

Guide de l'utilisateur de VLS

Cyril Deguet

Alexis de Lattre

Guide de l'utilisateur de VLS

by Cyril Deguet and Alexis de Lattre

Copyright © 2002-2004 Le projet VideoLAN

Ce document est le guide de l'utilisateur complet pour VLS .

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. The text of the license can be found in the appendix. *GNU General Public License*.

Table of Contents

1. Introduction.....	1
Qu'est ce que le projet VideoLAN ?.....	1
Qu'est-ce qu'un codec ?.....	3
Utilisation de VideoLAN ?.....	3
Utilisation de la ligne de commande.....	4
2. Installation de VLS.....	8
Installation de VLS.....	8
Désinstallation de VLS.....	9
3. Concepts et mode de fonctionnement.....	10
Structure de VLS.....	10
Interface d'administration.....	11
4. Configuration.....	12
Structure générale.....	12
Ecriture d'un <code>vls.cfg</code>	12
5. Utilisation de VLS.....	20
Lancement de VLS.....	20
Utilisation de l'interface telnet.....	20
Commandes de l'interface.....	21
A. GNU General Public License.....	23
Preamble.....	23
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION.....	23
How to Apply These Terms to Your New Programs.....	27

Chapter 1. Introduction

Qu'est ce que le projet VideoLAN ?

Aperçu

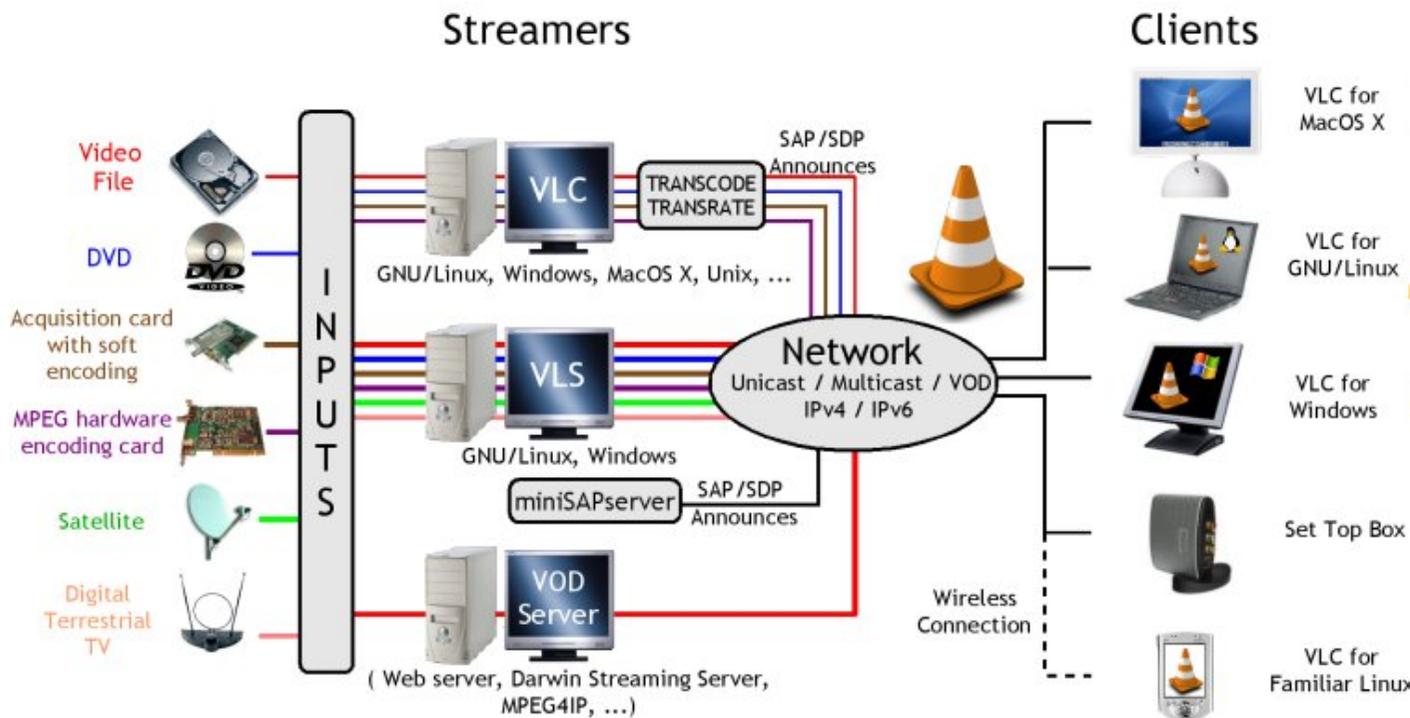
VideoLAN est une solution logicielle complète de diffusion vidéo, développée par des étudiants de l'Ecole Centrale Paris (<http://www.ecp.fr>) et des développeurs du monde entier, sous licence GNU General Public License (<http://www.gnu.org/copyleft/gpl.html>) (GPL). VideoLAN est conçu pour diffuser des vidéos MPEG sur des réseaux haut débit.

La solution VideoLAN comprend :

- VLS (VideoLAN Server), qui est capable de diffuser des fichiers MPEG-1, MPEG-2 et MPEG-4, des DVDs, des canaux satellite numériques, des chaînes de télévision numérique terrestre, et de la vidéo en temps réel sur un réseau en unicast ou multicast,
- VLC (à l'origine, VideoLAN Client), qui peut être utilisé en tant que serveur, pour diffuser des fichiers MPEG-1, MPEG-2 et MPEG-4, des DVDs, ou de la vidéo en temps réel sur un réseau en unicast ou multicast ; ou utilisé en temps que client pour recevoir, décoder et afficher des flux vidéo sous de nombreux systèmes d'exploitation.

Voici le schéma global de la solution VideoLAN complète :

Figure 1-1. La solution VideoLAN globale



De plus amples détails sur le projet peuvent être trouvés sur le Site web de VideoLAN (<http://www.videolan.org/>).

