Installing Links 2003 on Linux

There are a series of steps required to get Links 2003 running on Linux.

As Linux Mint 21.x was the distribution used in testing, the instructions here should be directly relevant for any Debian based distribution. For other distributions you may need to substitute the commands used, the basic methodology should be the same.

This document seeks to take you through the steps, the information has been pulled from many sources. It is not the intent here however to take you through the use of basic Linux commands like creating directories, changing to them, extracting archives etc, so that knowledge is assumed.

Test installations were carried out on an Acer Notebook and on a generic desktop. In all cases x.org has been used as the X server, no testing has been carried out under Wayland.

These instructions were developed using a Links 2003 CD kit. As I understand it the downloadable Links kit already includes the 1.07 update but not the “Nvidia“ crash fix.

It is recommended to read this document in its entirety before starting.

Set up the Environment

The first stage is to set up an environment in Linux in which to run Windows applications. This is achieved by using Wine.

Wine

Wine is a compatibility layer that allows Windows programs to run on Unix like Operating Systems. To achieve this it translates Windows API calls to POSIX calls, so there is no need for emulation or virtualisation and hence no need to install Windows.

At the initial time of writing the installation was completed with Wine 9.22 as was most testing. The instructions given here use a dedicated wineprefix, so one has been set up specifically for Links use. In this case Links is also the only application using the wineprefix so is isolated. The author follows this approach.

In later testing Wine 10 was used.

1. First check to see what version of Wine is installed, this is done using the following command in a terminal:

```
wine --version
```

Based on this information you may or may not need to upgrade. You could decide to see if the current version of Wine you have or your distribution provides works.

2. To remove an existing version of wine you can use the command:

```
sudo apt purge wine*
```

3. To install the newest stable release of Wine you can use the instructions at:

```
https://wiki.winehq.org/Download
```

In summary, on the test system, we executed the following commands:

```
sudo dpkg --add-architecture i386

sudo mkdir -pm755 /etc/apt/keyrings

sudo wget -O /etc/apt/keyrings/winehq-archive.key
 https://dl.winehq.org/wine-builds/winehq.key

sudo wget -NP /etc/apt/sources.list.d/
 https://dl.winehq.org/wine-builds/ubuntu/dists/jammy/winehq-jammy.sources

sudo apt update

sudo apt install --install-recommends winehq-stable
```

This is the relevant kit for Mint 21.x, substitute with the kit required for your system and the instructions to install it.

Using cut and paste of each line from the instructions provided on the winehq site into a terminal session works well and saves typing while reducing the likelihood of errors.

After installation you should be able to execute the command:

```
wine --version
```

and it should come back with the version just installed e.g.

```
~:$ wine --version
wine-9.22
```

Winetricks

Winetricks is a script that provides a menu interfaces for Wine setup and management, we are using it here to gain easy access to many of the setup tools for Wine that we need to use to get Links installed and working.

You should use the latest version of Winetricks:

To uninstall any old/existing version of Winetricks, on our test system we executed the command:

```
sudo apt-get purge winetricks
```

To Install the latest version

1. Download latest tar.gz file from

   ```
   https://github.com/Winetricks/winetricks/releases
   ```

2. Unpack the archive

3. In a terminal session move to the created folder it should have a Readme and a Makefile in it.

4. Use the following command to install it:

   ```
   sudo make install
   ```

5. Test installation by typing in the following on a command line:

   ```
   winetricks
   ```

6. Wine at this stage is likely to say it cannot find a wine-mono package, select to Install it, after performing the install Wine will start, and create a default “wineprefix”.

A Wineprefix is a folder that contains the Wine configuration as well as all of the Windows pieces that Wine uses for compatibility, including libraries and a registry. You can have multiple wine prefixes, for example you could potentially have one for each application. If you like you can consider each wineprefix to be equivalent to a Windows “system”.

I would suggest you may also like to install WineGUI which provides a Graphical Management System for Wine. The author has both installed and uses which ever tool makes sense for the job at hand and generaully uses WineGUI as the entry point.

When you wish to use both tools I suggest you change WineGUIs configuration so it places Wineprefixes where Winetricks expects them to be, this will ensure you can use both tools effectively.

There are pointers to both tools on the winehq site. For WineGUI on a Debian based system it is just a matter of downloading the .deb file and install it.

The following instructions make use of Winetricks only, you can use WinGUI if you prefer, and even where WineGUI cannot do a task it has an option to start Winetricks. As said the author tends to use WineGUI as the overall Wine Management interface.

Links Wine Prefix

The intention with this installation is to have a completely separate environment just for Links, doing this means that nothing else can interfere with Links and Links cannot interfere with anything else.

We can use Winetricks to do this as follows:

1. Start Winetricks
2. Choose the option to Create a new wineprefix and select OK
3. On the next select 32 bit and give it a name. In this example we are using the name “Links”

With Winetricks started and with its current focus being on the “Links” we can perform some more steps.

Initial Configuration of the “Links” Prefix

At this stage we need to perform some basic configuration task in our “Links” Wineprefix to make it a suitable environment for Links.

Start the Winetricks script, if it is not already running, and from the menu:

1. Select the Wineprefix being used for “Links” if necessary
2. Select “Change settings”
3. Scroll right to the bottom and select “winxp”
4. Click on OK

This sets the environment to a specific version of Windows. Windows XP is, of course, supported by Links and it was found that there could be some issues around the setup if later versions of Windows are selected.

Windows Media Player

Links 2003 will fail with the inbuilt Wine media player, wmvcore.dll. The system requires the Windows Media Player 9 to overcome this, if this is not installed Links 2003 will crash with an error like the following:

```
Exception Information
Code: 0xc0000005      Flags:0x00000000
Record: 0x0000000000000000     Address:0x0000000003d11004

Followed by some system information, a list of loaded modules, and a
thread dump.
```

To install Windows Media Player 9 you can use Winetricks. Start this script, if it is not already running, and from the menu:

1. Select the Wineprefix being used for “Links”
2. Select Install a Windows DLL or component
3. In the list scroll down to “wmp9” Windows Media Player 9 and select it
4. Click on OK and install it. You will have to respond by clicking yes a number of times during the install

Installing Links 2003

The steps given below are the basics that are needed. There has been much discussion over the years on what exactly needs to be installed with Links. The following list shows an installation using the CD base kit and the minimum number of patches:

1. Install Links 2003
2. Install 1.07 update. This incorporates the 1.06 mod, the “No CD” patch, the 1.02 patch, the 1.05 patch as well as its own patches and improvements.
3. Install the Nvidia Patch

The Nvidia patch can help with some video issues with other chip sets, not just Nvidia, so it should be installed as a matter of course. It also incorporates a fix that can resolve crashes with courses that have reflective water.

Step 1

Firstly insert Links 2003 CD 1 from the kit in the CD/DVD drive.

1. Start Winetricks
2. Select “Links” wineprefix
3. Select to “Run an arbitrary executable”
4. In the Browser Window that comes up navigate to the top level of the CD
5. Select on Setup.exe and click on OK

Links should proceed through a normal setup. You will probably have to manually open the tray to change CDs.

A program group and a desktop icon will have been created by the installation process and will include the needed wine commands.

The properties of the Links programs should be changed so that Links starts in XP compatibility mode.

