

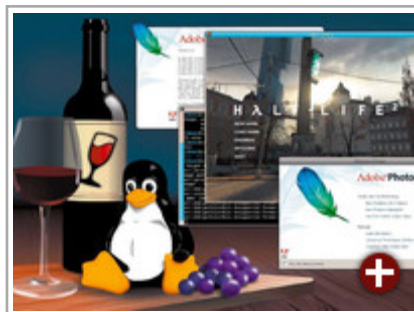
Kurz-Link: <http://www.pro-linux.de/-0102671>

Wine entkorkt: Der Windows-Ersatz im Detail



Mit jedem Versionssprung reift der Windows-Emulator Wine und verträgt sich mit weiteren Anwendungen. Doch Vorsicht: Oft ist noch behutsames Dekantieren nötig.

Von David Wolski[1]



DAVID WOLSKI

Gerade bei komplexeren Windows-Programmen muss man oft noch selbst Hand anlegen, um alle Kompatibilitätsprobleme auszumerzen

Der Jahrgang 2013 zeichnet sich durch die Veröffentlichung von [Wine 1.6\[2\]](#) aus. Seit Version 1.0, dessen Entwicklung von Google mitfinanziert wurde, hat Wine gute Fortschritte gemacht. Dahinter steht aber nicht weniger als eine Mammutaufgabe: Wine ist keine Virtualisierungsumgebung wie Virtualbox, sondern eine Abbildung der Windows-API (Application Programming Interface) in einer Laufzeitumgebung, um Windows-Programme auszuführen, die eine ausreichend komplette API brauchen.

Kurzer Rückblick: 20 Jahre Wine

Begonnen hatte Wine als Hobby-Projekt schon vor 20 Jahren zu Zeiten von Windows 3.1 mit seiner 16-Bit-API. Nach einem einfachen »Hello World«- Programm gelang es schnell, Solitär zum Laufen zu bringen. Mit Windows 95 musste Wine die neue 32-Bit-API unterstützen und zeigte schon bald beeindruckende Teilerfolge, so dass Corel von 1999 bis 2000 in die Weiterentwicklung von Wine investierte, um es für sein heute verschwundenes Word Perfect zu verwenden. Eine große Krise kam durch die erste Aufspaltung des Quellcodes, als

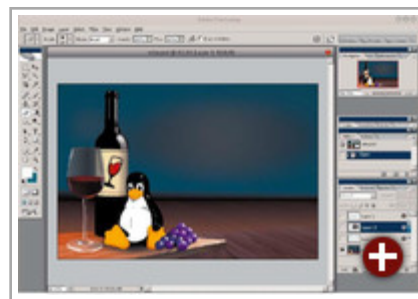
Transgaming die Direct-X-Unterstützung in sein eigenes, kommerzielles Projekt Wine-X und später in Cedega integrierte – eine kontroverse Aktion, die viele Open-Source-Entwickler vergraulte. Wiederbelebt wurde Wine 2006 von Google, das Wine damals für ihre Linux-Version von Picasa benötigte und vor allem die Wine-Unterstützung von Photoshop ein großes Stück weiterbrachte.

Nun ist die API von Microsoft zum einen hübsch umfangreich, zum Teil undokumentiert und nicht selten obskur. Dieses Gebilde hat Microsoft über Jahrzehnte mit Tausenden Mitarbeitern erschaffen, mit teilweise erheblichen Unterschieden zwischen unterschiedlichen Windows-Versionen. Die Wine-Entwickler haben weder (legalen) Quellcode von Microsoft, noch Zugriff auf den Quellcode der Programme, die unter Wine laufen sollen. Also muss die API Stück für Stück, Programm für Programm neu aufgebaut werden. Hier hilft nur Ausprobieren und Experimentieren weiter, was die lange Entwicklungszeit erklärt. Die API-Abbildung ist alles andere als komplett, aber inzwischen ausreichend, um viele verbreitete Programme fast wie unter Windows laufen zu lassen. Als Messlatte dienen den Entwicklern dabei Photoshop CS und ältere Office-Versionen sowie einige Spiele- Klassiker wie Halflife 2.

