Wine

Wine is a compatibility layer which translates Windows system calls into comparable Linux equivalents. This allows (most) Windows applications to run on Linux, including many games.

Wine can be installed from the repositories. Recommended additional components:

- `wine-gecko`: Components for displaying web content
- `wine-mono`: Components for running .NET applications
- `winetricks`: Install various tools and libraries in a Wine prefix

```
sudo pacman -S wine wine-gecko wine-mono winetricks
```

DXVK

Game performance can be significantly improved by installing *DXVK* in a Wine prefix. *DXVK* translates Direct3D calls from the DirectX 8/9/10/11 API to Vulkan to achieve improved 3D performance compared to WineD3D.

DXVK can be installed relatively easily in a Wine prefix using `winetricks`:

> **NOTE:** If you've set up Wine with a non-default prefix (i.e. your Wine "installation" does not reside under `~/.wine`) you will need to supply the path in an environment variable:
>
> ```
> WINEPREFIX=/path/to/your-prefix winetricks dxvk
> ```

> **WARNING:** DXVK overrides the DirectX 10 and 11 DLLs, which may be considered cheating in online multiplayer games, and may get your account **banned**. Use at your own risk!

```
winetricks dxvk
```

It's also possible to install a specific *DXVK* version, if needed:

```
winetricks dxvk1103
```

Alternatively, *DXVK* can also be installed via the AUR:

```
yay -S dxvk-bin
```

Install via the included helper program:

> **NOTE:** The same conditions about non-default prefix locations still apply:
>
> ```
> WINEPREFIX=/path/to/your-prefix setup_dxvk install --symlink
> ```

```
setup_dxvk install --symlink
```

This places symbolic links into the Wine prefix, which means when *DXVK* gets updated during system upgrades, all Wine prefixed are updated along with it.

VKD3D

VKD3D is the translation layer for Direct3D 12 to Vulkan. The latest version is installable through `winetricks`:

> **NOTE:** The same conditions about non-default prefix locations still apply:
>
> ```
> WINEPREFIX=/path/to/your-prefix winetricks vkd3d
> ```

```
winetricks vkd3d
```

VKD3D is also available from the AUR:

```
yay -S vkd3d-proton-bin
```

> **NOTE:** The same conditions about non-default prefix locations still apply:
>
> ```
> WINEPREFIX=/path/to/your-prefix setup_vkd3d_proton install --symlink
> ```

```
setup_vkd3d_proton install --symlink
```

This way when you upgrade to a new version of *VKD3D*, every prefix automatically gets updated as well.

Synchronization primitives

Games heavily rely on Windows synchronization primitives for multi-threaded workloads. Since Linux kernel version 6.14 the NTSync kernel module is available, which more closely resembles Windows synchronization primitives. Wine 10.16 and later automatically use NTSync when it is detected to improve performance in CPU-bound scenarios.

To load the kernel module at boot create a file in `/etc/modules-load.d/` with the content `ntsync`:

```
echo ntsync | sudo tee /etc/modules-load.d/ntsync.conf
```

MIDI Playback

Some Windows games still use MIDI playback for music. In order for this to work in Wine, a sequencer has to be installed, e.g. `fluidsynth`:

> **NOTE:** FluidSynth uses soundfonts to render MIDI music.

```
pacman -S fluidsynth soundfont-fluid
```

FluidSynth comes with a systemd user unit to run it in daemon mode. Edit the file `/etc/conf.d/fluidsynth` and uncomment the lines with the environment variables. Point the `SOUND_FONT` variable to a soundfont file in `*.sf2` format (refer to DOSBox for a list of available soundfonts for installation). Furthermore, adjust the `OTHER_OPTS` variable to use the appropriate audio backend that you are using, e.g. set parameter `-a pipewire` if you're using PipeWire instead of PulseAudio:

```
# Mandatory parameters (uncomment and edit)
SOUND_FONT=/usr/share/soundfonts/FluidR3_GM.sf2

# Additional optional parameters (may be useful, see 'man fluidsynth' for further info)
OTHER_OPTS='-a pipewire -m alsa_seq -p FluidSynth\ GM -r 48000'
```

After you've set everything up, enable/start the systemd user unit with:

> **ATTENTION:** Enable/start the unit as regular user, i.e. do not use `sudo`!

```
systemctl --user enable --now fluidsynth
```

Revision #4
Created 2023-03-26 03:01:23 UTC by Sebin
Updated 2026-03-15 22:00:20 UTC by Sebin