In Winetricks:

1. Select “links” prefix
2. Select run winecfg
3. On the Applications tab add LinksLauncher.exe and LinksMMIII.exe and set the Windows compatibility Version to: Windows XP

   You will need to navigate to the Links executables in the winecfg program, the path is Program Files (x86)\Microsoft Games\Links 2003

Keen observers may note that this is not dissimilar to setting the compatibility mode on Windows. Given Wine’s objective of running Windows applications you will see this mimicking of Windows actions with a Windows look within the software.

Step 2

Install the Links V1.07 update. This is named *Links_1-07_final.exe* and can be downloaded from https://www.linkscountryclub.com/.

To install using Winetricks:

1. Start Winetricks
2. Select the “links” wineprefix
3. Select to “Run an arbitrary executable”
4. In the Browser Window that comes up navigate to where Links_1-07_final.exe is, select it and click on OK.
5. You should see a typical Windows installation.

Select the overwrite options, these are files that the patch replaces. When in Links 2003 you can validate that 1.07 is installed by choosing Options then click on the About button.

In summary this patch should install:

- V1.02 patch
- V1.05 patch
- Patch to run without Links CD 1 in the drive
- MOD (V1.06 update)
- V1.07 update

Without this update Links will fail to run on Linux, and of course, without it, you can not play 2003T courses.

Step 3

Install the Nvidia patch. This patch is called *Links2k3_nvidia_win7_FINAL.exe*

To install using Winetricks:

1. Start Winetricks
2. Select the “links” wineprefix
3. Select to “Run an arbitrary executable”
4. In the Browser Window that comes up navigate to where Links2k3_nvidia_win7_FINAL.exe is, select it and click on OK.

   You should see a typical Windows installation.

While the name of the patch implies it is only for Nvidia equipped machines as stated earlier it does address other video issues.

Step 4

Before going any further you need to have started Links in order for the registry settings to be created for the user, the Wine environment does have a registry and it looks like the Windows registry, so should be familiar to Links users. The previous steps should mean that Links will start with default settings.

Links is likely to crash the first time you try to start it. If it does this, start it a second time and it should run. Anytime you change a core configuration option, and especially those related to Video, Links may well crash the first time it is run afterwards, starting Links a second time should result in it running.

After starting Links, you can try practice mode, if you like. It is expected that Links will operate with a relatively low resolution.

Step 5

Now you need to change the registry setting to match the video mode of your system, or in some cases what works best on your system, it has been noted during testing that a resolution of 1366 x 768 was problematic on two Linux systems while 1280 x 720 worked well on the same two systems, this resolution was also problematic under Windows as well with the same solution employed. Initially I suggest starting with you system's native resolution.

You need to run regedit within Wine, again you can use Winetricks:

1. Start Winetricks
2. Select the "links" wineprefix
3. Select and run regedit

This will give a window that looks like an older version of the registry editor on Windows

You take the path:

```
HKEY_CURRENT_USER
 Software
   Microsoft
     Microsoft Games
       Links 2003
         DisplaySettings
```

Change the setting for "Mode" to the appropriate value as per the table below.

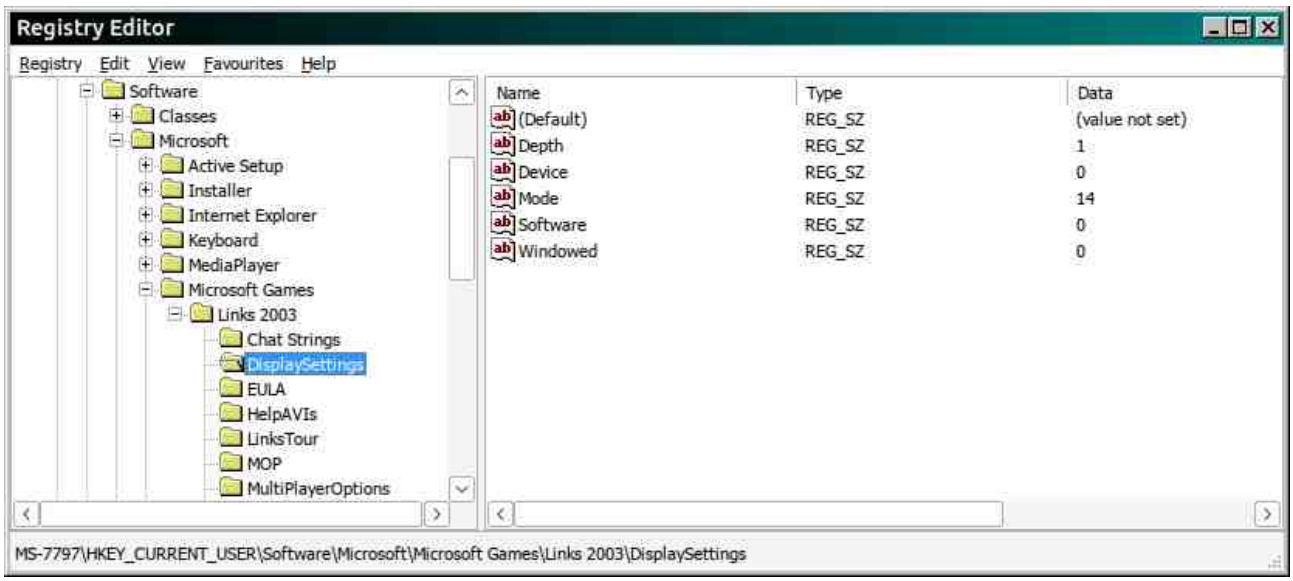
Mode	Resolution	Known As	Aspect
15	1920 x 1200	WUXGA	16:10
14	1920 x 1080	FHD 1080p	16:9
13 [3]	1808 x 1008		~16:9
12 [3]	1680 x 1050	WSXGA+	16:10
11 [3]	1440 x 900	WXGA+	16:10
10 [3 4]	1366 x 768	WXGA HD	~16:9
9 [1 3]	1360 x 768 [2]	FWXGA	~16:9
8 [3]	1280 x 960	SXGA-	4:3
7 [3]	1280 x 800	WXGA (16:10)	16:10
6 [1 3]	1280 x 768	WXGA (15:9)	5:3
5	1280 x 720	WXGA 720p	16:9
4 [3]	1600 x 1200	UXGA	4:3
3 [*]	1280 x 1024	SXGA	5:4
2 [*]	1024 x 768	XGA	4:3
1 [*]	800 x 600	SVGA	4:3

[*] These are the original values as of Links 2003 Patch 1.05 and generally 800 x 600 will be used on initial set up.
[1] Settings not working on a notebook with WXGA HD (1366 x 768). Links gives “A minimum setting of 800 x 600 is required” error, then exits. This notebook has both Nvidia and Intel GPU chipsets.
[2] Adjusted resolution to fit into an 8Mbit memory chip.
[3] Settings not working on a Desktop with FHD (1920 x 1080). Links gives a “A minimum setting of 800 x 600 is required” error, then exits. Modes 14 & 5 work. On the test system only a small subset of resolutions were reported by the Operating System and these where the ones where Links would start.
[4] While the setting does work on a notebook with WXGA HD (1366 x 768), it suffers from the disappearing golfer and missing menu items issue, Mode 5 does work.

In summary just because a resolution should work on a system you may or may not have problems. Trial and error can be required.

Ideally you would also use a 32 bit video mode, this is the “Depth” setting, 32 bit is a setting of 1.

