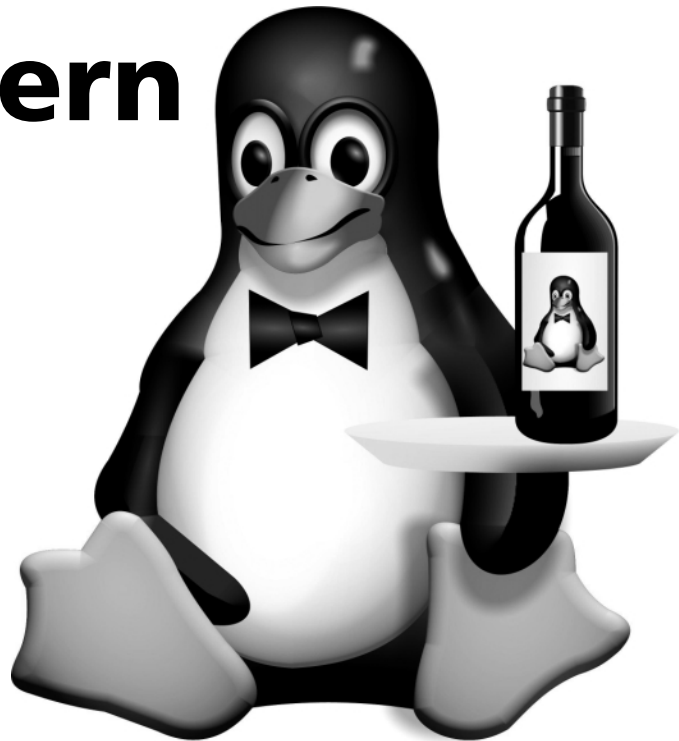


Wine 1.0: Feiern ohne Kater

DAVID M. GÜMBEL

Nach 15 Jahren Entwicklungszeit hat das Entwicklerteam von Wine das erste stabile Release 1.0 freigegeben. Damit ist es Zeit, sich das Projekt und die neue Version einmal genauer anzusehen.



Die Mail auf *wine-announce* liest sich ebenso bescheiden wie stolz: Alexandre Julliard, Maintainer des Projekts seit 1994, gab am 17. Juni 2008 die Freigabe der ersten stabilen Version von Wine bekannt. Bereits Anfang Mai war nach der letzten Betaversion 0.9.61 der erste Release Candidate erschienen und die Codebasis war lediglich noch für Bugfixes geöffnet. Doch was ist neu, was unterscheidet Wine 1.0 von früheren Versionen? Welche Erwartungen kann ein stabiles Release erfüllen, welche nicht? Um diese Fragen zu beantworten, muß zunächst ein kurzer Blick auf die Historie des Projekts geworfen werden.

Wine ist – eines populären Mißverständnisses zum Trotz – kein Emulator und wollte auch nie einer sein. Es handelt sich um eine Nachimplementierung des Windows-API unter Unix, das heißt: vor allem unter

Linux, aber auch Solaris und die BSD-Derivate werden grundsätzlich unterstützt. Mit Hilfe von Wine können also Programme, die auf dem 16- oder 32-Bit-API von Windows basieren, in unveränderter Form unter Linux ausgeführt werden. Dabei wird keine Windows-Lizenz benötigt, und die Programme integrieren sich gewissermaßen als »Bürger erster Klasse« genau wie native Linux-Applikationen mit ihren Fenstern, Icons, Menüs und der Zwischenablage in einen Desktop wie Gnome oder KDE.

Entwicklung der Codebasis

Ziel des Wine-Projekts ist es also, Windows-Programmen unter Unix das Interface (API) zur Verfügung zu stellen, das sie für ihre Ausführung benötigen. Als Wine im Jahre 1993 startete, war das ursprüngliche Entwicklungsziel das Win16-API, heute steht klar das Win32-API im Vordergrund. Lange Zeit war Wine dauerhaft im Alpha-Stadium und brachte deswegen keine Releases heraus, sondern lediglich monat-

liche Snapshots, die als Versionsnummer ihr jeweiliges Datum trugen. Die Version *wine-20020228* war die letzte, die unter der MIT-Lizenz veröffentlicht wurde, bevor die Entwickler beschlossen, auf die LGPL umzuschwenken. Damit sollte weiteren Forks vorgebeugt werden, die Weiterentwicklungen nicht an die Community zurückfließen lassen.

