

Wine FAQ

1. A propos de cette FAQ

1.1. Qui assure le suivi de cette FAQ ?

Dave Gardner l'a maintenu de 1995 à 1998.

Douglas Ridgway a pris le relais en 1999.

Andreas Mohr l'a converti en FAQ-O-Matic en 2000.

Dimitrie O. Paun, Keith Matthews et Tom Wickline (par ordre alphabétique) l'ont réorganisée en 2002.

Pour toute suggestion, ajout ou remarque concernant cette FAQ, n'hésitez pas à nous écrire à wine-faq@winehq.org (<mailto:wine-faq@winehq.org>)

1.2. Quel est le copyright de cette FAQ? Comment puis-je l'utiliser?

La FAQ originale de Wine, sur laquelle est basée la présente FAQ, a pour copyright © 1995-1998 David Gardner.

Elle peut être reproduite et modifiée selon les mêmes termes que Wine lui-même.

2. Questions générales sur Wine

2.1. Qu'est-ce que Wine et qu'est-il supposé faire ?

Wine est un programme qui permet d'exécuter les opérations DOS et les programmes MS Windows (Windows 3.x et exécutables Win32) sur les systèmes d'exploitation UNIX comme Linux. Il comprend un programme de chargement, qui charge et exécute un binaire Windows, et un ensemble de bibliothèques qui implémentent les appels aux API Windows en utilisant leurs équivalents UNIX ou X11. Les bibliothèques peuvent être également utilisées pour porter du code Win32 en exécutables UNIX, souvent sans beaucoup de changement au niveau du code source. Wine est un logiciel libre, et sa licence (qui se trouve dans le fichier LICENSE de chaque distribution) est la LGPL (GNU Lesser General Public License).

2.2. Est-ce que Wine émule complètement un ordinateur ?

Non, comme son nom l'indique, Wine n'est pas un émulateur [Wine Is Not a (CPU) Emulator]. Wine fournit simplement les API Windows. Cela signifie que vous aurez besoin d'un processeur compatible x86 pour exécuter une application Windows x86, par exemple un Intel ou un AMD. L'avantage est que, à la différence des solutions qui s'appuient sur une émulation CPU, Wine exécute des applications à pleine vitesse. Parfois un programme exécuté sous Wine sera plus lent

que s'il était exécuté sur une copie de Microsoft Windows, mais cela est principalement dû au fait que Microsoft a énormément optimisé des parties de son code, alors qu'essentiellement Wine n'est pas (pour le moment) bien optimisé. Occasionnellement, il arrive qu'une application tourne plus vite sous Wine que sous Windows. La plupart des applications tourne approximativement à la même vitesse.

2.3. Y-a-t-il d'autres solutions que Wine ?

Oui. Vous pouvez utiliser VMWare (<http://www.vmware.com>) pour exécuter une installation Windows au sein d'une machine virtuelle, ou utiliser Win4Lin (<http://www.win4lin.com>) pour exécuter une version spécialement adaptée de Windows sous Linux. Les deux solutions coûtent de l'argent pour à la fois le logiciel lui-même et la license Windows.

Notez que, comme Wine, ils peuvent seulement utiliser la plate-forme matérielle pour laquelle le programme cible a été compilé (voir ci-dessus).

2.4. Quelle est la différence entre Wine et les émulateurs matériels x86 ?

Ils existent deux émulateurs x86 libres: Bochs (<http://bochs.sourceforge.net>), et Plex86 (<http://savannah.nongnu.org/projects/plex86>).

Plex86 est un logiciel libre open-source offrant une solution de rechange pour VMWare, VirtualPC, et autre IA-32 sur IA-32 "Virtual PC products." Il peut uniquement fonctionner sur l'architecture IA-32.

Bochs est un émulateur de PC (x86) IA-32 open-source parfaitement portable écrit en C++, qui tourne sous les plate-formes les plus utilisées. Il permet l'émulation du CPU x86 d'Intel, des périphériques d'entrée/sortie courants, et d'un BIOS personnalisé. Actuellement, Bochs peut être compilé pour émuler un CPU 386, 486 ou Pentium. Bochs est capable de faire fonctionner la plupart des systèmes d'opérations par son émulation y compris Linux, Windows® 95, DOS, et récemment Windows® NT 4.

Les deux logiciels sont sous license GPL. Bochs est plus ancien que Plex86, semble être plus facile à installer, mais Plex86 se révélera plus rapide car il utilise un compilateur binaire en juste à temps.

L'inconvénient de tous les émulateurs est que vous avez besoin d'une version de Windows pour exécuter Windows, et qu'ils ont tous un impact sur la performance. Wine offre également une meilleure intégration du bureau- par exemple, les programmes utilisent votre gestionnaire de fenêtre standard, les icônes de la barre de lancement rapide [system tray] apparaîtront dans votre barre des tâches (si vous en avez une), et vous pouvez exécuter les programmes directement depuis la ligne de commande comme depuis les menus. La tablette électronique fonctionne aussi d'une façon homogène actuellement.

2.5. Quand Wine intégrera-t-il un émulateur CPU x86 de façon à faire tourner des applications Windows sur des machines non-x86 ?

Pour être bref, probablement jamais CPU. Souvenez-vous, Wine n'est pas un émulateur. Pour être plus précis disons que nous ne voulons probablement pas ou ne ressentons pas le besoin d'en intégrer un au sens traditionnel du terme.

Intégrer un émulateur CPU à Wine serait extrêmement difficile, à cause du grand nombre d'API Windows et de la complexité des types de données qu'elles échangent. Il n'est pas rare de voir dans une API Windows trois pointeurs ou plus vers des structures composées de nombreux champs, y compris des pointeurs vers d'autres structures complexes. Pour chacun d'entre eux, nous aurions besoin d'un mécanisme automatique de conversion qui puisse traiter les problèmes d'ordre des octets et d'alignement. De plus, Windows contient aussi de nombreux mécanismes de rappel automatique [callback] qui constituent autant d'occasions où nous aurions à essayer ces problèmes de conversion. Wine doit déjà faire avec les API 16 bits face aux API 32 bits et les API Ansi face aux API Unicode, des différences qui apportent leur lot de complexité. Ajouter un support pour les émulateurs CPU à l'intérieur de Wine compliquerait deux fois plus les choses et ne servirait qu'à ralentir le développement de Wine.

Heureusement, une autre solution existe pour faire fonctionner des applications Windows sous des plateformes non x86 : exécuter à la fois Wine et l'application au sein de l'émulateur CPU. Tant que l'émulateur offre un environnement UNIX standard, Wine devrait seulement avoir besoin de modifications minimales. La performance que vous perdez en faisant tourner Wine dans l'émulateur plutôt que nativement, vous la gagnez en complexité dans Wine. De plus, si l'émulateur est assez rapide pour faire tourner des applications Windows, Photoshop par exemple, alors il devrait être suffisamment rapide pour exécuter cette même application et Wine.