The registry should look something like:



Once you have made your changes using Wine's regedit, you can verify your Links screen resolution settings by starting Links and going into Links/Options and selecting the 'system info' button at the bottom of the screen. The 'Video Resolution' and Depth there should reflect the option you entered in regedit. Choosing 'Graphic Settings' in the Settings box on the same Options screen may or may not report the correct 'Display Mode' it can be hit or miss. Do not attempt to change your video resolution in here after you have used regedit, you may to need to go back and use regedit if you do.

From this you can see you may need to try different video modes in order to get the best overall result, the main thing to note is to not exceed the resolution you system is capable of, that will definitely not work. If you have a problem, start by dropping down to the next highest resolution your system could support. You may also need to use a resolution that has an aspect ratio that the screen has native support for.

Step 6

It can be a good idea, but is not mandatory, to add an entry for your Video card to Links video definition file videocard.txt. How to do this on Linux is noted here.

On Windows you use the DXDIAG utility to obtain the needed information, on Linux we use two utilities:

1. Firstly use the `lspci` command.

This gives information on all PCI devices in the machine, on the test machine, it has an inbuilt GPU plus an add on GPU, thus we are interested in two devices here.

Scanning the output, two graphic controllers stood out:

```
00:02.0 VGA compatible controller: Intel Corporation HD Graphics 620
```

```
                                :
                                :
01:00.0 3D controller: NVIDIA Corporation GM107 [GeForce 940MX]
```

So the machine has the inbuilt Intel 620 graphics GPU plus a Nvidia GeForce 940MX GPU. It is the text we are more interesting from this output as well as providing a checkpoint.

2. Next we use the command: `sudo lshw -numeric -C display`

This gives more detailed data on the graphics devices, the relevant lines are:

```
description: VGA compatible controller
product: HD Graphics 620 [8086:5916]
vendor: Intel Corporation [8086]

description: 3D controller
product: GM107 [GeForce 940MX] [10DE:179C]
vendor: NVIDIA Corporation [10DE]
```

We can tie the video GPUs between the two outputs, the significant details we are interested here are the Vendor IDs and the GPU IDs, for example 8086 is the ID for Intel and 5916 is the specific ID for an Intel 620. We now need to use this information to add suitable entries to Link's videocard.txt file.

So at the bottom on the Intel section the following entry was added.

```
0x5916 = "Intel(R) HD Graphics 620"
    DefaultBltNotFlip
    NoCopyZInVidMem
```

and at the bottom of the Nvidia section

```
0x179c = "NVIDIA GeForce 940MX"
    DefaultBltNotFlip
    NoCopyZInVidMem
    RenderEqualZ
```

The entries are based on other HBAs by the manufacturer as well as some other more recent GPU entries that have been seen.

Golf Course Locations

You may want to have courses located outside of the links folder and indeed outside of the wineprefix that Links is installed in. Often the easiest way to achieve this is to in Wine set up a drive letter that points to the location.

1. Start Winetricks
2. Select the "links" wineprefix
3. Select and run winecfg
4. Select the Drives tab in winecfg
5. There you can select a drive letter and assign a Linux path to the location

Now when installation courses, you can use this drive letter to gain easy access to where the course are. On the test systems drive “D” was set up to point to the course folder.

Also with Wine, drive Z points to the root location i.e. “/” so you can access any location on the system.

On the test systems the Operating System and applications are installed on a relatively small SSD drive. The courses are kept on a second HDD. The courses being on a hard disk has not caused any performance issues.

Integrated Intel GPUs

It is suggested to initially not set any setting specifically for any Intel GPU and to test Links. If you should encounter problems like missing golfers and menus or course crashes then refer to the Issues section below where some encountered problems are discussed along with solutions.

Nvidia GPUs

Under Linux you have the choice of using proprietary Nvidia drivers or the Open Source Nouveau driver.

As part of the testing both have been used and the results included here.

In summary, based on my testing, I cannot recommend the use of an Nvidia GPU with Links under Linux. In all tests there were performance problems with general slowness or an inconsistent slowdown in rendering at times, with Nvidia, it could become “painful” and certainly decreases the enjoyment, in all cases the Intel iGPU performed better.

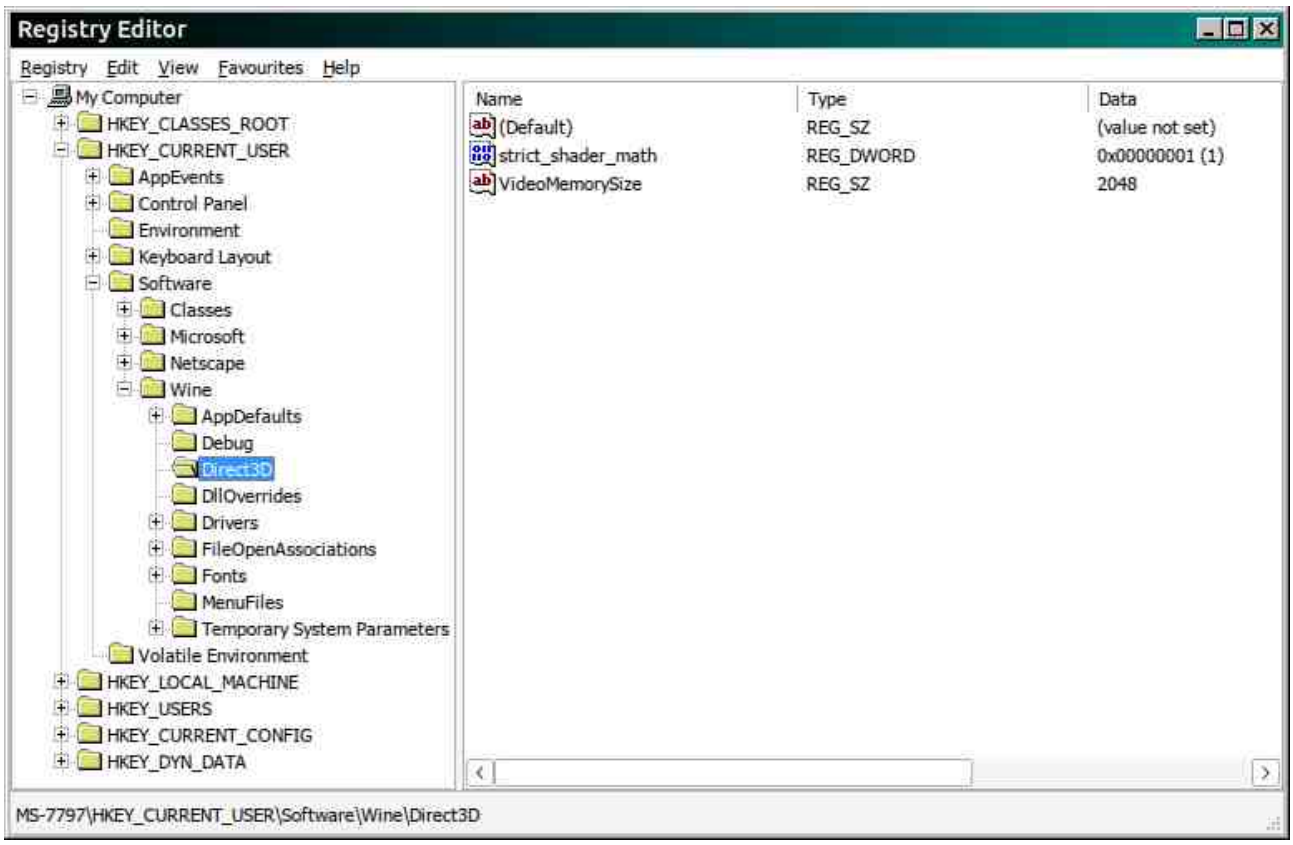
Nvidia Proprietary Drivers

We tested a range of Driver families, up to including the latest supported with the GPU in question. I cannot say any driver family works any better than the others.

