



- WineHQ (<https://www.winehq.org/>)
- Wiki (<http://wiki.winehq.org/>)
- AppDB (<//appdb.winehq.org/>)
- Bugzilla (<//bugs.winehq.org/>)
- Forums (<//forums.winehq.org/>)



(<https://www.winehq.org/>)

WINE^{HQ}

(<https://www.winehq.org/>)




Search

- Page (/FAQ)
- Discussion (/Talk:FAQ)
- View source (/index.php?title=FAQ&action=edit)
- History (/index.php?title=FAQ&action=history)

FAQ

From WineHQ Wiki

Translations of this page:  Français (/FAQ_fr) ; Not all ported. Translators, please see Discussion page.



(/File:Edit-find-replace.svg) This page has recently undergone extensive revision.

External links to specific sections of this page may now be broken. Consult the table of contents for current information.

This FAQ, or **F**requently **A**sKed **Q**uestions, covers general topics about Wine. For questions related to Wine software development, see the Developer FAQ (/Developer_FAQ).

Quick links: running a program in Wine, running Wine from a terminal, running program as root, uninstalling an application, getting a debugging log, using a wineprefix, creating a 32 bit wineprefix.

Contents

- 1 General questions
 - 1.1 Who is responsible for Wine?
 - 1.2 Does Wine hurt Linux or other free operating systems?
 - 1.3 Is Wine an emulator? There seems to be disagreement
 - 1.4 What is the difference between Wine, CrossOver, and Cedega?
 - 1.5 How does the Wine version numbering system work?
 - 1.6 Do I have to use the command line?
 - 1.7 What applications run well with Wine?
 - 1.8 How can I help contribute to the Wine project, and in what ways?
 - 1.9 I think I've found a bug. How do I report this to WineHQ?
 - 1.10 Where can I get further help?
- 2 Installing Wine
 - 2.1 What are the system requirements for Wine?
 - 2.1.1 Does Wine run on all Unix filesystems?
 - 2.1.2 Will Wine run only under X?
 - 2.2 Which version of Wine should I use?
 - 2.3 How do I install Wine?
 - 2.4 I have a problem installing my distro's Wine package and need help
 - 2.5 Can I install more than one Wine version on my system?
 - 2.6 Is there a 64 bit Wine?
 - 2.7 How can I download older versions of Wine for Ubuntu?
 - 2.8 How do I install Wine on my netbook (eeePC, Acer Aspire One, etc.)?
 - 2.9 Installing on Apple
 - 2.9.1 How do I install Wine on my Mac?
 - 2.9.2 Can I use Wine on an older Mac without an Intel chip?
- 3 Compiling Wine
 - 3.1 How do I compile Wine from source?
 - 3.2 How do I apply a patch?
- 4 Uninstalling
 - 4.1 How do I uninstall Wine?
 - 4.2 How do I uninstall individual Windows applications?
 - 4.3 How do I clean the Open With List?
 - 4.4 How do I wipe the virtual Windows installation?
- 5 Installing and Running Windows Applications
 - 5.1 I have lots of applications already installed in Windows. How do I run them in Wine?
 - 5.2 How do I run an installer using Wine?
 - 5.3 How do I run an application once I've installed it?
 - 5.4 How should I start Windows programs from the command line?
 - 5.5 How do I pass command line arguments to a program?
 - 5.6 How do I install/run a .msi file?
 - 5.7 How do I install/run a ClickOnce (.application) file?
 - 5.8 How do I launch native applications from a Windows application?
 - 5.9 My installer tells me I don't have enough free disk space
 - 5.10 When I double-click on a .exe file in my file manager, nothing happens
 - 5.11 I double-clicked on an .exe file, but the system said "The file foo.exe is not marked as executable..."

■ 6 Using Wine

- 6.1 How do I run Wine?
- 6.2 Should I run Wine as root?
- 6.3 I ran wine with sudo or as root. How do I fix my permission errors?
- 6.4 How do I know what version of Wine I have?
- 6.5 Wineprefixes
 - 6.5.1 Where is my C: drive?
 - 6.5.2 Can I store the virtual Windows installation somewhere other than ~/.wine?
 - 6.5.3 How do I create a 32 bit wineprefix on a 64 bit system?
 - 6.5.4 Why aren't versions of Windows prior to XP available in 64 bit wineprefixes?
 - 6.5.5 How can I run two programs as if they were on different computers?
- 6.6 Configuring
 - 6.6.1 How do I get Wine to launch an application in a virtual desktop?
 - 6.6.2 How do I edit the Wine registry?
 - 6.6.3 How do I associate a native program with a file type in Wine?
 - 6.6.4 How can I prevent Wine from changing the filetype associations on my system or adding unwanted menu entries/desktop links?
 - 6.6.5 How do I disable the GUI crash dialog?
 - 6.6.6 How can I make Wine fonts anti-aliased?
 - 6.6.7 How do I change the DPI (font size)?
 - 6.6.8 How do I configure a proxy?

■ 7 Limitations and Risks

- 7.1 Can Wine run in character mode?
- 7.2 Can I install applications to be shared by multiple users?
- 7.3 Can I use Wine to install drivers for my hardware?
- 7.4 Is Wine malware-compatible?
- 7.5 How good is Wine at sandboxing Windows apps?

■ 8 Applications

- 8.1 What applications/services conflict with Wine?
- 8.2 Does Wine support .NET? Should I install native .NET in Wine?
- 8.3 Does Wine support DirectX? Should I install Microsoft's DirectX under Wine?
- 8.4 Does Wine support Internet Explorer? Should I install Microsoft's Internet Explorer in Wine?
- 8.5 Why doesn't Wine work well with some copy protection?
- 8.6 I followed a howto I found on a website, and the application still doesn't work
- 8.7 I used a third party application (PlayOnLinux, WineBottler, etc.) to install or manage applications in Wine and need help
- 8.8 My application worked with an older version of Wine, but now it's broken!
- 8.9 I really need this particular application to work. How can I go about making this happen?
- 8.10 I'm writing a Windows app. How can it detect if it's running under Wine?

■ 9 Performance

- 9.1 My 3D application/game is very slow (FPS)
- 9.2 Is Wine slower than just using Windows?
- 9.3 I get lots of "fixme:" messages in the terminal and Wine runs a bit slow
- 10 Troubleshooting
 - 10.1 General
 - 10.1.1 How can I get a debugging log (a.k.a. terminal output)?
 - 10.1.2 How do I get a debug trace?
 - 10.2 Crashes and Freezes
 - 10.2.1 My program froze up, how do I close it?
 - 10.2.2 My whole computer freezes, reboots, or shuts off when I run my game in Wine!
 - 10.2.3 Every app I try to start crashes
 - 10.3 Warning and Error Messages
 - 10.3.1 My application says some DLL or font is missing. What do I do?
 - 10.3.2 Too many open files, ulimit -n probably needs to be increased
 - 10.3.3 preloader: Warning: failed to reserve range 00000000-60000000 or winevdm: unable to exec '*application name*': DOS memory range unavailable
 - 10.3.4 Failed to use ICMP (network ping), this requires special permissions
 - 10.3.5 err:virtual:map_image failed to set 60000020 protection on section .text, noexec filesystem?
 - 10.3.6 Broken NVIDIA RandR detected, falling back to RandR 1.0
 - 10.4 Graphics
 - 10.4.1 My application complains about being unable to change the resolution or color depth
 - 10.4.2 The application I am trying to run complains that it needs 256 colors but I have millions!
 - 10.4.3 My X screen won't go back to its normal resolution after running a game fullscreen
 - 10.4.4 I'm using Desktop Effects with Compiz, Fusion, or XGL and get poor performance/odd messages/broken applications
 - 10.4.5 Graphics in games with good ratings in AppDB are scrambled
 - 10.4.6 Wine displays corrupted or missing text
 - 10.4.7 Wine's windows and fonts are extremely large
 - 10.5 Sound
 - 10.5.1 Audio in games sounds different from running in Windows
 - 10.5.2 MP3s do not play in Windows Media Player or applications that depend on it
 - 10.5.3 Bad or no sound on systems using PulseAudio
 - 10.6 Networking
 - 10.6.1 My program fails to do networking, but my other applications can get online
 - 10.6.2 Why doesn't DNS resolve in 64-bit operating systems?
 - 10.7 Removable Media
 - 10.7.1 My CD or DVD disc won't eject
 - 10.7.2 The application's CD or DVD looks empty or is missing some

files!

- 10.8 Input Devices
 - 10.8.1 Some key combinations in my application do not work
 - 10.8.2 My mouse jumps around/resets its position often in games
- 10.9 Miscellaneous
 - 10.9.1 My application runs, but text areas act strangely (e.g. lines don't wrap, double-clicking doesn't select words)
 - 10.9.2 I deleted my Wine menu, and now I can't get it back
 - 10.9.3 16-bit applications fail to start
 - 10.9.4 Input doesn't work over SSH

1 General questions

1.1 Who is responsible for Wine?

Wine is available thanks to the work of many people around the world. Some companies that are or have been involved with Wine development are CodeWeavers (<http://www.codeweavers.com>), Bordeaux (<http://bordeauxgroup.com/>), TransGaming, Corel, Macadamian and Google. See Acknowledgements (/Acknowledgements) and Wine History (/Wine_History).