Logiciels VideoLAN

VLC

VLC fonctionne sur de nombreuses plateformes : Linux, Windows, Mac OS X, BeOS, *BSD, Solaris, Familiar Linux, Yopy/Linupy et QNX. Il est capable de lire :

- Des fichiers MPEG-1, MPEG-2 et MPEG-4 / DivX depuis un disque dur, un lecteur de CD-ROM, ...
- Des DVDs et VCDs,
- depuis une carte satellite (DVB-S),
- Des flux MPEG-1, MPEG-2 et MPEG-4 envoyés sur le réseau par un VLS ou un VLC.

VLC peut également être employé en tant que serveur pour diffuser:

- Des fichiers MPEG-1, MPEG-2 et MPEG-4 / DivX ,
- des DVDs,
- depuis une carte d'encodage MPEG,

vers :

- une machine (c'est à dire une adresse IP) : ceci est appelé *unicast*,
- Un groupe dynamique de machines que les clients rejoignent ou quittent (une adresse IP multicast): ceci est appelé *multicast*,

en IPv4 ou en IPv6 .

Pour la liste complète des fonctionnalités de VLC pour chaque plateforme, regardez la page des fonctionnalités (<http://www.videolan.org/vlc/features.html>) (en anglais).

Note: VLC ne fonctionna pas sous Mac OS 9, et ne fonctionnera probablement jamais sur cette plateforme.

VLS

VLS est capable de diffuser:

- des fichiers MPEG-1, MPEG-2 ou MPEG-4 stockés sur un disque dur, un CD, ...,
- un DVD dans un lecteur local ou copié sur un disque dur,
- une carte satellite (DVB-S) ou une carte de télévision numérique terrestre (DVB-T) ,
- une carte d'encodage MPEG ;

vers:

- une machine (c'est à dire une adresse IP) : ceci est appelé *unicast*,

- Un groupe dynamique de machines que les clients rejoignent ou quittent (une adresse IP multicast): ceci est appelé *multicast*,

en IPv4 ou en IPv6.

Une machine base de Pentium 100 MHz munie de 32 Mo de mémoire devrait suffire pour diffuser un flux sur un réseau. Si vous diffusez de nombreuses vidéos, stockées sur un disque dur, la vraie limitation ne vient pas du processeur mais du disque dur et de la connexion réseau.

VLS fonctionne sous Linux et Windows. Pour la liste complète des possibilités de VLS sur chaque plateforme, référez-vous à la page des fonctionnalités de diffusion (<http://www.videolan.org/streaming/features.html>).

Mini-SAP-server

Vous pouvez ajouter un service de découverte de services utilisant le standard SAP/SDP sur la solution VideoLAN. Le mini-SAP-server envoie des annonces indiquant les programmes multicast sur le réseau en IPv4 ou en IPv6 et les VLCs reçoivent ces annonces et ajoutent automatiquement les programmes annoncés à leur liste de lecture.

Le mini-SAP-server fonctionne sous Linux et Mac OS X.

Qu'est-ce qu'un codec ?

Afin de bien comprendre le fonctionnement de la solution VideoLAN, vous devez connaître la différence entre un *codec* et un *format conteneur*

- Un *codec* est un algorithme de compression, utilisé afin de réduire la taille du flux. Il existe des codecs audio et vidéo. MPEG-1, MPEG-2, MPEG-4, Vorbis, DivX, ... sont des codecs
- Un *format conteneur* contient un ou plusieurs flux déjà encodés. Très souvent, il y a un flux audio et un flux vidéo. AVI, Ogg, MOV, ASF, ... sont des formats conteneurs. Les flux contenus peuvent utiliser différents codecs. Si tout était parfait, n'importe quel codec pourrait être mis dans n'importe quel format conteneur, mais il existe des incompatibilités. Vous trouverez une matrice des codecs et formats conteneurs possible sur la page des fonctionnalités (<http://www.videolan.org/streaming/features.html>)

Pour décoder un flux, VLC commence par le *démultiplexer*. Cela signifie qu'il lit le format conteneur, et sépare l'audio, la vidéo, et éventuellement les sous-titres. Ensuite, chacun de ces flux séparés est envoyé aux *décodeurs* qui vont effectuer le processus mathématique de décompression .

MPEG est un cas particulier:

- MPEG est un codec. Il en existe plusieurs versions, appelées MPEG-1, MPEG-2, MPEG-4, ...
- MPEG est également un format de conteneur, parfois appelé *MPEG System*. Il en existe plusieurs types: ES, PS, et TS

Quand vous jouez une vidéo MPEG depuis un DVD, par exemple, le flux MPEG est en fait composé de plusieurs flux (appelés flux élémentaires, ES): un flux vidéo, un flux audio, les sous-titres, ... Tous ces flux sont multiplexés dans un seul flux Program Stream (PS). Ainsi, les fichiers .VOB du DVD sont en fait des fichiers MPEG-PS. Mais ce format PS n'est pas adapté pour la diffusion sur un réseau ou par un satellite, par exemple. Un autre format, appelé Transport Stream, ou TS, fut conçu pour la diffusion de vidéo MPEG.

Utilisation de VideoLAN ?

Documentation

La documentation utilisateur de VideoLAN comporte 4 documents :

- le *VideoLAN HOWTO*. Ce document est le guide complet de la solution de streaming VideoLAN. Ses exemples concrets vous aideront à mettre en place votre solution de diffusion.
- Le *VLC user guide*. Ce document est le guide complet de VLC
- Le *VLS user guide*. Ce document est le guide complet de VLS.
- La *FAQ de VideoLAN*. Ce document contient la Foire Aux Questions de VideoLAN

La dernière version de ces documents se trouve sur la page documentation (<http://www.videolan.org/doc/>).

Support aux utilisateurs

Si vous éprouvez des difficultés pour utiliser VideoLAN et que vous ne trouvez pas la réponse à vos problèmes dans la documentation, merci de regarder dans l'archive en-ligne des mailing-lists (<http://www.via.ecp.fr/via/ml/videolan-en.html>). Il existe deux listes anglophones pour les utilisateurs :

- vlc@videolan.org pour les questions relatives à VLC ,
- streaming@videolan.org pour les questions relatives à VLS, au mini-SAP-server, et au serveur.