Nvidia has been tested under Wine 9.22 and Wine 10, it was found the settings employed are inline with expectations when using an Nvidia HBA.

You should set the VideoMemorySize parameter to the amount of the dedicated video memory the GPU has. In theory Linux should be able to determine the amount of video memory, but there is no harm in setting it explicitly.

You should also add a Wine specific entry for Nvidia GPUs, the entry is named “`strict_shader_math`” it is of type DWORD and the value to set it to is `0x1`. So it looks something like:



The Nvidia specific entry enables the generation of NVIDIA specific shader code. This is used to work around bugs that often results in incorrect rendering of some objects which is reproducible with the Nvidia proprietary driver.

As a note, setting a severely restricted memory size when using an Nvidia GPU, has been seen to adversely affect performance. The HBAs used here have 2GB of dedicated Video memory.

In general with the Nvidia HBAs, performance with Links using the proprietary drivers under Linux has not been great, most notably the rendering speed can be quite inconsistent. This seems to be very much driver related, so different versions of drivers can give very different results. The speed changes and the, at times, quite lengthy delays, made for frustration. Given this on machines that also have an Intel iGPU the recommendation would be to use that.

On Linux, Nvidia on Demand, could not be made to work with Links during testing, the system had to be switched into Nvidia only mode in order to use the Nvidia GPU.

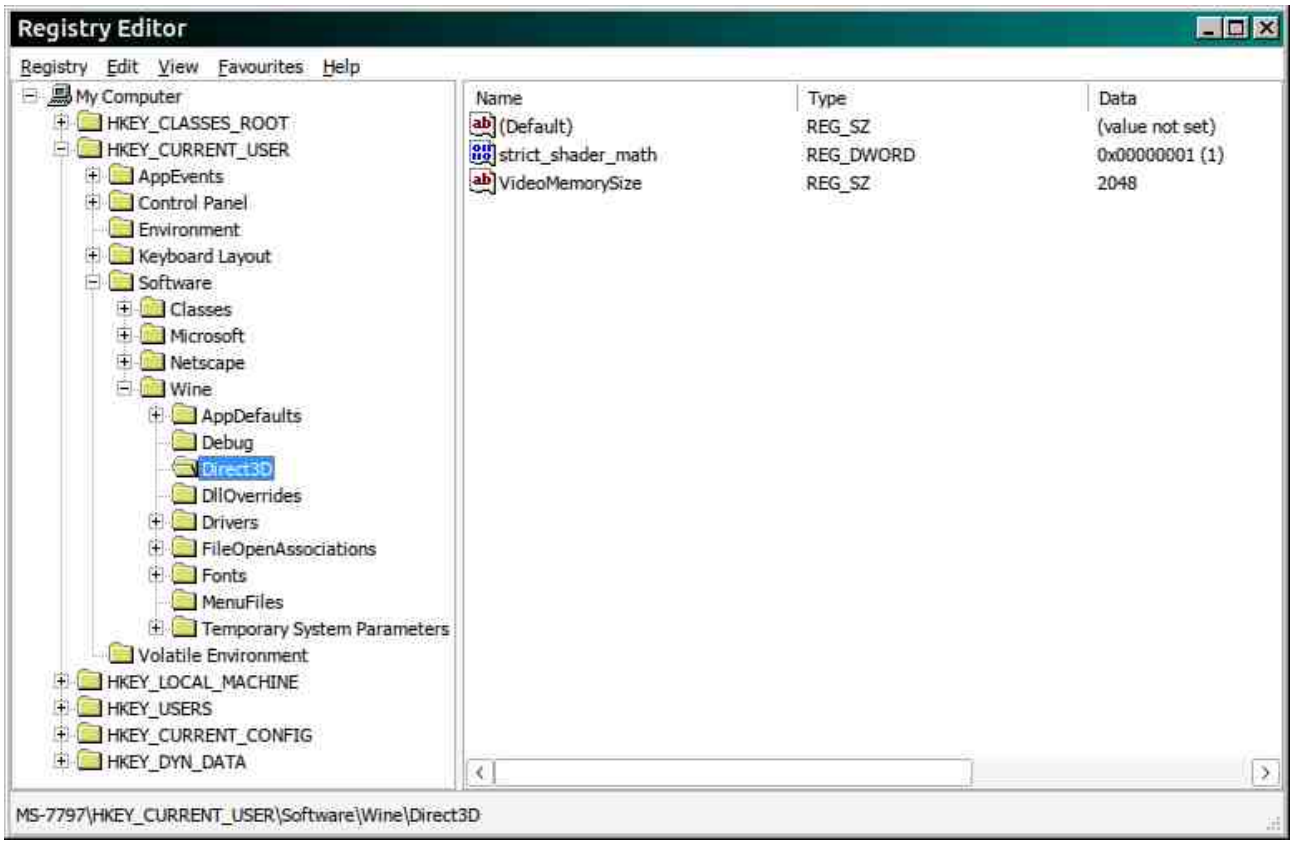
Nouveau Driver

The Nouveau driver is the open source driver for Nvidia GPUs, it is certainly worth a try if nothing more, if you have an Nvidia GPU and have not installed, or have removed the proprietary driver, then the Nouveau driver should be installed and active, you cannot have both the Nvidia proprietary and Nouveau drivers active at the same time.

The basic setup is the same as for the proprietary driver, it is repeated below.

You should set the VideoMemorySize parameter to the amount of the dedicated video memory the GPU has. In theory Linux should be able to determine the amount of video memory.

You should also add a Wine specific entry for Nvidia GPUs, the entry is named "`strict_shader_math`" it is of type DWORD and the value to set it to is `0x1`. So it looks something like:



The Nvidia specific entry enables the generation of NVIDIA specific shader code. This is used to work around bugs that often results in incorrect rendering of some objects which is reproducible with the Nvidia proprietary driver.

As a note, setting a severely restricted memory size when using an Nvidia GPU, has been noted to adversely affect performance. The HBAs used here have 2GB of dedicated Video memory.

There is a further change you may need to make. On the notebook used for testing it has both Intel and Nvidia GPUs, by default the Intel GPU is used at all times, we need to force to use of the Nvidia GPU with Links.

To do this we need to set the DRI_PRIME environment variable before Links is started.

This can be done on the Links command line something like:

```
env DRI_PRIME=<x> WINEPREFIX="<prefix path>/<prefix name>" wine C:\\users\\Public\\Desktop\\Links\
2003.lnk
```

Where:

<x> is the GPU to use
<prefix path> is the Linux path to where the prefix resides
<prefix name> is the name of the prefix

To work out which GPU is which you can use commands like:

```
DRI_PRIME=0 glxinfo | grep "OpenGL renderer"
DRI_PRIME=1 glxinfo | grep "OpenGL renderer"
```

In the testing performed the Nouveau driver seems to be fairly consistent in terms of rendering times without too much variation. Overall it is likely to be a better option than the Nvidia driver given the predictable performance. However it is still a relatively poor performer compared to an Intel iGPU therefore on machines that also have an Intel iGPU the recommendation would be to use that.