1.2 Does Wine hurt Linux or other free operating systems?

Wine increases the usefulness of Linux, makes it easier for users to switch to free operating systems, and for Windows developers to make applications that work on them. This greatly raises Linux marketshare, drawing more commercial and community developers to Linux.

1.3 Is Wine an emulator? There seems to be disagreement

There is a lot of confusion about this, particularly caused by people getting Wine's name wrong and calling it WINdows Emulator.

When users think of an emulator, they tend to think of things like game console emulators or virtualization software. However, Wine is a compatibility layer (http://en.wikipedia.org/wiki/Compatibility_layer) - it runs Windows applications in much the same way Windows does. There is no inherent loss of speed due to "emulation" when using Wine, nor is there a need to open Wine before running your application.

That said, Wine can be thought of as a Windows emulator in much the same way that Windows Vista can be thought of as a Windows XP emulator: both allow you to run the same applications by translating system calls in much the same way. Setting Wine to mimic Windows XP is not much different from setting Vista to launch an application in XP compatibility mode.

A few things make Wine more than just an emulator:

- Sections of Wine can be used on Windows. Some virtual machines use Wine's OpenGL-based implementation of Direct3D on Windows rather than truly emulate 3D hardware.
- Winelib can be used for porting Windows application source code to other operating systems that Wine supports to run on any processor, even processors

that Windows itself does not support.

"Wine is not just an emulator" is more accurate. Thinking of Wine as just an emulator is really forgetting about the other things it is. Wine's "emulator" is really just a binary loader that allows Windows applications to interface with the Wine API replacement.

1.4 What is the difference between Wine, CrossOver, and Cedega?

Wine is the base of the project, where most of the work is being done. Wine is not perfect, but tens of thousands (/Usage_Statistics) of people nevertheless use "vanilla" Wine successfully to run a large number of Windows programs.

CrossOver XI (formerly CrossOver Office) is a product made by a company called CodeWeavers that is based directly on Wine with a few tweaks and proprietary add-ons. Unlike the biweekly Wine releases, CrossOver releases are rigorously tested for compatibility with CodeWeavers' supported applications in order to prevent "regressions". CodeWeavers employs a large proportion of the Wine developers and provides a great deal of leadership for the project. All improvements to Wine eventually work their way into CrossOver.

Cedega (formerly WineX) is a product from a company called TransGaming. TransGaming based their product on Wine back in 2002 when Wine had a different license, closed their source code, and rebranded their version as specialized for gamers. In the years since Cedega was originally created from Wine, development on Wine and Cedega have continued mostly independently. TransGaming currently gives back very little code to Wine. Cedega is **not** "Wine with more gaming support" - because Wine has had years of development since Cedega was made, and many games actually run better under Wine than under Cedega. Currently, Wine has more advanced Direct3D support than Cedega, but Cedega still has more advanced copy protection support due to TransGaming's licensing of (closed source) code from a handful of copy protection companies. Unlike CrossOver, most improvements to Wine *don't* get into Cedega due to the license differences between Cedega and Wine.

For more information, see Wine History (/Wine_History).

1.5 How does the Wine version numbering system work?

See Wine_User's_Guide#Wine_from_WineHQ (/Wine_User%27s_Guide#Wine_from_WineHQ).

1.6 Do I have to use the command line?

Normally you do not have to use the command line to use Wine in Linux. You can use a graphical interface for most things, much like on Windows. In many cases you can right-click an installer and select "Open with Wine", or just double-click it. You can start installed programs using the shortcut icon or menu.

However, there are several situations when you might need to use the command line. The most common reason is to get debug output when your program does not run properly. You might also want to use utilities such as `regedit` that do not have menu shortcuts (note that you can create them using whatever tools your DE provides).

This does not hold true for Mac OS X Wine usage, which is all command line currently, unless you use a third party application (/Third_Party_Applications).

1.7 What applications run well with Wine?

Thousands of applications work well. As a general rule, simpler or older applications tend to work well, and the latest versions of complex applications or games tend to not

work well yet. See the Wine Application Database (<http://appdb.winehq.org>) for details on individual applications. If your application is rated Silver, Gold or Platinum, you're probably okay; if it's rated Bronze or Garbage, Wine isn't really ready to run it for most users. If there aren't any reports using a recent version of Wine, however, your best bet is to simply try and see. If it doesn't work, it probably isn't your fault, Wine is not yet complete. Ask for help on the forum if you get stuck.

1.8 How can I help contribute to the Wine project, and in what ways?

You can contribute programming or documentation skills, or monetary or equipment donations, to aid the Wine developers in reaching their goals.

One area where every Wine user can contribute to this project is by sending high quality bug reports to our Bugzilla (<http://bugs.winehq.org/>) and helping the developers with any followup questions that they may have about your bug. It is impossible and impractical for a developer to have a copy of every program on the market, so we need your help even after your initial bug report. If a developer has a good idea what might be causing the bug, he or she may ask if you can try a patch and see if it fixes the problem. If the patch works and then makes its way into our main development tree, the bug report will be closed, your help will be appreciated by everyone and your problem will be fixed.

For a list of ideas of how you can help, please consult the Contribute (/Main_Page#Contribute) section of the main page.

1.9 I think I've found a bug. How do I report this to WineHQ?

Bug reports should be submitted to our online Bugzilla system (<http://bugs.winehq.org/>). To increase developer productivity and facilitate a resolution to submitted bugs, **please read the Wiki article on Bugs (/Bugs)**. A poor bug report may be marked INVALID and closed, leaving you no closer to resolving your problem. High quality bug reports are an essential part of making Wine better.

Please note that you should *generally* avoid submitting bug reports if you have used any third party applications (/Third_Party_Applications) or native DLL overrides.

1.10 Where can I get further help?

In addition to this wiki (/Main_Page), check the Wine HQ Documentation (<http://www.winehq.org/documentation>) and the users forum (<http://forums.winehq.org/>). If you're an ISV looking at porting an application with Winelib you can also try wine-devel (<http://forums.winehq.org/>).

For help with a specific application, search the Application Database (<http://appdb.winehq.org>), a place where users share their experiences by submitting test data, tips and tricks, and asking questions.

The IRC channel: #WineHQ (<irc://irc.freenode.net/#winehq>) on irc.freenode.net. Knowledgeable Wine users hang out there, and often developers will lurk there too. See IRC (<https://www.winehq.org/irc>) for more important information.

2 Installing Wine

2.1 What are the system requirements for Wine?

Wine, along with the operating system you use to run it, generally requires less disk space and memory than Windows itself. If you're not currently running a Windows application, Wine won't consume any resources at all other than about 20 megabytes of

disk space.

Ideally, if an application runs fine in Windows, it should run fine on the same hardware using Wine, *provided native drivers for your hardware are installed and equivalent to the Windows drivers*. Open source Linux graphics drivers in particular are often inadequate to run games that work fine on the same hardware in Windows. If there is no native driver for your hardware, Wine will not be able to use it.

2.1.1 Does Wine run on all Unix filesystems?

Mostly. Wine is written to be filesystem independent so MS Windows applications should work on virtually any full-featured UNIX filesystem. The key exception is that not all filesystems / drivers support every feature of fat32 or NTFS. One example is that the ntfsv3 drivers do not support shared-write mmap, a feature that cannot be emulated and is used by applications such as Steam.

One other point is that Wine is a weird application in ways and some programs work *better* on case-insensitive filesystems (see Case Insensitive Filenames (/Case_Insensitive_Filenames) for more details).

2.1.2 Will Wine run only under X?

Until recently with projects such as Wayland, serious alternatives to x11drv weren't even on the horizon so development has focused on X. However, Wine's interface with the graphics driver is designed to be abstract so supporting future graphics systems will hopefully be straight-forward.

2.2 Which version of Wine should I use?

Short answer: Use the version that works best with the particular applications you want to run. In most cases, this will be the latest development version; however, in some cases it may take some experimenting to find it.

Longer answer: Wine development is rapid, with new releases in the development branch every two weeks or so. Functionality will usually be best with the most recent development version, however, there are cases where changes to existing code in Wine cause applications that worked well in older versions to not work in the new one (these are called regressions), as well as problems caused by the introduction of new, but as-yet-incomplete and untested, functions.

A good rule of thumb is to start with the version of Wine installed with your distro and see if that works with the applications you want to use. If it does, good! If it doesn't, upgrade. In most cases the upgrade should be to the latest development version, but it is a good idea to check Bugzilla and the AppDB for any known regressions and/or new bugs. If you find an existing bug marked STAGED, this means there is a patch for the problem in wine-staging (the experimental branch), and you should try the latest version in that branch. If there are known bugs without a STAGED patch or easy workaround, upgrade to the most recent version of Wine known to work for your application.

While Wine does have a "stable" branch, the term "stable" refers to the branch as a whole, which is infrequently updated, and (for the minor stable releases) only with bugfixes promised not to break functionality. Users of a development or staging release can achieve the same degree of stability by simply not upgrading. Note that user support for the stable branch is limited to the ability to file AppDB test reports. Users who ask for help on the forum/IRC or file bug reports for the stable branch will be asked to retest in the current development release.