Deux nouveaux projets s'inscrivent dans cette optique : QEMU (<http://fabrice.bellard.free.fr/qemu/>), projet open-source, et Dynamite (http://www.transitives.com/tech_overview.htm), environnement commercial émulant un CPU de chez Transitives Technologies (<http://www.transitives.com/>).

2.6. Pourquoi tout le monde veut Wine? Est-ce que Windows ne craint pas?

Tout d'abord Wine n'est pas là pour faire tourner Windows mais pour faire tourner des applications Windows.

Par conséquent si tous vos besoins informatiques sont satisfaits par des applications natives UNIX, alors vous n'avez pas besoin de Wine et vous ne devriez pas l'utiliser. Cependant, si vous avez vraiment besoin d'une ou plusieurs des dizaines de milliers d'applications Windows, alors Wine est le meilleur moyen de les utiliser sans abandonner Unix. Jetons un oeil sur toutes les autres solutions possibles pour voir pourquoi:

Parmi celles-ci, la plus évidente, est d'utiliser le double démarrage. C'est la solution qui offre la meilleure compatibilité. Cependant cela implique d'acquérir une licence Windows et de dédier une

bon morceau de votre disque dur à Windows. Mais le pire est encore à venir. Chaque fois que vous voudrez utiliser cette application vous devrez redémarrer sur Windows. C'est particulièrement désagréable si cette obligation vient de l'extérieur et vous obligent à utiliser cette application (e.g le traitement du carte de crédit, email a récupérer depuis un serveur Lotus Notes). Vous serez alors obligé de fermer toutes vos applications Linux juste pour ouvrir cette application Windows. Vous serez sans doute vite exaspéré de cette situation, ou vous trouverez qu'une telle situation n'est pas acceptable dans le cadre du travail.

L'autre solution est d'installer un logiciel d'émulation de machine virtuelle tel que VMWare, Win4Lin ou Plex86. Vous pouvez alors utiliser des applications Windows sans déplorer une telle interruption. Mais cela implique également d'acquérir une license Windows et autant d'espace disque pour Windows. De plus vous devrez payer pour cet inconvénient supplémentaire : si puisque vous utilisez VMWare ou Win4Lin vous devez acheter une nouvelle license, et plus important vous devez dédiez maintenant une bonne partie de votre mémoire vive à la machine virtuelle. Les performances vont en prendre un sérieux coup également.

L'utilisation de Wine vous permet d'éviter toutes ces charges : la license Windows, l'espace disque-dur pris par Windows, les conséquences sur la mémoire et les performances dues à l'émulation de la machine virtuelle. Vous pouvez maintenant démarrer votre application Windows depuis votre environnement bureau habituel, placer la fenêtre de cette application côte à côte avec d'autres applications natives, faire des copier/coller de l'une vers l'autre, et l'exécuter à pleine vitesse.

C'est aussi un élément assez important pour la migration d'un organisme de grande taille, on ne change pas en une nuit changez la configuration de 5000 bureaux sans risque.

2.7. Puis-je utiliser Wine pour que les pilotes Windows de ma carte réseau / carte graphique / scanner / etc. marchent sous Unix ?

Le but de Wine est de rendre possible l'exécution d'applications Windows sous Unix, mais pas les pilotes Windows ou les VxD.

Les pilotes et les applications Windows appartiennent à des mondes différents. Les applications tournent en mode utilisateur et utilisent les API fournies par le noyau et les autres dll mode utilisateur. A l'inverse, les pilotes sont chargés dans le noyau Windows, c.à.d. au niveau 0 au lieu du niveau 3, ils doivent faire face à des problèmes spécifiques de gestion de mémoire, et utilisent des instructions qui ne sont pas disponibles pour les applications classiques. Cela signifie qu'ils ne pourraient tourner dans Wine puisque ce dernier s'exécute en mode utilisateur. Il vous faudrait alors modifier le noyau Linux. Mais par ailleurs, les pilotes utilisent une API complètement différente des applications Windows classiques. Le travail réalisé sur Wine ne serait donc d'aucune utilité pour un tel projet. En d'autres termes, rendre possible l'utilisation de pilotes Windows ou VxDs sous Unix serait un tout autre projet.

Cependant, si vous voulez réutiliser des pilotes Windows sous un système d'exploitation non-Microsoft nous vous recommandons la lecture de ReactOS (<http://www.reactos.com/>).

2.8. Lequel des packages Wine existant me convient ?

Actuellement il existe un large choix de différentes versions/packages de Wine:

Wine (<http://www.winehq.org>)

C'est la distribution "standard" de Wine. Sa license est la LGPL, il peut être téléchargé gratuitement. Le code source et les binaires sont disponibles dans la section "download" du site.

TransGaming's Cedega (<http://www.transgaming.com>)

Il s'agit de la version TransGaming de Wine spécialement faite pour les jeux. Elle comprend un support plus mature du Direct3D que WineHQ, bien qu'actuellement WineHQ soit également très avancé pour le support D3D. La plupart du code est sous l'AFPL et peut être téléchargé gratuitement.

Cependant TransGaming distribue également des binaires pour le support de protection améliorée des copies (requis pour de nombreux jeux), une assistance, et d'autres avantages. Ces packages sont seulement disponibles sous forme de binaires pour les clients inscrits (5\$/mois, minimum de trois mois).

CodeWeavers' CrossOver Office (<http://www.codeweavers.com/site/products/cxoffice/>)

C'est la version Wine avec un assortiment spécial pour s'assurer que la plupart des programmes de la suite Office fonctionnent correctement. Elle coûte 74.95\$ pour la version Pro et 39.95\$ pour la version Standard. Elle semble véritablement valoir le coup, jusqu'à maintenant, si l'on s'en réfère à certains commentaires. (note: vous apporterez votre soutien à une entreprise contribuant activement à Wine si vous décidez d'acheter CrossOver.)

CodeWeavers' CrossOver Office Server Edition
(<http://www.codeweavers.com/site/products/cxserver/>)

Elle vous permet d'exécuter vos applications de productivité Windows favorites dans un environnement client léger distribué sous Linux. La Server Edition est aussi un bon ajout aux environnements Solaris, puisque le support intégré pour les bureaux Solaris permet de faire tourner les applications sur les stations de travail Sun. Pour les tarifs veuillez vous référer au lien suivant: Tarifs de la CrossOver Office Server Edition (<http://www.codeweavers.com/site/products/pricing/>)

2.9. Quelle est l'histoire de Wine ?

Le projet Wine a commencé en 1993 dans le but de supporter des programmes tournant sous Windows 3.1 pour Linux. Bob Amstadt a été le coordinateur original, mais a cédé assez rapidement la place à Alexandre Julliard, qui est toujours en place. Un groupe d'actualités (news:comp.emulators.ms-windows.wine) a été créé en Juillet 1994. Au fil des ans, des ports pour

d'autres Unix ont été ajoutés, avec le support pour Win32 quand les applications Win32 sont devenues très répandues.

