

VLS Anwenderhandbuch

Cyril Deguet

Alexis de Lattre

VLS Anwenderhandbuch

by Cyril Deguet and Alexis de Lattre

Copyright © 2002-2004 Das VideoLAN Projekt

Dieses Dokument ist das komplette VLS Anwenderhandbuch .

Ihnen ist die Erlaubnis erteilt, dieses Dokument unter den Bedingungen der GNU General Public License, wie sie von der Free Software Foundation veröffentlicht ist, zu kopieren, weiterzugeben und/oder zu verändern. Es gilt Version 2 der Lizenz oder (nach Ihrer Wahl) jede neuere Version. Der Text der Lizenz befindet sich im Appendix.
GNU General Public License.

Table of Contents

1. Einführung	1
Was ist das VideoLAN Projekt?	1
Was ist ein Codec?	3
Wie benutze ich VideoLAN?	3
Befehlszeilenbenutzung	4
2. Installation von VLS.....	8
Installation von VLS	8
Deinstallation von VLS.....	9
3. Überblick und grundlegende Konzepte	10
Struktur von VLS	10
Das Administrationsinterface.....	11
4. Konfiguration	12
Allgemeine Struktur	12
Schreiben einer vls.cfg-Datei.....	12
5. Benutzung von VLS.....	20
VLS starten	20
Benutzung des Telnet-Interfaces	20
Interfacebefehle.....	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. Einführung

Was ist das VideoLAN Projekt?

Überblick

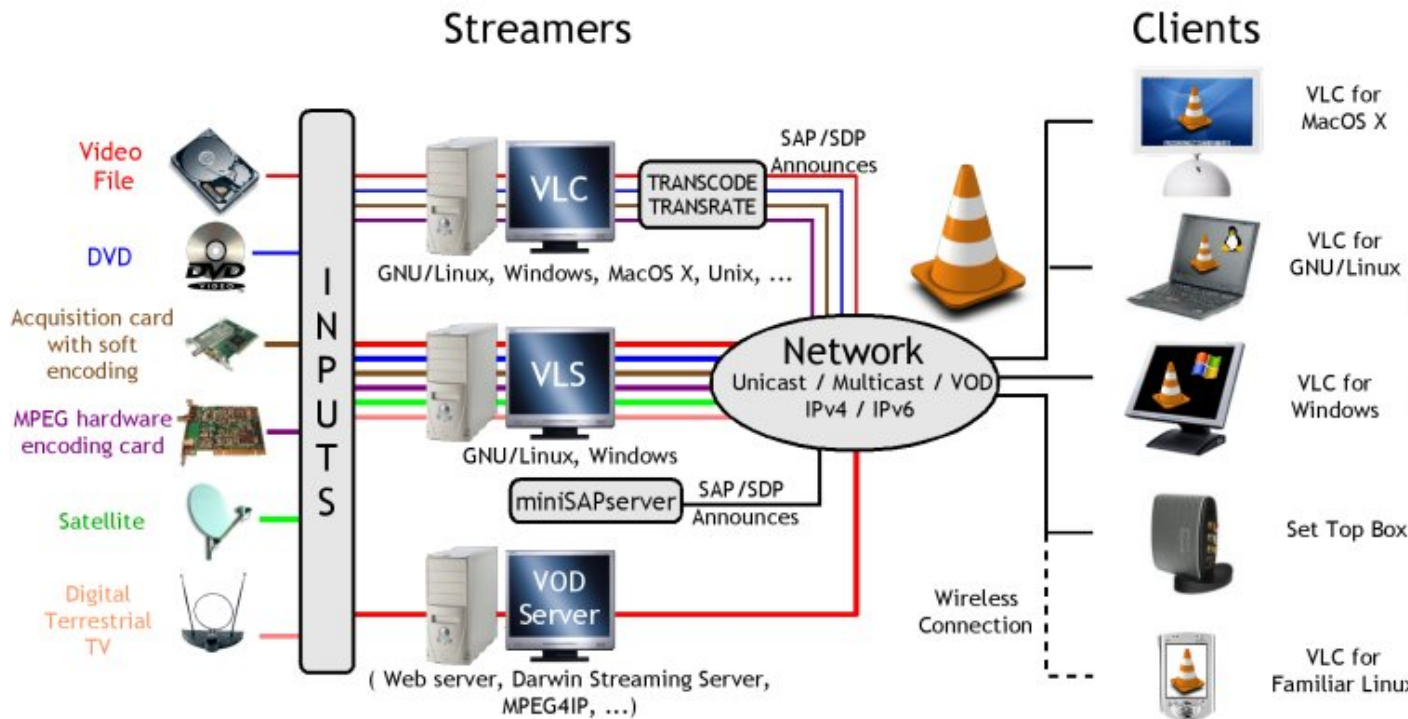
VideoLAN ist eine komplette Softwarelösung für Video-Streaming, entwickelt von Studenten der Ecole Centrale Paris (<http://www.ecp.fr>) und Entwicklern von der ganzen Welt, unter der GNU General Public License (<http://www.gnu.org/copyleft/gpl.html>) (GPL). VideoLAN ist ausgelegt auf das Streamen von MPEG-Videos in Netzwerken mit hoher Bandbreite.

Die VideoLAN-Lösung umfasst:

- VLS (VideoLAN Server), der MPEG-1, MPEG-2 und MPEG-4 Dateien, DVDs, digitale Satellitenkanäle, digitale terrestrische Fernsehkanäle und Live-Videos über das Netzwerk in Uni- oder Multicast streamen kann,
- VLC (anfänglich VideoLAN Client), welches als Server zum Streamen von MPEG-1, MPEG-2 und MPEG-4 Dateien, DVDs und Live-Videos über das Netzwerk per Uni- oder Multicast und als Client zum Empfangen, Dekodieren und Anzeigen von MPEG-Streams unter multiplen Betriebssystemen benutzt werden kann.

Hier ist eine Illustration der kompletten VideoLAN-Lösung:

Figure 1-1. Die globale VideoLAN-Lösung



Mehr Details über das Projekt können auf der VideoLAN Website (<http://www.videolan.org/>) gefunden werden.