The current stable, development, and staging releases are listed on the WineHQ home page (<http://www.winehq.org>). See Wine User's Guide#Wine from WineHQ (/Wine_User%27s_Guide#Wine_from_WineHQ) for a description of the three branches and the version numbering system.

2.3 How do I install Wine?

- Use a precompiled binary package for your operating system/distribution (see the Wine download page (<https://www.winehq.org/download>) for links and additional information).
- Compile Wine from source (/Building_Wine) - if you can't find an up-to-date package for your OS/distribution

2.4 I have a problem installing my distro's Wine package and need help

WineHQ only supports the binary packages that we build (available here (<http://www.winehq.org/download>), only the ones listed under 'WineHQ Binary Packages'). Consult your distro's support channels for help using your package manager and interpreting any error messages you may be receiving if you're having problems with distro packages. If you are an experienced user and believe there is a problem with the package itself and/or the repository, please report it to your distro's Wine package maintainer.

2.5 Can I install more than one Wine version on my system?

Yes, but you will have to build Wine yourself (see Building Wine (/Building_Wine)), as it is not possible to have multiple distro packages installed. The easiest way to do this is to run Wine from the build directory (don't do `make install`). If you want to actually install multiple versions, use `--prefix` when building Wine to designate a different install directory for each version, e.g.

```
./configure prefix=path_to_install_directory && make
```

then install it with

```
sudo make install
```

On Linux, even this step is not enough: you must also set environment variables so that the wine executable on your `$PATH` finds the right shared libraries. Assuming that the path to install directory is `$W`, then for Wine version 1.4, the following settings appear to be sufficient:

```
export WINEVERPATH=$W
export PATH=$W/bin:$PATH
export WINESERVER=$W/bin/wineserver
export WINELOADER=$W/bin/wine
export WINEDLLPATH=$W/lib/wine/fakedlls
export LD_LIBRARY_PATH="$W/lib:$LD_LIBRARY_PATH"
```

The `LD_LIBRARY_PATH` is definitely required.

Note that regardless of whether you install multiple versions or run them from the build directory, you will still have to designate which version of Wine you wish to use when running applications. It is also recommended that applications being run with different Wine versions be installed into separate wineprefixes.

2.6 Is there a 64 bit Wine?

Yes. 64 bit Wine has been available on Linux since 1.2, and most major distros now package it for users. Normally, installation should be as simple as installing the Wine package for your distribution through your package manager. Check the Downloads

(<http://www.winehq.org/download>) page. If you are building Wine from source, see Building Wine (/Building_Wine) for instructions on how to build 32 bit Wine on a 64 bit system and instructions on how to build 64 bit Wine in a shared WoW64 setup.

A few things to note:

- 32 bit Wine runs on both 32-bit and 64-bit Linux/Unix installations. 16-bit and 32-bit Windows applications will run on it.
- 64-bit Wine runs only on 64 bit installations, and so far has only been extensively tested on Linux. It requires the installation of 32 bit libraries in order to run 32 bit Windows applications. Both 32-bit and 64-bit Windows applications (should) work with it; however, there are still many bugs (<http://bugs.winehq.org/buglist.cgi?keywords=win64&resolution=--->).
- Current Wine includes support for 64 bit Wine on Mac OS X; however, this has not been tested very much, and some applications may never work due to an ABI incompatibility between Win64 and OS X (<https://www.winehq.org/wnn/364#Wine64%20on%20Mac%20OS%20X>).

2.7 How can I download older versions of Wine for Ubuntu?

Some old versions of the distro packages are archived at <http://wine.budgetdedicated.com/archive/binary/> (<http://wine.budgetdedicated.com/archive/binary/>). Please note that since these are not WineHQ packages, they are not supported here.

2.8 How do I install Wine on my netbook (eeePC, Acer Aspire One, etc.)?

If you have replaced the customized distro that came preinstalled on your netbook (Xandros, Linpus, etc.) with one of the mainstream distros that provide up-to-date Wine packages, you should be able to install Wine as normal for that distro.

If you are still using Xandros (eeePC), Linpus (Acer Aspire One (<http://www.aspireoneuser.com/forum/>)) or any other customized distro, you will have to ask on your netbook's support forum. Only other users of those distros can advise you on what, if any, binary packages will work on your system, where to find them, and how to install them.

You can also try building Wine from source following the instructions in Building Wine (/Building_Wine), but you will still need to consult your netbook's support forum regarding satisfying dependencies on your particular system.

2.9 Installing on Apple

2.9.1 How do I install Wine on my Mac?

- **WineHQ Packages (/MacOSX)** are available for Mac OS X 10.8 and higher.
- Build and install Wine using Homebrew (<http://brew.sh>), MacPorts (<http://www.macports.org>), or Fink (<http://www.finkproject.org>) to install Wine. All support the current releases of OSX. The MacPorts (<http://www.macports.org>) installation of Wine will automatically install any necessary Dependencies for a Wine installation.
- If this is too complicated, there are several third party apps (/Third_Party_Applications) you can use like Codeweavers' CrossOver Mac (<http://www.codeweavers.com/products/cxmac/>).
- **Linux:** If you are running Linux on your Mac, installing Wine is as simple as

installing it under Linux on a PC. Simply visit the downloads page (<http://winehq.org/download>).

2.9.2 Can I use Wine on an older Mac without an Intel chip?

No, not even in Linux. Older Macs used PowerPC processors are incompatible with code compiled for x86 (Intel and AMD) processors, unless the code is run under CPU emulation. Wine Is Not a (CPU) Emulator, nor does it include one. The Darwine project was an effort to do just that, but it has not been maintained in many years.

3 Compiling Wine

3.1 How do I compile Wine from source?

See Building Wine (/Building_Wine).

3.2 How do I apply a patch?

You have to build Wine from source; see above.

4 Uninstalling

4.1 How do I uninstall Wine?

Uninstalling Wine itself will not revert your Wine settings or uninstall your Windows apps, which are permanently stored in your user's home directory. Do not uninstall Wine if you only wish to remove all of your settings and apps. For instructions on removing your Wine settings and apps, see #How do I wipe the virtual Windows installation?

- If you installed Wine with your distribution's package manager, use it again to uninstall Wine.
- If you installed Wine from source, use `sudo make uninstall` in the source directory to remove it.

4.2 How do I uninstall individual Windows applications?

You can run Wine's uninstaller command; this is like Windows' "Add/Remove Programs" function. To uninstall 64 bit applications, including wine-mono, you need to run it with wine64. The uninstaller should remove menu and desktop entries... but it's not well tested; it might not work with all apps. See below for a reliable way to remove **all** Windows apps.

4.3 How do I clean the Open With List?

To clean **Open With List**, please **carefully** paste the following commands into a terminal:

```
rm -f ~/.local/share/mime/packages/x-wine*
rm -f ~/.local/share/applications/wine-extension*
rm -f ~/.local/share/icons/hicolor/*/*/application-x-wine-extension*
rm -f ~/.local/share/mime/application/x-wine-extension*
```

4.4 How do I wipe the virtual Windows installation?

You can remove your virtual Windows installation and start from scratch by deleting the hidden `.wine` directory in your user's home directory. This will remove all of your Wine settings and Windows applications. The simplest and safest way to do this is through your file manager. Simply set your file manager to show hidden files, browse to your home directory, and delete `.wine` the same way you would any other directory. If you want to keep it as a backup, you can rename or move it instead. To the host system, a `wineprefix` is just another directory that can be deleted, moved, renamed, etc., the same as any other directory.

If you prefer to do it from the command line, **carefully** paste the following commands into a terminal:

```
cd
rm -rf .wine
```

To rename it from the command line instead of deleting it:

```
mv ~/.wine ~/.wine-old
```

Your Windows applications, though deleted, will remain in your system menu (remaining desktop files and icons are located in `~/.local/share`).

To remove these leftover menu entries, **carefully** paste the following commands into a terminal:

```
rm -f ~/.config/menus/applications-merged/wine*
rm -rf ~/.local/share/applications/wine
rm -f ~/.local/share/desktop-directories/wine*
rm -f ~/.local/share/icons/????_*.{xpm,png}
rm -f ~/.local/share/icons/*-x-wine-*.{xpm,png} }}
```

Alternatively, you can stop Wine from creating menu entries in the first place.


5 Installing and Running Windows Applications

5.1 I have lots of applications already installed in Windows. How do I run them in Wine?

Short answer: you have to install them in Wine (`/Wine_User%27s_Guide#How_to_install_and_run_Windows_programs`) just like you did in Windows. Applications usually have a setup or installer program.

Long answer: some applications can be copied from Windows to Wine and still work, but don't try this unless you like tinkering under the hood of your car while it's running.

Wine is not designed to interact with an existing Windows installation. If you have any data you need from a Windows installation, browse your Windows filesystems in your normal file manager and copy the data to another location.

 Do not try to configure Wine to point to your actual Windows `C:\` drive. **This will break Windows and require a Windows reinstall.** We have tried to make this hard to do, so you probably cannot do it by accident. If you do manage this, Wine may or may not continue to operate, but your Windows install will be 100% dead due to critical parts of it being overwritten. The only way to fix Windows after this has happened is to reinstall it.