Pour plus d'informations, voyez <http://www.winehq.com/site/history>
(<http://www.winehq.com/site/history>)

2.10. Quelle est la version actuelle de Wine ?

Une nouvelle version de Wine sort à peu près chaque mois. Vous pourrez vous tenir au courant de toutes les dernières versions en lisant la newsgroup `comp.emulators.ms-windows.wine` (`news:comp.emulators.ms-windows.wine`), ou en visitant la page d'accueil Wine HQ (<http://www.winehq.org>). Quand vous téléchargez Wine depuis le site FTP de votre choix (voir la page de téléchargement (<http://www.winehq.org/site/download>) pour certains de ces choix), vous pouvez vérifier que vous récupérez bien la dernière version en regardant le numéro de version dans le nom du fichier. Par exemple, la distribution réalisée le 13 août 2004 s'appelle `Wine-20040813.tar.gz`. Les fichiers correctifs [patch] sont aussi disponibles. Si vous avez la précédente version, vous pouvez télécharger et appliquer le correctif actuel plutôt que de réinstaller entièrement la dernière distribution. Le nom du correctif suit les mêmes conventions que la distribution mensuelle. Un accès en lecture du CVS (<http://www.winehq.org/site/cvs>) est également disponible.

2.11. Quel est le statut actuel de Wine ?

À mi 2004, Wine c'est 1,6 million de lignes de code, écrites par plus de 600 développeurs de douzaines de pays à travers le monde. On estime que Wine est utilisé par 100 000 personnes. Wine implémente plus de 90% des appels dans les spécifications Windows répandues telles que ECMA-234 et Open32.

Vous pouvez également vous référer à la page du statut (<http://www.winehq.org/site/status>) pour avoir une vision globale de l'avancement des implémentations de Wine.

2.12. Quand Wine sera-t-il fini ?

Les grands projets informatiques ne sont jamais finis, ils n'existent que sous forme de versions. En tout cas, Wine court après une cible mouvante car chaque nouvelle révision de Windows contient de nouveaux appels aux API ou des variations sur celles existantes.

Puisque Wine est développé par des bénévoles, il est difficile de prédire quand il sera prêt pour une version générale. Mais l'intérêt croissant des entreprises pour porter des applications via Wine fait que, le développement de ce dernier est de plus en plus actif. À l'heure actuelle, nous sommes en train de travailler sur le lancement de Wine 0.9 Real Soon Now(tm).

2.13. Qui est responsable de Wine ?

Wine existe grâce au travail de nombreuses personnes. Voyez le fichier AUTHORS (<http://source.winehq.org/source/AUTHORS>) dans la distribution pour consulter. Parmi les

entreprises qui sont ou qui ont été impliquées dans le développement de Wine on trouve CodeWeavers, TransGaming, Corel, et Macadamian.

2.14. Parmi les API/interfaces non documentées, lesquelles ne le sont pas parce qu'elles sont difficiles à comprendre? Est-ce que regarder les sources de Microsoft pourrait aider ?

L'idéal serait que les API Windows soient complètement documentées; Wine serait alors une implémentation parfaite et nickel chrome. Avec un accès au code source, il pourrait être plus difficile de prouver qu'aucun copyright n'a été violé. Ceci dit, la documentation est souvent mal faite, inexistante, et même erronée quand elle existe, ce qui fait qu'un taux important de retro-engineering a été nécessaire, particulièrement dans l'interface shell. Le problème majeur concernant Wine est cependant le manque de ressources humaines. A un moment donné, plus de 5000 personnes travaillaient sur Windows 2000. Alors que Wine n'a pas besoin de répliquer Windows entièrement (nous reprenons simplement les parties nécessaires pour faire tourner les programmes Windows), ce chiffre, qui représente le nombre de personnes mobilisé pour une seule version, est 10 fois supérieur au nombre de personnes ayant travaillé sur Wine jusqu'alors, depuis le début du projet.

2.15. Est-ce que le dernier correctif de TransGaming est inclu dans la version Wine standard ?

Non.

TransGaming gagne de l'argent via un service d'abonnement et la license de son arbre Cedega est incompatible avec la license Wine. Les correctifs Cedega ne peuvent pas ainsi être intégré dans l'arbre de Wine sans la permission expresse de TransGaming. Ils ont proposé une partie de leur travail pour une intégration dans Wine, notamment DirectDraw et une partie de DirectSound, et celui-ci a été intégré dans l'arbre de Wine. Cependant, il semble très improbable qu'ils ne proposeront un jour leur travail sur Direct3D.

2.16. Y aura-t-il une version Windows de Wine ?

Certaines personnes sont en train de travailler sur la portabilité de Wine pour qu'il puisse être compilé sous Windows en utilisant l'un des projets suivants comme base :

- Cygwin (<http://www.cygwin.com> (<http://www.cygwin.com/>))
- MinGW (<http://www.mingw.org> (<http://www.mingw.org/>))
- ReactOS (<http://www.reactos.com> (<http://www.reactos.com/>))

Il y a des progrès, une version de Wine utilisable sous Wine devrait donc être disponible dans un futur proche.

La logique pour ces projets est en partie de trouver des zones où la portabilité de Wine est absente. C'est particulièrement vrai pour le projet ReactOS qui est une réimplémentation du noyau Windows et qui devrait ainsi être capable de réutiliser la plupart des dll de Wine.

Une autre raison de mener ces projets est la possibilité de remplacer une simple dll Windows par son alter ego Wine. Outre que c'est un bon test pour la dll Wine, cela nous laisse la possibilité de détecter les cas où nous nous trompons au sujet de l'interaction des dll entre elles.

2.17. Puis-je utiliser les pilotes d'impression Windows dans Wine?

Les pilotes d'impressions natifs ne sont pas supportés. Il fût un temps où Wine supportait les pilotes natifs 16bit mais cela remonte à longtemps. Wine utilise les imprimantes (et autres périphériques) installées sur votre système d'exploitation. La plupart du temps, si vous n'avez pas le matériel installé sur votre SE alors Wine ne peut pas l'utiliser.

3. De quoi ai-je besoin pour utiliser Wine?**3.1. Sous quelle(s) plateforme(s) et système(s) d'opération tournera Wine(Lib) ?**

Wine est développé pour tourner spécifiquement avec une classe de CPU/*Intel x86* sous certains UNIX qui tournent sous cette plate-forme. Winelib est cependant capable de porter le *code source* des applications Windows également vers d'autres plate-formes, et pas seulement vers les x86.

Ainsi faire tourner des binaires Windows sous d'autres plate-formes (par exemple: Mac OS X sous PowerPC) en utilisant seulement Wine est *impossible*. Il vous faudrait soit faire tourner Wine dans un environnement émulé x86 ou récupérer le code source de l'application Windows et le recompiler en utilisant Winelib.

Voici les plate-formes supportées par Wine. La compatibilité Winelib avec d'autres plate-formes est sans cesse en évolution, donc il n'en est pas fait spécialement mention dans cette liste.

NetBSD, OpenBSD, UnixWare, et SCO OpenServer 5 marchaient à une époque, mais Wine nécessite maintenant des unités d'exécution au niveau du programme résident [kernel-level threads] qui ne sont pas actuellement disponibles (ou pas bien maîtrisés par l'équipe Wine) sur ces plate-formes.