VideoLAN Software

VLC

VLC läuft auf vielen Plattformen: Linux, Windows, Mac OS X, BeOS, *BSD, Solaris, Familiar Linux, Yopy/Linupy und QNX. Es kann lesen:

- MPEG-1, MPEG-2 und MPEG-4 / DivX Dateien von einer Festplatte, einem CD-ROM-Laufwerk,...
- DVDs und VCDs,
- von einer Satellitenkarte (DVB-S),
- MPEG-1, MPEG-2 und MPEG-4 Streams aus dem Netzwerk, gesendet von VLS oder VLCs Streamausgabe.

VLC kann auch als Streaming-Server benutzt werden:

- MPEG-1, MPEG-2 und MPEG-4 / DivX-Dateien,
- DVDs,
- von einer MPEG-Kodierungskarte,

an:

- einen Rechner (d.h. an eine IP-Adresse): dies wird *Unicast* genannt,
- eine dynamische Gruppe von Rechner, an der Clients teilnehmen oder diese verlassen können (d.h. an eine Multicast-IP-Adresse): dies wird *Multicast* genannt,

per IPv4 oder IPv6.

Um die komplette Liste von VLCs Fähigkeiten auf jeder entsprechenden Plattform zu erhalten, schauen Sie auf die VLC-Features-Seite (<http://www.videolan.org/vlc/features.html>) (auf Englisch).

Note: VLC läuft nicht unter Mac OS 9 und wird es wahrscheinlich nie tun.

VLS

VLS kann streamen:

- MPEG-1, MPEG-2 oder MPEG-4 - Dateien, die auf einer Festplatte oder auf einer CD gespeichert sind,
- DVDs in einem lokalen DVD-Laufwerk oder kopiert auf eine Festplatte,
- Daten einer Satellitenkarte (DVB-S) oder einer digitalen terrestrischen TV-Karte (DVB-T) ,
- Daten einer MPEG-Kodierungskarte ;

an :

- einen Rechner (d.h. an eine IP-Adresse): dies wird *Unicast* genannt,
- eine dynamische Gruppe von Rechnern, an der die Clients teilnehmen, diese aber auch wieder verlassen können (d.h. an eine Multicast-IP-Adresse): dies wird *Multicast* genannt,

per IPv4 oder IPv6.

Ein Rechner mit einem Pentium-Prozessor mit 100 MHz und 32 MB Arbeitsspeicher sollte ausreichend sein, um einen Stream über das Netzwerk zu versenden. Wenn viele auf einer Festplatte gespeicherte Videos gestreamt werden, ist die wirkliche Beschränkung nicht der Prozessor, sondern die Festplatte und die Netzwerkanbindung.

VLS läuft unter Linux und Windows. Um die komplette Liste von VLS' Möglichkeiten auf der jeweiligen Plattform zu erhalten, schauen Sie auf die Streaming-Features-Seite (<http://www.videolan.org/streaming/features.html>).

Mini-SAP-server

Sie können einen Kanal-Informationdienst, basierend auf dem SAP/SDP-Standard, der VideoLAN-Lösung hinzufügen. Der mini-SAP-Server sendet Ankündigungen über die Multicast-Programme im Netzwerk in IPv4 und IPv6.

VLC-Anwendungen erhalten diese Ankündigungen und fügen die Programme automatisch ihrer Wiedergabeliste hinzu.

Der mini-SAP-Server läuft unter Linux und Mac OS X.

Was ist ein Codec?

Um die VideoLAN Lösung komplett zu verstehen, müssen Sie den Unterschied zwischen einem *Codec* und einem *Container-Format* kennen.

- Ein *Codec* ist ein Kompressionsalgorithmus, der zur Reduzierung der Größe eines Streams benutzt wird. Es gibt Ton- und Bildcodcs. MPEG-1, MPEG-2, MPEG-4, DivX, usw. sind Codcs.
- Ein *Container-Format* enthält einen oder mehrere Streams, die bereits von einem Codec kodiert worden sind. Sehr oft gibt es einen Bild- und einen Tonstream. AVI, Ogg, MOV, ASF, usw. sind Container-Formate. Die enthaltenen Streams können mit unterschiedlichen Codcs kodiert worden sein. In einer perfekten Welt könnten Sie jeden Codec in jedes Container-Format legen. Leider gibt es einige Inkompatibilitäten. Sie können eine Matrix von möglichen Codcs und Containerformaten auf der Featuresseite (<http://www.videolan.org/streaming/features.html>) finden.

Um einen Stream zu dekodieren, *demultiplex* VLC ihn erst. Das bedeutet, dass es das Containerformat liest und Ton, Bild und Untertitel trennt, wenn welche vorhanden sind. Danach wird jeder davon durch *Dekoder* geleitet, die die mathematischen Vorgänge zur Dekomprimierung ausführen .

MPEG ist ein spezieller Fall:

- MPEG ist ein *Codec*. Es gibt mehrere Versionen davon, genannt MPEG-1, MPEG-2, MPEG-4...
- MPEG ist außerdem ein Containerformat, manchmal bezeichnet als *MPEG System*. Es gibt mehrere Typen von MPEG: ES, PS und TS

Wenn Sie ein MPEG-Video beispielsweise von einer DVD abspielen, besteht der MPEG-Stream aus mehreren Streams ('Elementare Streams'/'ES' genannt): es gibt einen für Bild, einen für Ton, einen weiteren für Untertitel, usw. Diese unterschiedlichen Streams sind in einen einzelnen 'Program Stream' (PS) zusammengemischt. Die .VOB-Dateien, die Sie auf einer DVD finden, sind tatsächlich MPEG-PS-Dateien. Aber dieses PS-Format ist nicht für das Streamen von Videos beispielsweise über ein Netzwerk oder Satellit ausgelegt, sodass ein anderes Format, namens 'Transport Stream' (TS) für das Streamen von MPEG-Videos über solche Kanäle designed wurde.

Wie benutze ich VideoLAN?