Pour vous inscrire ou vous désinscrire de ces mailing-lists, merci de voir la page mailing-lists (<http://www.videolan.org/support/lists.html>).

Vous pouvez également parler aux utilisateurs et développeurs VideoLAN sur IRC, sur le serveur irc.freenode.net, et le channel `#videolan`.

Si vous découvrez un bug, merci de bien vouloir suivre les instructions de la page report de bug (<http://www.videolan.org/support/bug-reporting.html>).

Utilisation de la ligne de commande

- VLC possède un certain nombre d'interfaces graphiques différentes, qui sont organisées de façon assez diverse, de façon à s'harmoniser avec le style de chaque système d'exploitation. Documenter l'utilisation de chaque interface graphique serait très long et certaines fonctionnalités ne sont accessibles que depuis l'interface en ligne de commande. C'est pourquoi nous avons décidé de ne documenter que l'interface en ligne de commande, mais dans la plupart des cas, il devrait être assez facile de deviner comment utiliser l'interface graphique pour le même usage !
- VLS est dotée d'une interface en ligne de commande, et d'une interface Telnet, mais n'a pas d'interface graphique !

Toutes les commandes que vous voyez dans ce document doivent être tapées dans un terminal.

Ouvrez un terminal

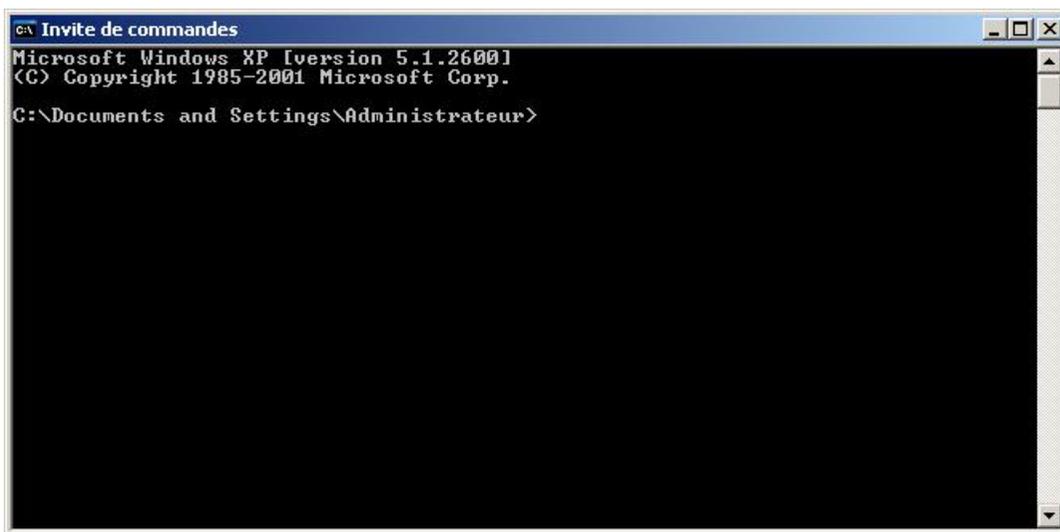
Windows

Cliquez sur *D* <65533> *marrer*, *Ex* <65533> *cuter* and tapez :

- **cmd Enter** (Windows 2000 / XP),
- **command Enter** (Windows 95 / 98 / ME).

Le terminal apparait

Figure 1-2. Windows terminal



Note: Sous Windows, vous devez vous situer dans le dossier o<65533> le programme est install<65533> pour le lancer

Linux / Unix

Ouvrez un terminal :

Figure 1-3. Linux X terminal



Dans la documentation, nous adoptons les conventions suivantes pour les commandes Unix :

- Les commandes devant être lancées en tant que *root* commencent par un # :

#

`commande_a_lancer_en_root`

- Les commandes devant être lancées en tant qu'utilisateur normal commencent par un % :

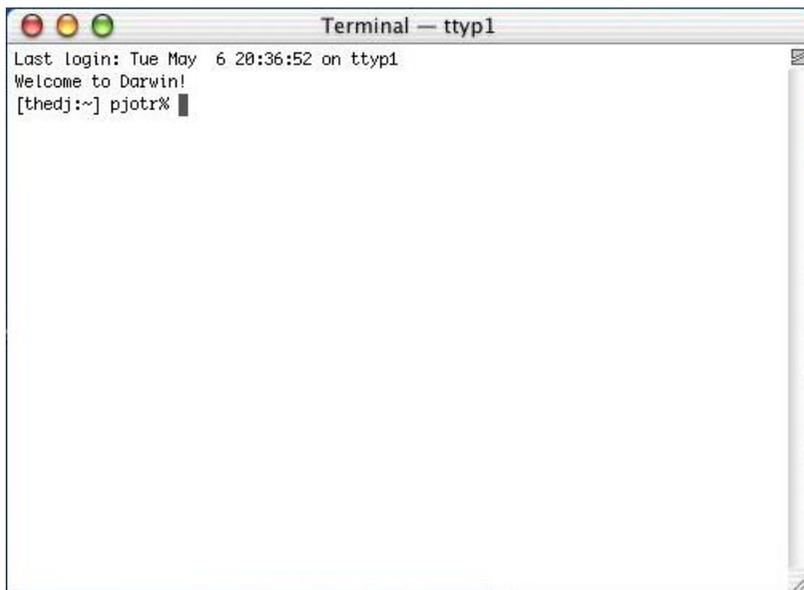
%

`commande_a_lancer_en_tant_qu_utilisateur`

Mac OS X

Allez dans *Applications*, ouvrez le dossier *Utilitaires* et double-cliquez sur *Terminal* :