L'équipe développement Wine espère également attirer l'intérêt d'autres revendeurs d'UNIX commercial et clone d'UNIX.

BeOS: efforts de portage assez sérieux autrefois (BeWine), mais BeOS connaît de sévères limitations dans le support d'appel Unix. La disparition de Be a généré le projet bien qu'il puisse revenir un jour parmi un des projets ouverts BeOS. En tout cas, il apparaît peu probable qu'un port fonctionnel voit le jour en l'état actuel.

Mac OS X / Darwin: Le Darwine (<http://darwine.sourceforge.net/project.html>) s'occupe actuellement du portage de Wine sur la plate-forme Darwin/x86. Leur but est de rendre finalement possible l'exécution d'applications x86 Windows sur Darwin/PPC puis ensuite sur Mac OS X en utilisant Bochs.

FreeBSD: Le suivi de ce portage est bien assuré malgré quelques limites dans certaines situations précises(principalement en l'absence de support périphérique/matériel).

Linux/x86: Pas de problème, et en tant que plate-forme préférée des développeurs et utilisateurs, c'est la plate-forme la mieux supportée de toutes.

3.2. Quelle puissance CPU minimale doit avoir mon ordinateur pour que Wine et les applications MS Windows tournent bien?

Nous devons différencier ici Wine et Winelib.

Wine ne tournera sur aucun CPU x86 inférieur au 80386 à cause des limitations de gestion d'adresse.

Il a également la réputation de tourner sur les 80486 et les CPU compatibles suivants. Le test simple est, si vous pouvez faire tourner X11 actuellement, vous devriez alors pouvoir faire tourner Wine et des applications MS Windows.

Comme toujours, plus votre CPU est puissant, mieux c'est. Avoir un coprocesseur math n'est pas important. Cependant, avoir une carte graphique avec accélération 3D supportée par le X sera un plus.

En fonction de votre application vous trouverez probablement que des vitesses plus grandes sont requises pour une utilisation judicieuse. On ne peut pas donner de conseils spécifiques sur ce sujet à cause du vaste choix des applications disponibles. Cependant la règle de base est que si votre application tourne correctement sous Windows, elle devrait tourner correctement sous la même plate-forme avec Wine.

3.3. Quel espace le code source de Wine et les binaires prendront-ils sur mon disque dur?

Il vous faut à peu près 250 mégaoctets d'espace libre sur le disque dur pour stocker et compiler le code source. Wine requiert également 18 Mo dans votre répertoire /tmp. Et environ 50 Mo sont requis pour faire un "make install".

Les paquets binaires, spécialement ceux ne contenant pas d'information sur le debugage, ont moins besoin d'espace disque, en général aux alentours de 20Mo.

3.4. De quel autre logiciel ai-je besoin pour installer, compiler et faire tourner Wine?

De nombreux outils de développement sont nécessaires pour compiler Wine. Une liste des paquets requis pour de nombreuses distributions se trouve dans le fichier README (<http://source.winehq.org/source/README> (<http://source.winehq.org/source/README>)).

Pour lancer Wine, vous aurez besoin des éléments suivants:

- Le binaire Wine compilé
- Un fichier wine.conf parfaitement configuré (ou le fichier ~/.winerc)

- Un system X Window installé et en état de marche
- Des programmes Windows pour faire des tests

3.5. De combien de RAM ai-je besoin sur mon système UNIX pour que Wine et les applications MS Windows tournent bien?

Si vous arrivez à faire tourner le X de manière fluide sur votre système UNIX actuellement, vous devriez pouvoir faire tourner Wine et des applications MS Windows tout aussi bien, en fonction de la gourmandise en RAM de l'application.

Les besoins mémoire de Wine dépendront de l'application ou du jeu que vous avez choisi d'exécuter. Il vous faudra répondre au besoin minimal de l'application ainsi que la surcharge de votre système d'exploitation sous-jacent. Vous pourrez voir avec le revendeur pour les besoins en mémoire indiqués pour cette application.

3.6. Combien de temps faut-il pour compiler Wine?

Wine tend à devenir assez important, et pour le compiler intégralement il faut un grand nombre de processus. Depuis mai 2004, les temps de compilation étaient d'environ 10 minutes sur un Athlon 2000 avec 512 Mo de RAM et 20 minutes sur un Athlon 1200 avec 640 Mo de RAM. Si vous avez une copie CVS de Wine, vous n'aurez sans doute pas besoin de tout reconstruire à chaque mise à jour.

3.7. J'ai une partition Drivespaced, Doublespaced ou Stackered. Est-ce que Wine peut exécuter des binaires MS Windows situés dans une de ces partitions?

Oui, mais seulement si le système d'opération supporte le montage de ces types de disques. Il y a un pilote de système de fichier Linux nommé dmsdos qui permettra les accès lecture/écriture vers des lecteurs Doublespaced et Drivespace 1.0. Plus spécifiquement, il supporte le montage de partitions compressées DOS 6.0 et 6.2 Doublespaced, DOS 6.22 Drivespaced, et Windows 95 Doublespaced (l'accès en lecture et en écriture marche correctement, mais l'accès en écriture est lent). On peut le trouver à <ftp://metalab.unc.edu/pub/Linux/system/filesystems/dosfs/> (<ftp://metalab.unc.edu/pub/Linux/system/filesystems/dosfs/>)

3.8. Ai-je besoin d'une partition DOS sur mon système pour utiliser Wine?

Vous n'avez pas besoin d'une copie installée et sous license de DOS ou de MS Windows pour installer, configurer et utiliser Wine. Cependant, Wine doit pouvoir 'voir' un binaire MS Windows (c.à.d. l'application) si elle doit être exécutée.

3.9. MS Windows doit-il être chargé dans cette partition pour exécuter des programmes MS Windows sous Wine?

Beaucoup de monde a installé et exécuté, avec succès, des programmes dans leur système de fichiers UNIX sans avoir une partition DOS ou MS Windows. Cependant, dans beaucoup de cas, il faut une infrastructure de répertoires et fichiers similaire à une installation existante Windows. Certains programmes d'installation d'applications veulent envoyer les fichiers de package dans les dossier /windows et /windows/system pour s'exécuter, et à moins que ceux-ci n'existent sur système de fichiers UNIX, ces programmes ne s'installeront pas correctement et ne tourneront

probablement pas bien, s'ils arrivent à tourner. La plupart des packages régleront cela pour vous en tant que partie du programme d'installation.

Si vous avez une partition DOS avec MS Windows installée dessus, assurez vous que votre système UNIX peut 'voir' cette partition (vérifiez votre fichier `/etc/fstab` ou montez la partition à la main) pour que Wine puisse exécuter les binaires MS Windows situés sur la partition DOS. Pour l'exécution sans une partition DOS, il faut fixer un chemin UNIX correspondant au lecteur C, et s'assurer que les dossiers `/windows` et `/windows/system` pointent sur un endroit qui existe bel et bien.