Ende 2005 erschien dann das erste Beta-Release mit der Versionsnummer 0.9. Nicht weniger als 61 weitere folgten, bevor schließlich der Sprung zur 1.0 gewagt wurde. Im Vorfeld hatten sich die Entwickler auf einige Kriterien geeinigt, die ein Wine 1.0 zu erfüllen hätte (siehe <http://wiki.winehq.org/WineReleaseCriteria>). Wichtige Punkte, die es zu erfüllen galt, waren das Ausführen bestimmter, besonders bedeutender Windows-Applikationen wie Photoshop CS2 oder verschiedener Office-Viewer von Microsoft, sowie ein Rückgang von Regressionen. Damit sollte sichergestellt werden, daß Wine mit zukünftigen (stabilen) Versionen möglichst keine bereits unter einer (stabilen) Vorversion lauffähig gemachten Applikationen wieder inkompatibel macht, weil Fehler in den Code eingeschleppt werden. Bei mehreren tausend implementierten Win-



Bild 1: Der Gecko-Installationsdialog

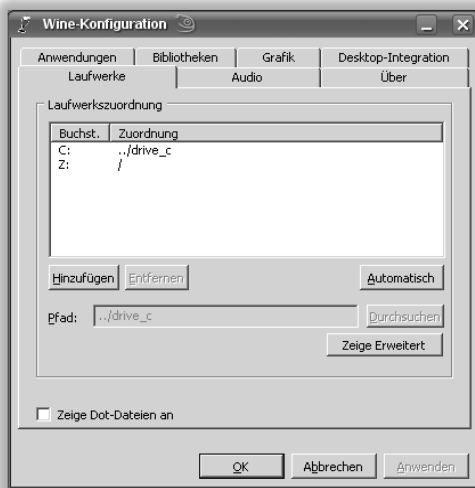


Bild 2: Laufwerkzuordnung in Wine

dows-API-Aufrufen und gut zwei Millionen Zeilen Quelltext, die Kompatibilität mit Windows-Versionen zwischen 2.0 und 2008 sicherstellen sollen, keine ganz leichte Aufgabe.

Stabilität

Für den Administrator hat das Release der Version 1.0 klar positive Effekte: Endlich liegt eine stabile Version vor, in die auch in Zukunft über längere Zeit keine neuen Features, sondern lediglich Bugfixes eingepflegt werden. 1.0 sollte also mit der Zeit mit den Freigaben der Versionen 1.0.x weiter reifen, was administrativen Aufwand für Tests der ausgeführten Windows-Software mit der jeweils neuesten Wine-Version spart. Zudem sind für die wesentlichen Distributionen Pakete in aktueller Version vorhanden und unter <http://www.winehq.org/site/download> erhältlich. Ein Selbstbauen von Wine aus den Quellen, mindestens bis zur Release von 0.9 noch das einzig probate Mittel, eine aktuelle Wine-Version für die eigene Distribution zu erhalten, entfällt also dauerhaft. Die Zahl der unterstützten Applikationen läßt sich schwer fassen. Die Applikationsdatenbank <http://appdb.winehq.org> kann helfen, einen Überblick über unterstützte Programme zu erhalten und gegebenenfalls auch einige Installationshinweise für problematische Anwendungen zur Verfügung stellen. Die Kompatibilität des jeweiligen Produkts wird von

