

UNIVERSIDAD POLITECNICA DE VALENCIA

ESCUELA POLITECNICA SUPERIOR DE GANDIA

MASTER EN POSTPRODUCCION DIGITAL



UNIVERSIDAD
POLITECNICA
DE VALENCIA



ESCUELA POLITECNICA
SUPERIOR DE GANDIA

Solución Cliente-Servidor para IPTV basada en VLC

TRABAJO FINAL DE MASTER

Autor: Alain Chas Arizcuren
Director: Jaime Lloret Mauri

Gandía, julio de 2015

Resumen

La evolución de la televisión tradicional al formato IPTV se está produciendo a una gran velocidad, debido a los profundos cambios aparecidos tanto a nivel de sistemas y redes de comunicaciones como de dispositivos. Esta evolución provoca que los nuevos proveedores y suministradores de servicios de IPTV necesiten disponer de herramientas que permitan la gestión de sus contenidos. Este proyecto ha consistido en el desarrollo de una nueva herramienta, de nombre 'AC TV Manager', que sirve para la gestión de los canales, la continuidad televisiva y la posterior emisión vía streaming que debe proporcionar un servicio de IPTV. Se ha desarrollado en lenguaje Java y sirve como interfaz entre un usuario administrador, VLC y FFmpeg, pues uno de los objetivos es que la herramienta no sea propietaria.

Abstract

The switch from traditional television to Internet Protocol television (IPTV) is accelerating as a result of profound changes to devices, systems and communication networks. Due to this evolution, IPTV service suppliers and providers require tools to manage their contents. This project has inspired the development of new software called 'AC TV Manager' which manages the channels, televised continuity and posterior broadcast via streaming provided by an IPTV service. It has been developed in Java and is an interface between the admin user, VLC and FFmpeg, as one of its objectives is for the tool not to be proprietary software.

Palabras Clave

IPTV, VLC, Gestión, Vídeo, Software

Keywords

IPTV, VLC, Management, Video, Software

Índice

1	Capítulo 1: Introducción.....	6
1.1	Introducción	6
1.2	Objetivos	7
1.3	Precedentes del proyecto	7
1.3.1	Motivación y punto de partida.....	7
1.3.2	Contexto	7
1.4	Estructura del proyecto.....	9
2	Estado del arte	9
3	Sistema servidor: Problemas afrontados y soluciones adoptadas.....	10
3.1	Programación horaria con código VLM.....	10
3.1.1	Media	11
3.1.2	Schedule	12
3.1.3	Conclusión	12
3.2	Conexión con la interfaz VLM.....	12
3.2.1	VLM vía telnet	13
3.2.2	Comandos que ofrece la interfaz telnet.....	13
3.3	Obtención de duración de los vídeos.....	13
3.4	Elección de la librería gráfica para la interfaz de usuario	14
3.4.1	Opciones barajadas en proyecto previo.....	14
3.4.2	Opciones barajadas para la nueva versión.....	17
3.4.3	Elección final: librería gráfica SWT	17
4	Sistema Servidor: Estructura general.....	19
4.1	Esquema gráfico del sistema.....	19
4.2	Software requerido	20
4.2.1	JDK.....	20
4.2.2	Shell (o consola)	20
4.2.3	FFmpeg.....	20
4.2.4	VLC.....	21
4.3	Comunicaciones entre software	21
4.3.1	Comunicación entre aplicación Java y Shell.....	21
4.3.2	Comunicación entre shell y FFmpeg	22
4.3.3	Comunicación entre aplicación Java y VLC.....	22
4.3.4	Load.txt.....	23
4.3.5	Save.txt.....	24
5	Sistema servidor: Visión técnica.....	25
5.1	Visión modular del sistema	25

5.1.1	GUI.....	25
5.1.2	VLC.....	26
5.1.3	Gestor de Canales.....	26
5.1.4	Sistema de importación de medios.....	28
5.1.5	Gestor de instancias de medios	28
5.1.6	Gestor de preferencias.....	29
5.1.7	Gestor de sesión.....	30
5.2	Servicio de streaming.....	31
5.3	Funcionamiento lógico.....	31
6	Manual del sistema servidor: AC TV Manager	33
6.1	Interfaz Inicial.....	33
6.2	Importación de medios	34
6.3	Información de los medios.....	35
6.4	Configuración de las instancias de medios	36
6.5	Configuración de los canales.....	37
6.6	Inicio de la emisión.....	41
6.7	Monitorización	42
6.8	Drag and Drop (D&D)	45
6.9	Intersección de medios	46
6.10	Gestión de la sesión	46
6.11	Gestión de preferencias	47
6.12	Información adicional.....	48
6.13	Limpieza de instancias de medios.....	49
7	Disposición de ficheros.....	50
7.1	autoSave.....	50
7.2	files	50
7.3	icons	51
7.4	manual.....	51
7.5	thumbnails.....	51
7.6	vlc	52
7.7	AC_TV_Manager.exe.....	52
7.8	AC_TV_Manager.jar	52
7.9	ffmpeg.exe	52
7.10	ffprobe.exe.....	52
7.11	Logo_50.ico	53
8	Sistema cliente	54
8.1	Solución cliente con VLC	54
8.2	Solución cliente con reproductor HTML5	57

9	Conclusiones.....	60
9.1	Cumplimiento del objetivo.....	60
9.2	Conclusiones sobre el proyecto	60
9.2.1	Conclusiones generales.....	60
9.2.2	Mejoras y cambios incorporados en la nueva versión.....	61
9.3	Problemas encontrados y cómo se han solucionado.....	62
9.4	Aportaciones personales.....	62
9.5	Líneas futuras	62
10	Bibliografía y referencias.....	64

1 Capítulo 1: Introducción

1.1 Introducción

IPTV significa Televisión sobre el protocolo IP [1], y es un servicio basado en la tecnología de video-streaming, que consiste en la distribución digital de vídeo a través de una red de computadoras, de manera que el usuario consume el producto en paralelo mientras se descarga.

El caso concreto del IPTV emula, mediante streaming, las emisiones broadcast de señal televisiva convencionales; de esta forma se hace llegar el contenido sobre protocolo IP a receptores (ya sean software o hardware) capaces de leer tal flujo de datos. Éstos, finalmente, se encargan de que el dispositivo asociado (televisor, monitor de ordenador, teléfono móvil, etc.) reproduzca correctamente el contenido. Por otra parte, cabe mencionar que IPTV también abarca el concepto de VoD (Video bajo demanda), un modelo de servicio que permite al usuario ver el contenido audiovisual en el momento deseado, y que está adoptando gran relevancia en los últimos años.

La aparición de cambios profundos tanto a nivel de sistemas y redes de comunicaciones como de dispositivos provocan que las redes de datos estén creciendo a un ritmo muy alto, así como sus capacidades, de modo que cada vez es más sencillo pensar en la posibilidad de migración a servicios IPTV que permitan garantizar una buena calidad de servicio. De esta forma, la evolución de la televisión tradicional al formato IPTV se está produciendo a una gran velocidad y se está haciendo patente en muchos países. Es por ello que los nuevos proveedores y suministradores de servicios de IPTV necesitan disponer de herramientas que permitan la gestión de sus contenidos, ya sean grandes corporaciones o entidades alternativas con pocos recursos y poco presupuesto.

Dependiendo de la tecnología utilizada para la difusión existen unos mecanismos para seleccionar diferentes canales dinámicamente así como administrar el ancho de banda disponible en el servicio IPTV [2], estas son algunas de las principales características de la oferta multimedia a través de multicast y que, dependiendo de la tecnología de transmisión usada (red cableada, Wifi, 3G, WiMAX, etc.) implicará una mayor o menor calidad en la QoE del usuario final [3]. Cada una de estas opciones multicast responde a un estándar y necesita la modulación adaptativa y la codificación para sus emisiones.

Uno de los problemas a los que se enfrenta la oferta multicast de televisión es el “cuello de botella” que puede originar una petición masiva del mismo, para ello es necesario multiplexar multicast [4]. Asimismo, la codificación del vídeo es fundamental ya que garantizará una mayor o menor calidad subjetiva final, menor tiempo de transcoding así como compatibilidad con navegadores, framerate de algunos países y compatibilidad con navegadores y sistemas operativos [5]. El flujo de compresión también es un punto importante en la emisión broadcast ya que puede solventar en parte problemas de ancho de banda, la combinación de CBR (Constant Bitrate) y VBR (Variable Bitrate) ayudarán a confeccionar un mapa de tramas más eficaz así como el número ideal de Intraframes dentro de la secuencia GOP (Group of Pictures), así es como funcionan actualmente diferentes sistemas de IPTV Multicast a través de proveedores ISP [6].

La implantación de servicios de IPTV va a crecer en los próximos años y la sensación actual es que la audiencia tenderá a servicios del tipo VoD (vídeo bajo demanda) aunque, hoy por hoy, el formato de emisión tradicional basado en continuidad televisiva sigue siendo patente. El

resultado es una constante demanda del uso de IPTV así como una transición de la emisión multimedia unicast a las grandes posibilidades del servicio multicast.

1.2 Objetivos

Es objeto del proyecto implementar una solución servidor-cliente de IPTV. Por un lado, se pretende desarrollar un sistema de gestión de IPTV (lado servidor) que cumpla con las funcionalidades de gestión del contenido de vídeo y audio, horarios de emisión, canales televisivos y salidas vía streaming asociadas a éstos. Se busca que el sistema servidor sea capaz de ofrecer streams unicast y multicast. Por otro lado, se desea hallar al menos una solución cliente mediante la cual se puedan visualizar los streams emitidos por la parte servidor.

También es objeto del proyecto que el lado servidor ofrezca al usuario una interfaz intuitiva con la que interactuar a la hora de gestionar los canales IPTV, por lo que se pretende desarrollar una aplicación que centralice todas aquellas herramientas que finalmente se utilicen; herramientas que serán basadas en software libre, ya que la filosofía de este proyecto es no ser ni depender de software propietario. De antemano se propone el software VLC (Video Lan Client) como herramienta de streaming, debido a sus ya conocidas características.

Finalmente, mencionar que se busca crear un sistema funcional sobre Windows, por ser el sistema operativo más utilizado.

1.3 Precedentes del proyecto

1.3.1 Motivación y punto de partida

El primer motivo de solicitar este proyecto ha sido las ganas de alcanzar una aplicación estable, funcional y definitiva a partir de la creada en un proyecto propio previo, la cual poseía serias limitaciones en sus funciones, bugs, y un modo de empleo poco intuitivo y amigable para el usuario.

1.3.2 Contexto

Es importante contextualizar bien el presente proyecto, de modo que el lector comprenda con exactitud cuál es el verdadero objetivo de éste.