Voici un exemple, copié depuis une machine qui n'a pas de partition DOS mais qui exécute avec succès Wine:

```
[Drive C]
  Path=/var/lib/wine
  Type=hd
[wine]
  Windows=c:\windows
  System=c:\windows\system
  Temp=e:\
  Path=c:\windows;c:\windows\system;c:
```

Dans `/var/lib/wine/windows`, il vous faudra installer un fichier de configuration `win.ini` que vous pourrez trouver sur une machine MS Windows 3.1 classique. Le dossier `/var/lib/wine/windows/system` devrait exister, mais il ne sera pas nécessaire qu'il contienne quoique ce soit. Cependant, pour utiliser des MS DLL, vous pouvez les copier dans ce dossier. Notez que ceci est une entorse à la licence Windows à moins que Windows soit installé sur la machine. Si vous avez un DOS/MS Windows installé sur votre système, vous pouvez monter cette partition au démarrage en modifiant le fichier `/etc/fstab` dans votre partition UNIX (en supposant que le noyau UNIX supporte le type de système de fichiers DOS/MS Windows).

Si vous éditez ce fichier à la main, il devrait afficher quelque chose du genre :

```
/dev/hda1 /dosd msdos uid=0,gid=100,umask=007 0 0
```

Cela vous permettra de lire et d'écrire sur la partition DOS sans être le root.

3.10. Si Wine remplace complètement MS Windows, est-ce qu'il dupliquera toutes les fonctions de MS Windows?

L'objectif de Wine est de rendre possible l'exécution d'applications Windows sous Unix. Dans cette perspective, il fournira des remplacements juste pour les DLL et API qui sont requises pour ces applications Windows. Cela signifie que Wine ne fournira aucun remplacement pour les DLL qui ne sont pas fournies avec Windows ou qui sont toujours fournies avec une application

Windows (par exemple l'exécutable de Visual Basic). Cela signifie également que l'implémentation d'une API qu'aucune application n'utilise jamais n'est pas une priorité. De la même manière, tant que des applications utilisant l'API Win64 ne sont pas disponibles, cela ne sera pas une priorité. Ceci dit, on essaiera certainement de garder des portes ouvertes et d'améliorer notre couverture d'API autant que possible.

Comme Wine n'est pas non plus un SE, écrire des pilotes de matériel ne fait pas partie des objectifs de Wine. Cependant, si vous êtes intéressé par les pilotes de matériel, les développeurs de noyau sur Linux (<http://www.kernel.org/>), FreeBSD (<http://www.freebsd.org/>) et ReactOS (<http://www.reactos.com/>) apprécieront certainement votre contribution.

De la même manière, Wine n'essaie pas d'être un environnement de bureau donc fournir des applets telle qu'une calculatrice, un gestionnaire de fichier ou même un gestionnaire de fenêtre qui ressemble à Windows n'est pas prioritaire ou serait même mieux fait dans un projet à part. De tels projets feraient double emploi pour une grande partie redondants avec d'autres projets open-source. Là encore, ils existent des projets où votre aide dans ce domaine seraient certainement appréciée, tels que les environnements de bureau Gnome (<http://www.gnome.org/>) ou KDE (<http://www.kde.org/>). Vous aurez en plus la satisfaction de voir que votre contribution est utile à tous et pas seulement aux utilisateurs de Wine.

3.11. Pourrai-je installer des applications MS Windows sur tout système UNIX quel qu'il soit?

Wine est écrit pour être indépendant du système de fichiers, donc les applications MS Windows s'installeront et s'exécuteront sous quasiment n'importe quel système de fichiers supporté par votre marque d'UNIX.

3.12. Est-ce que Wine ne s'exécutera que sous le X, ou peut-il s'exécuter en mode texte?

La plupart des efforts de développement de Wine est tournée vers l'interface graphique utilisateur de MS Windows, mais on a découvert une compatibilité, limitée, avec le mode de texte, en configurant `GraphicsDriver=ttydrv` dans la section `wine.conf [wine]`.

L'infrastructure de Wine est déjà relativement prête à supporter d'autres pilotes graphiques que `x11drv`, mais aucun pilote graphique 'de rechange' n'a encore été développé.

3.13. Est-ce que Wine tournera sur n'importe quel gestionnaire de fenêtres X? Est-ce qu'il nécessite un gestionnaire de fenêtre particulier?

Wine est indépendant du gestionnaire de fenêtre, donc le gestionnaire de fenêtres X que vous choisissez de lancer n'a presque aucune incidence sur votre capacité à exécuter des programmes MS Windows sous Wine. Wine utilise des bibliothèques X standard, aucun ajout n'est donc requis. Wine a son propre gestionnaire de fenêtres, qui agit comme MS Windows. Il peut être arrêté pour utiliser le gestionnaire de fenêtres natif en modifiant la configuration de `Managed` ou `Desktop` comme décrit dans **man wine.conf**.

3.14. Est-ce que les applications 32-bit de Windows 95/98/ME/NT/2000/XP tourneront sous Wine?

Oui, les programmes 32-bit sont bien supportés.

4. Récupérer Wine

4.1. Où puis-je récupérer Wine?

A cause de décalages dus à l'utilisation d'un site miroir, vous pourriez avoir vent de la dernière révision avant même que la révision ne soit disponible sur les sites ftp dans la liste ci-dessous. Les sources sont disponibles depuis les sites suivants:

- http://sourceforge.net/project/showfiles.php?group_id=6241&package_id=77449
(http://sourceforge.net/project/showfiles.php?group_id=6241&package_id=77449)
- <http://www.ibiblio.org/pub/Linux/ALPHA/wine/development/>
(<http://www.ibiblio.org/pub/Linux/ALPHA/wine/development/>)
- <ftp://ftp.infomagic.com/pub/mirrors/linux/sunsite/ALPHA/wine/development/>
(<ftp://ftp.infomagic.com/pub/mirrors/linux/sunsite/ALPHA/wine/development/>)
- <ftp://ftp.fu-berlin.de/unix/linux/mirrors/sunsite.unc.edu/ALPHA/wine/development/>
(<ftp://ftp.fu-berlin.de/unix/linux/mirrors/sunsite.unc.edu/ALPHA/wine/development/>)
- <ftp://orcus.progsoc.uts.edu.au/pub/Wine/development/>
(<ftp://orcus.progsoc.uts.edu.au/pub/Wine/development/>)

Il devrait être également disponible sur tout autre site miroir de [ibiblio.org](http://www.ibiblio.org), voyez <http://www.ibiblio.org/pub/Linux/MIRRORS.html>. Certains de ces sites archivent parfois des versions antérieures de Wine ainsi que la version courante. Pour savoir quelle est la dernière version, jetez un oeil sur le nom du fichier, qui aura la forme Wine-AAAAMMJJ.tar.gz. Remplacez simplement AAAAMMJJ dans le nom de la distribution par les chiffres respectivement de l'année, du mois et du jour. Le plus récent est celui à récupérer.

Les packages binaires de Wine sont disponibles pour différents SE et distributions. Voyez la page de téléchargement (<http://www.winehq.org/site/download>) pour trouver la liste la plus récente.