5.2 How do I run an installer using Wine?

See Wine User's Guide#How to install and run Windows programs (/Wine_User%27s_Guide#How_to_install_and_run_Windows_programs).

5.3 How do I run an application once I've installed it?

See Wine User's Guide#How to install and run Windows programs (/Wine_User%27s_Guide#How_to_install_and_run_Windows_programs)

5.4 How should I start Windows programs from the command line?

See Wine User's Guide#How to run Windows programs from the command line (/Wine_User%27s_Guide#How_to_run_Windows_programs_from_the_command_line).

5.5 How do I pass command line arguments to a program?

See Wine User's Guide#Passing Windows command-line arguments (/Wine_User%27s_Guide#Passing_Windows_command-line_arguments)

5.6 How do I install/run a .msi file?

See Wine User's Guide#Running .msi files (/Wine_User%27s_Guide#Running_.msi_files).


5.7 How do I install/run a ClickOnce (.application) file?

Use winetricks to install whatever version of .NET the program requires, cd to the directory containing the .application file, and run it with

```
wine start whatever.application
```

5.8 How do I launch native applications from a Windows application?

First, run regedit and add a dot character to the end of the list of extensions in HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\Environment\PATHEXT.

 Note that this change will have to be made every time you upgrade Wine, as it will be reverted whenever the wineprefix is updated.

You can then start native applications using **wine cmd** if you specify the full path or use the shell, e.g.:

```
wine cmd /c /usr/bin/glxgears
```

or

```
wine cmd /c /bin/sh -c glxgears
```

You might also need winepath to translate the filename from Windows format to Linux

format (see [How do I associate a native program with a file type in Wine?](#)).

5.9 My installer tells me I don't have enough free disk space

Usually, you really don't have enough free disk space. Wine's C: drive is located in your home directory. Whatever partition contains /home must have enough free space for your program. You can check by running:

```
df -h ~
```

If the amount of space you need is less than 1 GB, and df reports you have more than 1 GB available, try setting the Windows version to Windows 98. This will work around bugs in some old (Windows 98 era) installers that could not cope with large drives.

If you want to use a different partition that has more free space, use a wineprefix that is located on the other partition. Note that your other partition must be a Unix filesystem: FAT and NTFS partitions will not work. See [#How can I run two programs as if they were on different computers?](#) for instructions.

5.10 When I double-click on a .exe file in my file manager, nothing happens

Note: if you can, start applications by clicking on the application's icon in the Applications / Wine menu or desktop instead. Double-clicking .exe files is typically only needed for applications that aren't installed yet, e.g. to run the setup.exe on a CD-ROM game or a downloaded installer.

If double-clicking doesn't work, you might need to right-click the file and choose "Run with Wine". It depends on your file manager. If that also doesn't work, contact whoever built your Wine packages and let them know there's a problem.

You can work around this by using the command line instead of your file manager (see [Wine User's Guide#How to run Windows programs from the command line \(/Wine_User%27s_Guide#How_to_run_Windows_programs_from_the_command_line\)](#)).

5.11 I double-clicked on an .exe file, but the system said "The file foo.exe is not marked as executable..."

If the dialog says "Read about the executable bit", with a hyperlink, try clicking on the hyperlink and reading about the executable bit.

If the file is on your hard drive, right-click it, choose Properties / Permissions, and check "Allow executing file as program".

If the file is on a CD-ROM, you can run it from the command line as described above. Or, if you know how to use mount, remount the cd-rom to mark all files as executable with a command like `mount -o remount,mode=0777,exec /media/cdrom` but using the real mount point if it's different from /media/cdrom.

6 Using Wine

6.1 How do I run Wine?

Wine is not an application you run. Wine enables your computer to run Windows applications. Simply install and run your applications as you would in Windows. See [#How do I run an installer using Wine?](#).

6.2 Should I run Wine as root?

⚠ NEVER run Wine as root! Doing so gives Windows programs (and viruses) full access to your computer and every piece of media attached to it. Running with `sudo` also has these same risks but with the added bonus of breaking the permissions on your `~/.wine` folder in the process. If you have run Wine with `sudo` you need to fix the permission errors as described in the next question, and then run `winecfg` to set Wine up again. You should always run Wine as the normal user you use to login.

For Linux systems, all ideas that Wine needs root can be solved through Posix Capabilities (<http://www.linuxjournal.com/article/5737>) or Posix File Capabilities (<http://www.ibm.com/developerworks/linux/library/l-posixcap/index.html>) or correcting other security settings.

As far as Windows programs are concerned, you are running with administrator privileges. If an application complains about a lack of administrator privileges, file a bug; running Wine as root probably won't help.

6.3 I ran wine with sudo or as root. How do I fix my permission errors?

You need to fix the permissions on your `~/.wine` directory, this is where all Wine state, configuration and any important data you might have such as installed programs, saved data within Wine programs, etc. are stored. Once you delete or fix the permissions on this directory, rerun Wine as a regular user always! Run the following to fix the permissions on your `~/.wine` directory if it now has root permissions.

```
cd $HOME
sudo chown -R $USER:$USER ~/.wine
```

6.4 How do I know what version of Wine I have?

Open up a terminal and run `wine --version`. It will say something like "wine-1.9.2"; if you are using Git (`/Git_Wine_Tutorial`) then you will have a version along the lines of "wine-1.9.2-231-gb922b55".

TIP: You can find out what the latest release of Wine is from WineHQ's main page (<http://www.winehq.org>). Currently, Wine development releases come out every two weeks. Your operating system may ship with an out of date (obsolete) version of Wine. Depending on what OS you use, you may be able to add an update source to your package management system to keep up to date. Check the downloads (<http://www.winehq.org/download>) page for details.

6.5 Wineprefixes

6.5.1 Where is my C: drive?

Wine uses a virtual C: drive instead of your real C: drive. The directory in which this is located is called a 'wineprefix.'

By default, it's in your home directory's `~/.wine/drive_c` subdirectory. (On MacOSX, see the MacOSX FAQ (`/MacOSX_FAQ`) for how to find this.)

See also the `WINEPREFIX` environment variable; if this is set, wine uses it to find the wineprefix.

6.5.2 Can I store the virtual Windows installation somewhere other than ~/.wine?

Yes: `~/.wine` is just the default wineprefix (a.k.a. "configuration directory" or "bottle").

You can change which prefix Wine uses by changing the WINEPREFIX environment variable (outside Wine). To do this, run something like the following in a terminal:

```
export WINEPREFIX=~/.wine-new
wine winecfg
```

Wine will then create a new prefix in ~/.wine-new.

To use the default prefix, use the command `unset WINEPREFIX`. Or just set WINEPREFIX to ~/.wine.

Alternatively, you can specify the wine prefix in each command, e.g.

```
WINEPREFIX=path_to_wineprefix wine winecfg
```

You can rename, move, copy and delete prefixes without affecting others, and each prefix has its own wineserver instance.

Wherever you see ~/.wine or \$HOME/.wine in this Wiki, you can usually replace it with \$WINEPREFIX.

6.5.3 How do I create a 32 bit wineprefix on a 64 bit system?

At present there are some significant bugs (<http://bugs.winehq.org/buglist.cgi?keywords=win64&resolution=--->) that prevent many 32 bit applications from working in a 64 bit wineprefix. To work around this, you can create a new 32 bit wineprefix using the WINEARCH environment variable. In a terminal, type:

```
WINEARCH=win32 WINEPREFIX=path_to_wineprefix winecfg
```

(use the actual path to the wineprefix) Do **not** use an existing directory for the new wineprefix: Wine must create it.

Once a 32 bit wineprefix is created, you no longer have to specify WINEARCH in the command line to use it, as the architecture of an existing wineprefix cannot be changed. Note that if the wineprefix is not the default (~/.wine, you **do** have to specify the WINEPREFIX variable when installing anything (including winetricks) to it:

```
WINEPREFIX=path_to_wineprefix wine start /unix path_to_installer
```

6.5.4 Why aren't versions of Windows prior to XP available in 64 bit wineprefixes?

Short answer: because there were no 64 bit versions of Windows prior to XP.

If you are on a 64 bit system, you will have to create a 32 bit wineprefix (/FAQ#How_do_I_create_a_32_bit_wineprefix_on_a_64_bit_system.3F) to be able to select a version of Windows older than XP in winecfg.

6.5.5 How can I run two programs as if they were on different computers?

Example: You have server and client programs. One won't run in the presence of the other.

Using different Wine prefixes will help you here, since they simulate two Windows computers, in essence.

Run the first program as normal:

```
wine first-program.exe
```


The second needs to be run in a different prefix, so we need to change the WINEPREFIX environment variable:

```
WINEPREFIX="$HOME/.wine-second" wine second-program.exe
```

The *first-program.exe* and *second-program.exe* can be two copies of the same program.

6.6 Configuring

6.6.1 How do I get Wine to launch an application in a virtual desktop?

You can do this with `winecfg` (`/Winecfg`). Add the application in the Applications tab and then, in the Graphics tab, enable "Emulate a virtual desktop".