Using a Hybrid Nvidia/Intel Setup on a Desktop

On a Notebook a Hybrid set up works with the inbuilt screen or external monitor there are no separate video connections. On a notebook the Intel always drives the display, so it is the display controller, but you can switch between the GPUs for the screen rendering work. So the task can be divided between the two GPUs, or Intel can do everything, the Nvidia can never do all since it is not connected to the display, it is only a rendering engine in this instance not a display controller, it can only tell the Intel to “here display this”.

However on a desktop you have separate Video controllers with separate connectors.

There are various reasons to add a separate video HBA to a system, on the system in question these are only two video connectors a digital one (HDMI) and an analogue one (VGA), the requirement was to have two monitors and they have digital input, however, the most common reason is for better performance than you normally get from Integrated graphics

However the performance of Links under Linux with Nvidia is quite poor and the Integrated Intel GPU outperforms it easily.

The solution in this case is use a hybrid set up and to connect one monitor to the iGPU and one to the dGPU, the primary display to the iGPU and use this for Links with the Intel GPU doing the rendering rather than the Nvidia.

Many systems will disable the iGPU when you add a separate HBA, you will need to go into the BIOS to enable both.

There is still the problem however that by default the Nvidia is being used for the OpenGL rendering whereas we need the Intel to do this.

Initial testing was carried out using Nvidia proprietary drivers, however despite trying various methods Links would not use the Intel GPU for rendering.

Testing was then carried out using the Nouveau driver for the Nvidia hardware With this configuration were were able to successfully use either GPU depending in the command line used to start Links.

It is advisable to not assume which will be the Primary and which will be the Secondary GPU, so best to ask Linux to tell us. We can do this as follows, with the indicated results.

```
$ DRI_PRIME=0 | grep "OpenGL renderer"
OpenGL renderer string: NVE7

$ DRI_PRIME=1 | grep "OpenGL renderer"
OpenGL renderer string: Mesa Intel(R) HD Graphics 2500 (IVB GT1)
```

So from this example we need to use DRI_PRIME=1 as part of the Links command line, it will look something like:

```
env WINEPREFIX="<prefix path>/<prefix name>" DRI_PRIME=<x> wine C:\\users\\Public\\Desktop\\Links\
2003.lnk
```

Where:

<x>	is the GPU to use
<prefix path>	is the Linux path to where the prefix resides
<prefix name>	is the name of the prefix

When Links start it reports the rendering engine in use and in this case reports the Intel, playing a few holes confirms the improved performance.

With the test system the Nvidia HBA is only being used as a simple display controller and the Nouveau driver is perfectly adequate for this.

Radeon GPUs

No Radeon GPUs were available for testing. For GPUs with discrete memory it is suggested you set the memory parameter initially to the actual amount of video memory and work from there.

Summary

After following this process you should have Links installed and running.

You may face some issues as you are playing and the next section discusses those found during testing and how they were resolved.

Addressing Issues

This section deal with issues that were encountered during testing and offers potential solutions.

Reflective Water Crashes

This was an extremely consistent crash seen during testing, the reported crash looks something like:

```
Appname: linksmmiii.exe  AppVer: 22.10.2.41  Modename: linksmmiii.exe

ModVer: 22.10.2.41  Offset:  0029defc


Exception Information:

Code: 0xc0000005  Flags:0x00000000

Record 0x0000000000000000  Address: 0x000000000069defc
```

The cause of this crash turns out to be that the Nvidia patch has not been installed.

On the test system the correct linksMMIII.exe file has a date of March 31 2010.

Incorrect Rendering, Links Hang & Potential crash

Some courses showed scenes rendering with large patches of black in the main view. Areas of black were also seen in the Top Camera view. At times other video corruption was also seen, for example the golfer overlaying the hole previews, or each movement of the golf swing staying on the screen.

Invariably Links would at least hang a short time later and in some cases crash. These issues were observed with all Wine versions tested.

After an upgrade to Wine 10 some rendering issues appeared to be resolved however Links was still hanging on one particular course in the set being used for testing, so the newer wine version appears to have resolved some issues but not all.

The crash that has been seen at times is on some but not all systems with integrated Intel GPUs, it has not been noted under Nvidia, the crash is:

```
Appname: linksmmiii.exe  AppVer: 22.10.2.41  Modename: ucrtbase.dll

ModVer: 10.0.14393.2247  Offset:  000149b9


Exception Information:

Code: 0xc0000005  Flags:0x00000000

Record 0x0000000000000000  Address: 0x000000007acf49b9
```

Testing showed that we could resolve this issue with changes to the Wine registry.

The settings are in the registry location:

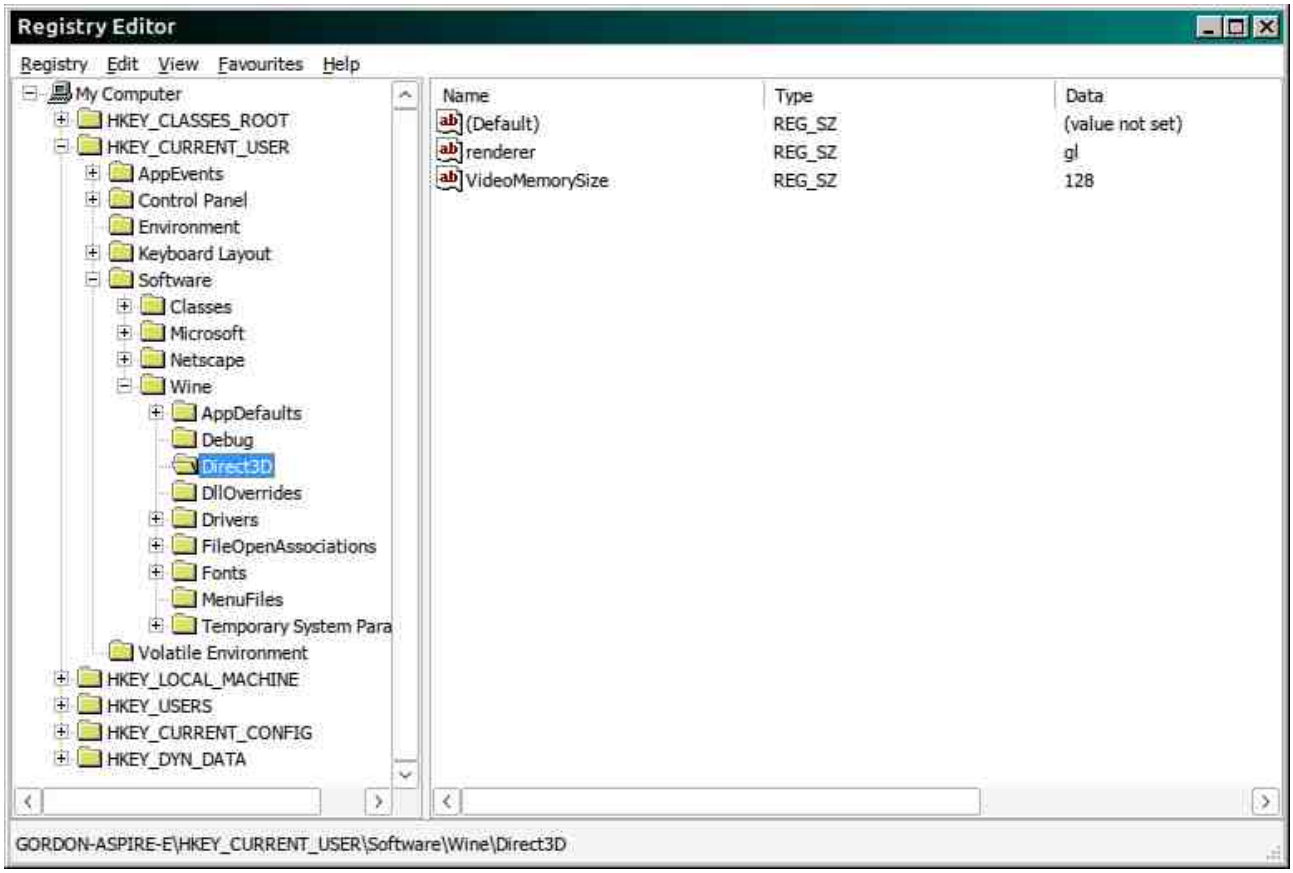
```
HKEY_CURRENT_USER
 Software
```

```
Wine
  Direct3D
```