4.2. Y-a-t-il un arbre CVS?

Les sources actuelles de Wine sont également disponibles via un accès anonyme client/serveur sur le CVS. Il vous faudra la version 1.9 du CVS ou toute autre version supérieure. Si vous êtes derrière un parefeu, il vous faudra soit ouvrir le port 2401 de votre parefeu soit utiliser le protocole SOCKS.

Pour vous connecter au serveur CVS, faites

```
export CVSROOT=:pserver:cvs@cvs.winehq.org/home/wine
cvs login
```

Utilisez "cvs" comme mot de passe (sans les guillemets). Notez que /home/wine est un chemin sur le serveur, pas sur votre machine. Pour vérifier entièrement l'arbre source de Wine (ce qui peut être très lent), utilisez

```
cvs -z 3 checkout wine
```

ou si vous voulez juste un sous-arbre, ou un fichier particulier, vous pouvez faire

```
cvs -z 3 checkout wine/ANNOUNCE
```

Il faut savoir, cependant, que récupérer entièrement l'arbre source de Wine via CVS est assez lent, spécialement si l'on compare par rapport à la récupération depuis un miroir FTP situé près de chez vous. Pour une liste de sites miroirs CVS, voyez <http://www.winehq.org/site/cvs#cvsservers> (<http://www.winehq.org/site/cvs#cvsservers>)

Des fichiers correctifs sont aussi disponibles, pour que vous n'ayez pas à télécharger, installer et configurer la distribution entière chaque mois si vous êtes à jour dans la version précédente. Le nom des versions des fichiers patch suit la même règle que les révisions générales, et prend donc la forme

Wine-AAAAMMJJ.diff.gz

Les correctifs sont disponibles sur les mêmes sites que ceux qui distribuent la version complète. Pour se mettre à jour avec une nouvelle révision avec un correctif, changez d'abord de dossier jusqu'au répertoire racine de la révision (celui contenant le fichier README), faites ensuite un "make clean", et mettez à jour la version avec

```
gunzip -c patch-file | patch -p1
```

où patch-file est le nom du fichier correctif sous la forme Wine-AAAAMMJJ.diff.gz. Vous pouvez ensuite relancer ./configure, et ensuite exécuter make depend && make

Si vous voulez créer un site miroir de la distribution Wine depuis le site tsx-11 et que vous voulez faire partie de la liste de cette FAQ, veuillez l'ajouter dans la partie "things to go into the documentation".

4.3. Puis-je récupérer Wine en utilisant cvsup?

Les miroirs CVS n'offrent pas encore le support cvsup, mais le serveur principal oui. Utilisez le fichier wine.sup avec:

```
*default host=cvs.winehq.org
```

```

*default base=/cvs
*default prefix=/cvs/wine
*default release=wine
*default delete

# If your network link is a T1 or faster, comment out the following line.*
# *si votre connection Internet est une ligne T1 ou supérieure, commentez la ligne suiv
#*default compress

*default use-rel-suffix
wine

```

5. Installation et configuration de Wine

5.1. Comment dois-je compiler le code source de Wine?

La marche à suivre se trouve dans le fichier README (<http://source.winehq.org/source/README>) pour toutes les instructions. Par ailleurs, si vous le souhaitez vous pouvez régler la variable d'environnement `TMPDIR` **TMPDIR=~/tmp** ou **TMPDIR=/tmp** (si vous êtes root).

5.2. Comment installer Windows dans Wine sous Linux?

Réponse simple: IMPOSSIBLE. Windows a besoin d'un accès direct au matériel, ce qui n'est pas possible si Wine et UNIX se trouvent en travers de la route.

A la base Wine est censé être utilisé SANS Windows. Si vous voulez utiliser une installation Windows, utilisez alors une installation existante en parallèle de l'installation UNIX (vous trouverez plus d'informations dans le HOW-TO sur le double démarrage [dual-boot HOWTO] concernant votre SE). Ou utilisez autrement l'utilitaire `cabextract` pour extraire les archives d'installation Windows vers un répertoire que vous voulez utiliser comme arbre Windows de Wine.

5.3. Comment dois-je configurer Wine pour qu'il tourne sur mon système?

Wine vous oblige à avoir un fichier de configuration comme `~/ .wine/config`. Le format de ce fichier est expliqué dans la page d'aide [man page] `wine.conf`. Le fichier `documentation/samples/config` (<http://source.winehq.org/source/documentation/samples/config>) (<http://source.winehq.org/source/documentation/samples/config>) contient un exemple de fichier de configuration. Des conseils plus explicites se trouvent dans le fichier README (<http://source.winehq.org/source/README>) (<http://source.winehq.org/source/README>) qui se trouvera dans le répertoire de base de Wine une fois que vous aurez décompressé le fichier de distribution avec l'une des deux commandes `gunzip` ou `untar`.

5.4. Comment puis-je améliorer Wine sans perdre la configuration en place?

L'amélioration de l'installation de Wine n'affecte pas la configuration de Wine existante. Par conséquent après l'amélioration de Wine vous avez toujours (en place) votre ancienne configuration.

5.5. Si je veux utiliser une installation de Windows, quelles versions marchent bien?

Utilisez soit une installation classique non-Windows (Wine s'améliore de jour en jour) soit une installation Win9x (Win95, 98, 98SE, ME). NE configurez PAS Wine pour utiliser une installation Windows basée sur la technologie NT (NT, Win2K, WinXP, Win2K3).

En général, la plupart des installations Windows contiennent une quantité monstrueuse de saloperies qui peut perturber Wine et le rendre moins fiable.

5.6. Si j'utilise une installation Windows avec Wine, laquelle marche le mieux?

Au 02/2002:

Je dirais que Win98SE est la meilleure version à utiliser avec Wine, puisqu'elle est pas mal répandue chez les développeurs et qu'elle est relativement mature. L'utilisation de fichiers Win2K est *assurément* pire qu'une installation de Wine sans fenêtre, et Win ME est connu pour être problématique, également (vu qu'aucun développeur ne l'utilise). Pour faire bref: tout Win9x <= W98SE est bon.

5.7. Impossible de faire tourner/d'exécuter l'installation d'applications générées par Visual Basic. Que faire?

Assurez vous que vous avez bien toutes les bibliothèques d'exécution VB d'installées. Vous pouvez obtenir la dernière version depuis le site de Microsoft.

5.8. Quand je clique sur un fichier *.exe dans mon gestionnaire de fichier, rien ne se passe.

Les versions normales de Wine n'ont pas encore d'extensions .exe enregistrées pour Wine dans KDE/Gnome. Vous devez à la place ouvrir une fenêtre de terminal (souvent une icône représentant un "écran noir") et taper quelque chose ressemblant à ça:

```
cd /mon/répertoire/de/programmes/windows  
wine monprogramme.exe
```

5.9. bash me répond "wine: Command not found" Que dois-je faire?

Essayez de vous déconnecter puis de vous reconnecter sous bash. Cela peut résoudre le problème.

Si ce n'est pas le cas, alors assurez-vous que le binaire wine est dans votre *PATH*.