1.3.2.1 Proyecto previo

Se parte de un proyecto propio anterior, como PFC (Proyecto Final de Carrera) en la UPNA (Universidad Pública de Navarra) en 2013 [7], dentro de la titulación 'Ingeniería Técnica de Telecomunicación: Especialidad Sonido e Imagen'. Su objetivo fue el desarrollo de un sistema de gestión de una parrilla de contenidos de vídeo y audio en red. La *figura 1* muestra la versión resultante del software servidor en el proyecto previo, aplicación que se pretende mejorar con este proyecto.

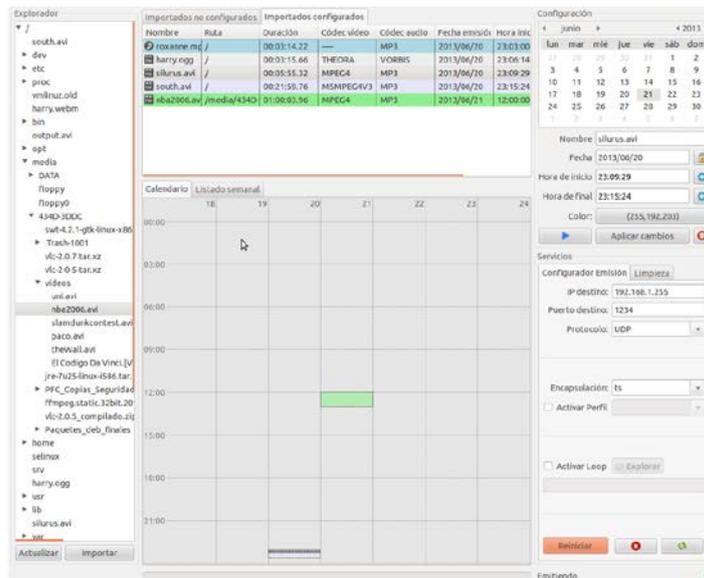


Figura 1 - Interfaz gráfica de la versión anterior del programa (para Ubuntu 12.04).

1.3.2.2 El porqué de una nueva versión

El principal motivo es la carencia de ciertas características funcionales de la aplicación servidor previa, dado que quedó incompleta y no ofrecía la consistencia suficiente para ser un programa utilizable en un caso real.

Aunque la aplicación del proyecto previo va a ser remodelada, la idea es que ésta siga manteniendo su estructura general. A continuación se detallan aquellas partes y mecanismos del sistema a mantener en la nueva versión:

- Uso de la consola del sistema para instanciación de VLC (Video Lan Client) y FFmpeg.
- Uso de VLC como servidor de streaming.
- Comunicación vía telnet entre aplicación y VLC.
- Interfaz de usuario programada en Java [8] y haciendo uso de la librería gráfica SWT.
- Uso de la interfaz VLM (Video Lan Manager) del proyecto VLC.

Puede verse la estructura del sistema en la *figura 2*.

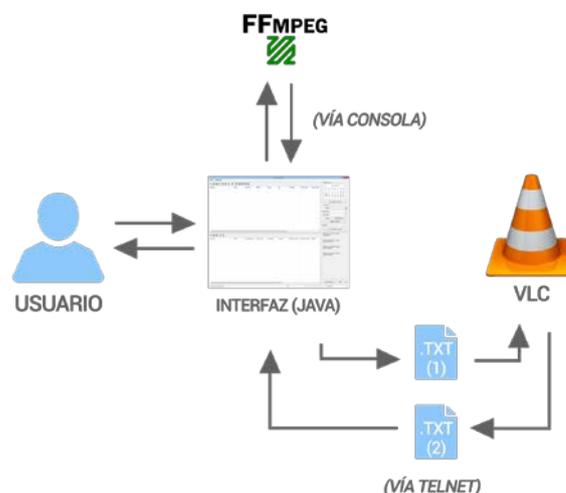


Figura 2 - Estructura general del sistema servidor con interacción de usuario.

1.4 Estructura del proyecto

Este proyecto comienza presentando el estado actual del arte a tratar en el capítulo 2, apartado que se centra en destacar qué soluciones relacionadas a la propuesta en este proyecto se pueden encontrar hoy día.

Posteriormente, los capítulos 3, 4, 5, 6 y 7 se centran describir todos los aspectos relativos al lado servidor.

El capítulo 3 (Problemas afrontados y soluciones adoptadas) presenta los pasos previos a la realización de la aplicación, tales como elección de las herramientas softwares de las que ayudarse, lenguaje de programación a utilizar, elección de interfaz gráfica, etc.

El capítulo 4 (Estructura general) documenta la estructura del sistema servidor desde el punto de vista de aplicaciones software involucradas en el sistema, sus cometidos, y los modos en que estas se comunican entre sí para lograr el objetivo deseado.

El capítulo 5 (Visión técnica) profundiza más en aspectos como funcionamiento lógico del sistema y estructura a nivel modular de la aplicación software desarrollada.

El capítulo 6 (Manual de AC TV Manager) presenta las pautas a seguir a la hora de utilizar el sistema servidor como programa de ordenador; además detalla todas las opciones de las que dispone el sistema software desarrollado, bautizado como 'AC TV Manager'.

El capítulo 7 (Disposición de ficheros) muestra la composición de ficheros de ordenador que componen el sistema, el cual se presenta como aplicación portable para Windows. Se detalla qué es cada uno y para qué sirve.

Posteriormente, en el capítulo 8, se presentan dos sencillas soluciones cliente para recibir los streams emitidos por el lado servidor; una basada en el propio VLC y otra basada en reproductores HTML5.

Finalmente, los capítulos 9 y 10 son referentes a las conclusiones y a la bibliografía, respectivamente.

2 Estado del arte

En el mercado existen múltiples sistemas privados/propietarios que son utilizados entre los distribuidores de IPTV. El problema que encontramos es que todas ellas son soluciones propietarias y de costes muy elevados.

A continuación se mencionan los softwares de gestión de IPTV más relevantes del mercado:

- *Cisco Systems* [9] ha creado una arquitectura para reducir el tiempo de envío y transportar video de manera eficiente. Cisco IPTV multicast está siendo utilizado para videoconferencias (e-learning) así como video por demanda (VoD), el cuello de botella está solucionado con la mutiplexación de hasta 50 usuarios simultáneos, a partir de este momento se necesita una mayor codificación de las tramas de audio y vídeo. La infraestructura de Cisco para este tipo de servicio requiere de servidores de streaming, backbones para multicast a 100, 300, 900 Kbps, una red ACNS (Cisco

Application and Content Networking System) así como un servicio al cliente a través de LAN, Wireless, 3G o WiMax y de VPN para acceso remoto de los trabajadores.

- La empresa *Exterity* propone *AvediaServer* [10], que es un sistema centralizado de gestión de IPTV. Proponen tres opciones de plataformas y software que pueden servir en función de su escalado dentro de cualquier ámbito empresarial. Transporta el contenido de cualquier fuente a través de una red a cualquier dispositivo sin comprometer la disponibilidad o el rendimiento del sistema IPTV.
- *Antik Technology* propone *Antik IPTV Management Server* [11], una solución Software que permite configurarlo todo, desde la gestión de canales hasta la configuración de los STB (Set-Top Boxes) de los clientes. Incluso permite conexiones a aplicaciones de facturación de clientes.
- *MWAREIPTV* presenta *IPTV Middleware System* [12], que presenta una arquitectura modular para el control total del servicio IPTV. Permite configurar y gestionar varios servicios IPTV desde una consola de administración basada en web central.
- *Snap.TV* ofrece *IPTV Content Management System* [13] su sistema de gestión de contenidos de IPTV. Permite crear y distribuir infochannels combinando el material local, basado en weby en directo o diferido, de IPTV.
- *ENCODEDMEDIA* presenta *IPTV Manager* [14], que proporciona un análisis IPTV remoto, software de monitorización y administración desde sistemas Windows. Ofrece una vista de los dispositivos Encoded Media en la red. Permite obtener información de estado, en tiempo real, sobre los canales, streams y ancho de banda utilizado, además de información de hardware que incluye el uso de CPU y las temperaturas.

AC TV Manager pretende ser una alternativa basada en software libre que, al menos, cumpla con las funcionalidades básicas de un gestor IPTV: gestión de canales y contenido y configuración de la emisión.

3 Sistema servidor: Problemas afrontados y soluciones adoptadas

La principal tarea fue encontrar el modo concreto de decirle a VLC qué contenidos debía emitir y cuándo. Se conocía la capacidad de servidor de streaming del VLC (y esta era la principal apuesta), pero no la forma de gestionarla. Se optó por el estudio de la herramienta VLM (Video Lan Manager) que integra el propio VLC, la cual permite, mediante un tipo de código específico, gestionar el horario de los contenidos a emitir (entre otras cosas). Los resultados fueron satisfactorios y por ello se ha decidido utilizar esta opción en el proyecto actual.

Las tareas que continuaron a la de la programación de contenidos a emitir fueron: conseguir hallar la duración de los vídeos y decidir qué librería gráfica de Java utilizar en la aplicación.

3.1 Programación horaria con código VLM

Puede entenderse como VLM (Video Lan Manager) una interfaz específica que propone el proyecto de código libre Video Lan Client para la gestión de emisiones de vídeo y audio. Provee

un conjunto de instrucciones que el usuario puede utilizar para dar ciertas órdenes avanzadas de comportamiento al software VLC.

A pesar de no haber mucha documentación, el lenguaje no es demasiado extenso, y uno puede familiarizarse rápidamente con él. Es suficiente la documentación que puede hallarse en la wiki del proyecto VideoLan [15].

Desde el punto de vista de la gestión de contenidos, el lenguaje cuenta con 2 elementos esenciales: los *media* y los *schedule*.

3.1.1 Media

Los *media* se componen de una entrada (por ejemplo, un fichero de vídeo en el PC) y de una salida (que define cómo y a dónde se quiere enviar el stream).

Hay dos tipos de media:

- *vod*: este tipo de elemento es usado para vídeo bajo demanda, y no toma parte dentro del alcance de este proyecto. Son medios que se envían por streaming al cliente que los solicite.
- *broadcast*: este tipo de elementos se acercan a lo que sería un canal de TV, ya que provocan que un medio empiece a emitirse cuando el administrador lo desee, y el cliente lo recibe si quiere (sin tener control sobre tal medio).

A continuación podemos ver el ejemplo de declaración de un elemento de tipo *broadcast*:

```
new medio1 broadcast enabled
setup medio1 input http://host.mydomain/movie.mpeg
setup medio1 output
#rtp{mux=ts,dst=239.255.1.1,port=5004,sdp=sap://,name="Medio 1"}
```

Pasos para la configuración, mediante código VLM, de un elemento *broadcast*:

1. Se declara un nuevo elemento mediante la etiqueta *new*, se le da el nombre deseado (en este caso *medio1*) y se define de qué tipo es (*broadcast* en este caso). Luego, se activa mediante la etiqueta *enabled*.
2. Se configura la entrada del *broadcast* declarado mediante *setup*, nombre del *broadcast* e *input*, poniendo al final la URI del recurso (puede ser un archivo local, en nuestra computadora).
3. Se configura la salida del *broadcast* declarado nuevamente mediante *setup*, nombre del *broadcast* y *output*; ahora, en vez de añadir el origen, añadimos el destino, que posee variables tales como: protocolo (*rtp* en este caso), tipo de encapsulación del stream (*ts*), IP destino (*239.255.1.1*), puerto (*5004*), etc.