Figure 1-4. Mac OS X terminal

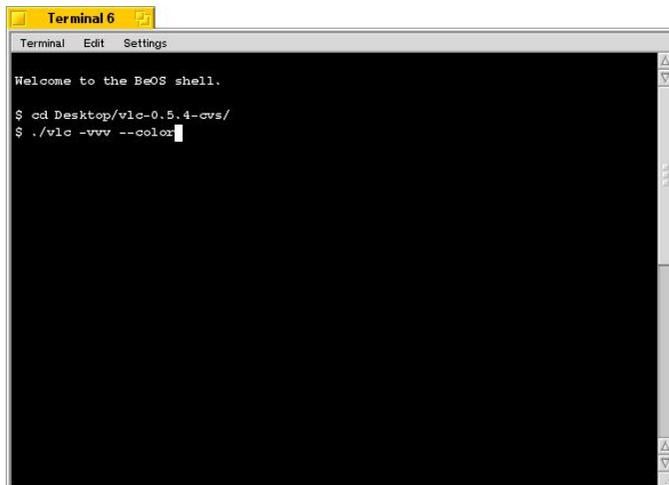


Note: Sous Mac OS X, vous devez vous situer dans le dossier où le programme est installé pour le lancer, et le lancer avec `./`.

BeOS

Dans la barre de bureau, allez dans *Application* et sur *Terminal* :

Figure 1-5. BeOS terminal



Note: Sous BeOS, vous devez vous situer dans le dossier où le programme est installé pour le lancer, et le lancer avec `./`.

Chapter 2. Installation de VLS

Installation de VLS

Windows

Téléchargez le fichier Zip depuis la page de téléchargement de VLS pour Windows (<http://www.videolan.org/streaming/download-vls-windows.html>), décompressez-le, et lancez `setup.exe`.

GNULinux & Mac OS X

Installez les bibliothèques

Plusieurs bibliothèques sont requises :

- *libdvbpsi* (obligatoire)
- *libdvdcss* pour lire des DVDs cryptés ,
- *libdvread* pour diffuser des DVDs ,
- *libdvb* pour pouvoir diffuser depuis une carte DVB (satellite ou télévision numérique terrestre) .

Téléchargez les bibliothèques depuis la page des sources de VLS (<http://www.videolan.org/streaming/download-vls-sources.html>) .

Pour chaque bibliothèque, décompressez, configurez (sauf pour *libdvb* qui n'a pas de *./configure*), compilez et installez :

```
% tar xvzf library.tar.gz
% cd library
% ./configure
% make
# make install
```

Vérifiez que le fichier de configuration `/etc/ld.so.conf` contient la ligne :

```
/usr/local/lib
```

Dans le cas contraire, ajoutez-la, et lancez `starten Sie dann` :

```
# ldconfig
```

Installez VLS

Téléchargez les sources de la dernière version : récupérez le fichier `vls-version.tar.gz` de la page des sources de VLS (<http://www.videolan.org/streaming/download-vls-sources.html>). Décompressez-la et générez *./configure* :

```
% tar xvzf vls-version.tar.gz
% cd vls-version
```

Pour avoir la liste des options, faites :

```
% ./configure --help
```

Configurez ensuite VLS :

- pour un VLS simple, sans support de DVD, faites :

```
% ./configure --disable-dvd
```
- pour un VLS avec support du DVD, faites :

```
% ./configure
```
- pour un VLS avec support du DVB, faites :

```
% ./configure --enable-dvb --with-dvb=PATH_TO_DVB_DRIVERS --with-libdvd=PATH_TO_LIBDVB
```

Puis, compilez et installez :

```
% make
# make install
```

Vous pouvez également faire **make uninstall**, **make clean** ou **make distclean** .

Désinstallation de VLS

Windows

Allez dans le *Panneau de contrôle*, cliquez sur *Ajoutez et supprimez des programmes*, choisissez *VLS* et cliquez sur *Modifier/Enlever* et suivez les étapes pour désinstaller le programme .

Si vous avez compilé VLS depuis les sources

Allez dans le répertoire contenant les sources et faites :

```
# make uninstall
```

Vous pouvez ensuite enlever les sources de VLS .

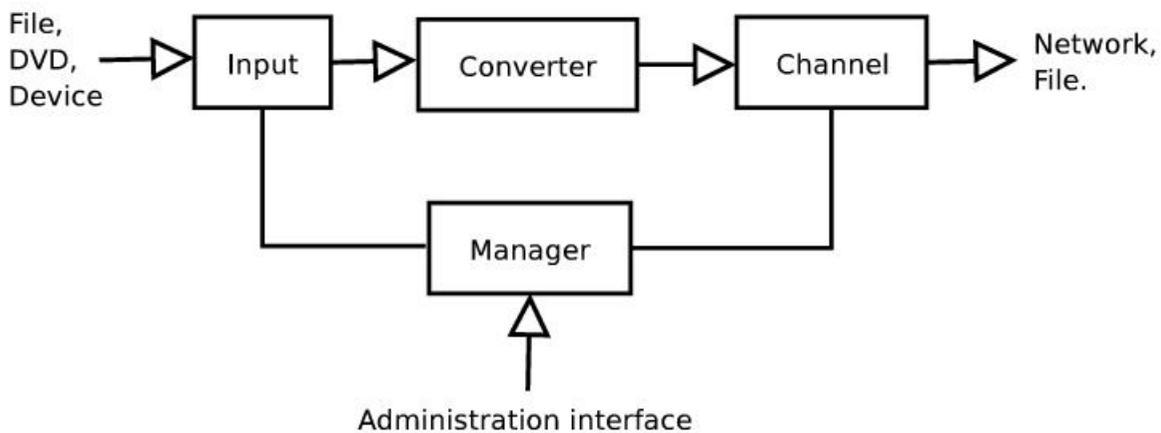
Chapter 3. Concepts et mode de fonctionnement

Structure de VLS

D'un point de vue utilisateur, VLS peut être divisé en quatre composants majeurs:

- un *gestionnaire* ,
- *des entrées* ,
- *des convertisseurs* ,
- et *des sorties* .