Dokumentation

Die Benutzerdokumentation von VideoLAN besteht aus 4 Dokumenten:

- Das *VideoLAN HOWTO*-Dokument ist das komplette Handbuch über die VideoLAN-Streaming-Lösung. Es enthält praktische Beispiele zur Aufsetzung Ihrer Streaming-Lösung.
- Der *VLC user guide*. Dieses Dokument ist das komplette Handbuch für VLC.
- Der *VLS user guide*. Dieses Dokument ist das komplette Handbuch für VLS.
- Die *VideoLAN-FAQ*. Dieses Dokument enthält oft gestellte Fragen über VideoLAN.

Die neueste Version dieser Dokumente kann auf der Dokumentationseite (<http://www.videolan.org/doc/>) gefunden werden.

Benutzersupport

Wenn Sie Probleme bei der Benutzung von VideoLAN haben und die Antwort auf Ihre Probleme in der Dokumentation nicht finden können, auf Sie bitte in das Onlinearchiv der Mailinglisten (<http://www.via.ecp.fr/via/ml/videolan-en.html>). Es gibt zwei englischsprachige Mailinglisten für die Benutzer:

- vlc@videolan.org für die Fragen über VLC ,
- streaming@videolan.org für die Fragen über VLS, mini-SAP-Server und das Netzwerk.

Wenn Sie sich bei einer dieser Listen einschreiben oder austragen möchten, gehen Sie bitte auf die Mailinglisten Seite (<http://www.videolan.org/support/lists.html>).

Sie können auch mit VideoLAN Benutzern und Entwicklern über IRC sprechen: Server *irc.freenode.net*, Channel *#videolan*.

Wenn Sie einen Fehler finden, folgen Sie bitte der Anleitung auf der Fehlermeldeseite (<http://www.videolan.org/support/bug-reporting.html>).

Befehlszeilenbenutzung

- VLC hat viele unterschiedliche grafische Interfaces, die sehr unterschiedlich aufgebaut sind, um mit den Richtlinien des jeweiligen System zu entsprechen. Die Funktionalität jedes Interfaces zu dokumentieren, wäre zu aufwendig und einige Features sind nur über die Befehlszeile verfügbar. Deshalb haben wir uns entschieden, nur das Befehlszeileninterface zu dokumentieren, aber in vielen Fällen sollte es einfach zu erraten sein, wie das grafische Interface entsprechend zu benutzen ist!
- VLS hat ein Befehlszeilen- und ein Telnet-Interface, aber keine grafische!

Alle Befehle, die in diesem Dokument auftauchen, sollten in ein Terminal eingeben werden..

Öffnen Sie ein Terminal

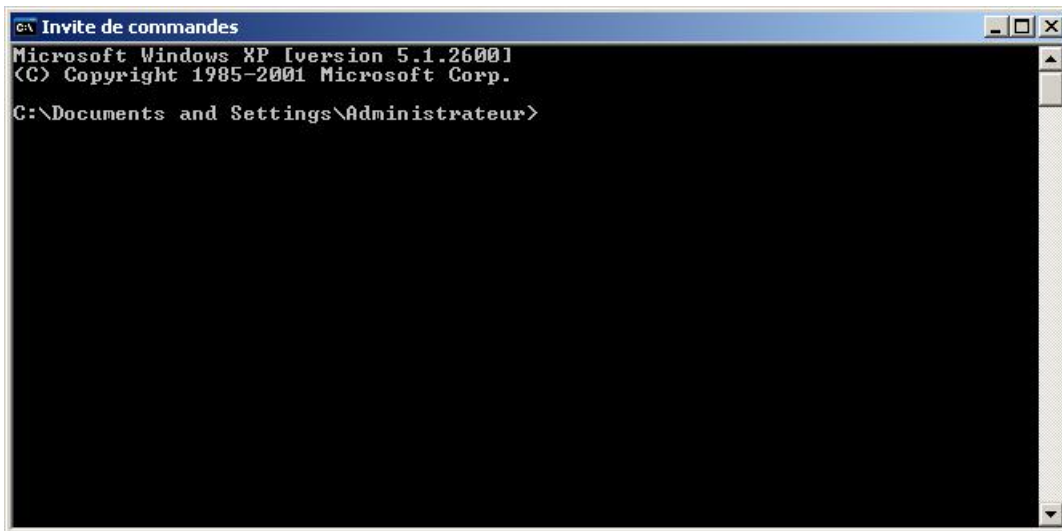
Windows

Klicken Sie auf *Start, Ausführen* und geben Sie ein:

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

Das Terminal erscheint

Figure 1-2. Windows terminal



Note: Unter Windows müssen Sie in dem Verzeichnis sein, wohin das Programm installiert wurde, um es auszuführen.

Linux / Unix

Öffnen Sie ein Terminal :

Figure 1-3. Linux X terminal



In der Dokumentation adoptieren wir die folgenden Konventionen für die Unix-Befehle:

- Befehle, die als *root* eingegeben werden sollten, beginnen mit einem #:

```
#
```

```
befehl_der_als_root_eingegeben_werden_soll
```

- Befehle, die als normaler Benutzer eingegeben werden sollten, beginnen mit einem %:

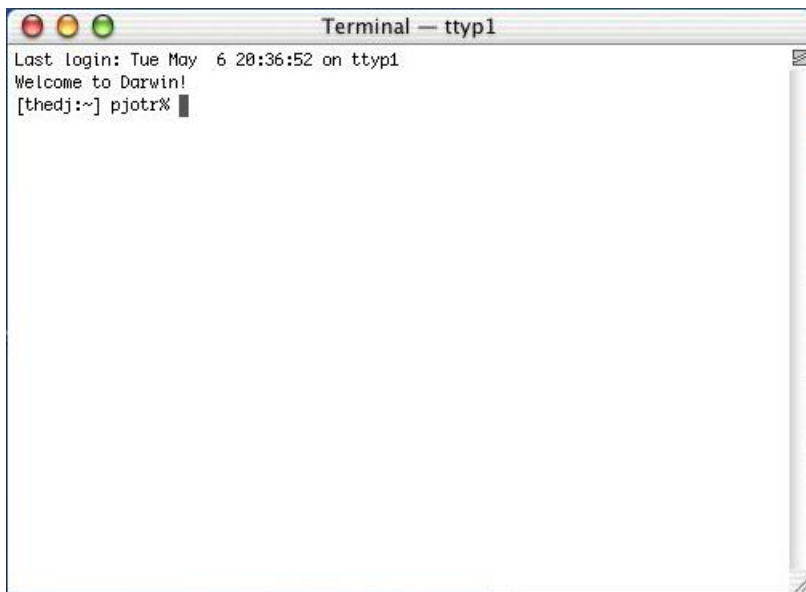
```
%
```

```
befehl_der_als_normaler_benutzer_eingegeben_werden_sollte
```