3.1.2 Schedule

Los *schedule* son aquellos elementos de tipo calendario, que obligan a VLC a realizar determinada acción cuando se alcanza cierta hora en cierto día.

A continuación puede verse el ejemplo de declaración de un elemento *schedule*:

```
new sched_1 schedule enabled
setup sched_1 date 2015/05/18-12:00:00
setup sched_1 append control medio1 play
```

Pasos para la configuración, mediante código VLM, de un *schedule*:

1. Se declara un nuevo elemento mediante la etiqueta *new*, se le da el nombre deseado (*sched_1* en este caso), se pone *schedule* y se activa (con *enabled*).
2. Se configura la fecha y hora del Schedule con *setup*, nombre del *schedule* y *date*; se declara, después, la fecha y la hora a la que se quiere que el evento tenga lugar.
3. Se asigna un comportamiento para cuando se lance el evento; en este caso se ha querido que el *broadcast* antes declarado (*channel1*) comience a emitirse. Esto se consigue mediante *setup*, nombre del Schedule, *append*, *control* y el comando de control que se desea (normalmente “nombre del broadcast” + *play* para iniciar la emisión o “nombre del broadcast” + *stop* para detenerla).

3.1.3 Conclusión

Con los dos elementos analizados en el apartado anterior (*broadcast* y *schedule*) puede generarse un fichero de texto que contenga toda la programación de un canal de TV.

Por otro lado, la interfaz VLM permite la carga de ficheros con programación en tal código (*punto 2.2.2* de la memoria).

El uso de la herramienta VLM es, por tanto, una solución viable y efectiva ante el problema de cómo hacer que el contenido se emita de una forma organizada.

3.2 Conexión con la interfaz VLM

El proyecto VLC determina que hay dos modos de conectar con la interfaz VLM. Una de ellas es vía web y otra vía telnet.

Esta última fue la elección adoptada, por ofrecer sencillez de control de la herramienta VLM y por su eficacia. Puede realizarse toda la labor por comandos, mientras que vía web se requiere el uso de un navegador.

3.2.1 VLM vía telnet

Telnet es un protocolo de red que permite conectar un ordenador a otro de modo que el primero (cliente) pueda manejar al segundo (servidor).

En este caso se ha creado una clase en Java que se encarga de la conexión telnet (mediante *socket*) con la interfaz VLM. De este modo se puede utilizar la característica de un modo muy dinámico gracias a los comandos que ofrece.

3.2.2 Comandos que ofrece la interfaz telnet

Aunque el código VLM visto antes (relativo a los *broadcast* y *schedule*) puede enviarse a modo de comando por interfaz VLM (vía telnet), resulta más cómodo y óptimo hacérselo llegar mediante la carga de un fichero.

La interfaz VLM ofrece gran variedad de comandos; a continuación se destacan los que han resultado de utilidad en el proyecto:

- *load*: permite cargar un fichero con configuración VLM en el VLC.
- *save*: el comando *save* genera un archivo de configuración de toda la programación guardada en VLC. Esto permitirá monitorizar que la programación es la adecuada en todo momento.
- *del*: con *del* se puede borrar todo aquel vídeo ya programado que se desee. Para ello hay que especificar el nombre de la instancia que VLC tiene asignado a tal vídeo.
- *logout*: para una desconexión segura de *Telnet* se debe usar este comando. Si ésta no es la adecuada, al querer conectarse nuevamente puede no hacerlo.
- *show*: muestra los elementos (*broadcast* y *schedule*) cargados en el VLC y el estado de cada uno, de modo que se puede saber que vídeo se está emitiendo.

3.3 Obtención de duración de los vídeos

Esta es una característica indispensable en el proyecto, ya que debe conocerse para poder crear los elementos *schedule* anteriormente comentados de forma correcta. Se necesita saber cuándo debe pararse la emisión de un vídeo tras comenzar ésta.

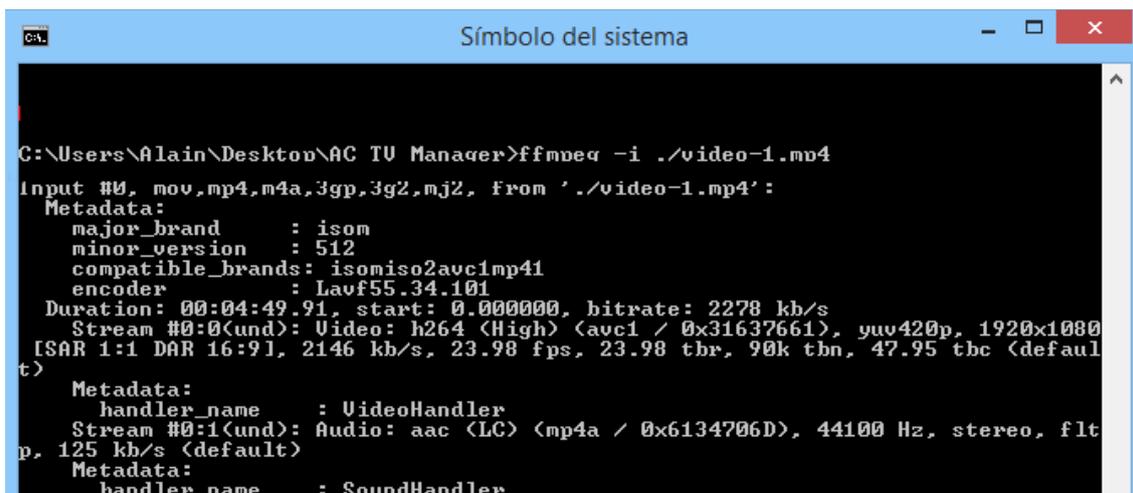
Acompañando a la anterior exigencia, la idea del programa es crear tanto un listado horario como un visor de contenidos programados (una escaleta gráfica) de forma que el usuario pueda ver qué franjas horarias están ocupadas por algún vídeo ya programado. Por estas razones debe hallarse un método eficaz de obtención de la duración de un vídeo.

Hubo intentos previos para la obtención de duración de los medios, que posteriormente fueron desechadas por dificultad:

- El primero consistió en intentar obtener la duración de un archivo de vídeo por medio del análisis de los datos de su cabecera.

- El segundo consistió en intentar obtener la duración mediante la obtención de los frames por segundo del vídeo y el nº de frames de éste, y relacionando ambos parámetros.
- El tercero fue una solución totalmente diferente, que consistió en hacer uso de un software externo llamado *FFmpeg* (se detalla en el apartado 3.2.3).

La solución adoptada ha sido el uso del software *FFmpeg*. Por un lado, es capaz de dar la duración del vídeo con resolución de centésima de segundo (resolución superior a la necesaria para el proyecto, que es de un segundo, debido a que VLC así lo limita). Por otro, posee un gran conjunto de librerías de códecs, por lo que es capaz de leer prácticamente cualquier formato.



```

C:\Users\Alain\Desktop\AC TV Manager>ffprobe -i ./video-1.mp4
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from './video-1.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2avc1mp41
    encoder          : Lavf55.34.101
  Duration: 00:04:49.91, start: 0.000000, bitrate: 2278 kb/s
  Stream #0:0(und): Video: h264 (High) (avc1 / 0x31637661), yuv420p, 1920x1080
  [SAR 1:1 DAR 16:9], 2146 kb/s, 23.98 fps, 23.98 tbr, 90k tbn, 47.95 tbc (default)
  Metadata:
    handler_name     : VideoHandler
  Stream #0:1(und): Audio: aac (LC) (mp4a / 0x6134706D), 44100 Hz, stereo, flt
  p, 125 kb/s (default)
  Metadata:
    handler_name     : SoundHandler
  
```

Figura 3 - Obtención de información de un vídeo con *FFmpeg*.

En la *figura 3* se puede observar cómo *FFmpeg* devuelve (entre otros datos) una duración de 00:03:02 segundos y 48 centésimas ante la petición de información de cierto vídeo, por línea de comandos.

3.4 Elección de la librería gráfica para la interfaz de usuario

El desarrollo de una interfaz gráfica potente compite en importancia con la manejabilidad y la fiabilidad de las operaciones internas que ésta realiza. Una buena interfaz debe ser simple, intuitiva y dinámica, a fin de que el usuario en cuestión sea capaz de expresar sus posibilidades sin necesidad de revisar el manual de usuario repetidamente.

3.4.1 Opciones barajadas en proyecto previo

En el proyecto previo se barajaron las siguientes opciones:

- Búsqueda de aplicaciones libres del tipo “Google Calendar” (*figura 4*), que se pudieran utilizar directamente en el proyecto o de las cuales se pudieran extraer propiedades que facilitarían la creación de un calendario de contenidos gráfico. Se descartó esta opción por la falta de software libre relacionado con esta idea.

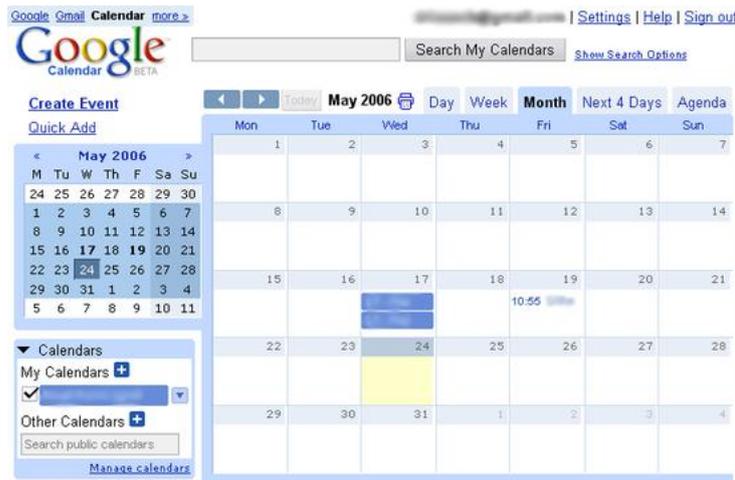


Figura 4 - Google calendar.

- Java FX (figura 5):* es una familia de productos de Sun Microsystems, en principio creada para aplicaciones web con características y capacidades de aplicaciones de escritorio, aunque está orientado a interfaces altamente animadas. En el fondo fue la respuesta de Sun Microsystems para competir con Flash de Adobe. Se descartó la opción debido a su falta de APIs optimizadas para labores de softwares de escritorio como tablas, calendarios, etc.



Figura 5 - Java FX.