Lancez en tant que root:

```
find / -name "wine" -type f -perm +111
```

pour trouver le chemin où le binaire Wine se trouve. Vérifiez ensuite que *PATH* l'inclut:

```
echo $PATH
```

Sinon, ajoutez le par exemple dans */etc/profile* en faisant:

```
export PATH=$PATH:/chemin/du/binaire/wine
```

Cela devrait vous dépanner.

Si vous avez utilisé un gestionnaire de paquets (**rpm** ou **apt**) - Vérifiez vos paquets. Le paquet `winesetuptk.rpm` est seulement une interface [front-end] pour créer un fichier de configuration qui tienne la route, il N'INSTALLE PAS le paquetage Wine...

Pour les paquets complets, utilisez <http://rpmseek.com/> (<http://rpmseek.com/rpm-pl/wine.html?hl=com&cx=0::>) ou la section Download (<http://www.winehq.org/site/download>).

5.10. Comment désinstaller Wine de mon ordinateur?

Cela dépend de la façon dont vous l'avez installé. Si vous avez utilisé un RPM, la commande à faire est celle-ci: **rpm -e wine (en tant que root)**

Si vous l'avez installé depuis la source (le fichier `.tar.gz`), la bonne façon de faire est d'aller à la racine de l'arbre source (le répertoire avec le script de configuration, le fichier `readme` etc) et de lancer ensuite (en root): **make uninstall**

6. A propos de l'exécution de Wine

6.1. Comment lancer un programme MS Windows sous Wine?

En invoquant Wine, vous devez spécifier le chemin entier de l'exécutable, ou le nom de fichier seulement. Par exemple pour lancer le solitaire de Windows tapez l'une des commandes suivantes:

- **wine sol** ou **wine sol.exe** (en utilisant le chemin de recherche pour localiser le fichier).
- **wine c:\\windows\\sol.exe** (en utilisant le nom de fichier DOS).
- **wine /usr/windows/sol.exe** (en utilisant le nom de fichier UNIX).

- **wine "c:\windows\sol.exe"** (en utilisant le nom de fichier DOS entre guillemets).

Le chemin du fichier sera aussi ajouté au chemin [path] quand un nom complet est utilisé en ligne de commande.

6.2. J'ai installé et configuré Wine, mais Wine ne trouve pas MS Windows sur mon disque. Où est l'erreur?

Si vous avez une partition DOS, assurez vous d'abord que vous l'avez montée, soit en rajoutant l'entrée dans le fichier `/etc/fstab`, ou en la montant à la main.

Rappelez vous également qu'à moins que votre version d'UNIX puisse voir à travers, ou que vous utilisiez un utilitaire qui puisse voir à travers, votre partition DOS ne doit pas se situer sur une partition Drivespaced, Doublespaced ou Stackered, car par nature ni Linux, ni FreeBSD, ni NetBSD ou Wine ne peuvent 'voir' les fichiers situés sur ces partitions DOS compressées.

Vérifiez la syntaxe de votre chemin dans le fichier `wine.conf`. Les lettres capitales ne doivent pas être utilisées dans les chemins, car elles sont automatiquement converties en minuscules.

6.3. J'ai réussi à exécuter différents programmes MS Windows, mais tout ne marche pas dans ces programmes. Qu'est ce qui cloche?

Wine n'est pour le moment pas complet, par conséquent dans chaque programme certaines caractéristiques peuvent ne pas fonctionner. Cela finira par se régler car de plus en plus d'appels API MS Windows inclus dans Wine.

6.4. J'ai lancé différents programmes MS Windows, mais comme les menus de ces programmes ne fonctionnent pas, comment quitter ces programmes?

Arrêtez la fenêtre xterm shell que vous avez utilisé pour lancer le programme MS Windows, et la fenêtre X qui est apparue avec le programme s'arrêtera également.

6.5. Mon programme ne marche pas, que faire?

Si vous êtes un développeur et que vous avez des connaissances en C, alors vous pouvez commencer à déboguer Wine et nous aider à l'améliorer! Si ce n'est pas le cas, vous devrez alors soit convaincre un développeur Wine d'essayer de faire marcher votre programme (il doit y avoir une version ou une démonstration téléchargeable pour cela).

Vous pouvez soumettre votre application à la base de données Wine Application DB (<http://appdb.winehq.org/>) et récupérer des tuyaux sur la façon d'améliorer le fonctionnement de votre application.

Vous pouvez également soumettre votre application au CodeWeavers CrossOver Compatibility Center (<http://www.codeweavers.com/site/compatibility/>). Où vous pouvez voter pour soutenir votre application favorite.

Egalement, il se peut que vous puissiez faire marcher l'application en prenant les DLL natives depuis une installation Microsoft Windows, et en les utilisant (régler les DLL à natif dans le fichier de configuration). Toutes les DLL ne peuvent pas être remplacées de la sorte - en particulier DirectX pas plus que certaines DLL du système central [core system] comme gdi32, user, ntdll, kernel32 etc.

6.6. Puis-je utiliser Wine avec SUSE, RedHat ou d'autres distrib Linux?

Vous pouvez utiliser Wine sur n'importe quelle installation Linux suffisamment récente. La quantité de travail pour mettre Wine en service et l'utiliser varie selon que vous l'installez avec les paquets binaires ou que vous faites une installation depuis la source.

6.7. Est-ce que Wine marche avec des processeurs AMD?

Oui. Wine devrait marcher avec n'importe quel processeur compatible Pentium ou plus.

6.8. Puis-je lancer un programme Unix depuis un programme Windows?

Bien sûr, Wine supporte cela. Entrez juste le nom du programme Unix là où le programme propose l'exécution de quelque chose, et cela devrait marcher.

6.9. J'obtiens "Error installing iKernel.exe: (0x1400)" quand je lance un installeur InstallShield 6.

Si vous obtenez l'erreur "Error installing iKernel.exe: (0x1400)" à chaque fois, c'est probablement parcequ'ils restent certaines processus d'une précédente tentative. Vous pouvez le vérifier avec la commande

```
$ ps auxxw | grep wine
```

Si cette commande affiche d'anciennes copies de Wine lançant votre installation, vous devez les arrêter [kill] avant de pouvoir lancer le programme d'installation. S'il n'y a pas d'autres programmes Wine en cours d'exécution, vous pouvez tous les arrêter [kill] avec la commande

```
$ killall wine
```

Si vous être en train de faire tourner des programmes Wine auquel vous tenez, vous devrez arrêter les anciennes instances d'installation une à une en utilisant la commande kill et les PID de chaque processus PID (ou peut-être le très chic Gestionnaire de tâches de Wine qui n'existe pas encore).

Répétez la commande **ps** pour vous assurer que tous les anciens processus Wine sont clos.

7. Récupérer de l'aide

7.1. Y-at-il de la documentation sur Wine?

Oui, voir <http://www.winehq.org/site/documentation>. (<http://www.winehq.org/site/documentation>)

7.2. Je n'ai pas trouvé la réponse à ma question dans la documentation, mais j'ai écrit une documentation expliquant comment résoudre ce problème. Que dois-je faire?