You can also use the following command:

```
wine explorer /desktop=name,1024x768 program.exe
```

Replace *program.exe* with the name of your program, and change the resolution to the size of the virtual desktop you want. Changing *name* allows you to open several desktops simultaneously.

6.6.2 How do I edit the Wine registry?

The Wine registry is stored in the `.reg` files in `~/.wine`, however you should not edit these files by hand due to the encoding that they use. Always use the `regedit` program that comes with Wine. This can be run by typing `wine regedit` in the terminal. Wine's `regedit` is virtually identical to the Windows version of `regedit` and also supports importing and exporting of registry files. NEVER try and import your entire Windows registry, this will just break Wine.

See also: Useful Registry Keys (`/Useful_Registry_Keys`)

6.6.3 How do I associate a native program with a file type in Wine?

There are two ways using which you can associate a native program with a file type. The first method is to use `winebrowser` and an alternative would be to write a shell script.

The example below uses `winebrowser` to launch the default PDF handler on your system (on a Unix desktop it uses `xdg-open`). Save the lines below to a file `pdf.reg`.

```
[HKEY_CLASSES_ROOT\.pdf]
@="PDFfile"
"Content Type"="application/pdf"
[HKEY_CLASSES_ROOT\PDFfile\Shell\Open\command]
@="winebrowser \"%1\""
```

Import the `.reg` file into the registry using the command

```
regedit $HOME/pdf.reg
```

Another option is to use a shell script to call a native application. Save it as `run_linux_program` under `$HOME/bin`:

```
#!/bin/sh $1 "`wine winepath -u "$2"`"
```

Don't forget to run `chmod a+x $HOME/bin/run_linux_program` to make it executable. Also make sure that `$HOME/bin` directory is in your `$PATH`. Otherwise it will not work.

To associate (say) .pdf files with the `acroread` Linux program save the following as `$HOME/pdf.reg` and then import it with the command `regedit $HOME/pdf.reg` :

```
[HKEY_CLASSES_ROOT\.pdf]
@="PDFfile"
"Content Type"="application/pdf"
[HKEY_CLASSES_ROOT\PDFfile\Shell\Open\command]
@="/bin/sh run_linux_program acroread \"%1\""
```

You can reuse this script and just edit the registry file. For example to associate .doc documents with OpenOffice (`ooffice`):

```
[HKEY_CLASSES_ROOT\.doc]
@="DOCfile"
"Content Type"="application/msword"
[HKEY_CLASSES_ROOT\DOCfile\Shell\Open\command]
@="/bin/sh run_linux_program ooffice \"%1\""
```

6.6.4 How can I prevent Wine from changing the filetype associations on my system or adding unwanted menu entries/desktop links?

Users who do not want the installer for a Windows app to change filetype associations, add menu items, or create desktop links, can disable `winemenubuilder.exe` . There are several ways to do this:

- *In `winecfg`*: before running the installer, run `winecfg`. Go to the Libraries tab and type `winemenubuilder.exe` into the "New overrides" box (it is not in the dropdown list). Click add, then select it from the "Existing overrides" box. Click "Edit" and select "Disable" from the list, then click "OK".
- *Registry file*: If you need to apply the setting many times (e.g. every time you recreate the Wine prefix), this approach may be more convenient. Create a text file named with extension .reg (e.g., `disable-winemenubuilder.reg`) containing the following:

```
[HKEY_CURRENT_USER\Software\Wine\DllOverrides]
"winemenubuilder.exe"=""
```

To apply the setting, run:

```
regedit disable-winemenubuilder.reg
```

- *Environment variable*: set the `WINEDLLOVERRIDES` environment variable when you run the installer, e.g.,

```
WINEDLLOVERRIDES=winemenubuilder.exe=d wine setup.exe
```

Users who frequently create new wineprefixes may wish to put `WINEDLLOVERRIDES=winemenubuilder.exe=d` in their `.bashrc` to avoid having to specify it for every wineprefix.

6.6.5 How do I disable the GUI crash dialog?

Wine includes a GUI crash dialog that is turned on by default. Users of apps that work despite a background crash may find the GUI dialog annoying, and in some cases the dialog itself has been reported to prevent an app from working.

The easiest way to disable the crash dialog is with winetricks (/Winetricks):

```
sh winetricks nocrashdialog
```

If you prefer to do it manually, copy the following key to a text editor:

```
[HKEY_CURRENT_USER\Software\Wine\WineDbg]
"ShowCrashDialog"=dword:00000000
```

Save the file with a .reg extension (e.g. crashdialog.reg), then apply it with regedit:

```
regedit crashdialog.reg
```

(You may need to specify the full path to the file, depending on where you saved it.)

To turn the GUI crash dialog back on, change 00000000 to 00000001 and reapply.

These changes can also be made by simply running regedit and adding/changing the appropriate key the point-and-click way.

6.6.6 How can I make Wine fonts anti-aliased?

Support for subpixel font rendering was added to Wine in version 1.1.12, but it may not be enabled. Use winetricks and select one of the fontsmooth-gray, fontsmooth-rgb or fontsmooth-bgr options.

6.6.7 How do I change the DPI (font size)?

First, you should try editing with winecfg. Go to the Graphics tab, and slide the "Screen Resolution" slider accordingly. Changes will not effect the winecfg window until you restart it.

If windows and fonts are so big you can't get to the controls in winecfg, see #Wine's windows and fonts are extremely large.

6.6.8 How do I configure a proxy?

If you want to use a proxy server for all HTTP connections, simply set the http_proxy environment variable. On many Linux distributions, configuring a network proxy, e.g. with the Network Proxy tool, does this for you automatically.

Alternatively, you can configure a proxy in the registry. There are separate locations for wininet.dll and winhttp.dll.

For wininet, use regedit to add the following values to the [HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings] key:

```
"ProxyEnable"=dword:00000001
"ProxyServer"="proxy-server-address:port"
```

At present, Wine's wininet.dll does not support proxy autoconfiguration (PAC) scripts.

For winhttp, you'll need to use the proxycfg.exe utility to configure the registry. This utility is available in the system32 directory of a Windows installation, and MSDN describes its usage.

7 Limitations and Risks

7.1 Can Wine run in character mode?

Most of Wine's development effort is geared towards programs written for the Windows GUI, but some limited support for character mode is available with the "null" driver. Wine automatically activates "null" whenever x11 driver isn't loaded, but even then, Wine depends on the xorg libraries.

Also the "null" driver will only work for *pure* console applications that never use any windowing functions (for example, parts of OLE create purely internal windows).

See Wine User's Guide#Text mode programs (CUI: Console User Interface) (/Wine_User%27s_Guide#Text_mode_programs_.28CUI:_Console_User_Interface.29) for more information.

7.2 Can I install applications to be shared by multiple users?

Wine does not currently allow sharing its configuration ("prefixes") between users, due to the risk of registry corruption from running multiple wineservers simultaneously bug #11112 (https://bugs.winehq.org/show_bug.cgi?id=11112). At present, applications must be installed separately for each user.

However, you can copy Wine prefixes; you can install everything to one prefix, then make a copy of it in each user's home directory. This saves running installers repeatedly.

7.3 Can I use Wine to install drivers for my hardware?

No. Wine requires your hardware to already be working on your operating system. The technical reason for this is that Wine, like most applications, runs in user mode and not kernel mode (http://en.wikipedia.org/wiki/Kernel_mode).

7.4 Is Wine malware-compatible?

Yes. Just because Wine runs on a non-Windows OS doesn't mean you're protected from viruses, trojans, and other forms of malware.

There are several things you can do to protect yourself:

- Never run executables from sites you don't trust. Infections have already happened. (<http://www.winehq.org/pipermail/wine-devel/2007-January/053719.html>)
- In web browsers and mail clients, be suspicious of links to URLs you don't understand and trust.
- Never run any application (including Wine applications) as root (see above).
- Use a virus scanner, e.g. ClamAV (<http://www.clamav.org>) is a free virus scanner you might consider using if you are worried about an infection; see also Ubuntu's notes on how to use ClamAV (<https://help.ubuntu.com/community/ClamAV>). No virus scanner is 100% effective, though.
- Removing the default Wine Z: drive, which maps to the unix root directory, is a weak defense. It will not prevent Windows applications from reading your entire filesystem, and **will prevent you from running Windows applications that aren't reachable from a Wine drive (like C: or D:)**. A workaround is to copy/move /symlink downloaded installers to ~/ .wine/drive_c before you can run them.
- If you're running applications that you suspect to be infected, run them as their own Linux user or in a virtual machine (the ZeroWine malware analyzer works this way).

7.5 How good is Wine at sandboxing Windows apps?

Wine does not sandbox in any way at all. When run under Wine, a Windows app can do anything your user can. Wine does not (and cannot) stop a Windows app directly making native syscalls, messing with your files, altering your startup scripts, or doing other nasty things.

You need to use AppArmor, SELinux or some type of virtual machine if you want to properly sandbox Windows apps.

Note that the winetricks (/Winetricks) sandbox verb merely removes the desktop integration and Z: drive symlinks and is not a true sandbox. It protects against errors rather than malice. It's useful for, e.g., keeping games from saving their settings in random subdirectories of your home directory.