- Swing y AWT (figura 6):* son las dos librerías gráficas principales de Java, prácticamente independientes al sistema operativo, y que permiten una visualización exacta independientemente de la plataforma. Se utilizan ambas, ya que se complementan para extender su funcionalidad. Ofrecen un conjunto de herramientas suficiente para crear interfaces competitivas, pero los resultados son poco vistosos, y éste fue el principal motivo por el que se desechó esta opción.

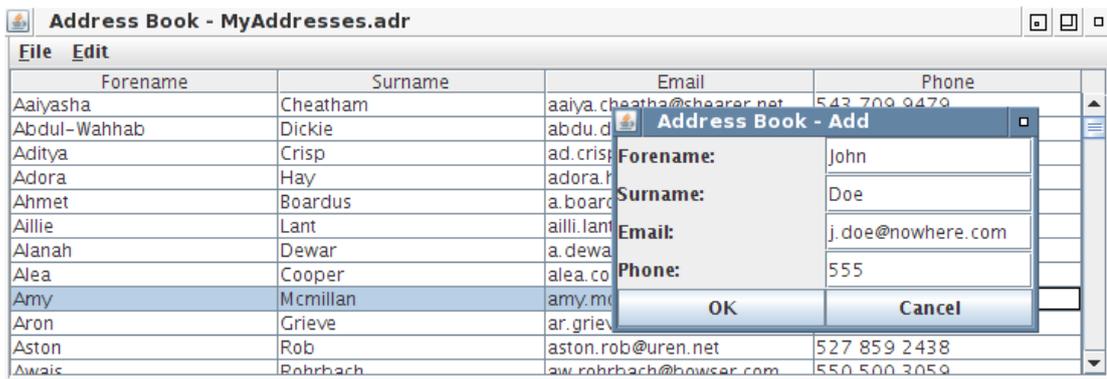


Figura 6 - Swing y AWT.

- *Interfascia (figura 7):* una librería gráfica con poca presencia en la comunidad Java y sobre la cual no existe apenas documentación; se descartó su uso por falta de ésta.



Figura 7 - Interfascia.

- *SWT (figura 8):* es una librería para construir interfaces gráficas desarrollada por el proyecto Eclipse¹. Con ella se consigue que las interfaces tengan el mismo aspecto que las ventanas y los componentes de sus sistemas operativos; además ofrece grandes posibilidades y cuenta con diversa documentación y multitud de tutoriales.

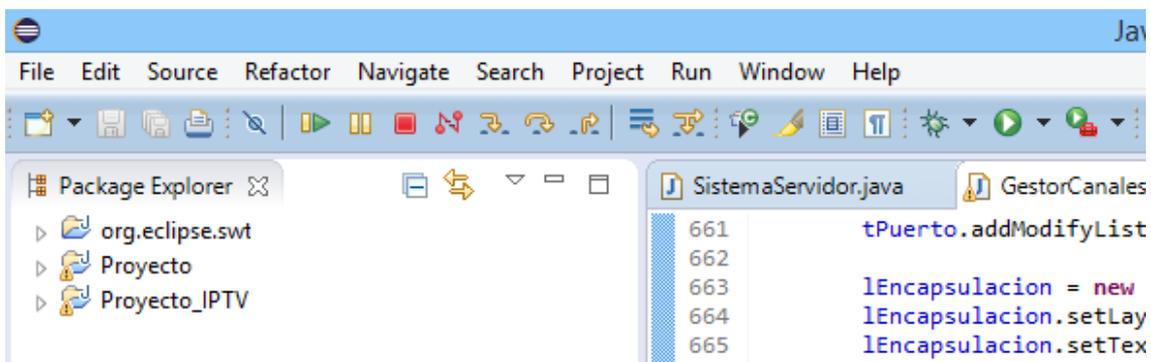


Figura 8 - SWT.

¹ Eclipse: es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados.

3.4.2 Opciones barajadas para la nueva versión

Se decidió realizar un nuevo intento con java FX, debido a que es una tecnología más reciente, y habiendo pasado dos años desde la anterior versión de la aplicación se creyó que su uso y documentación habrían aumentado notablemente.

Tras varias pruebas, análisis de snippets² y vídeos demostrativos de la tecnología se concluyó que:

- a) A pesar de ser una tecnología potente y apuesta sólida por parte de Oracle, no parece resultar una buena opción para aplicaciones de escritorio, sino más bien para aplicaciones de internet enriquecidas; es decir, para tareas que vino haciendo flash años atrás.
- b) Posee buenas herramientas de dibujo y animación que superan con creces las exigencias de dibujo de un calendario de contenidos gráfico simple. Sin embargo, no posee una librería tan completa y sencilla de widgets gráficos (botones, tablas, listas, etc.) como SWT o Swing.

Descartada la tecnología Java FX, la segunda apuesta para el proyecto fue SWT, y con ella se ha proseguido hasta su conclusión.

Aunque SWT es parecida en cuanto a prestaciones a Swing y AWT, se consiguen resultados mucho más vistosos, ya que los componentes (botones, campos de texto, etc.) cogen el aspecto por defecto del sistema operativo, y se pierde esa sensación de “programa Java” que se obtiene al utilizar Swing, que gráficamente es más plano y aburrido.

3.4.3 Elección final: librería gráfica SWT

SWT [16] es una librería gráfica para crear interfaces gráficas en Java (como Swing o Awt) que crea a través de JNI (Java Native Interface) interfaces nativas del Sistema Operativo (obsérvese *figura 9*) en donde se ejecuta la aplicación SWT en cuestión. Esto quiere decir que con el mismo código se visualiza la ventana como si hubiera sido creada para determinado SO en concreto.



Figura 9 - SWT en diferentes SO.

Posee la gran ventaja de que crea nativamente una GUI dependiente del SO, haciéndolo más rápido que Swing y permitiendo un look acorde con la plataforma software sobre la que corre.

² Fragmentos de código funcionales que generalmente se usan a modo de ejemplo o demostración de una característica.

Por otra parte, consume menos recursos que Swing y hoy día se sigue desarrollando. Las únicas desventajas son que se encuentra mucha menos documentación que para Swing y que las clases Java relativas a la librería gráfica deben ser añadidas al proyecto, sin tener un soporte nativo como ocurre con Swing.

Ejemplos de aplicaciones creadas con SWT:

- Eclipse IDE, uno de los más completos IDE (Integrated Development Environment) para desarrollo en Java.
- Azureus, uno de los más conocidos clientes torrent.

4 Sistema Servidor: Estructura general

En este capítulo se presenta la estructura general del proyecto, sin entrar a detalles demasiado técnicos.

Se explica brevemente qué software entra en juego, qué funcionalidades tiene y cuáles de ellas se han usado en la aplicación. También se especifica cómo interactúan las diferentes piezas de software entre sí, con el fin de generar una idea generalizada, pero clara y concisa a la vez, del funcionamiento del sistema de gestión.

4.1 Esquema gráfico del sistema

Se presenta en la *figura 10* la estructura básica del sistema de gestión de IPTV, al que en su conjunto llamaremos *AC TV Manager*, con el objetivo de que los apartados siguientes sean más digeribles y fáciles de entender para el lector, nada más que remitiéndose a esta imagen.

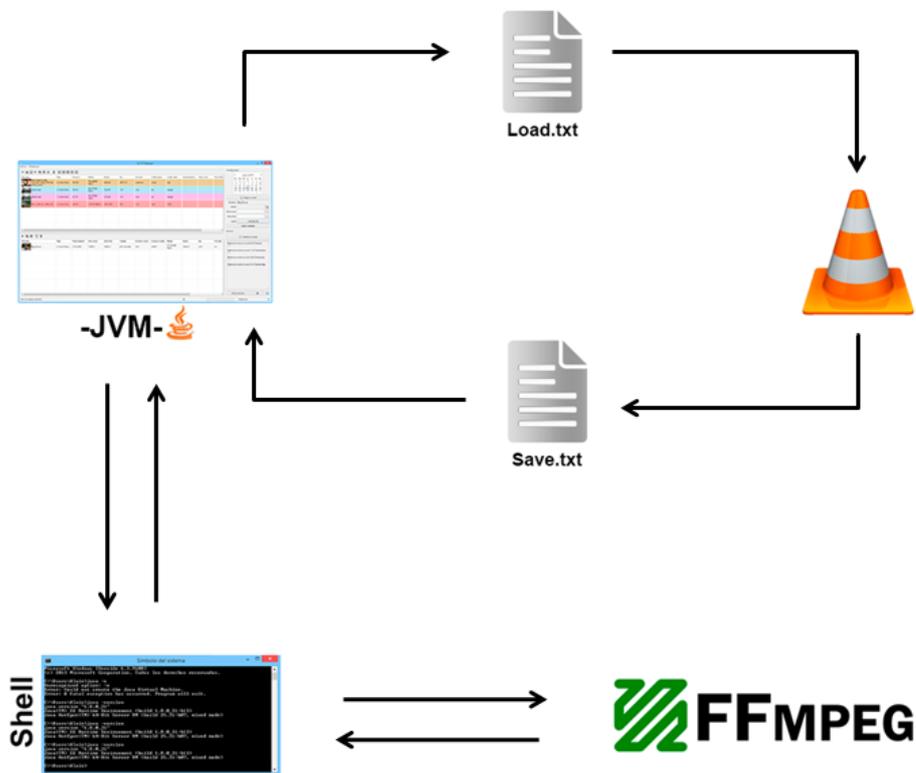


Figura 10 - Estructura general del sistema de gestión.

En la *figura 10*, que muestra la estructura del sistema de gestión, pueden observarse los siguientes elementos:

- *Aplicación Java*: corre sobre la JVM. A lo largo de la memoria se hará distinción entre "aplicación", que hará referencia al software desarrollado en Java y que funciona como interfaz de usuario y centro de control del sistema completo, y "sistema", que hará referencia al conjunto de softwares que forman el sistema completo de gestión y servidor de streaming. La aplicación Java es la encargada de generar una GUI y de procesar la información que el usuario transmite a través de ésta, así como de conectar con las diferentes piezas de software para poder crear un sistema funcional.

- *Consola/terminal/shell*: todas hacen referencia al intérprete de comandos del sistema operativo.
- FFmpeg: se detalla en el punto 4.2.3.
- VLC: se detalla en el punto 4.2.4.
- 2 ficheros de texto, de nombre Load.txt y Save.txt.

4.2 Software requerido

Es importante tener en cuenta que el sistema operativo (SO) sobre el que se ha desarrollado (y para el que está dirigido) este proyecto ha sido Windows 8, aunque también ha sido objeto de éste ser suficientemente versátil como para ejecutarse correctamente en otras versiones del SO. Se comprobó correctamente la compatibilidad en sistemas Windows XP y Windows 7.

A continuación se lista y se explica el software que compone el proyecto.