With an embedded Intel GPU that uses shared memory it was found on a test system that initially limiting the amount of graphics memory in Wine to 128MB, no matter what amount of memory it has access, or is reserved for it, resolved the problems across various versions of Wine.

To do this add a new string value of VideoMemorySize, it will be a type of REG-SZ and it was set initially to 128.

It should look like:



Different versions of Wine and different machines impacted what setting could be used. Two different problems emerged at some settings. Video corruption would be seen then a hang, while at some settings a crash occurred. It was thought that setting the memory size to the amount of reserved memory for video may give a good result, but it did not necessarily follow. This is the integrated GPU and to determine the memory allocated the following commands were used.

To show the devices:

```
~:$ lspci
00:00.0 Host bridge: Intel Corporation Xeon E3-1200 v6/7th Gen Core Processor Host Bridge/DRAM
Registers (rev 02)
00:02.0 VGA compatible controller: Intel Corporation HD Graphics 620 (rev 02)
00:14.0 USB controller: Intel Corporation Sunrise Point-LP USB 3.0 xHCI Controller (rev 21)
00:14.2 Signal processing controller: Intel Corporation Sunrise Point-LP Thermal subsystem (rev 21)
00:15.0 Signal processing controller: Intel Corporation Sunrise Point-LP Serial IO I2C Controller #0
(rev 21)
00:16.0 Communication controller: Intel Corporation Sunrise Point-LP CSME HECI #1 (rev 21)
```

```
00:17.0 SATA controller: Intel Corporation Sunrise Point-LP SATA Controller [AHCI mode] (rev 21)
00:1c.0 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #1 (rev f1)
00:1d.0 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #9 (rev f1)
00:1d.2 PCI bridge: Intel Corporation Sunrise Point-LP PCI Express Root Port #11 (rev f1)
00:1d.3 PCI bridge: Intel Corporation Device 9d1b (rev f1)
00:1f.0 ISA bridge: Intel Corporation Sunrise Point-LP LPC Controller (rev 21)
00:1f.2 Memory controller: Intel Corporation Sunrise Point-LP PMC (rev 21)
00:1f.3 Audio device: Intel Corporation Sunrise Point-LP HD Audio (rev 21)
00:1f.4 SMBus: Intel Corporation Sunrise Point-LP SMBus (rev 21)
01:00.0 3D controller: NVIDIA Corporation GM107 [GeForce 940MX] (rev a2)
01:00.1 Audio device: NVIDIA Corporation GM107 High Definition Audio Controller [GeForce 940MX] (rev
ff)
02:00.0 Non-Volatile memory controller: Micron/Crucial Technology P2 NVMe PCIe SSD (rev 01)
03:00.0 Network controller: Qualcomm Atheros QCA9377 802.11ac Wireless Network Adapter (rev 31)
04:00.0 Unassigned class [ff00]: Realtek Semiconductor Co., Ltd. RTL8411B PCI Express Card Reader
(rev 01)
04:00.1 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL8111/8168/8411 PCI Express Gigabit
Ethernet Controller (rev 12)
```

The Intel video controller can be seen at 00:02.0, it can be interrogated as follows:

```
~:$ lspci -v -s 00:02.0
00:02.0 VGA compatible controller: Intel Corporation HD Graphics 620 (rev 02) (prog-if 00 [VGA
controller])
        Subsystem: Acer Incorporated [ALI] Aspire E5-575G
        Flags: bus master, fast devsel, latency 0, IRQ 135
        Memory at b2000000 (64-bit, non-prefetchable) [size=16M]
        Memory at c0000000 (64-bit, prefetchable) [size=256M]
        I/O ports at 5000 [size=64]
        Expansion ROM at 000c0000 [virtual] [disabled] [size=128K]
        Capabilities: <access denied>
        Kernel driver in use: i915
        Kernel modules: i915
```

The amount of pre-allocated Memory can be seen as 256MB, when this amount of memory was specified Links hung and/or crashed, it did not when the amount was set to 128MB in Wine as said.

On this specific machine it was found 160 - 192MB worked most of the time and seemed to improve performance, however it proved that 128MB gave the most stability. Yet on another machine, with embedded Intel graphics, we could set the Memory to 512MB or higher and on that machine and it worked well with improved performance.

A greater amount of memory assign to the video does not necessary mean better performance however, It has been noted, at times, that reducing the amount of memory assigned to the shared video can improve performances.

There is no rule of thumb that can be applied here, it is more trial and error. On the machine discussed here it appears, in early testing, that 160MB performs better than 192MB. Performance changes are more apparent on some courses than others, courses that are typically slow may show more dramatic differences.

In summary it can be very much trial and error to come to the optimum amount of shared memory to use for the graphics.

You can initially not set it explicitly which should mean “automatic” If you then see a crash, that is not resolved with the Nvidia patch, and with that crash we typically see a memory address of 69defc, then try setting the memory to 1024MB, if it still fails try 512MB, and so on. On the test machine 128MB

worked and 256MB failed, so it became a matter of scaling it up and down in that range to see if it results in any further improvement or indeed problems.

Disappearing Golfer/Incomplete Menus

A problem with a disappearing golfer and incomplete menus was seen when using some resolutions in the WXGA family.

The only resolution in this family that has been seen to work reliability is 1280 x 720 i.e. Mode 5. This worked well with Links set to full screen mode, so Links scaled the image to fit the screen.

Some resolutions in this family gave the Links error “A minimum setting of 800 x 600 is required” and Links would not start.

Links Reports an Incorrect HBA on its Starting Screen

It has been seen on test systems that Links was stating an incorrect HBA when starting. The HBA was in the same family, but an incorrect model.

No operational problems were observed but it was felt that it may be worthwhile to correct this. This situation can be corrected as followed.

In Step 6 above it was shown how to determine to the video Vendor ID and Device ID for use in Links’ Videocard.txt file. We can use this same information to add specific settings to Wine in order to set the correct HBA. Wine has the capability to get the information from the system but there are times when this does not work correctly.