Mac OS X

Gehen Sie in den Ordner *Programme*, öffnen Sie den Ordner *Dienstprogramme* und doppelklicken Sie auf *Terminal* :

Figure 1-4. Mac OS X terminal

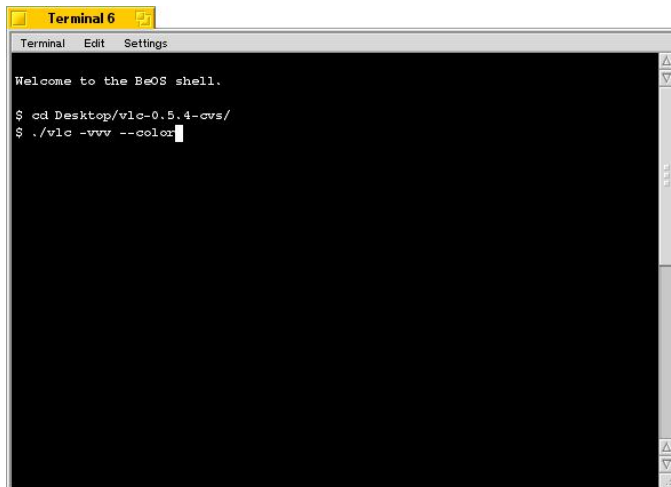


Note: Unter Mac OS X müssen Sie in dem Ordner sein, in dem das Programm installiert wurde, um es auszuführen und der Befehl muss mit `./` beginnen.

BeOS

In der Schreibtischleiste gehen Sie auf *Application* und dann auf *Terminal*:

Figure 1-5. BeOS terminal



Note: Unter BeOS müssen Sie in dem Verzeichnis sein, wo das Programm installiert wurde, um es auszuführen und der Befehl muss mit `./` beginnen.

Chapter 2. Installation von VLS

Installation von VLS

Windows

Laden Sie die ZIP-Datei von der VLS Windows Downloadseite (<http://www.videolan.org/streaming/download-vls-windows.html>) herunter, entzippen Sie sie und führen Sie `setup.exe` aus.

GNU/Linux & Mac OS X

Installation der Bibliotheken/Libraries

Viele Bibliotheken werden für bestimmte Zwecke benötigt:

- *libdvbpsi* (immer benötigt)
- *libdvdcss* wenn Sie auf verschlüsselte DVDs zugreifen möchten ,
- *libdvbread* wenn Sie in der Lage sein möchten, DVDs zu streamen ,
- *libdvb* wenn Sie in der Lage sein möchten, von einer DVB-Karte streamen zu können (eine Satelliten- oder eine digitale terrestrische TV-Karte).

Laden Sie die Bibliotheken von der VLS Quellen Downloadseite (<http://www.videolan.org/streaming/download-vls-sources.html>) herunter.

Dekomprimieren, konfigurieren (außer *libdvb*, welche kein *./configure* hat), kompilieren und installieren Sie jede Bibliothek:

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

Überprüfen Sie unter GNU/Linux, ob die Konfigurationsdatei `/etc/ld.so.conf` die folgende Zeile enthält:

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

Wenn dies nicht der Fall ist, fügen Sie diese hinzu und starten Sie dann:

```
# ldconfig
```

Führen Sie unter Mac OS X aus:

```
# ranlib /usr/local/lib/*.a
```

Installation von VLS

Laden Sie die Quellen der neuesten Version herunter: holen Sie sich die Datei `vls-version.tar.gz` von der VLS-Quellen Downloadseite (<http://www.videolan.org/streaming/download-vls-sources.html>). Dekomprimieren Sie es und generieren Sie `./configure`:

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

Um die Liste der Konfigurationsoptionen, machen Sie folgendes:

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

Konfigurieren Sie dann VLS:

- wenn Sie ein einfaches VLS ohne DVD-Unterstützung möchten, machen Sie:

```
% ./configure --disable-dvd
```

- wenn Sie ein VLS mit DVD-Unterstützung möchten, machen Sie:

```
% ./configure
```

- wenn Sie ein VLS mit DVB-Unterstützung möchten, machen Sie:

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

Kompilieren und installieren Sie danach:

```
% make
# make install
```

Sie können auch **make uninstall**, **make clean** oder **make distclean** nach Bedarf ausführen.

Deinstallation von VLS

Windows

Gehen Sie in die *Systemsteuerung*, klicken Sie auf *Programme hinzufügen und entfernen*, wählen Sie *VLS* und klicken Sie auf *Bearbeiten/Entfernen* und folgen Sie den Schritten zur Deinstallation des Programms.

Wenn Sie VLS aus den Quellen kompiliert haben

Gehen Sie in das Verzeichnis, das die Quellen von VLS enthält und führen Sie aus:

```
# make uninstall
```

Danach können Sie die Quellen von VLS entfernen.