den Benutzern bewertet – allerdings sind die Bewertungen meist mit Vorsicht zu genießen, da sie stark subjektiv geprägt sind und nicht jeder Nutzer alle Teilfunktionalitäten eines Programms als gleich wichtig wahrnimmt. Mit den oben genannten »Elefanten«, die für 1.0 sauber unterstützt werden, sind jedoch bereits wichtige Eckpfeiler abgesteckt. Grundsätzlich gilt: Je älter ein Programm ist, je weniger Abhängigkeiten von Hardware oder Fremdkomponenten es mitbringt, desto größer die Wahrscheinlichkeit, daß es sich unter Wine stabil nutzen läßt. Die Installation der Programme selbst bereitet mittlerweile nur noch in den seltensten Fällen Probleme, so daß sich der Versuch auf jeden Fall lohnt.

Nach wie vor problematisch ist allerdings die Wartung und Paketierung von unter Wine installierten Programmen. Sie erwarten als Default eine Verzeichnisstruktur unter `$HOME/.wine`, in der sie ähnlich zur Struktur unterhalb von C: unter Windows die Dateien und Registry-Informationen ablegen. Zwar kann man diesen Pfad durch Setzen der Umgebungsvariable `WINEPREFIX` problemlos verändern, jedoch erwartet Wine unglücklicherweise, daß das jeweilige Verzeichnis dem Wine startenden Benutzer gehört. Nutzen mehrere verschiedene User ein Programm, das man zum Beispiel unterhalb von `/opt/` installiert hat, ergeben sich daraus Probleme.

Um bei konstanter Wine-Version Updates zu unter Unix betriebener Windows-Software einzuspielen, kann die Vorversion mittels des mitgelieferten `uninstaller` entfernt und dann die neue Version eingespielt werden. Dieses Vorgehen ist in etwa vergleichbar zu dem unter Windows, integriert sich jedoch nicht in die Paketverwaltung der jeweiligen Distribution – Handarbeit für den Administrator ist hier also noch nötig. Alternativ kann auch kommerzieller Support erworben werden – die Seite des Entwicklers Dan Kegel

gibt hierzu einen guten Überblick (<http://www.kegel.com/wine/isv/#consultants>).

Daß Wine nach anderthalb Jahrzehnten bei der Version 1.0 angekommen ist, ist eine beeindruckende Leistung der Community. Perfekte Kompatibilität aller Windows-Applikationen mit Unix wird Wine prinzipbedingt wohl nie gewährleisten können. Für den stabilen Betrieb von Legacy-Applikationen unter Linux ist es jedoch eine gute Basis, die mit Erscheinen von Wine 1.0 noch weiter an Stabilität und Reife gewonnen hat. Auch das zukünftige Mo-

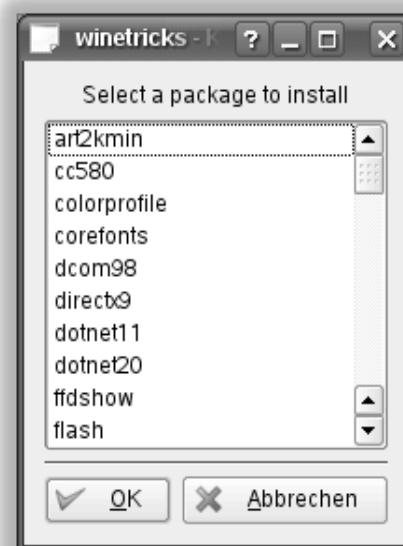


Bild 3: Das winetricks-Skript zur Installation von Windows-Systemteilen

dell mit einem stabilen und einem Entwicklungszweig sowie die konsequente Durchführung von Regressionstests trägt hierzu bei.

Was neue Features angeht, so fließen sie zunächst in den Entwicklungszweig der Version 1.1. Wesentliche Großbaustellen sind das Einbinden der immer häufiger aufzufindenden Win32-.NET-Hybridapplikationen und die schrittweise Implementierung einer DIB-Engine (siehe <http://wiki.winehq.org/DIBEngine>). ♦

David M. Gümbel berät Unternehmen und Kommunen bei der Einführung von Open Source-Lösungen und offenen Standards