4.2.1 JDK

Es la infraestructura software y conjunto de herramientas que nos permiten crear y correr aplicaciones programadas en Java en nuestro equipo. Contiene la Java Virtual Machine (JVM), que es la máquina virtual que permite a los programas Java (Java bytecode) correr bajo cierta máquina física. Nos servirá para correr la aplicación, puesto que está programada en lenguaje Java.

4.2.2 Shell (o consola)

Llamaremos shell al intérprete de comandos nativo del SO (en este caso Windows); es la herramienta que permite comunicarse con el núcleo del sistema operativo mediante una línea de comandos.

Entre sus cometidos dentro del proyecto están:

- Instanciar VLC y FFmpeg cuando sea necesario.
- Encontrar procesos en ejecución de VLC (representados por su PID o identificador de programa).
- Matar procesos en ejecución de VLC.
- Comprobar disponibilidad de puertos TCP al arrancar VLC, con el objetivo de una correcta comunicación Telnet entre éste y la aplicación.

4.2.3 FFmpeg

FFmpeg [17] es una colección de software libre que puede grabar, convertir (transcodificar) y hacer streaming de audio y vídeo. Está orientado a su uso a través de la línea de comandos, de

modo que se hace uso de la shell para comunicarse con este software. Es utilizado, dentro del proyecto, para las tareas de importación de medios:

- Determinar si un medio es vídeo, audio o ninguno de los anteriores.
- Obtener información acerca de los vídeos (duración, formato contenedor, códec, etc.).
- Obtener un thumbnail representativo de todo medio importado.

4.2.4 VLC

VLC media player [12] es un reproductor y framework multimedia libre de código abierto desarrollado por el proyecto VideoLan.

Posee la capacidad de streaming, además de una herramienta a modo de interfaz para la gestión de ésta: VLM, de "Video Lan Manager". Se puede interactuar con ella a través de una interfaz telnet del propio VLC. Se hace uso de tal herramienta en este proyecto.

4.3 Comunicaciones entre software

Partiendo del modelo visto en el esquema general (*figura 10*), se detallan a continuación los tipos de comunicación que se llevan a cabo dentro del sistema.

4.3.1 Comunicación entre aplicación Java y Shell

Se consigue a través de servicios que nos provee la clase "Process" de Java (deriva de java.lang.Object).

Es una comunicación bidireccional (véase *figura 11*), ya que desde la interfaz gráfica (GUI) se le dice a la terminal (o shell) qué instrucción debe ejecutar, y posteriormente se lee la salida que ésta da.

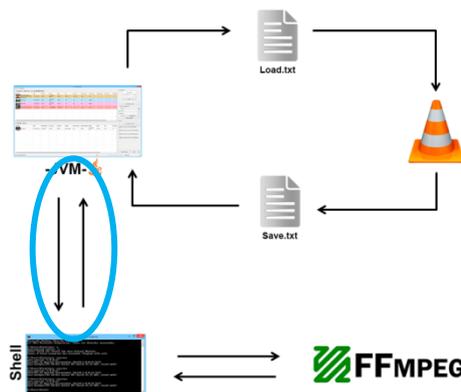


Figura 11 - Conexión aplicación Java-Shell, bidireccional.

El envío sirve para:

- Instanciar VLC.
- Llamadas a FFmpeg.

- Solicitar los puertos activos al SO.
- Matar procesos VLC.

La recepción sirve para leer la respuesta proveniente de FFmpeg y procesarla.

Ha de tenerse en cuenta que esta comunicación es un proceso transparente para el usuario, ya que la Shell corre internamente sobre el SO, sin necesidad de mostrar resultados gráficamente.

4.3.2 Comunicación entre shell y FFmpeg

La shell instancia FFmpeg (que no tiene GUI) con información añadida y recibe su salida (véase *figura 12*), por lo que también es considerada una comunicación bidireccional. Dado que el proceso de la shell es transparente para el usuario, al final aparenta ser una comunicación directa entre la aplicación Java y el FFmpeg, del que tampoco tiene por qué ser consciente el usuario.

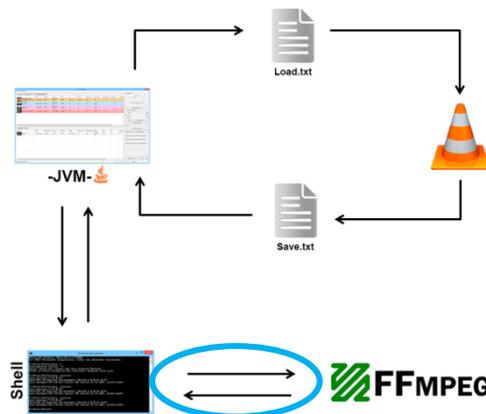


Figura 12 - Conexión Shell-FFmpeg, bidireccional.

En resumen, toda la comunicación referente al FFmpeg resulta en un proceso automático y transparente para el usuario, que simplemente ve cómo campos de información de la GUI acerca del vídeo importado son rellenados por el propio programa.

4.3.3 Comunicación entre aplicación Java y VLC

A partir de tal conexión (*figura 13*) se puede considerar una comunicación unidireccional, ya que se da órdenes a la herramienta VLM del VLC, y se confía en que ésta cumpla con su cometido.

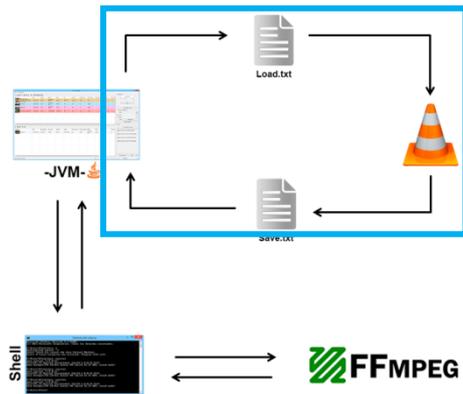


Figura 13 - Conexión aplicación Java-VLC.

Prerrequisitos:

- VLC debe haber sido instanciado con interfaz telnet activa.
- Debe haberse establecido conexión telnet.

Básicamente, los comandos que se utilizan y que interpretará la herramienta VLM son:

- load "Load.txt"
- save "Save.txt"

En los puntos 4.3.4 y 4.3.5 se detalla qué es cada uno de los archivos ".txt", y qué funciones cumplen dentro del proyecto.

4.3.4 Load.txt

Load.txt (figura 14) es el fichero de texto en el que debe aparecer toda la programación/configuración para el streaming de vídeo escrita de tal modo que la herramienta VLM del VLC pueda interpretarla.

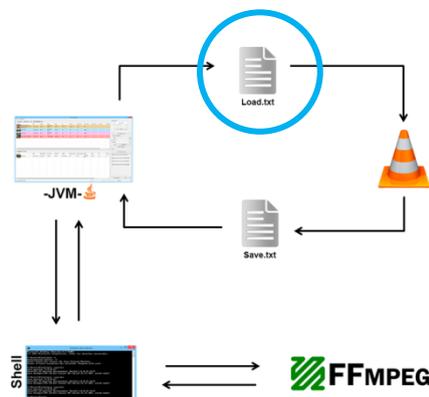


Figura 14 - Load.txt

En este fichero tan sólo hay información del tipo visto en el apartado de la memoria relativo al VLM.

El objetivo de la aplicación es crear este fichero, valiéndose de los medios importados y los horarios dados por parte del usuario (además de otras opciones como multiplexación, dirección IP destino, transcodificación, etc).

Se carga vía telnet en el VLC, y este último adquiere el comportamiento que se le haya especificado en el fichero.

4.3.5 Save.txt

Save.txt (figura 15) es el fichero que podemos pedir a la herramienta VLM que genere; muestra la configuración de todo el streaming programado en ese instante.

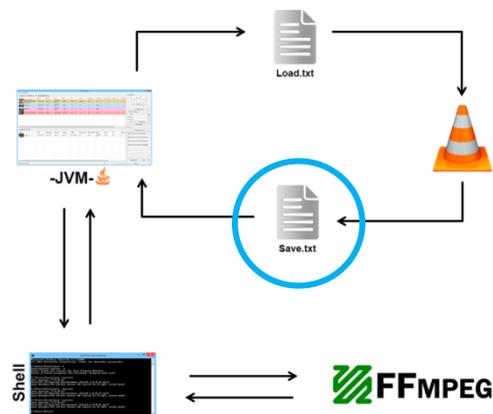


Figura 15 - Save.txt

En teoría debiera tener la misma configuración que Load.txt, y sirve para asegurar que la configuración se ha cargado correctamente.

Sirve, una vez procesado por la aplicación, para refrescar la parte de la GUI que muestra los contenidos programados gráficamente, así como para el refresco del listado de vídeos por fecha y hora. También sirve de base para el fichero de guardado de la configuración del proyecto.

5 Sistema servidor: Visión técnica

AC TV Manager es un sistema gestor de contenidos de IPTV cuyo núcleo es un programa desarrollado en Java que funciona como interfaz entre el usuario y VLC (Video Lan Client) con el objetivo de facilitar la configuración de la emisión de medios en streaming a través de datagramas IP. En resumidas cuentas, el programa permite configurar los horarios y los destinos a los que los medios deben emitirse, de modo que resulte sencillo gestionar un canal de IPTV.

El programa da soporte a los siguientes protocolos: UDP, RTP, RTSP y HTTP, permitiendo además la creación de hasta N canales funcionando simultáneamente. También incluye la opción de una transcodificación en vivo independiente para cada canal.

El programa funciona como gestor y servidor Broadcast (a modo de la TV convencional) en red. Por otra parte, el software también se ayuda de la funcionalidad del proyecto FFmpeg para labores de importación de medios (obtención de información e imágenes).

5.1 Visión modular del sistema

En el apartado anterior se ha visto el programa desde el punto de vista de su estructura general y comunicación entre softwares. A continuación se muestra un punto de vista más específico, haciendo hincapié en la visión modular del sistema (*figura 16*) y en las labores de la aplicación programada en Java.

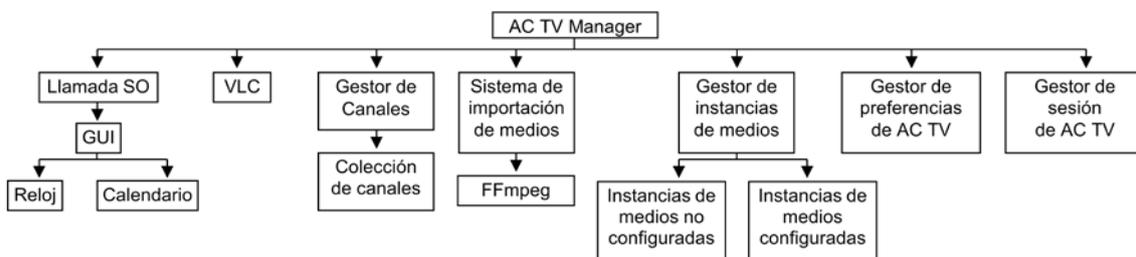


Figura 16 - Estructura del sistema desde el punto de vista modular.