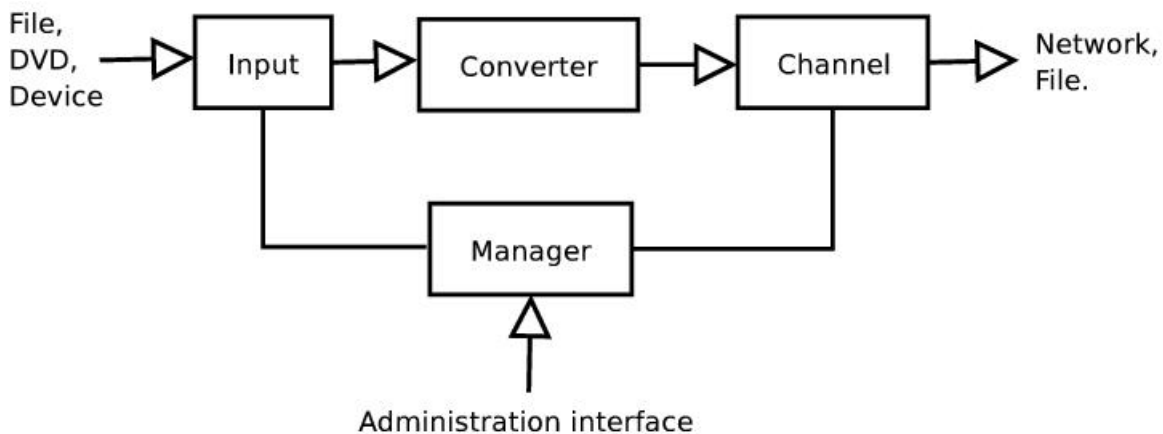
Chapter 3. Überblick und grundlegende Konzepte

Struktur von VLS

Aus der Sicht eines Benutzer kann VLS in vier Hauptkomponenten aufgeteilt werden:

- ein *Manager*,
- *Inputs*,
- *Konverter*,
- und *Channel* bzw. Ausgabekanäle.

Figure 3-1. Struktur von VLS



Die Inputs

Die Rolle eines *Inputs* ist MPEG Streams aus einer gegebenen Quelle (Datei, DVD, DVB-Karte, Gerät...) zu lesen und die richtigen *Konverter* damit zu "füttern". Ein Input kann in der Lage sein mehrere Streams, *Programme* genannt, zu lesen. Es gibt mehrere Arten von Inputs:

- den *local*-Input, der Videos aus Dateien oder DVDs lesen kann,
- den *Video*-Input, der Videos aus MPEG-Kodierungskarten lesen kann,
- den *DVB*-Input, der Videos von DVB-Karten lesen kann,
- den *v4l*-Input, der Videos aus Zugangskarten lesen kann, die von den Video4Linux-Treibern unterstützt werden.

Sie können gleichzeitig mehrere Inputs benutzen und mehrere Programme abspielen.

Die Konverter

Die Rolle eines *Konverters* ist einen Stream von einem Input zu erhalten und ihn in das MPEG-TS-Format zu konvertieren. VLS ist in der Lage PS-Streams (beispielsweise von DVDs) in TS-Streams zu konvertieren (ps2ts-Konverter). Natürlich kann es auch TS-Streams lesen und sie reparieren, in dem Streamunterbrechungen berücksichtigt werden (ts2ts-Konverter)..

Die Channel

Ein *Channel* erhält einen Stream von einem Konverter und sendet ihn an ein gegebenes Ziel (Netzwerk, Datei...). Wenn Sie möchten, können Sie einen "Channel" auch einen "Output" nennen. Dies ist das gleiche. Derzeit gibt es zwei unterstützte Arten von Channeln: *network* und *file*. Beachten Sie das VLS derzeit nur einen Output pro Stream unterstützt. Sie können also nicht einen Streamen über das Netzwerk senden und ihn gleichzeitig in eine Datei speichern. Der Netzwerk-Output umfassend konfigurierbar. Sie können wählen, welches Netzwerk-Interface Sie benutzen, sowie Quell- und Ziel-IP-Adressen angeben .

Der Manager

Der *Manager* kontrolliert die gesendeten Streams. Durch ein *Administrationsinterface* können Sie dem Manager mitteilen, die verschiedenen Programme zu starten, stoppen, anzuhalten, wiederaufzunehmen, zu beschleunigen oder zu verlangsamen. Sie können auch eine Liste aller Programme erhalten. Der Manager liest diese aus der VLS-Konfigurationsdatei (`vls.cfg`), weshalb diese nicht geändert werden darf, nachdem VLS gestartet wurde. Derzeit können Sie den Manager nicht fragen, ob ein angegebener Stream gesendet wird. Sie erhalten aber eine Fehlermeldung, wenn Sie versuchen einen Stream anzuhalten, der nicht gesendet wird .

Das Administrationsinterface

Es gibt derzeit zwei Wege, das Streamen zu starten :

- Sie können die *Befehlszeile* benutzen, um Argumente beim Start zu übergeben
- oder Sie können das *Telnet-Interface* benutzen, um das Streamen zu starten/anzuhalten/zu pausieren wann immer Sie möchten.

Wenn Sie das Telnet-Interface benutzen, müssen Sie sich authentifizieren, bevor sie einen Befehl angeben, weil nicht jedem Benutzer erlaubt werden darf, einen Befehl auszuführen (dies kann in der Konfigurationsdatei `vls.cfg` eingestellt werden).

Chapter 4. Konfiguration

VLS liest seine Einstellungen aus der Konfigurationsdatei `vls.cfg`, welche im aktuellen Verzeichnis oder in `SYSCONF_DIR/videolan/vls` sein sollte (wobei `SYSCONF_DIR /usr/local/etc`, wenn Sie VLS von Hand erstellt und installiert haben, oder `/etc` ist, wenn Sie das binäre Debianpaket installiert haben.).

Um eine `vls.cfg`-Datei zuschreiben, nehmen Sie am Besten die mit VLS mitgelieferte als Ausgangspunkt.

Allgemeine Struktur

Die Konfigurationsdatei `vls.cfg` ist in Sektionen aufgeteilt. Jede darf mehrere Variablen enthalten:

```
BEGIN "ErsteSektion"
  Variable1 = "Wert1"
  Variable2 = "Wert2"
  [...]
END
```

```
BEGIN "ZweiteSektion"
  Variable1 = "Wert1"
  Variable3 = "Wert3"
  [...]
END
```

```
[...]
```

Alle Sektionsnamen, Variablennamen und Werte sind nicht case-sensitive, d.h. sie werden unabhängig von Groß oder Kleinschreibung ausgewertet. Es darf leere Sektionen und Untersektionen geben. Kommentare müssen einem `#` folgen. Einige Variablen haben einen standardmäßigen Wert. Dies bedeutet, dass Sie die Deklaration dieser Variablen weglassen können. Dann wird diesen ihr Standardwert zugewiesen.

