Program Files/Common Files/FabFilter/Shared Content/

Name Abbreviations:

For parameter display on the v-machine, a plugin name abbreviation is created. This name is comprised of 3-4 characters and by default is generated from the plugins unique id. Some manufacturers have a unique id associated with their plug which decodes to a 4 letter short name. Other do not and result in numbers and nonsense names. To override the automatic name abbreviation, specify it by:

```
name_abbreviation: FOO
```


Using the wizard file

Once you have completed a wizard file, test it by moving the plugin directory from the plugs directory elsewhere (e.g. desktop) Delete the old vfxinfo.txt file from this directory. Use the pc app "Import VST Instrument or Effect" feature. Select the dll in from moved directory. Select the wizard vfxinfo.txt file when prompted. Make sure you can use the plugins on the pc app.

Testing on V-Machine

Copy the plugins directory to a FAT formatted usb mass storage device into a directory named plugs/

e.g. plugs/fooplug

power on your v-machine with the learn buttons pressed. Insert the usb drive when "V-MACHINE Update.." is displayed.

Change bank to "Plugin Parade", change preset to the new plug and test operation.

Note that some plugins require a preset to be created before sound may be heard.

Future wizard extensions:

The following features will be added with later versions of the wizard:

--We won't need to include the basic parameter list unless we want to overwrite them.
--Prompting the user for a manual search if a directory or file to sync can't be found.
--copying files and folders to a unique location within the plugins dir e.g:

/Program Files/foo/foo.txt:New Folder/foo.txt
/Program Files/foo/folder:New Folder/

This will allow us to strip unessesary files.

--Registry templates so that a user may be able to insert personalized info such as serial numbers.
--Text File templates so that again for license and serial info.
--Additional synchronization dirs -- ability to update a copied v-machine dir on sync.