8 Applications

8.1 What applications/services conflict with Wine?

Many distributions are moving towards enabling Compiz/Beryl/other eye candy by default. The problem is that these programs conflict with Wine over the display driver. Disable these programs before using any applications with Wine (especially games, or when noticing weird window problems). Also, disabling the Composite extension in /etc/X11/xorg.conf will most certainly prevent compositing from affecting Wine.

Keyboard tools such as IBUS, xneur (gxneur, kxneur) and SCIM have been reported to conflict with keyboard input in Wine.

8.2 Does Wine support .NET? Should I install native .NET in Wine?

Recent versions of Wine will prompt you to download wine-mono (/Mono) on wineprefix creation. For many .NET apps, particularly older ones, this is sufficient.

If wine-mono does not work for your app/game, file a bug (if one has not been already filed), and try native .NET. You can install it by running winetricks (/Winetricks) and selecting the appropriate .NET version. Note that native .NET can at present only be installed to a 32 bit wineprefix, so if your app/game is 64 bit, you are out of luck for now.

8.3 Does Wine support DirectX? Should I install Microsoft's DirectX under Wine?

Wine itself provides a DirectX implementation that, although it has a few bugs left, should run fine. Wine supports DirectX 9.0c at this time. Work on DirectX 10 is underway.

If you attempt to install Microsoft's DirectX, you will run into problems. It is not recommended nor supported by Wine HQ to attempt this. You can install the runtime, but it will not run. The runtime needs access to the Windows drivers, and Wine cannot access them for obvious reasons. The only native Microsoft DLLs that could be useful anyway are the d3dx9_xx.dll type ones, and these require you to accept Microsoft's license. Additionally, versions of these DLLs are now part of the Wine tree. So, as Wine improves these DLLs will only become less relevant.

That said, there are some guides out there which describe how you can install Microsoft's DirectX. We reiterate: it is not recommended nor supported by Wine HQ to attempt this. Furthermore it is considered off topic in Wine HQ support mediums (such as the forums). Please use a clean Wine configuration folder before seeking help (delete your ~/.wine or use another wineprefix and re-install your Windows applications).

8.4 Does Wine support Internet Explorer? Should I install Microsoft's Internet Explorer in Wine?

Wine uses the core of Firefox to implement its own Internet Explorer replacement (wine-gecko). Recent versions of Wine should prompt you to install it on wineprefix creation. In most cases Wine's IE replacement (wine-gecko) is sufficient.

If you really need the real IE for a specific application to work, use `winetricks` (`/Winetricks`) to install `ie6`, `ie7` or `ie8`. You should put IE in a separate wineprefix, with whatever app really needs it. Note that the Wine project does not support installing the real Internet Explorer, as it requires a huge number of native DLLs, which is hard to configure. Please do not ask the Wine project for help if you run into problems.

You may also try commercial solutions, such as CrossOver (<http://www.codeweavers.com>), but if you do so and run into problems, do not seek help on the forum/ mailing list or IRC, as third party applications such as these are not supported here.

8.5 Why doesn't Wine work well with some copy protection?

Copy protection uses several "dirty" methods to detect if discs are "real" or not. Some of these methods work in Wine but many do not, such as the extremely invasive StarForce system. Wine would need to be altered to allow for almost rootkit-like functionality of programs to get some of these copy protection schemes to work, so support is likely to be a long time off if ever (some people use illegally modified or "cracked" games with the copy protection removed, but the Wine project cannot advocate that).

See also our page on copy protection (`/Copy_Protection`).

8.6 I followed a howto I found on a website, and the application still doesn't work

There are many unofficial howtos for various apps posted on blogs and forums. They are rarely maintained by their authors, and because of Wine's rapid rate of development, even ones that were correct at the time they were posted can quickly become outdated. Applying tweaks that are no longer needed in current Wine can actually prevent an app that now runs fine without tweaks from working at all (Office 2007 is a prime example). The only howtos supported here are the ones on this site. If you insist on following an outside one, ask its author for help.

If in doubt, start again installing the app in a fresh wineprefix and ask for help on the user forum.

8.7 I used a third party application (PlayOnLinux, WineBottler, etc.) to install or manage applications in Wine and need help

Consult the support channels for whatever third party application you used. Links to some of these outside applications are provided as a convenience on the Third Party Applications (`/Third_Party_Applications`) page, but they are not supported here. Before seeking help on the forum/ mailing list/IRC or filing bugs, reinstall your Windows application using plain Wine.

8.8 My application worked with an older version of Wine, but now it's broken!

This is called a regression. Please perform a regression test (/Regression_Testing) to identify which patch caused it, then file a bug report (<http://bugs.winehq.org>) and add the "regression" keyword: we'll pay special attention to it, since regressions are a lot easier to fix when caught early.

8.9 I really need this particular application to work. How can I go about making this happen?

Briefly, you can improve Wine yourself, convince someone else to do it for you, or try some complicated hacks involving native Windows DLLs rather than Wine's unfinished ones.

8.10 I'm writing a Windows app. How can it detect if it's running under Wine?

This is a bad idea. See the Developer FAQ (/Developer_FAQ#How_can_I_detect_Wine.3F).

9 Performance

9.1 My 3D application/game is very slow (FPS)

Usually a 3D performance issue, indicates that something is wrong with your OpenGL 3D drivers. See 3D Driver Issues (/3D_Driver_Issues) for more information.

9.2 Is Wine slower than just using Windows?

Actually, Wine is sometimes faster. The speed of an application depends on a lot of factors: the available hardware and their drivers, the quality of the code in the APIs the application uses, and the quality of the code in the underlying operating system.

Driver code matters a lot. If you're running a graphics-heavy application using a video card with very poor drivers such as an ATI card under Linux, performance will degrade substantially. On the other hand, Linux has superior memory management, and comes out ahead of Windows in many CPU-related tasks.

Sometimes, bugs in Wine can make applications excessively slow. That said, Wine works on correctness first, performance second.

See Performance (</Performance>) for more information.

9.3 I get lots of "fixme:" messages in the terminal and Wine runs a bit slow

Ignore them. Generally speaking, a `fixme` message will not make much sense to someone unfamiliar with Wine development. They are for Wine developers (</Developers>). Also, it's important to understand that `fixme` messages often *do not* indicate a serious problem. Many applications will work fine even though Wine prints a few `fixme` messages. That said, they can still sometimes provide insight into how a particular application works (or doesn't work).

If there are a very large number of these messages scrolling repeatedly, you can sometimes speed Wine up a little by turning them off altogether. You can do so by setting the `WINEDEBUG` environment variable to `-all`. For example, your command line could look something like:

```
WINEDEBUG=-all wine program.exe
```

More advanced users and programmers interested in debugging Wine should see Debug Channels (/Debug_Channels) and the Developers (/Developers) wiki pages. Here's an example to selectively turn off fixme messages from dsound and part of D3D only:

```
WINEDEBUG=fixme-dsound,fixme-d3d_surface wine program.exe
```

10 Troubleshooting

10.1 General

10.1.1 How can I get a debugging log (a.k.a. terminal output)?

Run your application from the command line (see Wine_User's_Guide#How_to_run_Windows_programs_from_the_command_line (/Wine_User%27s_Guide#How_to_run_Windows_programs_from_the_command_line)).

You can then copy the log from the screen and paste it into a file if it's short; otherwise redirect the output of wine to a file using a Unix shell redirection ([http://en.wikipedia.org/wiki/Redirection_\(computing\)](http://en.wikipedia.org/wiki/Redirection_(computing))), e.g.

```
cd ~/.wine/drive_c/Games/Tron  
wine tron.exe &> log.txt
```

Important: Unless you have been asked specifically to provide a debug trace then do not do so. Simply follow the instructions above.

Important: If you get a crash dialog while doing this, please click **Close**. Otherwise your log will be incomplete.

Important: If the resulting text file doesn't have names of C source files in it, your copy of wine probably lacks debugging symbols. Please either build wine from source, or install the debug symbols package. (In Ubuntu, the debug symbols for package wine1.4 are in package wine1.4-dbg.)

10.1.2 How do I get a debug trace?

Note: Please only use this procedure when instructed. In most cases a regular terminal output is enough (see above). When filing bugs, it is often necessary to get additional debug trace (generally +relay, +seh, but you might be asked for specific debug channels (/Debug_Channels)). To retrieve a trace, run:

```
WINEDEBUG=+relay,+seh,+tid wine your_program.exe >> /tmp/output.txt 2>&
```

Then **attach** /tmp/output.txt to the bug. If the resulting file is larger than 1 MB, compress it with bzip2 or rzip before attaching. There are some cases where the bug seems to disappear when WINEDEBUG is used with the right channel. Please mention that in the bug report. For a list of debug channels that are available in Wine, see Debug Channels (/Debug_Channels)

10.2 Crashes and Freezes

10.2.1 My program froze up, how do I close it?

If you ran the program from a terminal window by typing `wine program.exe`, you can usually just go back to that terminal window and press Ctrl+C. If you ran the application some other way, such as from a launcher shortcut, then you can open up a terminal and forcibly kill the process:


```
killall -9 Application.exe
```

If you want to kill all Wine programs at once, you can run:

```
wineserver -k
```

You can also open up a Wine version of the Windows task manager by running `wine taskmgr` in a terminal. This will allow you to kill individual Wine processes.