Schreiben einer `vls.cfg`-Datei

Nun folgt eine Erklärung aller Sektionen, die in einer `vls.cfg`-Datei gefunden werden können:

Section "VLS"

Diese Sektion enthält programmweite Einstellungen.

```
LogFile = "name"
```

Der Name der VLS-Logdatei. Wenn er leer ("") gelassen wird, wird kein Protokoll in einer Datei gespeichert. Der Standardwert ist `"vls.log"`.

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

Protokollierung im Log des Systems. Derzeit wird nur `syslog` unterstützt. Kompilieren Sie mit `./configure --enable-syslog`.

Caution

Wenn VLS durch **vlsd** gestartet wird, ist die folgende Konfiguration verbindlich:

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

ScreenLog = "[disable|enable]"

Protokollierung in der Konsole.

Beispiel:

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

Sektion "Groups"

In dieser Sektion können Sie Benutzergruppen und die Befehle, die diese ausführen dürfen, festlegen. Für jede Gruppe, die Sie festlegen möchten, müssen Sie eine Zeile im folgenden Format hinzufügen:

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

Diese fügt eine Gruppe "groupname" hinzu, deren Nutzern erlaubt ist, command1, command2 usw. auszuführen. Derzeit sind die verfügbaren Befehle help, browse, start, suspend, resume, forward, rewind, stop, shutdown und logout.

Beispiel:

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

Sektion "Users"

Diese Sektion enthält eine Liste von Benutzern, denen es gestattet ist, VLS durch ein Administrationsinterface zu steuern. Fügen Sie für jeden Benutzer eine Zeile im folgenden Format hinzu:

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

Dies fügt einen Benutzer "username" hinzu, der zur Gruppe "groupname" gehört, wie sie in der "Groups"-Sektion festgelegt wurde, und sich mit dem Passwort "password" anmelden kann.

- Unter Unix/Linux muss das Passwort mit einem Tool wie **mkpasswd** oder mit der Unix-Funktion "crypt" verschlüsselt werden.
- Unter Windows muss das Passwort in klaren Text sein.

Beispiel für Unix/Linux:

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

Sektion "Telnet"

In dieser Sektion können Sie das Telnetadministrationsinterface einstellen.

```
LocalPort = "port"
```

Legt fest, welcher Port für den Telnetserver benutzt werden wird. Der Standardport ist "9999".

```
Domain = "domain"
```

Entweder "inet4" oder "inet6" (der Standard ist "inet4"). Wenn Sie IPv4-Adressen benutzen möchten, nehmen Sie "inet4". Wenn Sie IPv6 benutzen möchten, nehmen Sie "inet6".

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

Legt fest, auf welcher IP-Adresse der Telnetserver nach Befehlen hören wird. Die Standardadresse ist "0.0.0.0" (oder "0::0" bei IPv6).

Beispiel:

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

Sektion "NativeAdmin"

Die gleiche Syntax wie "Telnet". Wird noch nicht benutzt.

Sektion "Inputs"

In dieser Sektion legen Sie fest, welche Inputs Sie benutzen möchten. Fügen Sie für jeden Input, den Sie benötigen, eine Zeile im folgenden Format hinzu:

```
InputName = "Type"
```

Dies fügt einen Input namens "InputName" hinzu, dessen Typ "Type" ist. Wie bereits erklärt, gibt es mehrere Arten von Inputs:

- "local", um einen Stream aus einer Datei oder einer DVD wiederzugeben ,
- "video", um einen Stream einer MPEG-Kodierungskarte wiederzugeben ,
- "dvb", um einen Stream einer DVB-Karte wiederzugeben ,
- "v4l" um einen Stream eines Video4Linux-Gerätes wiederzugeben.

Jeder Input muss in seiner eigenen Sektion konfiguriert werden (siehe den nächsten Abschnitt).

Beispiel:

```
BEGIN "Inputs"
```

```

local1 = "local"
pvr     = "video"
dvb1    = "dvb"
tuner   = "v4l"
END

```

Konfiguration der Inputs

Für jeden Input, der in der "Inputs"-Sektion deklariert wurde, müssen Sie, abgesehen von "local"-Inputs, eine Sektion mit dem gleichen Namen hinzufügen. Wenn Sie beispielsweise den Input "pvr" deklariert haben, sollte es auch eine Sektion namens "pvr" geben. Die Syntax solcher Sektionen hängt vom Typus des Inputs ab.

Um einen lokalen Input zu konfigurieren, müssen Sie nichts tun, es sei denn, dass eine andere "trickplay"-Strategie benutzt werden muss:

```

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

```

"Local1" ist der Name des lokalen Inputs, den Sie konfigurieren möchten. "ProgramCount" ist die Anzahl von Programmen, die diesem Input zugewiesen worden sind. "TrickPlay" ist die trickplay-Strategie, die von diesem Input benutzt wird (standardmäßig "normal").

Um einen Video-Input zu konfigurieren, fügen Sie eine Sektion im folgenden Format hinzu:

```

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

```

"VideoInputName" ist der Name des Video-Inputs, den Sie konfigurieren möchten. "Device" ist der Pfad der MPEG-Kodierungskarte, von der Sie lesen möchten (standardmäßig "/dev/video"). "Type" ist entweder "Mpeg2-PS" oder "Mpeg2-TS", entsprechend Ihrer Geräteeinstellung (standardmäßig "Mpeg2-PS").