Auch andere, einfache Windows- Anwendungen laufen meist problemlos. Bei neueren und komplexen Programmen müssen Anwender selbst experimentieren und die Konfiguration von Wine vervollständigen.

Installation: Aktuelle Versionen bevorzugt

Bei allen populären Linux-Distributionen finden Sie Wine über den dort eingesetzten Paketmanager zur einfachen Installation. Auch wenn Wine bereits in Version 1.6 vorliegt, so sind die Pakete in den Linux-Distributionen meist ein gutes Stück älter, da als stabil markierte Wine-Versionen den Vorzug haben. Wenn ein Programm in einer älteren Wine-Version nicht läuft, lohnt sich der Versuch, es nochmal mit einer aktuelleren Entwicklerversion von



DAVID WOLSKI

Auf manche Windows-

Wine zu versuchen. Für Ubuntu und Mint bietet ein PPA als inoffizielles Repository Entwicklerversionen an. In der Kommandozeile nehmen Sie das PPA mit dem Befehl

```
sudo add-apt-repository ppa:ubuntu-wine/ppa sudo apt-get update
```

auf und installieren dann über

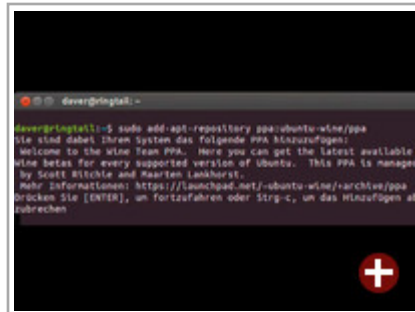
```
sudo apt-get install wine1.6
```

die neueste dort verfügbare Wine-Version. Debian-Anwender finden frische Wine-Pakete im Unstable-Zweig von Debian Sid. Wer ein stabiles Debian nicht zu einem gemischten System umwandeln möchte, kann aber auch gezielt die inoffiziellen Pakete von [http://dev.carbon-project.org/debian/wine-unstable\[3\]](http://dev.carbon-project.org/debian/wine-unstable[3]) installieren. Dazu laden Sie die dort angebotenen Pakete für die genutzte Plattform (amd64 oder i386) in ein Verzeichnis herunter. In der Kommandozeile gehen Sie in dieses Verzeichnis und bereiten die DEB-Pakete als root oder mit sudo nacheinander über diese drei Befehle für die Installation vor:

```
dpkg -i libwine*
dpkg -i wine-bin-unstable*
dpkg -i wine-unstable*
```

In den meisten Fällen wird sich der Paketmanager über fehlende Abhängigkeiten beschweren, wenn Wine bisher noch nicht installiert war. Dies ist aber kein Hindernis, denn mit dem nachfolgenden Kommando `apt-get install -f` weisen Sie den Paketmanager einfach an, die Abhängigkeiten selbständig aufzulösen und die Wine-Installation abzuschließen.

In Fedora ist Wine immer auf einem recht neuen Stand, und auch unter Opensuse ist die Entwicklerversion von Wine bereits in den Standardpaketquellen enthalten.



DAVID WOLSKI

Frisch abgefülltes Wine: Ubuntu und Debian bieten nur ältere Versionen von Wine. Neuere Entwicklerversionen bekommen Ubuntu-Anwender über ein PPA nachgeliefert

Funktioniert die gewünschte Anwendung?

Ob ein Windows-Programm in Wine läuft, können Sie nicht nur durch Ausprobieren ermitteln. Es lohnt sich vor der Installation immer ein Blick in die durchsuchbare Anwendungsdatenbank [http://appdb.winehq.org\[4\]](http://appdb.winehq.org[4]), die zeigt, wie gut – oder schlecht – es um die aktuelle Unterstützung einer Software steht.