5.1.1 GUI

Es la interfaz gráfica de la aplicación, mediante la cual interactuará el usuario. Está compuesta por widgets de la librería gráfica SWT (una librería open-source para Java creada por el proyecto *Eclipse*), los cuales son llamadas directas a widgets gráficos intrínsecos del sistema operativo (*Windows* en este caso).

Entre los elementos gráficos pueden resaltarse:

- El reloj (*figura 17*), que mostrará la hora del sistema en el que se esté ejecutando la aplicación) y el calendario. También resultará una herramienta base para la gestión horaria de las emisiones, ya que ayuda a detectar qué instancias de medios ya han sido emitidas y permite, por tanto, la limpieza automática.

Hora de sistema: 11:16:43

Figura 17 - Hora del sistema.

- El calendario (*figura 18*) es otra pieza fundamental, permitiendo al usuario la selección del día en que se emitirá el medio a configurar. Además, permite una rápida navegación para la visualización (en el panel de escaleta gráfica) de la programación creada.

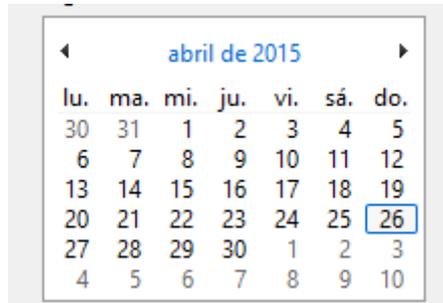


Figura 18 - Calendario gráfico.

5.1.2 VLC

AC TV Manager delega las funciones de streaming al software VLC. Se comunica con él vía Telnet tal como muestra la *figura 19* para cargarle la nueva configuración horaria y de emisión cada vez que el usuario realiza un cambio en el programa.

Desde el panel de preferencias puede asignarse la ruta del ejecutable 'vlc.exe' deseado, por lo que el sistema podría utilizar nuevas versiones del software VLC a medida que estas vayan apareciendo.

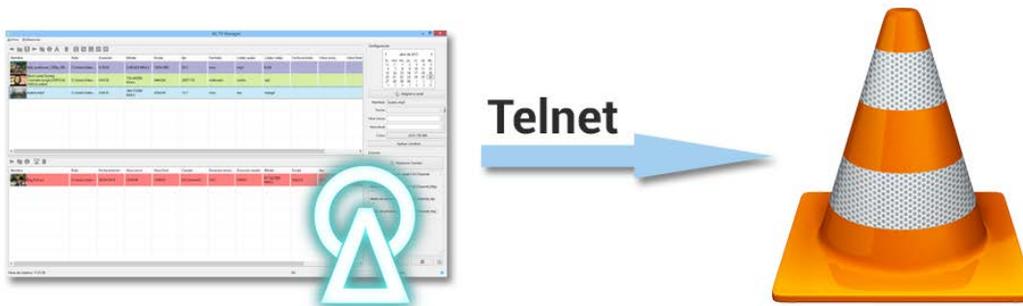


Figura 19 - Comunicación entre aplicación y VLC.

5.1.3 Gestor de Canales

El gestor de canales mostrado en la *figura 20* es el conjunto de herramientas que nos permitirán añadir vías de salida de stream. El programa permite la creación de hasta N canales mediante el panel de creación de canales de la *figura 21*; es decir, tantos canales como desee el usuario.

A la hora de configurar las instancias de medios, será indispensable asignarles al menos un canal de salida. El canal se compone de:

- Número identificador.
- Nombre.
- Protocolo (UDP, HTTP, RTP o RTSP).
- IP destino (sólo afectará sobre los protocolos UDP y RTP; HTTP y RTSP actúan como servidor de stream bajo demanda, por lo que la IP será la del propio ordenador donde esté corriendo la aplicación).
- El modelo de encapsulación (o muxer) del stream. Los modos disponibles son 'ts' (transport stream), 'ogg', 'webm', 'asf' y 'mkv'. No todos los muxers funcionan con todos los protocolos, de hecho el único muxer válido para todos los protocolos es el 'ts' (transport stream), mientras que los demás se limitan a funcionar únicamente bajo protocolo HTTP.
- Es un campo optativo. La extensión es el conjunto de caracteres que acompañarán a la dirección por defecto del stream. Imaginemos un canal UDP con IP destino 239.0.0.1 y puerto 7050; su dirección sería *udp://239.0.0.1:7050*. Si le añadiéramos la extensión 'stream-pruebas' su dirección de acceso quedaría de la siguiente forma: *udp://239.0.0.1:7050/stream-pruebas*.

Nº	Nombre Canal	Protocolo	IP	Puerto	Encapsulación	Extensión	Canal activado	Transcoding
0	ACChannel	UDP	localhost	7000	ts		Sí	Desactivado
1	ACChannel_ht...	HTTP	localhost	7001	ts		Sí	Desactivado
2	ACChannel_rtp	RTP	localhost	7002	ts		Sí	Desactivado
3	ACChannel_rtsp	RTSP	localhost	7003	ts		Sí	Desactivado

Figura 20 - Panel de gestión de canales.

Nº:
 Nombre:
 Protocolo:
 IP:
 *El campo IP no tendrá efecto sobre los protocolos HTTP y RTSP
 Puerto:
 Encapsulación:
 Extensión:

Figura 21 - Panel de creación de nuevo canal.

5.1.4 Sistema de importación de medios

Es el conjunto de herramientas que el programa ofrece para la importación de los medios. Puede hacerse mediante el botón destinado a ello, el cual abrirá un explorador de Windows para la selección de los archivos a importar, o mediante Drag&Drop desde cualquier ventana de exploración del sistema operativo.

Cada vez que se importan archivos, estos pasan por un 'filtrado' para determinar si son medios de audio o vídeo válidos y obtener, de paso, sus propiedades relevantes (duración, bitrate, resolución, fps, formato, códec de audio y códec de vídeo). El filtrado se crea a través de la información devuelta por una llamada al software *ffprobe* (parte del proyecto *FFmpeg*).

Finalmente, el software *FFmpeg* se encarga de obtener un 'thumbnail' representativo del vídeo, antes de crearse la entrada de tabla representada en la *figura 22*.

Nombre	Ruta	Duración	Bitrate	Escala	fps	Formato	Códec audio	Códec vídeo
 Big_Fish.avi	C:\Users\Alain...	2:00:01	817.627000 Kbit/s	520x272	25/1	avi	mp3	mpeg4

Figura 22- Instancia de medio (representada por entrada de tabla); se muestran parámetros importantes acerca del medio al que referencia: duración, códecs, etc.

5.1.5 Gestor de instancias de medios

Las instancias de medios son las referencias a los medios almacenados en el equipo, discos duros, red, etc. Las instancias vienen representadas por una entrada de tabla, y son los elementos que deben ser configurados (mediante asignación de fecha y hora) para una correcta emisión del medio por alguno (o varios) de los canales.

El gestor de instancias se encarga de mover éstas de la tabla de no-configurados a la tabla de configurados (ejemplo mostrado en *figura 23*) si han sido preparadas para su emisión. Las instancias están configuradas cuando:

- a) Tienen una hora de inicio asignada.
- b) Tienen una hora de final asignada.
- c) Están asignadas a uno (o varios) canales.

Nombre	Ruta	Duración	Bitrate	Escala	fps	Formato	Códec audio	Códec vídeo	Fecha emisión	Hora inicio	Hora final
bb6_sunflower_1080p_30f...	C:\Users\Alain...	0:10:34	3.481424 Mbit/s	1920x1080	30/1	mov	mp3	h264			
Black Label Society Concrete Jungle [OFFICIAL VIDEO].webm	C:\Users\Alain...	0:03:38	753.42000 Kbit/s	448x336	2997/125	matroska	vorbis	vp8			
buena.mp4	C:\Users\Alain...	0:04:52	368.372000 Kbit/s	320x240	15/1	mov	aac	mpeg4			

Nombre	Ruta	Fecha emisión	Hora inicio	Hora final	Canales	Duración emisi...	Duración medio	Bitrate	Escala	fps	Formato
Big_Fish.avi	C:\Users\Alain...	26/04/2015	12:00:00	14:00:01	[ACChannel]	2:01	2:00:01	817.627000 Kbit/s	320x272	25/1	avi

Figura 23 Tablas de la interfaz gráfica. En la tabla superior las instancias no configuradas (3 en este ejemplo) y en la tabla inferior las ya configuradas (1 en este ejemplo).

5.1.6 Gestor de preferencias

El gestor de preferencias mostrado en la *figura 24* se encarga de aplicar cambios globales en ciertos comportamientos de la aplicación. Las preferencias no se almacenan en una sesión (es decir, no son almacenadas en el fichero creado al guardar una sesión), sino que son intrínsecas a la aplicación en sí, y se mantienen al cerrar y volver a abrir el programa.

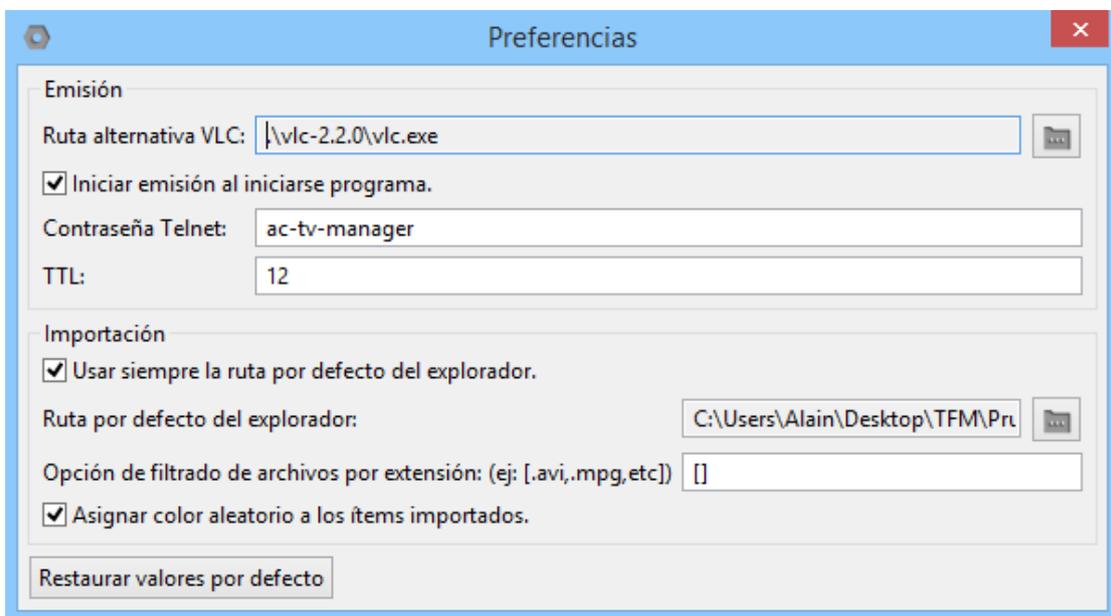


Figura 24 - Panel de preferencias de AC TV Manager.