Figure 3-1. Structure de VLS



Entrées

Le rôle d'une *entrée* est de lire des flux MPEG depuis une source donnée (fichier, DVD, carte DVB, ...), et de les envoyer aux *convertisseurs*. Une entrée peut lire plusieurs flux. Ceux-ci sont alors appelés des *programmes*. Il existe plusieurs types d'entrées :

- l'entrée *local*, qui peut lire depuis des fichiers et des DVDs ,
- l'entrée *video*, qui peut lire des vidéos depuis des cartes d'encodage MPEG ,
- , l'entrée *dvb*, qui peut lire depuis des cartes DVB ,
- l'entrée *v4l*, qui peut lire depuis les cartes d'acquisition supportées par les pilotes Video4Linux .

Vous pouvez utiliser plusieurs entrées et jouer plusieurs programmes en même temps.

Les convertisseurs

Le rôle d'un *convertisseur* est de recevoir un flux depuis une entrée et de le convertir au format MPEG-TS. VLS peut convertir des flux PS (provenant d'un DVD, par exemple) en flux TS, en utilisant le convertisseur ps2ts. Bien sûr, il peut également lire les flux TS, et les réparer en prenant en charge les discontinuités du flux (convertisseur ts2ts).

Les sorties

Une sortie reçoit le flux depuis le convertisseur, et l'envoie vers une destination donnée (réseau, fichier). Actuellement, il existe deux types de sorties: *network* et *file*. Pour l'instant, VLS ne peut utiliser qu'une sortie par flux, donc vous ne pouvez pas diffuser un flux sur le réseau et l'enregistrer dans un fichier en même temps. La sortie réseau est très configurable: vous pouvez choisir l'interface réseau, et spécifier les adresses IP source et destination .

Le gestionnaire

Le *gestionnaire* contrôle l'envoi des flux. En utilisant une *interface d'administration*, vous pouvez lancer, arrêter, interrompre, ou reprendre les programmes. Vous pouvez également afficher la liste des programmes disponibles, lue depuis le fichier de configuration de VLS (`vls.cfg`) (c'est pourquoi elle ne peut pas être changée après le démarrage). Pour l'instant, vous ne pouvez pas demander au gestionnaire si un flux est en cours de diffusion, mais une erreur se produira si vous tentez de stopper un flux qui ne l'est pas .

Interface d'administration

Il existe deux moyens de contrôler VLS :

- vous pouvez utiliser la *ligne de commande* pour passer des arguments au démarrage;
- ou vous pouvez utiliser l'*interface Telnet* pour contrôler la diffusion à tout moment .

Afin d'utiliser l'interface Telnet, vous devez tout d'abord vous authentifier, car tous les utilisateurs ne peuvent exécuter toutes les commandes (ceci se règle dans le fichier de configuration `vls.cfg`) .

Chapter 4. Configuration

VLS lit sa configuration depuis le fichier `vls.cfg`, qui doit se trouver dans le répertoire courant ou dans `SYSCONF_DIR/videolan/vls` où `SYSCONF_DIR` est `/usr/local/etc` si vous avez installé VLS à la main, ou `/etc` si vous avez utilisé le package Debian .

Pour écrire un fichier `vls.cfg`, utilisez celui fourni avec VLS comme point de départ .

Structure générale

Le fichier de configuration `vls.cfg` est divisé en section, chacune contenant plusieurs variables :

```
BEGIN "Section1"
    Variable1 = "valeur1"
    Variable2 = "valeur2"
    [...]
END

BEGIN "Section2"
    Variable1 = "valeur1"
    Variable3 = "valeur3"
    [...]
END
```

[...]

Tous les noms de section, noms de variable, et valeurs ne sont pas sensibles à la case. Il peut y avoir des sections et sous-sections vides. Les commentaires doivent être placés après un `#`. Certaines variables ont une valeur par défaut; ce qui signifie que vous pouvez ne pas les déclarer, leur valeur sera alors la valeur par défaut .

Ecriture d'un `vls.cfg`

Nous allons maintenant détailler toutes les sections du fichier `vls.cfg` :

Section "VLS"

Cette section contient des paramètres globaux .

```
LogFile = "name"
```

Nom du fichier de log de VLS. Si une valeur vide ("") est entrée, aucun log n'est écrit. Le défaut est "vls.log" .

```
SystemLog = "[disable|enable]"
```

Log dans le mécanisme de log du système. Actuellement, seul syslogd est supporté: compilez avec `./configure --enable-syslog` .

Caution

Si VLS est lancé en tant que **vlsd**, la configuration suivante est requise :

```
BEGIN "Vls"
  LogFile   = ""
  SystemLog = "enabled"
  ScreenLog = "disabled"
END
```

ScreenLog = "[disable|enable]"

Affichage des logs sur la console .

Exemple :

```
BEGIN "Vls"
  LogFile   = "vls.log"
  SystemLog = "disable"
  ScreenLog = "enable"
END
```

Section "Groups"

Dans cette section, vous pouvez définir des groupes d'utilisateurs et quelles commandes ils sont autorisés à exécuter. Pour chaque groupe, vous devez ajouter une ligne du type :

```
groupname = "command1|command2|..."
```

Ceci ajoute le groupe "groupname", dont les utilisateurs peuvent exécuter command1,command2, ... Les commandes disponibles sont: browse, start, suspend, resume, forward, rewind, stop, shutdown, logout .

Exemple :

```
BEGIN "Groups"
  monitor = "help|browse|logout"
  master  = "help|browse|start|resume|suspend|forward|rewind|stop|shutdown|logout"
END
```

Section "Users"

Cette section contient la liste des utilisateurs autorisés à contrôler VLS via l'interface d'administration. Pour chaque utilisateur, vous devez ajouter une ligne du type :

```
username = "password:groupname"
```

Ceci ajoute un utilisateur "username", qui appartient au groupe "groupname", défini dans la section "Groups", et qui peut se logger avec le mot de passe "password" .

- Sous Unix/Linux, le mot de passe doit être encrypté, avec un outil tel que **mkpasswd**, ou à l'aide de la fonction unix "crypt" .
- Sous Windows, le mot de passe doit être en clair .