Anwendungen erhalten je nach unterstütztem Funktionsumfang, den experimentierfreudige Wine-Anwender der Webseite melden, einen Status nach der Kategorie Platin, Gold, Silber, Bronze oder Garbage (Müll). Alle Anwendungen, die weniger als Gold-Status haben, laufen üblicherweise erst nach einigen Klimmzügen. Viele Einträge zeigen deshalb nach einem Klick auf »Show« eine Minianleitung mit Installationshinweisen. Wichtig ist hier auch immer die angegebene Versionsnummer von Wine, denn viele Programme laufen erst ab neueren Versionen. Auf welchem Stand das installierte Wine gegenwärtig ist, zeigt der Befehl `wine --version` in einem Terminal-Fenster.

Die Konfiguration von

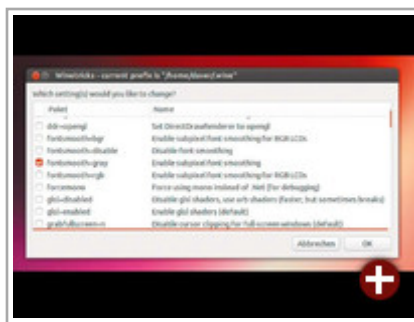
Wine erstellen

Nach der Installation von Wine ist der erste Schritt der Start des Konfigurationsprogramms `winecfg`. Das Programm legt eine Umgebung für Wine im versteckten Ordner `.wine` in Ihrem Home-Verzeichnis an, in dem eine Simulation der Windows-Registry in Form von Textdateien angelegt wird, sowie unter `~/.wine/drive_c` die Anwendungsdateien zukünftig installierter Programme.



Zudem überprüft winecfg die Laufzeitumgebung und bietet beispielsweise an, die Browser-Engine Gecko nachzurüsten, damit Windows-Programmen eine HTML-Engine zur Verfügung steht, was beispielsweise für die Anzeige von Hilfetexten wichtig ist. Eine Standard-Konfiguration erstellt das Programm automatisch, und Sie können viele Optionen hier ändern: So etwa, um die Darstellung anzupassen, ein Audiogerät für die Soundausgabe auszuwählen oder originale Windows-Bibliotheken hinzuzufügen. Ein wichtiger Punkt ist die Zuordnung von virtuellen Laufwerksbuchstaben zu Verzeichnissen, die Sie unter »Laufwerk« vornehmen. Dies ist nötig, wenn ein Windows-Programm zur Installation oder zum Start nach einer eingelegten Setup-CD fragt.

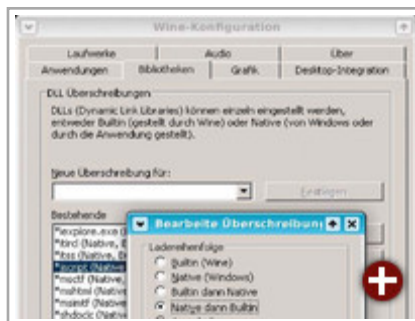
Erweiterte Einstellungen: Das Skript Winetricks



DAVID WOLSKI

Wine anpassen: Das Shell-Skript winetricks bietet Zugriff auf viele wichtige Wine-Optionen, die sich mit winecfg nicht ändern lassen, so etwa die Installation von Schriften

Optionen erreichen Sie über den Punkt *Select the default wineprefix*. Das folgende Menü erlaubt die Installation von häufig benötigten DLLs und Schriftarten. Mit *Change Wine settings* legen Sie interne Wine-Parameter fest. Es empfiehlt sich hier, in jedem Fall `fontsmooth=gray` zu aktivieren, da dies die Darstellung von Schriften in Windows-Programmen deutlich verbessert.

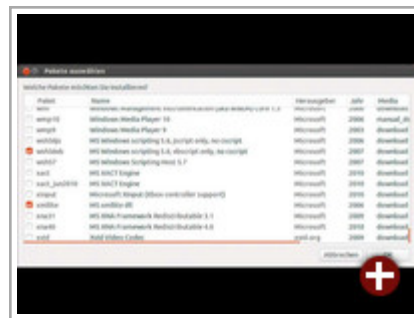


DAVID WOLSKI

Einzelne Windows-Bibliotheken ersetzen: Einige wählerische Programme erfordern den