Las preferencias configurables son:

- Ruta alternativa VLC. Por defecto, el programa busca el software VLC que tenga instalado en su PC. Si no lo tiene instalado en un directorio reconocible o simplemente desea que el programa instancie el VLC que usted desea, explore en su sistema mediante el botón de la derecha. La imagen muestra un ejemplo válido de ruta de VLC. Asegúrese de que el ejecutable posee el nombre *vlc.exe*.

- El campo Iniciar emisión al iniciar programa obliga a la aplicación a iniciar directamente la instancia de VLC servidor nada más arrancarla.
- El campo de Contraseña telnet permite asignar la contraseña Telnet que el programa utilizará para comunicarse con la instancia VLC. Tal comunicación es un proceso transparente para el usuario, no se debe realizar ningún paso más.
- Opción de ajuste de TTL. Es el número de routers que cada datagrama IP del stream puede atravesar sin ser eliminado. Por defecto, su valor es 12.
- Usar siempre la ruta por defecto del explorador provoca que cada vez que se abra una ventana de exploración de Windows para importar medios se vea el contenido de la ruta configurada en el siguiente campo.
- Ruta por defecto del explorador permite seleccionar la ruta por defecto a utilizar en la ventana de importación de medios.
- La opción de Filtrado permite filtrar automáticamente ficheros por extensión en la ventana de importación de medios.
- La opción de Asignación de color aleatorio a ítems importados ayuda a la diferenciación de éstos en todo el flujo de trabajo del programa. Por defecto viene activada. Desactivado, provoca que los medios tengan fondo blanco (color de tabla).
- Puede volver a los valores predeterminados mediante el botón Restaurar valores por defecto.

5.1.7 Gestor de sesión

El gestor de sesión se encarga de transcribir toda la información de la sesión a un fichero. Por una parte, el usuario puede establecer una ruta de guardado de tal fichero, aunque el programa provee un sistema de auto-guardado para evitar pérdida de información ante un cierre inesperado.

Se llama sesión al estado en que se encuentra el programa en determinado momento, e incluye:

- Instancias de medios no configuradas.
- Instancias de medios configuradas.
- Canales (y parámetros de transcoding de cada uno).

La ruta de autoguardado es aquella en la que se halle la aplicación, dentro de la carpeta 'autoSave'. Si el programa se encuentra en la siguiente ruta: *C:\Users\Alain\Desktop*, el fichero de auto-guardado se encontrará en: *C:\Users\Alain\Desktop\AC_TV_Manager\autoSave*.

El gestor de sesión también es el encargado de recrear la sesión ante la apertura de un fichero previamente guardado.

5.2 Servicio de streaming

Como se ha mencionado en un apartado anterior (4.1.3) *AC TV Manager* permite configurar canales de salida; estos permiten funcionar bajo uno de los 4 protocolos siguientes: UDP, RTP, HTTP y RTSP.

VLC no se comporta de la misma manera con los diferentes protocolos; la *figura 25* presenta la diferencia.

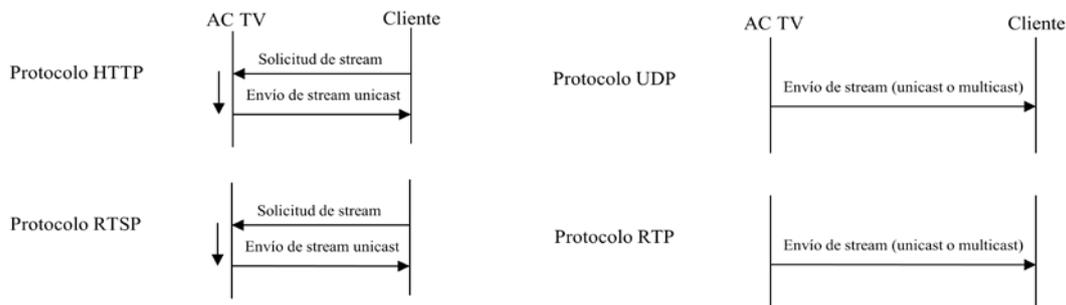


Figura 25 - Comportamiento de AC TV Manager como servidor de streaming en función del protocolo utilizado.

HTTP y RTSP provocan que VLC actúe como servidor de stream sólo ante la petición de un cliente. Cuando un cliente solicita el stream, VLC le envía una réplica del stream que internamente está procesando. Hay que recordar que este es un sistema broadcast, por lo que el usuario empezará a recibir la emisión, tal como si acabara de elegir en su televisor un determinado canal. Lo que se envía a cada cliente es, por tanto, un stream unicast independiente (aunque todos ellos llevarán la misma información de vídeo, ya que son copias).

UDP y RTP son utilizados para el envío directo de stream a una dirección unicast o multicast; no existe para ello ninguna solicitud previa por parte de ningún cliente.

Estos comportamientos frente al protocolo son debidos al funcionamiento interno de VLC. Otros sistemas o softwares podrán tener un modo de funcionamiento totalmente diferente.

Los streams pueden verse afectados por varias configuraciones (multiplexación, transcoding, etc.). Se detallan las capacidades sobre éstos en el manual del sistema (punto 5 de la memoria).

5.3 Funcionamiento lógico

El usuario accede al sistema completo (*AC TV Manager*) a través de la interfaz de usuario, es decir, la GUI de la aplicación Java. A partir de aquí el usuario puede gestionar todos los aspectos y módulos vistos anteriormente.

Las acciones lógicas a realizar por parte del usuario para acabar obteniendo una programación televisiva vienen dadas en la *figura 26*.

Figura 26 - Diagrama de funcionamiento lógico de AC TV Manager.

6 Manual del sistema servidor: AC TV Manager

6.1 Interfaz Inicial

Se presenta el aspecto de la interfaz de usuario inicial en la *figura 27*.

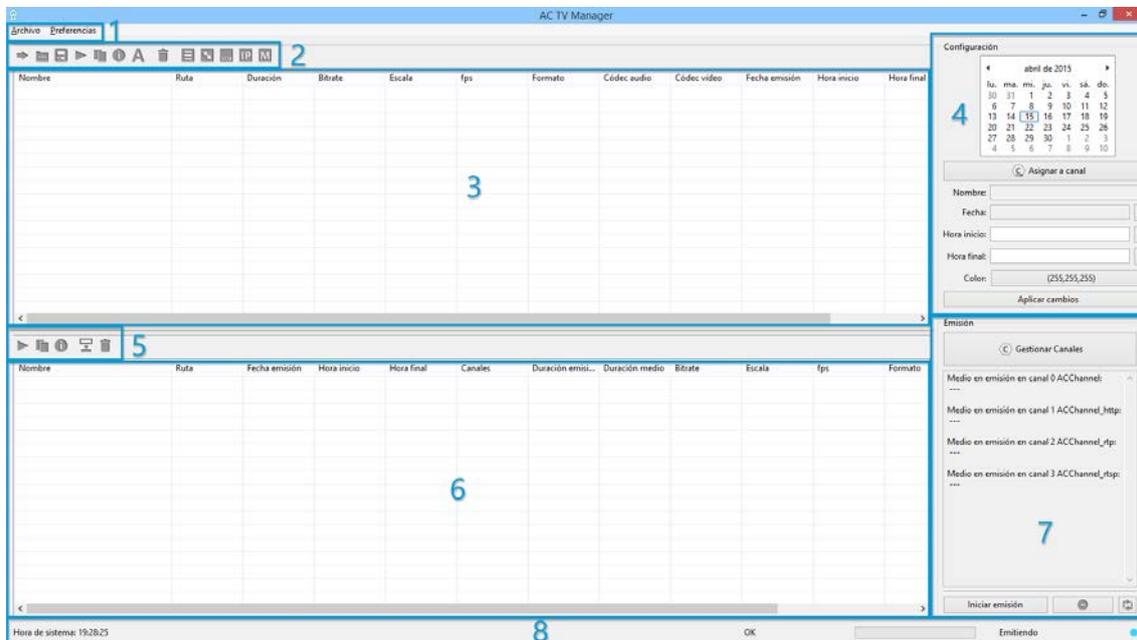


Figura 27 - Interfaz inicial del programa.

1. **Menú superior.** Ofrece opciones generales tales como guardado y apertura de la sesión, o la gestión de preferencias de la aplicación.
2. **Barra de Herramientas 1.** Barra de herramientas relativa a la tabla 1, con utilidades que afectan a las instancias de medios que se encuentran en ésta.
3. **Tabla 1 (tabla de instancias no configuradas).** Es la tabla que almacena las instancias de medios que aún no han sido configuradas para su emisión.
4. **Panel de configuración de instancias.** Panel con opciones de configuración de la instancia de medio seleccionada (fecha de inicio, hora, etc.).
5. **Barra de Herramientas 2.** Barra de herramientas relativa a la tabla 2, con utilidades que afectan a las instancias de medios que se encuentran en ésta.
6. **Tabla 2 (tabla de instancias configuradas).** Es la tabla que almacena las instancias de medios que ya han sido configuradas para su emisión.
7. **Panel de emisión.** Es el panel que permite la adición o borrado de canales, así como su configuración. Además, posee los botones de inicio y parada generales, que provocarán que se instancie VLC corriendo en segundo plano (realizando las labores de servidor).
8. **Reloj, etiquetas y barra de progreso.** Reloj y conjunto de etiquetas gráficas que sirven para monitorización de procesos del programa (tales como la inicialización de la emisión o la importación de archivos).

La interfaz es escalable, aunque se recomienda utilizarla en un monitor con resolución mayor a 960x640 píxeles. La interfaz es de márgenes fijos con excepción de las tablas, las cuales son escalables y están divididas por un separador que el usuario puede ajustar mediante arrastre.

6.2 Importación de medios

El primer paso dentro del programa es la importación de medios; existen dos formas de hacerlo:

- Botón **Importación** ➔ (barra de herramientas 1).
- Mediante Drag&Drop desde el escritorio o cualquier ventana de exploración de Windows a la tabla 1.

El primer método provoca la apertura de una ventana de exploración de Windows, la cual permitirá selección de ficheros múltiple.

El usuario no debe preocuparse de seleccionar únicamente archivos de vídeo o audio, ya que el programa filtrará cuáles son válidos y los incorporará a la sesión. El programa es capaz de importar todo aquello que las librerías del proyecto FFmpeg son capaces de interpretar como vídeo y/o audio.

Una vez seleccionados y abiertos, se iniciará el proceso de importación, monitoreado por la barra de progreso de la parte inferior de la interfaz.

Tras la importación, se crean **instancias de medios**, una por cada vídeo o audio importado. Estas instancias son referencias al fichero de vídeo o audio original, y se presentan en la tabla 1 tal como muestra la *figura 28*.