Exemple pour Unix/Linux :

```
BEGIN "Users"
  monitor = "3BcKWoiQn0vi6:monitor"      # password is 'monitor'
  admin   = "42BKicGuFAL/c:master"      # password is 'Vir4Gv5S'
END
```

Section "Telnet"

Dans cette section, vous pouvez configurer l'interface d'administration telnet .

```
LocalPort = "port"
```

Définit le port qui sera utilisé pour le serveur telnet. Le port par défaut est "9999" .

```
Domain = "domain"
```

Soit "inet4", soit "inet6" (défaut = "inet4"). Si vous voulez utiliser IPv4, mettez "inet4", et pour l'IPv6, mettez "inet6" .

```
LocalAddress = "IP address"
```

Définit sur quelle adresse IP le serveur Telnet va écouter les requêtes. L'adresse par défaut est "0.0.0.0" (ou "0::0" avec l'IPv6) .

Exemple :

```
BEGIN "Telnet"
  LocalPort = "9999"
END
```

Section "NativeAdmin"

Même syntaxe que "Telnet". Inutilisé .

Section "Inputs"

Dans cette section, vous pouvez définir quelles entrées vous désirez utiliser. Pour chaque entrée, ajoutez une ligne du type :

```
InputName = "Type"
```

Ceci ajoute une entrée nommée "InputName", de type "Type". Comme expliqué, il existe plusieurs types d'entrées :

- "local" pour jouer des fichiers ou des DVDs ,
- "video" pour lire depuis une carte d'encodage MPEG ,
- "dwb" pour lire depuis une carte DVB ,
- "v4l" pour lire depuis un périphérique Video4Linux .

Chaque entrée doit être configurée dans sa propre section (voir la suite) .

Exemple :

```
BEGIN "Inputs"
  local1 = "local"
  pvr    = "video"
  dvb1   = "dwb"
```

```
tuner = "v4l"
END
```

Configuration des entrées

Pour chaque entrée déclarée dans la section "Inputs", à l'exception des entrées "local", vous devez ajouter une section avec le même nom que l'entrée correspondante. Par exemple, si vous déclarez une entrée "pvr", il doit également y avoir une section "pvr". La syntaxe de ces sections dépend du type de l'entrée correspondante.

Pour configurer une entrée locale, vous n'avez rien à faire, sauf si vous utilisez une stratégie "trickplay" différente :

```
BEGIN "Local1"
  ProgramCount = "1"
  TrickPlay    = "normal"
END
```

"Local1" est le nom de l'entrée que vous configurez. "ProgramCount" est le nombre de programmes correspondant à cette entrée. "TrickPlay" est la stratégie trickplay utilisée par l'entrée (le défaut est "normal").

Pour configurer une entrée vidéo, ajoutez une section dans ce format :

```
BEGIN "VideoInputName"
  Device = "device"
  Type   = "type"
END
```

"VideoInputName" est le nom de l'entrée vidéo que vous configurez. "Device" est l'adresse de la carte d'encodage MPEG que vous désirez utiliser (le défaut est "/dev/video"). "Type" est soit "MPEG2-PS", soit "MPEG2-TS", selon la configuration de votre périphérique (le défaut est "MPEG2-PS").

Exemple pour une carte Hauppauge WinTV-PVR-250 :

```
BEGIN "pvr"
  Device = "/dev/video0"
  Type   = "Mpeg2-PS"
END
```

Pour configurer une entrée dvb, ajoutez une section du type suivant :

```
BEGIN "DvbInputName"
  DeviceNumber = "devicenumber"
  SendMethod   = "0"
END
```

"DvbInputName" est le nom de l'entrée dvb que vous configurez. Réglez "SendMethod" à "0" si vous désirez diffuser le flux DVB complet et à "1" si vous désirez diffuser les flux MPEG audio et vidéo (le défaut est "0"). "DeviceNumber" est le numéro du périphérique DVB depuis lequel vous désirez lire (lecture depuis /dev/ost/dvr<devicenumber>, le défaut étant ""). Le fichier de configuration de dvb est défini par le périphérique. Vous pouvez le trouver dans \$HOME/.dvbrc pour /dev/dvb/adapter0 ou dans \$HOME/.dvbrc.X pour /dev/dvb/adapterX.

Exemple :

```
BEGIN "dvb1"
  DeviceNumber = "0"
  TrickPlay    = "normal"
END
```

Section "Channels"

Dans cette section, vous pouvez définir les sorties que vous désirez utiliser. Pour chacune, ajoutez une ligne du type :

```
ChannelName = "Type"
```

Ceci ajoute une sortie appelée "ChannelName", de type "Type". Ceci doit être "network" ou "file". Comme les entrées, les sorties doivent être configurées dans leur propre section .

Exemple :

```
BEGIN "Channels"
  localhost = "network"
  client1   = "network"
  client2   = "network"
  multicast1 = "network"
  multicast2 = "network"
  localfile = "file"
END
```

Configuration des sorties

Pour chaque sortie déclarée dans la section "Channels", vous devez ajouter une section avec le nom correspondant. La syntaxe de cette section dépend du type de la sortie correspondante .

Pour configurer une sortie network, ajoutez une section dans le format suivant :

```
BEGIN "NetChannelName"
  Domain    = "Domain"
  Type      = "Type"
  SrcHost   = "SourceHost"
  SrcPort   = "SourcePort"
  DstHost   = "DestHost"
  DstPort   = "DestPort"
  TTL       = "ttl"
  Interface = "Interface"
END
```

- "NetChannelName" est le nom de la sortie que vous configurez .
- "Domain" est soit "inet4", pour des adresses IPv4, ou "inet6", pour des adresses IPv6 (le défaut est "inet4") .
- "Type" est soit, "unicast", "broadcast" ou "multicast" (le défaut est "unicast"), selon ce que vous désirez faire, et votre adresse "DstHost") .
- "SourceHost" est l'adresse IP (ou le nom DNS) depuis lequel VLS va envoyer le flux .
- "SourcePort" est le port UDP depuis lequel le flux sera envoyé .
- "DestHost" est l'adresse IP (ou le nom DNS) vers lequel le flux sera envoyé .
- "DestPort" est le port UDP vers lequel le flux sera envoyé (le défaut est "1234") .
- "TTL" est une option utile seulement si "Type" est "multicast" (le défaut est "0"). Vous pouvez l'utiliser pour modifier le nombre de routeur que le paquet pourra téléviser .
- "Interface" est une option qui ne fonctionne que sous GNU/Linux, pour forcer l'interface réseau de sortie ("eth1", par exemple). Vous devez être root pour utiliser cette option .