Das Werkzeug winecfg bildet nur einen kleinen Teil der Einstellungen ab, und viele Optionen lassen sich nur direkt in den Konfigurationsdateien von Wine bearbeiten. Ein handliches Skript namens **winetricks** mit grafischen Menüs nimmt Anwendern dabei aber typische, häufig nötige Aufgaben und Anpassungen ab. Sie installieren das Skript am besten in seiner aktuellsten Version direkt vom Server der Entwickler. Auf der Kommandozeile verwenden Sie dazu `wget http://winetricks.org/winetricks` für den Download der Skript-Datei **winetricks** in das aktuelle Verzeichnis. Dort starten Sie es dann mit `sh winetricks`.

In Debian müssen Sie meist zuvor noch das Programmpaket **zenity** mit `apt-get install zenity` nachrüsten. Unter *Install an App* und *Install a Game* finden Sie Installationshilfen für populäre, aber meist ältere Windows-Programme.



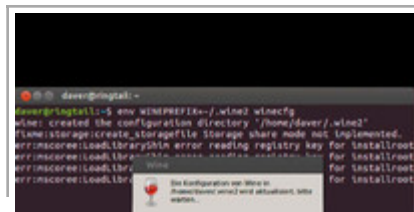
DAVID WOLSKI

Bibliotheken nachinstallieren: Mit winetricks laden und installieren Sie allgemeine Bibliotheken und Runtimes von Microsoft-Servern, falls Programme diese benötigen

Leihgaben: Originale Windows-DLLs nachrüsten

Wine darf aus lizenzrechtlichen Gründen keine Windows-Komponenten mit ausliefern. Einige Windows-Programme haben aber spezielle Ansprüche, denen die Wine-API (noch) nicht genügt, und verlangen nach originalen Bibliotheken von Windows.

Eine Reihe von Bibliotheken bietet das Konfigurations-Skript über sein Menü *Select the default wineprefix > Windows-DLL installieren* zur Installation und lädt dazu in den meisten Fällen ein



Austausch der mitgelieferten Wine-Bibliotheken mit den originalen Windows-DLLs. Diese Einstellungen nehmen Sie in winecfg vor

Update-Paket von den Microsoft-Servern herunter. Stellen Sie zuvor sicher, dass im Linux-System der Entpacker **cabextract** vorhanden ist.

Unter Debian müssen Sie das

Programm meist noch mit `apt-get install cabextract` installieren. Auch bei anderen Linux-Distributionen liegt der Entpacker in den Standard-Paketquellen, wird aber beispielsweise bei Ubuntu gleich zusammen mit Wine installiert.

Verlangt ein Programm eine sehr spezielle DLL, dann kopieren Sie die Bibliothek von einem bestehenden Windows-System in das Wine-Verzeichnis `~/.wine/drive_c/windows/system32`. Starten Sie dann winecfg, um dort unter *Bibliotheken* > *Neue Überschreibung für die gewünschte Bibliothek* auszuwählen. Mit *Festlegen* erstellen Sie eine neue Regel, die Sie mit *Bearbeiten* ändern. Die Einstellung *Native* nutzt dann die vorhandene, originale Windows-DLL anstelle der eingebauten (Builtin) Bibliothek von Wine. Das Skript winetricks erstellt diese Einträge übrigens selbständig bei der Installation von DLLs.

Eigene Umgebung für widerspenstige Programme

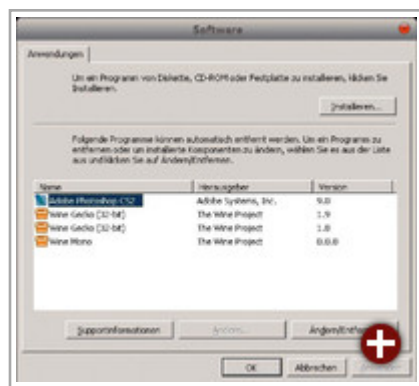
Windows-Programme, die viele eigene Bibliotheken und Einstellungen brauchen, können damit wiederum anderen, funktionierenden Anwendungen in Wine gefährlich werden. Wine bietet für diesen Zweck die Möglichkeit mehrerer unabhängiger Umgebungen mit eigenen Verzeichnissen an und nennt diese »**Prefix**«. Um eine neue Wine-Umgebung zu erstellen, beispielsweise als **.wine2**, geben Sie im Terminal-Fenster

```
env WINEPREFIX=~/.wine2 winecfg
```