10.2.2 My whole computer freezes, reboots, or shuts off when I run my game in Wine!

If you are getting a complete deadlock and are unable to even use your mouse after running Wine, it's probably not a specific problem with the Wine software. Wine is a user-level process, and shouldn't be able to completely hang the operating system under any circumstances. Instead, Wine is likely exposing a deeper problem with the system, such as a defective hardware driver, a bad memory stick, or overclocking flakiness.

It's often a graphics driver problem, in which case non-Wine apps might also be affected. If running `glxgears` also crashes, it's definitely a graphics driver problem. The most common cause is upgrading to a new kernel without also updating the graphics drivers to match. Try reinstalling your graphics drivers.

If the computer is a laptop and shutting itself off entirely, a likely cause is overheating. Some laptops have problems with cooling to begin with, and the Linux ACPI code controlling fans is known to be buggy.

If that doesn't help, ask for help on the wine-users forum. Be sure to mention the name of the app, the version of Wine, the output of `cat /etc/issue`, `lspci | grep -i vga`, and, if you're using the proprietary NVidia drivers, `cat /proc/driver/nvidia/version`. Maybe someone can help.

10.2.3 Every app I try to start crashes

You may have a messed-up `.wine` directory (aka `wineprefix`). For instance, if you install an app that starts a service when the system boots, and that service crashes, you'll see a crash every time you start wine.

To work around this, try removing your `.wine` directory and reinstall your apps, but skip the one that broke things.

10.3 Warning and Error Messages

10.3.1 My application says some DLL or font is missing. What do I do?

Applications *should* come with all the DLLs they need (except for core Windows DLLs). They sometimes forget to, and rely on you to already have the DLL or font installed. You can install the missing DLL or font in several ways:

- downloading it from the original creators of the runtime (e.g. Microsoft). The easiest way to do this is with `winetricks (/Winetricks)`.
- install other applications which do include them.
- copy it from a licensed version of Windows installed on the same machine.

Do not download DLLs or scripts from websites you do not know and trust! Fake or infected DLLs can cause you great pain, even on Wine.

See the `winetricks` wiki page (`/Winetricks`) for more information on `winetricks`.

10.3.2 Too many open files, `ulimit -n` probably needs

to be increased

Your operating system is probably living in the past, and has too low a hard limit on the number of open file descriptors. (See <https://bugs.launchpad.net/ubuntu/+bug/663090> (<https://bugs.launchpad.net/ubuntu/+bug/663090>) for why raising the hard limit is the right thing to do, and why raising the soft limit by default is dangerous.)

For Ubuntu and most modern versions of Linux, you can edit `/etc/security/limits.conf` as root, and change the line

```
* hard nofile 2048 (or whatever the current limit is)
```

to

```
* hard nofile 8192
```

(The asterisk means 'for all users'.)

Then log out and log in again, and do `ulimit -H -n`. It should show 8192 now, and Wine should have access to more file descriptors.

Here's another method that's more portable (might even work on Mac OS X), but only works temporarily, and only raises the limit for apps started from the current terminal window:

```
$ sudo bash
# ulimit -n 8192
# su yourusername $ wine yourprogram.exe
```

10.3.3 preloader: Warning: failed to reserve range 00000000-60000000 or winevdm: unable to exec '*application name*': DOS memory range unavailable

This issue is being followed in bug 12516 (http://bugs.winehq.org/show_bug.cgi?id=12516).

The cause is a Linux kernel setting. Run `cat /proc/sys/vm/mmap_min_addr` as root: if it does not equal 0 then running `sysctl -w vm.mmap_min_addr=0` as root can be used to temporarily fix the issue; to fix it permanently, add the line `vm.mmap_min_addr=0` to `/etc/sysctl.conf`. Please record if you do this alteration, as the area Wine needs may change.

See Preloader Page Zero Problem ([/Preloader_Page_Zero_Problem](#)) for more information.

10.3.4 Failed to use ICMP (network ping), this requires special permissions

On *NIX systems ICMP ping requires use of raw sockets, which is limited to super user (root) only. And running Wine as root is a bad idea. Fortunately newer versions of Linux allow granular permission control to grant only required permissions to specified files.

To allow Wine opening raw sockets run this command:

```
sudo setcap cap_net_raw+epi /usr/bin/wine-preloader
```

Note: This works with default binary Wine install only on most distros. Self-compiled Wine will be located under `/usr/local/bin`. The 64-Bit name is `wine64-preloader`. Third party Wine wrappers (such as PlayOnLinux) might keep Wine binary in other places. You will need to rerun the command after updating Wine.

To remove the permissions use:

```
sudo setcap -r /usr/bin/wine-preloader
```

10.3.5 err:virtual:map_image failed to set 60000020 protection on section .text, noexec filesystem?

This can be caused by filesystems mounted with user or noexec options, or by SELinux. Make sure the app in question isn't on a funny filesystem, or try disabling SELinux temporarily.

10.3.6 Broken NVIDIA RandR detected, falling back to RandR 1.0

RandR (<https://en.wikipedia.org/wiki/RandR>) is a protocol used by applications to talk to the X server to change screen resolution, among other things. nVidia's proprietary drivers for GNU/Linux intentionally do not properly implement newer versions of RandR, which Wine normally relies upon. This could present problems, particularly in software that attempts to change resolution or output to multiple monitors. (See bug 34348 (https://bugs.winehq.org/show_bug.cgi?id=34348))

The open source Nouveau driver is not affected by this issue, and is the recommended solution for users who do not need the proprietary driver for any of their apps or games. Users who require the proprietary driver should try the workaround described below.

Many applications will attempt to start in lower resolutions if (and only if) they are available - but will happily use a higher resolution if that is the only option. In this case, you may be able to avoid a crash on nVidia drivers by forcing your X server to only support your monitor's native resolution and nothing else. This has proven to work for a range of games.

To make this change to Xorg, edit (as root) /etc/X11/xorg.conf and add the following line:

Option	"metamodes" "1920x1080 +0+0"
--------	------------------------------

within the Screen section (changing 1920x1080 to whatever supported resolution you require).

Upon restarting Xorg, you can test your changes using the xrandr command. xrandr should list your chosen resolution, and xrandr --q1 should do the same. There should not be any additional resolutions listed. You may now re-test your Windows application, hopefully with more success. Remember to comment out the line and restart Xorg when you are finished if you need other resolutions working for other software.

10.4 Graphics

10.4.1 My application complains about being unable to change the resolution or color depth

You generally need to edit the Screen section of your /etc/X11/xorg.conf to support additional color depths and resolutions. There may also be a problem with Xrandr.

10.4.2 The application I am trying to run complains that it needs 256 colors but I have millions!

The inability to switch from 24bpp mode to 8bpp mode is a limitation of X, not a bug in Wine. See 256 Color Mode (/256_Color_Mode) for some possible workarounds.

10.4.3 My X screen won't go back to its normal resolution after running a game fullscreen

You can often work around this by changing the screen resolution and then changing it back again under the system preferences.

Alternately, you can run this terminal command to restore your X settings:

```
xrandr -s 0
```

10.4.4 I'm using Desktop Effects with Compiz, Fusion, or XGL and get poor performance/odd messages/broken applications

Using compositing managers in X11 tends to cripple OpenGL performance or break OpenGL entirely (this does not apply to the Mac OS X compositor, which cannot be disabled). We recommend that you disable them entirely before trying to use Wine. If you are using one and experiencing slow performance then please do not file bugs in Wine, as these are bugs in your window manager or your video drivers. Also, disabling the Composite extension within `/etc/X11/xorg.conf` will most certainly prevent any compositing from affecting Wine.

10.4.5 Graphics in games with good ratings in AppDB are scrambled

- Retry using the latest graphics drivers.
- Most AppDB (<http://appdb.winehq.org/>) entries are based on **NVIDIA/GeForce** hardware running the proprietary driver.
- **ATI/AMD/Radeon** cards running the proprietary fglrx driver have problems in Wine. As a rule of thumb, at least games that use shaders are broken. See this posting (<http://article.gmane.org/gmane.comp.emulators.wine.user/36669>) and Wine bug 7411 (http://bugs.winehq.org/show_bug.cgi?id=7411) for details.
- **Other hardware** (Intel/S3/Matrox etc.) will probably run only old (non-shader) games. Compatibility is not well tested.
- Same for **open source** drivers as their 3D support is typically basic only.

10.4.6 Wine displays corrupted or missing text

This may be bug 16146 (http://bugs.winehq.org/show_bug.cgi?id=16146), caused by the `nvidia-96xx` legacy driver, or bug 18120 (http://bugs.winehq.org/show_bug.cgi?id=18120), which affects QT 4.5.0 applications. It could also be caused by missing fonts, font conflicts, or adding new fonts to Wine.

Try using a fresh Wine prefix (by moving or deleting `~/.wine`, or changing the `$WINEPREFIX` environment variable). If you still have this problem, try setting the following in the Wine registry:

```
[HKEY_CURRENT_USER\Software\Wine\X11 Driver]
"ClientSideWithRender"="N"
```

Place above in a text file called `norender.txt` and it can be inserted into the registry with the command `regedit norender.txt`. Please apply only as required. (This was reported as being required of OS X on the 1 Dec 2007, and more recently on other platforms, such as Ubuntu.)