Note: "SrcHost" et "SrcPort" sont optionnels (si vous ne les réglez pas, VLS n'attachera pas la socket) .

Pour configurer une sortie file, ajoutez une section dans le format suivant :

```
BEGIN "FileChannelName"
  FileName = "file"
  Append   = "append"
END
```

"FileChannelName" est le nom de la sortie file que vous configurez. "file" est le nom du fichier où le flux sera stocké (le défaut est "fileout.ts"). "append" est soit "yes", soit "no", et indique si VLS ajoute le flux à la fin du fichier, ou le réécrire .

Exemple :

```
BEGIN "localhost"           # The client is on the same host as the server
  DstHost = "localhost"
  DstPort = "1234"
END
```

```
BEGIN "client1"            # unicast towards client1
  DstHost = "192.168.1.2"
  DstPort = "1234"
END
```

```
BEGIN "client2"           # unicast towards client2 in IPv6
  Domain  = "inet6"
  DstHost = "3ffe:ffff::2:12:42"
  DstPort = "1234"
END
```

```
BEGIN "multicast1"        # multicast streaming
  Type    = "multicast"
  DstHost = "239.2.12.42"
  DstPort = "1234"
  TTL     = "2"
END
```

```
BEGIN "multicast2"        # multicast streaming in IPv6
  Domain  = "inet6"
  Type    = "multicast"
  DstHost = "ff08::1"
  DstPort = "1234"
  TTL     = "12"
END
```

```
BEGIN "localfile"         # file output
  FileName = "stream.ts"
  Append   = "no"
END
```

Caution

Si vous utilisez Windows, vous devez spécifier les valeurs "SrcHost" et "SrcPort". Par exemple :

```
BEGIN "client1"          # The client is on the same host as the server
  SrcHost = "192.168.1.1" # IP of VLS
  SrcPort = "1242"        # Source port : the value is not important
  DstHost = "192.168.1.2" # IP of the client
  DstPort = "1234"
END
```

Configuration des programmes

Comme expliqué précédemment, vous devez définir les programmes. Chacun est un flux MPEG (un fichier, par exemple). Pour cela, vous devez ajouter une section "Input" dans votre fichier `vls.cfg`. Chaque section Input doit avoir la syntaxe suivante :

```
BEGIN "Input"
  FilePath    = "path"
  ProgramCount = "count"
END
```

"path" est le chemin où vos fichiers MPEG sont situés (par défaut, c'est le répertoire courant). "count" est le nombre de programmes ("0" par défaut) .

Pour chaque programme, vous devez ajouter une section avec la syntaxe suivante :

```
BEGIN "number"
  Name      = "name"
  Type     = "type"
  FileName = "file"
  Device   = "device"
END
```

- "number" est le numéro de programme: le premier est 1, le second 2, ...
- "name" est le nom du programme, que vous utiliserez pour lancer le programme (voir la section suivante) .
- "type" peut être "Mpeg1-PS", "Mpeg2-PS", "Mpeg2-TS", ou "DVD". Si votre flux est stocké dans un fichier MPEG (*.mpeg, *.mpg, *.vob, and so on...), il es probablement au format Mpeg1-PS ou Mpeg2-PS .
- si type n'est pas "dvd", VLS supposera que votre fichier est stocké dans le fichier "file", dans le répertoire "path" (où "path" est le répertoire défini dans la section "Input") .
- dans le cas contraire, la variable "Device" sera utilisée à la place de "FileName". Elle représente l'adresse de votre lecteur de DVD ("/dev/hdc" ou "/dev/cdrom", par exemple). Vous pouvez également jouer un DVD copié sur un disque dur: "device" est alors l'endroit où les fichiers .vob sont stockés ("/mnt/data/VIDEO_TS", par exemple) .

Note: VLS peut diffuser des fichiers MPEG suivant deux critères

- le fichier doit être MPEG *PS* (Program Stream) ou MPEG *TS* (Transport Stream), qui contiennent la vidéo et l'audio multiplexés. VLS ne peut pas treamer des fichiers MPEG *ES* (Elementary Stream), càd, un fichier avec seulement l'audio ou la vidéo .

Pour savoir si votre fichier MPEG est du MPEG *PS*, MPEG *TS* or MPEG *ES*, lisez le fichier avec VLC et regardez les messages (regardez dans la fenêtre de messages, ou utilisez la ligne de commande `vlc -vvv`) .

- Si vous voyez :

```
[00000107] main module debug: using demux module "ts_dvbpsi"
alors, le fichier est du MPEG TS .
```

- Si vous voyez :

```
[00000109] main module debug: using demux module "ps"
alors, le fichier est du MPEG PS .
```

- Si vous voyez :

```
[00000109] main module debug: using demux module "es"
alors, le fichier est du MPEG ES, VLS ne peut pas le diffuser .
```

- l'en-tête de séquence de la vidéo doit se répéter périodiquement. Ceci est très fréquent avec le MPEG-2, mais rare avec le MPEG-1. Il n'est pas facile de savoir si l'en-tête de séquence se répète. Les fichiers avec une extension `.vob` sont normalement des fichiers MPEG-2, et les fichiers avec une extension `.mpg` ou `.mpeg` sont généralement MPEG-1 .

Vous pouvez télécharger ce fichier MPEG-PS diffusable pour vos tests : `presentation_short.vob` (ftp://ftp.videolan.org/pub/videolan/streams/presentation/presentation_short.vob) .