ein. Das Verzeichnis `~/.wine2` wird dabei automatisch erstellt und mit einer Standard-Konfiguration ausgestattet. Damit diese alternative Umgebung auch bei der weiteren Konfiguration mit winecfg oder winetricks verwendet wird, müssen Sie stets das Prefix angeben – also beispielsweise `env WINEPREFIX=~/.wine2 sh winetricks` für den Start des Skripts winetricks. Und auch zum Ausführen von Setup-Programmen von Windows-Anwendungen geben Sie das Prefix als Start-Parameter an:

```
env WINEPREFIX=~/.wine2 wine start /Unix "/media/cdrom/setup.exe"
```

Bei Programmstartern für eine so installierte Anwendung ergänzt Wine das passende Prefix als Startparameter allerdings automatisch, damit die Programme in ihrer jeweiligen Umgebung ausgeführt werden.



DAVID WOLSKI

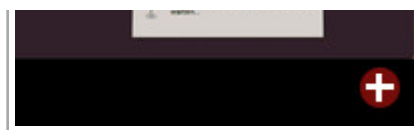
Der Uninstaller von Wine: Um

Umgekehrt ist es aber nicht möglich, eine 64-Bit-Umgebung in einem 32-Bit-Wine zu erzeugen. Ein Prefix dient auch dazu, getrennte Umgebungen für Wine 32 Bit unter 64-Bit-Systemen anzulegen. Hier erstellt winecfg nämlich standardmäßig eine 64-Bit-Umgebung für Wine, unter der viele 32-Bit-Programme nicht laufen. Um gezielt eine 32-Bit-Umgebung anzulegen, dient der Befehl

```
WINEPREFIX=~/.wine32 WINEARCH='win32' winecfg
```

Aufräumen: Programme wieder deinstallieren

Windows-Anwendungen, die nicht zufriedenstellend laufen oder nicht mehr benötigt werden, lassen sich auch wieder deinstallieren. Wine bringt dazu einen Uninstaller mit, den Sie in einem Terminal-Fenster mittels `wine uninstaller` starten. Sollte das Setup-Programm zum Deinstallieren nicht



DAVID WOLSKI

Ein neues Prefix erstellen: Wine kann für mehrere Umgebungen (Prefixe) in getrennten Verzeichnissen konfiguriert werden, hier im Verzeichnis `»~/.wine2«`

Windows- Programme aus der Wine-Umgebung wieder zu entfernen, bringt Wine dieses Deinstallationsprogramm mit I

funktionieren, können Sie auch die gesamte Wine- Umgebung zurücksetzen. Starten Sie dazu das Skript winetricks, wählen Sie das Standard-Prefix oder das gewünschte Wine-Prefix aus, und wählen Sie ferner im nächsten Menü *Delete all data and applications inside this wineprefix*.

Beachten Sie, dass Sie danach wieder mit einer leeren Wine-Umgebung beginnen. Es empfiehlt sich deshalb, Programme zu Experimenten in ihre eigenen Wine-Prefixe zu installieren, um andere Windows-Anwendungen nicht zu beeinträchtigen.

Dieser Artikel stammt von unserem Kooperationspartner [PC-WELT](#)[5].

Linkverweise:

[1] <mailto:info@pro-linux.de>

[2] <http://www.pro-linux.de/?=0.1.0.2.0.20033.0.0..00>

[3] <http://dev.carbon-project.org/debian/wine-unstable/>

[4] <http://appdb.winehq.org>

[5] http://www.pcwelt.de/ratgeber/Wine_entkorkt__Der_Windows-Ersatz_im_Detail-Distribution-8211010.html

[Mediadaten](#) [RSS/Feeds](#) [Datenschutz](#) [Impressum](#)

© 2013 Pro-Linux