Beispiel für eine Hauppauge WinTV-PVR-250 - Karte:

```

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

```

Um einen DVB-Input zu konfigurieren, fügen Sie eine Sektion im folgenden Format hinzu:

```

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

```

Beispiel:

```

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

```

Sektion "Channels"

In dieser Sektion können Sie die Channel (Outputs), die Sie benutzen möchten, definieren. Fügen Sie pro Channel eine Zeile im folgenden Format hinzu:

```
ChannelName = "Type"
```

Dies fügt einen Channel namens "ChannelName" hinzu, dessen Typ "Type" ist. "Type" muss entweder "network" oder "file". Wie Inputs müssen Channel in ihrer eigenen Sektion konfiguriert werden.

Beispiel:

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

Konfiguration der Channel

Für jeden in der "Channels"-Sektion deklarierten Channel müssen Sie eine Sektion mit dem gleichen Namen hinzufügen. Die Syntax einer solchen Sektion hängt vom Typus des entsprechenden Channels ab.

Um einen network-Channel zu konfigurieren, fügen Sie eine Sektion im folgenden Format hinzu:

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

- "NetChannelName" ist der Name des network-Channels, den Sie einstellen möchten.
- "Domain" ist entweder "inet4", wenn Sie IPv4-Adressen benutzen, oder "inet6", wenn Sie IPv6 benutzen (standardmäßig "inet4").
- "Type" ist entweder "unicast", "broadcast" oder "multicast" (standardmäßig "multicast"), abhängig von dem, was Sie machen möchten (und Ihrer "DstHost"-Adresse).
- "SourceHost" ist die IP-Adresse (oder der DNS-Name) von der VLS den Stream senden wird.
- "SourcePort" ist der UDP-Port von dem der Stream gesendet werden wird.
- "DestHost" ist die IP-Adresse (oder der DNS-Name), an den der Stream gesendet werden wird.
- "DestPort" ist der UDP-Port, an den der Stream gesendet werden wird (standardmäßig "1234").
- "TTL" ist nur dann eine nützliche Option, wenn "Type" "multicast" ist. Der Standardwert ist "0". Sie können ihn zur Erhöhung der TTL Ihrer Multicast-Pakete benutzen, wenn diese mehrere Router überqueren müssen.
- "Interface" ist eine nur unter GNU/Linux unterstützte Option, um das Senden des Streams über ein bestimmtes Netzwerk-Interface, beispielsweise "eth1", zu erzwingen. Zur Benutzung dieser Option benötigen Sie root-Rechte.

Note: "SrcHost" und "SrcPort" sind optional (wenn Sie sie nicht einstellen, wird VLS das Socket nicht "binden").

Um einen file-Channel einzustellen, fügen Sie eine Sektion im folgenden Format hinzu:

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

"FileChannelName" ist der Name des file-Channels, den Sie einstellen möchten. "file" ist der Name der Datei, in der Stream gespeichert werden wird (standardmäßig "fileout.ts"). "append" ist entweder "yes" oder "no" und legt fest, ob VLS den Stream an der Ende der Datei anhängen oder sie neu schreiben wird.

Beispiel:

```
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

Wenn Sie Windows benutzen, sollten Sie die "SrcHost" und "SrcPort" Werte festlegen. Zum Beispiel:

```
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
```

Konfiguration der Programme

Wie bereits erklärt, müssen Sie die Programme definieren. Jedes ist ein MPEG-Stream (eine Datei beispielsweise). Um dies zu tun, müssen Sie eine "Input"-Sektion Ihrer `vls.cfg`-Datei hinzufügen. Jede "Input"-Sektion muss die folgende Syntax haben:

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

"path" ist der Pfad, in dem Ihre MPEG-Dateien gespeichert sind (standardmäßig das aktuelle Verzeichnis). "count" ist die Anzahl von definierten Programmen (standardmäßig "0").

Für jedes Programm, das Sie definieren möchten, müssen Sie eine Sektion im folgenden Format hinzufügen:

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

- "number" ist die Programm-Nummer: das erste Programm hat die Nummer 1, das Zweite die Nummer 2, usw.
- "name" ist der Name des Programms, den Sie zum Start des Programms benutzen (siehe dazu das nächste Kapitel).
- "type" kann "Mpeg1-PS", "Mpeg2-PS", "Mpeg2-TS", oder "DVD" sein. Wenn Ihr Stream in einer MPEG-Datei (*.mpeg, *.mpg, *.vob, usw.) gespeichert ist, wird es wahrscheinlich im Mpeg1-PS oder Mpeg2-PS - Format sein.
- Falls "type" auf Mpeg1-PS", "Mpeg2-PS" oder "Mpeg2-TS" eingestellt, wird VLS annehmen, dass der Stream in der Datei "file" im Verzeichnis "path" gespeichert ist (die "path"-Variable wurde in der "Input"-Sektion festgelegt).
- Falls "type" "DVD" ist, wird die Variable "Device" anstatt von "FileName" benutzt werden (die Variable "FilePath" wird nicht auf den Devicenamen angewendet!). Die Variable "Device" ist das Device Ihres DVD-Laufwerkes (beispielsweise "/dev/hdc" oder "/dev/cdrom"). Sie können auch eine auf eine Festplatte kopierte DVD wiedergeben. Dann ist "Device" das Verzeichnis, in dem die .vob-Dateien gespeichert sind (beispielsweise "/mnt/data/VIDEO_TS").

Note: VLS kann MPEG-Dateien streamen, die zwei Kriterien entsprechen

- Die Datei muss entweder MPEG *PS* (Program Stream) oder MPEG *TS* (Transport Stream) sein, das gemultiplextes Bild und Ton enthält. VLS kann kein MPEG *ES* (Elementary Stream), d.h. eine Datei mit nur Ton oder Bild, streamen.. Um zu erfahren, ob eine MPEG-Datei MPEG *PS*, MPEG *TS* oder MPEG *ES*, lesen Sie die Datei mit VLC und schauen Sie auf die Meldungen (im Meldungenfenster oder benutzen Sie die Befehlszeile **vlc -vvv**).

- Wenn Sie sehen:

```
[00000107] main module debug: using demux module "ts_dvbpsi"
```

ist die Datei MPEG TS.

- Wenn Sie sehen :