Note: Pour pouvoir lire des DVDs, vous devez compiler VLS avec le support des DVD, qui utilise `libdvdread` et `libdvdcss`. Vous devez avoir l'accès en lecture *et* en écriture à votre lecteur de DVD .

Exemple complet :

```
BEGIN "Input"
  FilePath = "/home/videolan/streams"
  ProgramCount = "4"
END

BEGIN "1"      # MPEG2 stream stored in /home/videolan/streams/Dolby.vob
  Name       = "dolby"
  FileName   = "Dolby.vob"
  Type       = "Mpeg2-PS"
END

BEGIN "2"      # another file
  Name       = "canyon"
  FileName   = "Dolby_Canyon.vob"
  Type       = "Mpeg2-PS"
END

BEGIN "3"      # DVD
  Name       = "dvd"
  Device     = "/dev/cdrom"
  Type       = "Dvd"
END

BEGIN "4"      # DVD stored on a hard disk
  Name       = "matrix"
  Device     = "/mnt/data/matrix/VIDEO_TS"
  Type       = "Dvd"
END
```

Chapter 5. Utilisation de VLS

Lancement de VLS

Si vous désirez utiliser l'interface telnet, lancer VLS est très simple: tapez juste **vls** dans une console et c'est tout. Lancer **vlsd** lancera VLS en tant que démon et le détachera de la console. Souvenez-vous que VLS va essayer de charger son fichier de configuration (`vls.cfg`) depuis le répertoire courant, et si il ne l'y trouve pas, depuis `SYSCONF_DIR/etc/videolan` (voir la section *Configuration*).

Caution

Si votre fichier de log est `vls.log`, comme dans l'exemple, VLS aura besoin de l'accès en écriture au répertoire courant, ou vous verrez quelque chose comme :

```
*** Exception *** in copy constructor (0xbffffc98, copy of 0x80e30a8)
Unable to open the log file "vls.log": Error: Could not open file 'vls.log':
Permission denied
```

Souvenez-vous que vous devez être root quand vous utilisez l'option "Interface" du fichier `vls.cfg`.

Si tout se passe bien, vous verrez quelque chose comme :

```
VideoLAN Server v 0.5.3 (Jun  6 2003) - (c)1999-2003 VideoLAN
2002-03-09 17:24:51 [INFO/Vls]  Module "channel:file" registered
2002-03-09 17:24:51 [INFO/Vls]  Module "channel:network" registered
2002-03-09 17:24:51 [INFO/Vls]  Module "mpegreader:file" registered
2002-03-09 17:24:51 [INFO/Vls]  Module "mpegconverter:ts2ts" registered
[...]
```

Ce que vous verrez à l'écran (stderr) est exactement ce que vous verrez dans le fichier de log `vls.log`.

Une fois que VLS a démarré, il ne prend pas d'argument sur son entrée standard, et vous pouvez le mettre en tâche de fond (vous pouvez utiliser l'utilitaire *screen* pour cela).

Si vous désirez utiliser l'interface en ligne de commande, merci de vous référer au VideoLAN HOWTO (<http://www.videolan.org/doc/>).

Utilisation de l'interface telnet

Après le lancement de VLS, un serveur telnet est ouvert, sur le port 9999 par défaut. Vous pouvez vous y connecter à l'aide de la commande :

```
% telnet localhost 9999
```

Vous devriez voir quelque chose comme :

```
Trying 127.0.0.1...
Connected to vls.
Escape character is '^]'.

Videolan Server Administration System

Login:
```

Vous devez ensuite vous authentifier à l'aide du couple login/password défini dans le fichier de configuration. Une fois l'authentification réalisée, vous devriez voir une invite du type :

```
admin@vls>
>
```

Vous pouvez ensuite taper les commandes définies dans le paragraphe suivant. Pour vous déconnecter, tapez **logout** à l'invite .

Commandes de l'interface

help

Utilisation: help [commande] .

Appellée sans argument, la commande donne la liste de toutes les commandes (disponibles ou non). Avec un argument, elle donne le détail de l'utilisation de la commande spécifiée .

browse

Utilisation: browse [entrée] .

Sans argument, la commande liste les programmes des entrées. Avec un argument, elle donne les programmes de l'entrée spécifiée. Le status de chaque programme est donnée .

start

Utilisation: start <programme> <sortie> <entrée> [--loop] [--rtp]

lance le programme spécifié de l'entrée spécifiée sur la sortie spécifiée. L'option --loop fait que le programme se répète. L'option --rtp fait que les paquets TS sont envoyés en utilisant le protocole RTP, défini dans les RFC 1889 et 2250 .

stop

Utilisation: stop <sortie>

"stop" arrête la diffusion sur la sortie spécifiée .

forward

Utilisation: forward <sortie> <vitesse>

Avance dans la sortie donnée à la vitesse donnée. Ceci ne fonctionne pas lors de la lecture depuis une carte d'acquisition, d'encodage, ou une carte DVB .

rewind

Utilisation: rewind <sortie> <vitesse>

Recule dans la sortie donnée à la vitesse donnée. Ceci ne fonctionne pas lors de la lecture depuis une carte d'acquisition, d'encodage, ou une carte DVB .

suspend

Utilisation: suspend <sortie>

interrompt la diffusion sur la sortie spécifiée .

resume

Utilisation: resume <sortie>

Reprend la diffusion de la sortie spécifiée .

logout

Utilisation: logout

Ferme la session et la connexion .

shutdown

Utilisation: shutdown

Stoppe les diffusions et arrête VLS .

Appendix A. GNU General Public License

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software - to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps:

1. copyright the software, and
2. offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

Section 0

This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a

work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

Section 1

You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Section 2

You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

1. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
2. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
3. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.

Exception:: If the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

Section 3

You may copy and distribute the Program (or a work based on it, under Section 2 in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

1. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
2. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
3. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

Section 4

You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

Section 5

You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

Section 6

Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

Section 7

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

Section 8

If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

Section 9

The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

Section 10

If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY Section 11

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

Section 12

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989 Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.