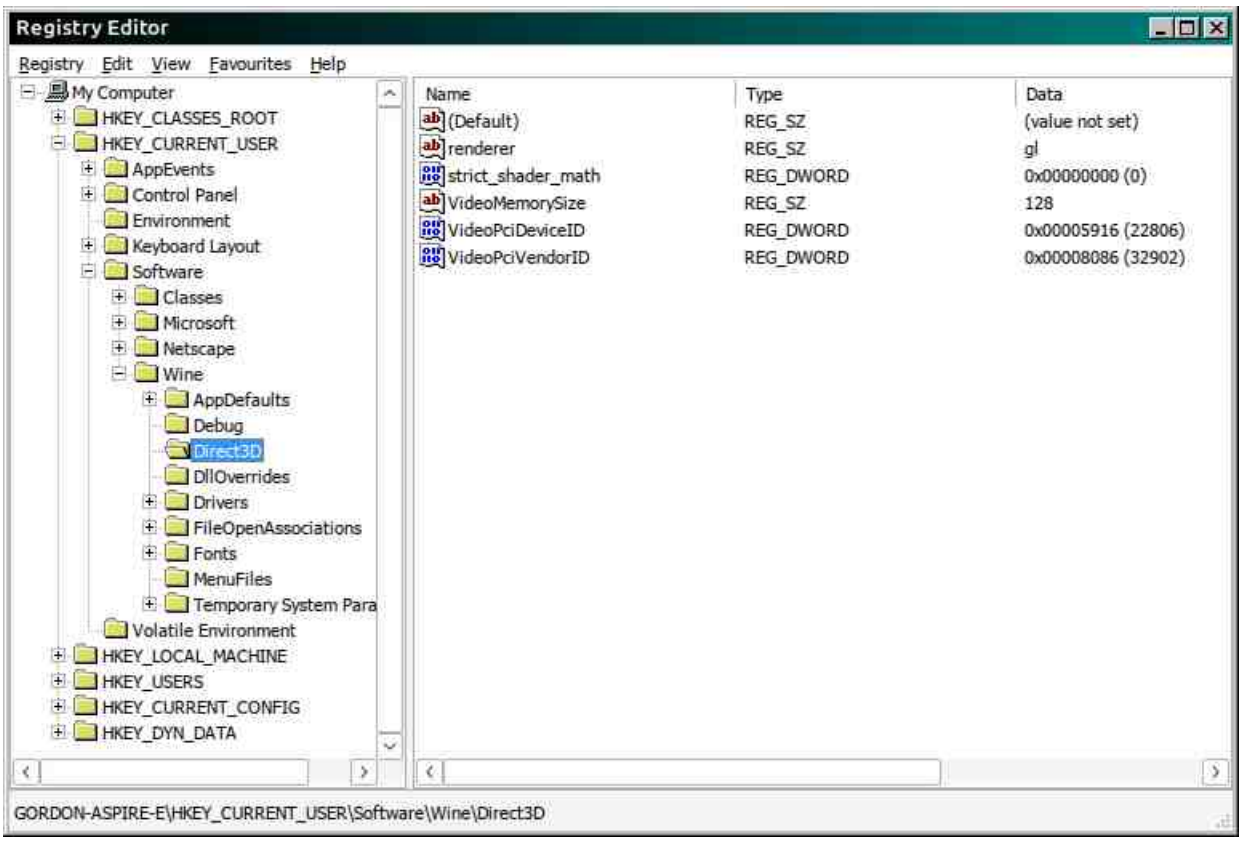
Within Wine’s Direct3D registry section, which was shown earlier, we can add an additional two entries. They are both of type DWORD are should be named:

VideoPciDeviceID
VideoPciVendorID

As an example in the case of the Intel HD 620 CPU above we would set these values as:

VideoPciDeviceID = 0x00005916
VideoPciVendorID = 0x00008086

So with this example the registry looks like:



This example also shows the Wine Video Memory Size workaround.

Direct3D Wrapper

It should not really be necessary to use the wrapper on Linux, Wine effectively has its own DirectX wrapper built in, and DirectX will be converted to OpenGL so any use of the wrapper should be redundant.

For completeness and, “just in case”, instructions have been included here.

Links has been tested in Linux both with and without the wrapper. The author’s preference is without, it makes for a simpler setup and as stated it should not be needed and in any case one of the objectives of Wine is to be able to run old Windows games so it has much of the core functionality of the wrapper built in as said.

However, it is acknowledged that it may be necessary, in certain circumstances, to try to use the wrapper to resolve issues, hence the following set up instructions are provided.

Direct3D is the specific dgVoodoo software version (V2.7) that is known to work with Links. Testing has shown that versions up to 2.81.3 work. The earliest version that has worked is 2.55.3.

The 2.7 version can be sourced via linkcorner.org.

As indicated a number of other versions have also been tested, the latest version does not work under Linux, the developer has no interest in Linux so I

am not expecting this to change, later versions also do not work with Links under Windows correctly either, they have water rendering issues. The author felt that Version 2.55 seemed to be particularly suitable for Linux, it is also a slightly lighter install, with only 3 DLLs, all of which are smaller than the 2.7 release and Links appears to run a little quicker.

The installation of this software is really the copying of a number of files and then performing a configuration as follows:

1. From the MS sub-folder, copy the x86 DLLs into your Links 2003 folder. By default, this is C:\Program Files (x86)\Microsoft Games\Links 2003 directory within the Wine folders.

   The ‘bitness’ of the DLLs needs to match the ‘bitness’ of Links, and of course Links 2003 is a 16/32 bit application.

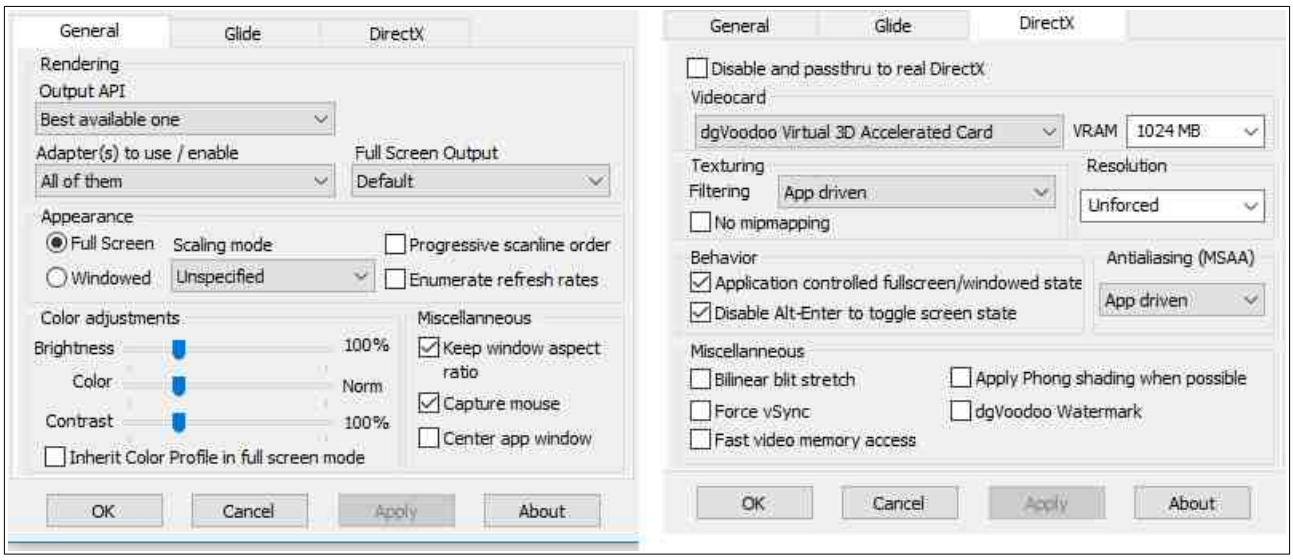
2. dgVoodoo reads the configuration from a file named dgVoodoo.conf. While dgVoodoo includes a configuration file, you should use the dgVoodoo Control Panel Application to at least modify, if not create, the configuration file. The Control Panel Application allows the setting of the content of the configuration files in a convenient GUI way, you can possibly have multiple configuration files perhaps one for each application.