Des mises à jour et des ajouts concernant la documentation de Wine doivent être envoyés à la liste de diffusion "wine-patches" à <http://www.winehq.org/site/forums> (<http://www.winehq.org/site/forums>). Les ajouts sur la FAQ ou le site Web doivent être déposés dans le répertoire de base "Wine knowledge" approprié.

7.3. Y-a-t-il une newsgroup Usenet pour Wine?

Oui, elle s'appelle `comp.emulators.ms-windows.wine` (`news:comp.emulators.ms-windows.wine`). La newsgroup est un espace ouvert aux utilisateurs et aux développeurs qui veulent discuter de Wine, et pour certaines informations non importantes grand public. Les annonces importantes seront postées en parallèle sur d'autres newsgroup appropriées, telles que les suivantes:

- `comp.os.linux.announce` (`news:comp.os.linux.announce`)
- `comp.windows.x.announce` (`news:comp.windows.x.announce`)
- `comp.emulators.announce` (`news:comp.emulators.announce`)

Si votre site Usenet ne dispose pas de ces newsgroups, demandez expressément à votre admin FAI de les ajouter et/ou de mettre les liens pour s'y connecter.

7.4. Y-a-t-il un site Web pour Wine?

Wine HQ (<http://www.winehq.org>) est le site officiel.

7.5. Y-a-t-il un canal IRC pour Wine?

Bien sûr. C'est le canal `#wineHQ` sur `irc.freenode.net` voir (<http://freenode.net>). Habituellement plusieurs développeurs Wine sont présents juste pour VOUS aider, vous! ;-)

7.6. Je crois que j'ai trouvé un bogue. Comment puis-je prévenir l'équipe de programmeurs de Wine de celui-ci?

Les rapports de bogue doivent être soumis à notre système en ligne Bugzilla (<http://bugs.winehq.org/>). Vous devrez au minimum préciser les choses suivantes:

- La version Wine testée
- Le nom de l'application Windows, y compris la version, et, si possible, une URL où l'application peut être téléchargée
- Une brève description du bogue
- Les parties importantes de la sortie du débogueur de Wine
- Une capture d'écran du problème, si possible

Pour plus d'informations sur la manière de rapporter les bogues voir la section `How to report a bug` (<http://www.winehq.org/Docs/wine-user/bug-reporting.shtml>) du guide de l'utilisateur Wine [the Wine Users Guide].

8. Aider Wine ou devenir un développeur Wine

8.1. Comment devenir développeur Wine? Qu'ai-je besoin de savoir?

Si vous pouvez programmer en C, c'est déjà ça. Téléchargez les sources via (CVS, (<http://www.winehq.org/site/cvs>)) inscrivez-vous aux listes de diffusion, jetez un oeil au code source, et intéressez-vous au groupe de nouvelles `comp.emulators.ms-windows.wine` et aux listes de diffusion (<http://www.winehq.org/site/forums>). Regardez si vous voyez quelque chose que vous pensez pouvoir réparer ou dont vous pouvez vous occuper. Vous n'aurez pas de problèmes pour trouver des domaines où il y a du travail à faire dans Wine (filtrer [grep] les FIXME dans la source).

8.2. Comment puis-je contribuer au projet Wine, et de quelle(s) manière(s)?

Vous pouvez apporter vos (connaissances en programmation et en documentation, ou faire des dons en argent ou matériel, pour aider les développeurs de Wine à atteindre leurs buts.

Pour trouver des idées sur la manière dont vous pouvez aider, consultez la page de contribution de Wine (<http://www.winehq.org/site/contributing>).

8.3. Je veux apporter mon aide sur les tests beta de Wine. Comment faire?

Wine est encore fait de code Alpha en ce moment. Cependant, n'importe qui est le bienvenu pour télécharger la dernière version, et l'essayer à tout moment.

8.4. J'ai écrit du code que je voudrais soumettre au projet Wine. Comment faire?

Soumettre un correctif pour un ajout dans Wine est très simple. En gros tout ce que vous avez à faire est d'envoyer le correctif à la liste diffusion `wine-patches` (<http://www.winehq.org/mailman/listinfo/wine-patches>). Il y a cependant des recommandations sur le format du correctif et par conséquent il est conseillé de lire la page où nous décrivons comment soumettre les correctifs (http://www.winehq.org/site/sending_patches). Cela vous donnera également plus de détails sur l'ensemble du processus et en particulier sur ce que va devenir votre correctif une fois soumis.

9. Développement de programmes utilisant Wine/WineLib

9.1. Puis-je utiliser Wine pour porter mes sources Win32 vers Unix?

C'est justement ça, WineLib. Pour le moment c'est peut-être encore un peu difficile, mais cela évolue tout le temps. Lisez le Guide d'utilisateur de WineLib [WineLib User's Guide] (<http://www.winehq.org/Docs/winelib-user/>) pour information.

9.2. Est ce que MFC marchera avec Wine? Que dois-je faire?

Wine n'implémente pas de remplacements MFC et ne prévoit pas d'en faire. Cependant il est possible (avec beaucoup de travail) de compiler MFC depuis les sources et de créer ainsi une bibliothèque `mfc42.dll.so`.

Reportez-vous au Winelib User's Guide (<http://www.winehq.org/Docs/winelib-user/>) pour savoir comment faire.

9.3. Existe-t-il des applications commerciales qui ont été portées en utilisant Wine?

Voici quelques exemples d'applications portées en utilisant Wine ou Winelib:

- Corel's WordPerfect Office Suite 2000 a été porté sous Linux en utilisant Wine.
- Kylix, la version Linux de Delphi, a été portée sous Linux en utilisant Winelib. L'IDE utilise en fait une combinaison de QT et Winelib alors qu'il n'aurait pas été possible d'y arriver en utilisant uniquement Wine. Cependant, les applications générées ne dépendent absolument pas de Wine.
- MusicMatch Jukebox 5 a également été porté (http://www.itworld.com/nl/lrx_desktop/01042001/) sous Linux en utilisant Winelib. Cependant des versions plus récentes ne le sont pas, et la version 5 n'est plus disponible.
- Ability Office (<http://www.ability.com/linux/abilitylinux.php>)
- IBM's Websphere (<http://www7b.boulder.ibm.com/dl/swws/swwsgddb-p>)

Beaucoup d'autres applications ont déjà été portées. (on parle de plusieurs applications dans le top 500 ici)

9.4. Comment puis-je détecter Wine?

Franchement, vous n'avez pas trop intérêt à faire cela. Si il y a un truc bizarre à corriger dans Wine, il est vraiment préférable de le réparer dans Wine.

10. Points concernant Wine HQ

10.1. Pourquoi la liste de diffusion est-elle réglée pour répondre à l'auteur et pas à la liste?

Il a de bonnes raisons pour procéder de la sorte.

10.2. Comment se désinscrire de la mailing liste ?

Voir: <http://www.winehq.org/site/forums> et sélectionnez [(Un-)Subscribe]