```
[00000109] main module debug: using demux module "ps"
```

ist die Datei MPEG PS.

- Wenn Sie sehen:

```
[00000109] main module debug: using demux module "es"
```

ist die Datei MPEG ES, was VLS nicht streamen kann.

- Der Sequenz-Header des Videos muss sich selbst regelmäßig wiederholen, was bei MPEG-2 oft der Fall ist, jedoch selten bei MPEG-1. Es gibt keinen einfachen Weg zu erfahren, ob der Sequenz-Header regelmäßig wiederholt wird. Datei mit einem *.vob*-Suffix sind normalerweise MPEG-2 - Dateien und Dateien mit dem *.mpg*. oder *.mpeg*-Suffix sind üblicherweise MPEG-1 - Dateien.

Sie können diese streambare MPEG-2 PS - Datei für Ihre Tests herunterladen: *presentation_short.vob* (ftp://ftp.videolan.org/pub/videolan/streams/presentation/presentation_short.vob).

Note: Um DVDs wiedergeben zu können, müssen Sie VLS mit DVD-Unterstützung, die *libdvdread* und *libdvdcss* benutzt, kompilieren. Sie werden Lese- *und* Schreibzugriff auf das DVD-Device benötigen.

Komplettes Beispiel:

```
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. Benutzung von VLS

VLS starten

Wenn Sie das Telnet-Interface benutzen, ist das Starten von VLS sehr einfach: geben Sie einfach **vls** in eine Shellconsole ein. Das ist alles. Das Starten von **vlsd** wird VLS als Daemon starten und sich aus der startenden Shell ausklinken. Beachten Sie, dass VLS versuchen wird seine Konfigurationsdatei (`vls.cfg`) aus dem aktuellen Verzeichnis zu lesen. Falls keine `vls.cfg`-Datei dort sein sollte, wird es versuchen, sie aus dem Verzeichnis `SYSCONF_DIR/etc/videolan` zu laden (siehe Sektion *Konfiguration*).

Caution

Wenn Ihre Log-Datei `vls.log` wie im Beispiel ist, benötigt VLS Schreibzugriff auf das aktuelle Verzeichnis oder Sie werden etwas vergleichbares mit dem Folgenden sehen. :

```
*** 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
```

Beachten Sie außerdem, dass Sie Sie root sein müssen, wenn Sie die "Interface"-Option in `vls.cfg` benutzen.

Wenn alles in Ordnung ist, werden Sie ähnliches wie das Folgende sehen :

```
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
[...]
```

Was Sie auf dem Bildschirm (stderr) sehen, ist genau dasselbe wie in der Log-Datei `vls.log`.

Wenn VLS erfolgreich gestartet wurde, übernimmt es keinen Befehl aus dessen Standardinput, sodass Sie ihn in den Hintergrund verlagern können (sie können das *screen*-Utility dazu benutzen).

Wenn Sie aber das Befehlszeileninterface benutzen möchten, schauen Sie bitte in das Dokument VideoLAN HOWTO (<http://www.videolan.org/doc/>).

Benutzung des Telnet-Interfaces

Nachdem VLS gestartet wurde, öffnet es einen Telnet-Server (standardmäßig auf dem Port 9999). Sie können sich mit dem folgenden Befehl mit diesem Server in Verbindung setzen :

```
% telnet localhost 9999
```

Sie sollten etwas ähnliches wie das Folgende sehen :

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

```
VideoLAN Server Administration System
```

Login:

Danach müssen Sie sich mit einem Loginnamen und entsprechendem Passwort authentifizieren, wie Sie es in der Datei `vls.cfg` festgelegt haben. Wenn Sie sich erfolgreich authentifiziert haben, sollten Sie in etwa folgende Eingabeaufforderung sehen :

```
admin@vls>
>
```

Sie können nun die im folgenden Paragraph erklärten Befehle benutzen. Um sich abzumelden, geben Sie **logout** nach der Telnet-Eingabeaufforderung ein.

Interfacebefehle

help

Benutzung: `help [Befehl]`.

Ohne Argument aufgerufen, gibt "help" die Liste aller Befehle aus, ob sie nun verfügbar sind oder nicht. Mit einem Argument aufgerufen, zeigt es an, wie der jeweilige Befehl benutzt wird.

browse

Benutzung: `browse [Input]`.

Ohne Argument aufgerufen, gibt "browse" alle Programme der Inputs aus. Mit einem Argument aufgerufen, gibt es nur die Programme des angegebenen Inputs an. Jedes Programm wird mit dessen Status angezeigt.

start

Benutzung: `start <Programm> <Channel> <Input> [--loop] [--rtp]`

"start" startet das angegebene Programm des angegebenen Inputs und gibt es über den angegebenen Channel aus. Durch die "--loop"-Option wird das Programm unendlich wiederholt. Durch die "--rtp"-Option werden die TS-Pakete über das RTP-Protokoll gesendet, wie es in RFC 1889 und 2250 festgelegt ist .

stop

Benutzung: `stop <Channel>`

"stop" beendet das Senden des angegebenen Channels.

forward

Benutzung: `forward <Channel> <Geschwindigkeit>`

"forward" beschleunigt den Channel um die angegebene Geschwindigkeit. Dies funktioniert nicht, wenn direkt aus einem Gerät, wie einer MPEG-Kodierungskarte, einer DVB-Karte oder einer Eingangskarte, gelesen wird.

rewind

Benutzung: rewind <Channel> <Geschwindigkeit>

"rewind" verlangsamt den Channel um die angebene Geschwindigkeit. Dies funktioniert nicht, wenn direkt aus einem Gerät, wie einer MPEG-Kodierungskarte, einer DVB-Karte oder einer Eingangskarte, gelesen wird.

suspend

"suspend" unterbricht das Streamen des angegebenen Channels.

resume

Benutzung: resume <Channel>

"resume" nimmt das Streamen des angegebenen Channels wieder auf.

logout

Benutzung: logout

"logout" schließt die derzeitige Administrationssitzung und die entsprechende Verbindung.

shutdown

Benutzung: shutdown

"shutdown" stoppt alle Sendungen und beendet 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.