   So, copy dgVoodooCpl.exe to your desktop or any other folder you prefer and you can start it with the command “wine dgVoodooCpl.exe”. For convenience it was placed it into the Links 2003 folder in order to keep all files together.

   You will see three different tabs with the settings as default. If you change something and press the OK or Apply buttons then dgVoodooCPL writes the current configuration into a configuration file. It saves this to the folder you select on the Config folder/running instance roll down menu at the top of the Control Panel.

   By default it chooses your login user’s appdata folder but you can add extra ones manually, it is recommended to add one and choose your game folder to create your configuration file there. Therefore for Links usage it is better to save the configuration file to the Links Directory, where it can live with the DLLs.

   So run dgvoodooCpl.exe to configure the wrapper. You will be configuring the settings on the General and Direct X tabs. Glide is of no relevance to Links. The relevant tabs are shown below:



There are a small number of settings we are specifically interested in

1. Deselect the dgVoodoo Watermark checkbox to prevents having a translucent imprint in the lower right of your screen that says "dgvoodoo."

2. We noted that a number of courses consistently crashed or hung usually accompanied by video corruption of some sort. This was resolved by setting the amount of VRAM to 128MB. Note that no parameters had been set in Wine. Using the Wine parameters discussed in the main body of this document may make this setting redundant.

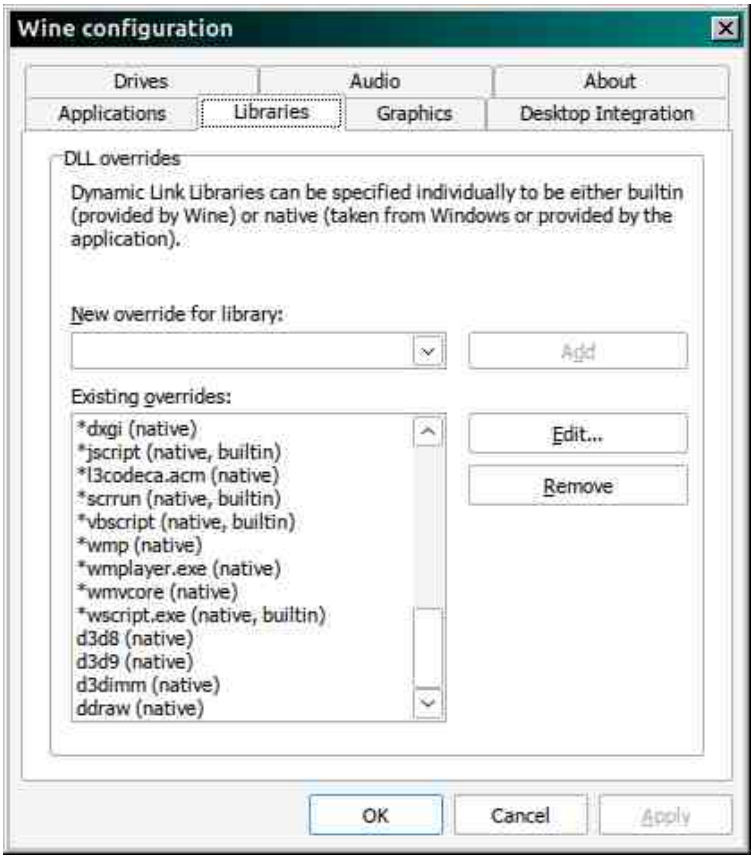
In addition, to allow Wine to use the Voodoo drivers, it is necessary to configure Wine to allow the use of the DLLs rather than its internal implementation.

1. Run winecfg, either directly or through winetricks.

2. You need to add overrides for the Voodoo DLLs:

   Navigate to the "Libraries" tab, type the DLL file names you copied (without the .dll extension, you only need to type the names of DLLs you copied, click "Add", find the added DLL names in the box below the one where you added the names, scroll down until you find the names, select the DLL, click "Edit", then change the "Load order" to "Native"; do this for every DLL you added. Save the changes once you're done.

   The window should look something like:



The DLLs in the kit are:

D3D8.dll
D3D9.dll
D3DImm.dll
DDraw.dll

Wine will list the DLL's in lowercase, and will not accept uppercase letters when entered, Windows of course is not case sensitive. For information Version 2.55 does not have a D3D9.dll file. In any case Links should not require the D3D8 and D3D9 DLL files.

Links should now start using the Wrapper and should report dgVoodoo as providing the 3D Acceleration on Link's loading screen. It is expected that Links will crash the first time it is run after installing the wrapper, just run it a second time.

For Information, other wrappers have been investigated, but to date none have proven to be viable.

With the Wrapper, Links Starts with a Black Screen

It has been observed on a system using an on board Intel GPU with the wrapper that when Links starts the sound track plays but what is displayed is just a black screen with some lines on it.

The lines appear to correspond with the underlines that appear under the short cut keys for the options on the main menu.

To date no solution has been found for the problem under this configuration.

On the same system, without the wrapper Links starts correctly, hence is the solution of choice. Different versions of the wrapper made no difference. The short cut keys do work, for example “X” to exit. On the same system running Windows 10 the wrapper worked as expected.

General Notes

The following points were noted during the installation and testing:

- Sometimes you need to restart links more than once after making changes. This was especially noted on the first starting of Links after making any video configuration changes.
- The native resolution can be problematic in some instances, hence testing with other resolutions can be needed.
- You can set Links to either use windowed mode or full screen mode which will scale images to fit the screen in need.
- When the chosen resolution is close to the native then Links scaling can work well.
- Translucent shadows work on Intel and Nvidia GPUs when not using the Direct3D fix.
- When not using the Direct3D fix adding graphic card specific entries to videocard.txt may be desirable.
- Getting reliability and good performance can be a matter of trial and error, trying different video solutions, in terms of driver, resolution and memory allocation. The native driver may work best In certain situations yet in others the wrapper could work. Under Linux you also have another layer involved in Wine, so Wine version can be a consideration.
- For testing a sample of about 18 courses were used. A mixture of 2001, 2003 and 2003T courses.
- To improve performance removing the entry in the registry that sets the sky to use can help i.e. make it blank in the registry. This is not unique to Links on Linux.
- The more cameras used the slower the rendering. Again this is not unique to Links on Linux.
- Some cameras did not display properly during testing, there were no problems with main and top cameras.
- Under Linux, it was found on the test systems, that Links could perform notably better with the Integrated Intel Graphics than with Nvidia HBAs or on-board Nvidia chipsets using proprietary drivers. Later drivers are generally “better” but even with the latest driver versions available that were tested (Version 580) it was noted that Links suffered unexplained “slowdowns” at times, this has been reported with other program as well.