10.4.7 Wine's windows and fonts are extremely large

Sometimes you can use the Alt key and the mouse to move the `winecfg` window so you can reach the "Screen Resolution" slider on the Graphics tab; slide it down. Changes will not effect the `winecfg` window until it's restarted.

If that doesn't work, you can use this one line registry change (all one line):

```
echo -e "[HKEY_LOCAL_MACHINE\\System\\CurrentControlSet\\Hardware Profil
```

If all fails, you could remove your `~/wine` directory and reinstall your Windows applications.

10.5 Sound

10.5.1 Audio in games sounds different from running in Windows

A number of DirectSound features are not implemented (correctly) in Wine yet.

- Sound effects (echo etc.) are not implemented (Bug 10603 (http://bugs.winehq.org/show_bug.cgi?id=10603))
- Doppler effect is not implemented (source (<http://source.winehq.org/git/wine.git/?a=blob;f=dlls/dsound/sound3d.c;h=252c7632667fb1b49556e1f281f8232e42e1c345;hb=HEAD#l266>))

10.5.2 MP3s do not play in Windows Media Player or applications that depend on it

For MP3 sound to play out of the box in apps that use the WMP engine and codecs, you must have 32-bit `libmpg123` installed on your system and Wine must have been compiled with MP3 support. Not all distro packages provide this; openSUSE packages in particular are known to be compiled without MP3 support. If your distro's Wine package provides no MP3 support, consult your distribution's user forum to see if any package provides it. (For openSUSE, `wine-mp3` package is available at Packman (<http://packman.links2linux.org/>) which adds mp3 support using `libmpg123` to Wine.)

The workaround for lack of `libmpg123` and/or `winemp3.acm` is to use the codec installed by WMP9, `l3codeca.acm`. Copy `l3codeca.acm` to the wineprefix's `/windows/system32` directory (or use `winetricks` to install WMP9), then create a symlink to it named `winemp3.acm` in the same directory. Wine will then use the native codec to play MP3s.

This only affects WMP and apps that rely on it for MP3 playback (including Powerpoint). Apps that install their own MP3 codec, such as Winamp or VLC Player, should be able to play MP3s without this workaround.

10.5.3 Bad or no sound on systems using PulseAudio

Bad or no sound was a problem on some systems with Pulseaudio in older versions of Wine using the `winealsa` driver. If you are using a version of Wine older than 1.8, please upgrade, as most such problems should be solved by the `winepulse` driver. If the problem still occurs, try unsetting the `PULSE_LATENCY_MSEC` environment variable. Some distro packages (notably Debian) formerly set this variable to work around the sound issue with the `winealsa` driver, but it is not needed with the `winepulse` driver and may prevent sound from working.

10.6 Networking

10.6.1 My program fails to do networking, but my other applications can get online

Note: These instructions are for older Wine installs. If you're using Wine 1.x and your application still fails to do networking, you can give this a try as well. If you're running Wine 1.x and below instructions work for you, **file a bug** so we can fix Wine to improve other people's experience.

You need to make sure that your hostname resolves to the IP address of your network interface. To verify if you have this problem run `hostname -i`. If it returned IP address starting from "127." then read on.

To set this up correctly, you can type the following from a terminal:

```
hostname
```

This will return your hostname the way your computer sees it. Now, you need to open an editor with system administrator privileges, how you do this will depend on the distribution you are using. Open the file `/etc/hosts` and see if there is an entry for your hostname. Assuming your hostname is "yourhost" and your network IP address is 192.168.0.23, the entry might look like this:

```
127.0.0.1 yourhost.yourdomain.com yourhost
```

Change this to (or add, if there is no such line):

```
192.168.0.23 yourhost.yourdomain.com yourhost
```

For most Windows games with networking problems, this is all you need to get networking to work.

10.6.2 Why doesn't DNS resolve in 64-bit operating systems?

Many distributions don't provide all the 32-bit compatibility libraries that wine needs. In this case, wine needs 32-bit DNS libraries. On Ubuntu/Debian, this package is `lib32nss-mdns`. To install it on those OSes, use:

```
sudo apt-get install lib32nss-mdns
```

For other operating systems, the package name and installation method may differ. Consult your distribution's support channels.

10.7 Removable Media

10.7.1 My CD or DVD disc won't eject

Try `wine eject`. It is a function to free up, unlock, and eject the drive. Make sure that the drive is mapped as a CD-ROM in `winecfg` and specify the drive letter in the command line, e.g.

```
wine eject d:
```

10.7.2 The application's CD or DVD looks empty or is missing some files!

Some discs are poorly mastered in a way that affects only Unix-based operating systems.

You may need to use the "unhide" or "norock" mount options for these discs.

Run `mount` once to see where the disc is mounted, then mount it again in the same place with the needed option (and the "remount" option). Some assembly required.

Examples:

```
sudo mount -o remount,unhide /dev/sr0 /mnt/cdrom
```

or

```
sudo mount -t iso9660 -o ro,unhide /dev/cdrom /media/cdrom0
```

See also your operating system's documentation, e.g. <https://help.ubuntu.com>

/community/Mount (<https://help.ubuntu.com/community/Mount>) or <http://manpages.ubuntu.com/manpages/natty/man8/mount.8.html> (<http://manpages.ubuntu.com/manpages/natty/man8/mount.8.html>)

10.8 Input Devices

10.8.1 Some key combinations in my application do not work

Even in full screen mode, window managers typically capture some keys. For example, in KDE and GNOME, Alt+Left Click is used to move the whole application window by default. Thus, this key combination is not available to applications in Wine. You have to disable the colliding combinations in your window manager. For KDE, see Control Center/Window Behaviour or (better) Window-specific settings/Workarounds/Block global shortcuts. For GNOME, see System/Preferences/Windows and change the "Movement Key" setting. Also see System/Preferences/Keyboard Shortcuts for specific keyboard combinations.

(Keywords: Keyboard, Shortcut, Modifier, Alt, Ctrl, Control.)

10.8.2 My mouse jumps around/resets its position often in games

See http://bugs.winehq.org/show_bug.cgi?id=27156 (http://bugs.winehq.org/show_bug.cgi?id=27156) and https://bugs.freedesktop.org/show_bug.cgi?id=30068 (https://bugs.freedesktop.org/show_bug.cgi?id=30068).

10.9 Miscellaneous

10.9.1 My application runs, but text areas act strangely (e.g. lines don't wrap, double-clicking doesn't select words)

You may have run into a bug in Wine's RICHED20.DLL. You can try using Microsoft's RICHED20.DLL by running winetricks (/Winetricks) and selecting riched20. This may let you work around the problem until the Wine developers fix the bug.

10.9.2 I deleted my Wine menu, and now I can't get it back

Rather than actually delete anything, menu editors on Unix desktops simply mark menu entries as hidden so that they don't show up in the menu. Thus, they remain hidden after reinstalling the application. First, see if these menu entries can be found your menu editor and re-enabled.

This information is stored in `~/.config/menus/applications.menu`. Edit `~/.config/menus/applications.menu` and you should find a section near the end that looks similar to this:

```
<Menu>
  <Name>wine-wine</Name>
  <Deleted/>
</Menu>
```

or perhaps this:

```
<Menu>
  <Name>wine-wine</Name>
  <Menu>
    <Name>wine-Programs</Name>
    <Menu>
      <Name>wine-Programs-AutoHotkey</Name>
      <DirectoryDir>/home/user/.local/share/desktop-directories
    </Menu>
  </Menu>
  <Deleted/>
</Menu>
```

Remove the `<Deleted/>` line and your Wine menu will reappear.

10.9.3 16-bit applications fail to start

See https://bugs.winehq.org/show_bug.cgi?id=36664 (https://bugs.winehq.org/show_bug.cgi?id=36664). For 64 bit Linux Kernels 3.15+, 3.14.6+, 3.10.42+, 3.13.22+, 3.4.92+ and 3.2.60+. You'll need to run:

```
# echo 1 > /proc/sys/abi/ldt16
```

as root, but be aware that this has security implications.

If your kernel is older than the above stated versions `/proc/sys/abi/ldt16` most likely will be missing.

The security fix is `espfix64` and as of yet only exists in development branch for 3.16 Linux Kernel. At some point in a future I will put a list of kernels containing the security fix as it most likely will be backported when it is tested. Doing `/proc/sys/abi/ldt16` will be required on the security fixed kernels just like the current insecure ones.

10.9.4 Input doesn't work over SSH

When using Wine over SSH, you must run in 'trusted X11 forwarding' mode. For OpenSSH, this means using `-Y` instead of `-X`. See for example https://bugs.winehq.org/show_bug.cgi?id=41476 (https://bugs.winehq.org/show_bug.cgi?id=41476).

Retrieved from "<https://wiki.winehq.org/index.php?title=FAQ&oldid=2377>
(<https://wiki.winehq.org/index.php?title=FAQ&oldid=2377>)"

■ This page was last modified on 20 December 2016, at 12:31.

Hosted By  (<https://www.codeweavers.com/>)