Nombre	Ruta	Duración	Bitrate	Escala	fps	Formato	Códec audio	Códec vídeo	Fecha emisión	Hora inicio	Hora final
 bbb_sunflower_1080p_30f...	C:\Users\Alain...	0:10:34	3.481424 Mbit/s	1920x1080	30/1	mov	mp3	h264			
 bueno.mp4	C:\Users\Alain...	0:04:52	369.372000 Kbit/s	320x240	15/1	mov	aac	mpeg4			
 Black Label Society Concrete Jungle [OFFICIAL VIDEO].webm	C:\Users\Alain...	0:03:38	753.442000 Kbit/s	448x336	2997/125	matroska	vorbis	vp8			
 Big_Fish.avi	C:\Users\Alain...	2:00:01	817.627000 Kbit/s	520x272	25/1	avi	mp3	mpeg4			

Figura 28 - Tabla 1 (instancias de medios no configuradas) tras la importación de 4 medios.

Otra forma de incorporar nuevas instancias de medios al programa es crear un duplicado de alguna de las instancias existentes mediante el botón **duplicación** (barra de herramientas 1 o 2, en función de la tabla). Esta función crea una nueva entrada en la tabla 1, duplica de aquella seleccionada, referenciando al mismo medio. Esta instancia es completamente independiente a la original dentro del programa.

Si un medio no posee información de vídeo, aparecerá el thumbnail de la *figura 29* en su respectiva instancia.



Figura 29 - Thumbnail por defecto para los medios sin información de vídeo reconocible.

6.3 Información de los medios

Las tablas presentan información relevante acerca del medio, así como sus propiedades de emisión. Los parámetros que se muestran repartidos en columnas son:

1. **Nombre** (acompañado de un thumbnail extraído del medio). Es el nombre del fichero; éste puede ser modificado dentro del programa mediante el botón **Alias A** (barra de herramientas 1), sin afectar al nombre original del fichero.
2. **Ruta**. Representa la dirección donde se encuentra el fichero dentro del sistema operativo.
3. **Duración del medio**. Dada en formato h:mm:ss.
4. **Bitrate**. Tasa de bits del medio completo (vídeo+audio).
5. **Escala**. Representa la resolución, en px, del medio.
6. **fps**. Representa los fotogramas por segundo del medio.
7. **Formato**. Hace referencia al formato contenedor (p. ej. AVI).
8. **Codec audio**. Especifica el códec de audio con el que se comprimió el sonido del medio.
9. **Codec vídeo**. Especifica el códec de vídeo con el que se comprimió la información visual del medio.
10. **Fecha emisión**. Fecha en formato dd/mm/aaaa en la que se va a emitir tal medio.
11. **Hora inicio**. Hora a la que comenzará a emitirse el medio.
12. **Hora final**. Hora en la que finalizará la emisión del medio.
13. **Canales**. Canales por los que el medio será emitido.

La tabla 2 presenta una 14ª columna con información de **Duración de la emisión**; esta información se obtiene de la diferencia de los parámetros Hora final y Hora inicio.

Los instancias de medios de las tablas pueden ser ordenadas en función del parámetro de cada columna si se pulsa sobre ésta.

Para obtener información extendida (dada por el software FFprobe del proyecto FFmpeg) de un determinado medio, basta con clicar sobre su instancia en la tabla y pulsar el botón **Información**  (barra de herramientas 1 o 2, en función de la tabla). Se abrirá un panel como el de la *figura 30*.

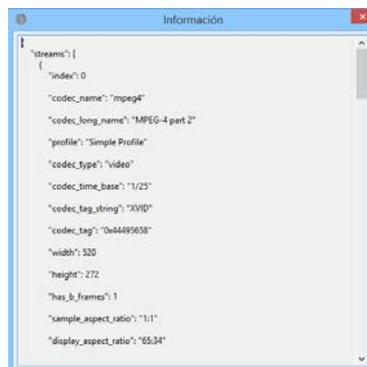
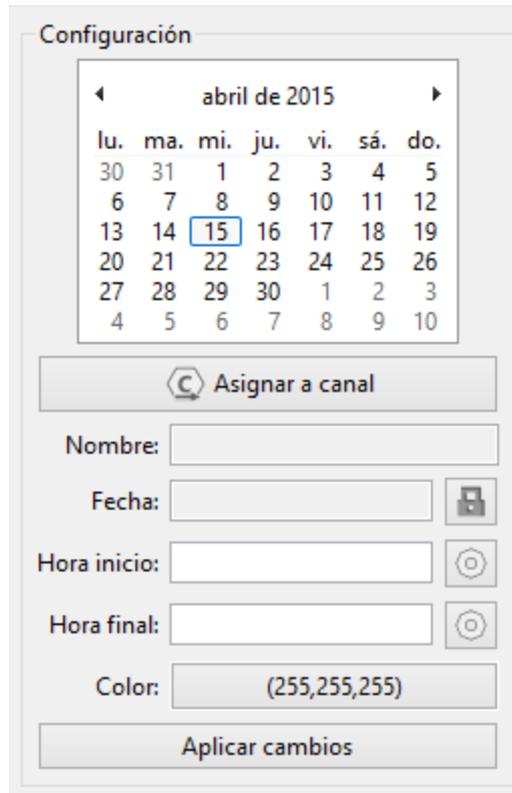


Figura 30 - Ventana de información de medios.

También puede realizarse un preview del medio mediante el botón **Previsualización**  (barra de herramientas 1 o 2, en función de la tabla).

6.4 Configuración de las instancias de medios

Clique en una de las entradas de la tabla 1 para proceder a la configuración de la correspondiente instancia de medio. Verá que el campo de **Nombre** del panel de configuración (figura 31) se actualiza sólo.



Configuración

← abril de 2015 →

lu.	ma.	mi.	ju.	vi.	sá.	do.
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

Asignar a canal

Nombre:

Fecha: 

Hora inicio: 

Hora final: 

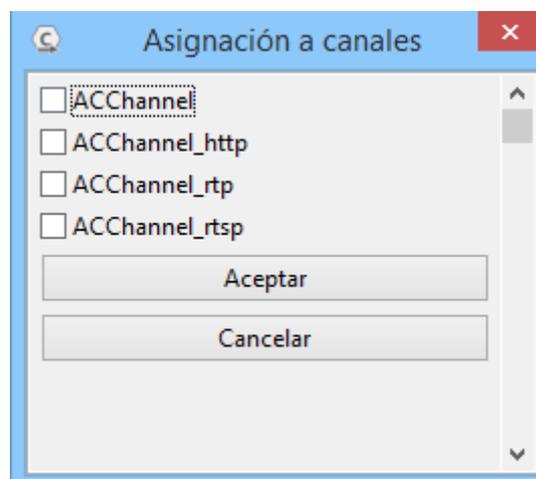
Color:

Aplicar cambios

Figura 31 - Panel de configuración de instancias.

A continuación siga los siguientes pasos:

1. Presione el botón **Asignar a Canal** para seleccionar por qué canal(es) se emitirá el medio. Se abrirá un panel (figura 32) con el listado de canales creados. Seleccione los deseados y presione **Aceptar** para completar la acción.



Asignación a canales

- ACChannel
- ACChannel_http
- ACChannel_rtp
- ACChannel_rtsp

Aceptar

Cancelar

Figura 32 - Ventana de asignación a canal.

2. Seleccione el día en que quiera emitir el medio clicando sobre el calendario gráfico.

3. Escriba la hora de inicio de la emisión con el formato hh:mm:ss. Si la hora no está correctamente escrita o no es válida, este campo aparecerá en rojo (figura 33). En cambio, si es una hora válida, los caracteres se verán en negro (figura 34).

Una interfaz de usuario que muestra un campo de texto etiquetado como 'Hora inicio:' con el valor '12:0:00' escrito en rojo. A la derecha del campo hay un botón con un icono de configuración.

Figura 33 - Hora no válida o con formato erróneo.

Una interfaz de usuario que muestra un campo de texto etiquetado como 'Hora inicio:' con el valor '12:00:00' escrito en negro. A la derecha del campo hay un botón con un icono de configuración.

Figura 34 - Hora válida.

4. Siga el mismo criterio para asignar una hora de final. Si desea autocompletar la hora de final en función de la duración del medio, pulse sobre el botón **autocompletar** . Esta función es recíproca para la hora de inicio.
5. Pulse el botón de **color** si desea cambiar el color de fondo de la entrada de tabla asociada a la instancia de medio. El color también afectará al panel de escaleta gráfica (hará que el recuadro dibujado se rellene con el mismo color).
6. Finalmente, presione el botón **Aplicar cambios**. Si todos los campos son correctos, la instancia de medio habrá quedado configurada y pasará a la tabla 2.

El botón **candado**  permite bloquear el campo de fecha para cada instancia de medio. Esto es útil para que el parámetro no se vea alterado cuando navegamos por el calendario para visualizar diferentes días en la **escaleta gráfica** que se verá más adelante.

6.5 Configuración de los canales

El segundo punto clave, junto con la configuración de instancias de medios previamente vista, es la creación y configuración de canales de emisión. Para acceder al panel de gestión de

canales presione el correspondiente botón del panel Emisión

Un botón rectangular con un icono de configuración a la izquierda y el texto 'Gestionar Canales' a la derecha.

Se abrirá la ventana de la *figura 35*.

letras, dígitos, espacios, guiones (-) y barras bajas (_). A continuación un ejemplo de dirección de stream multicast con y sin extensión:

▲ <udp://@:239.0.0.1:7000/extension-1>

▲ <udp://@:239.0.0.1:7000/>

- Finalmente, pulse sobre el botón **Crear** para añadir el canal a la tabla de la ventana de gestión de canales. El canal sólo se creará si todos los campos son correctos. Los campos incorrectos aparecerán en rojo.
 - Los canales no pueden ser modificados posteriormente. Si desea un canal con un nombre, destino, multiplexación, etc. diferentes, deberá crear uno nuevo.
-
- Presione el botón **Copia de canal**  para crear un nuevo canal con la misma planificación de emisión que aquél seleccionado en la tabla. Técnicamente, todas las instancias de medios asignadas al canal seleccionado en la tabla se asignarán también al nuevo canal.
 - Clique sobre el botón **Previsualización de canal**  para abrir una instancia de VLC (independiente al programa) que abra la dirección de stream del canal seleccionado en la tabla. La *figura 37* muestra la ventana relativa a la instancia de VLC, solicitando el stream <udp://@:localhost:7000/>.

Figura 37 - Instancia cliente VLC (independiente al programa) recibiendo un stream emitido por AC TV Manager.

- El botón **Activación/desactivación**  permite conmutar el estado de activación del canal. Si está activado, el canal estará en funcionamiento; en caso contrario, no emitirá contenido. Por defecto los canales se crean en modo activado.
- El botón **Transcoding**  permite abrir la ventana de configuración de codificación en vivo (*figura 38*) para el canal seleccionado. Se mostrará lo siguiente:

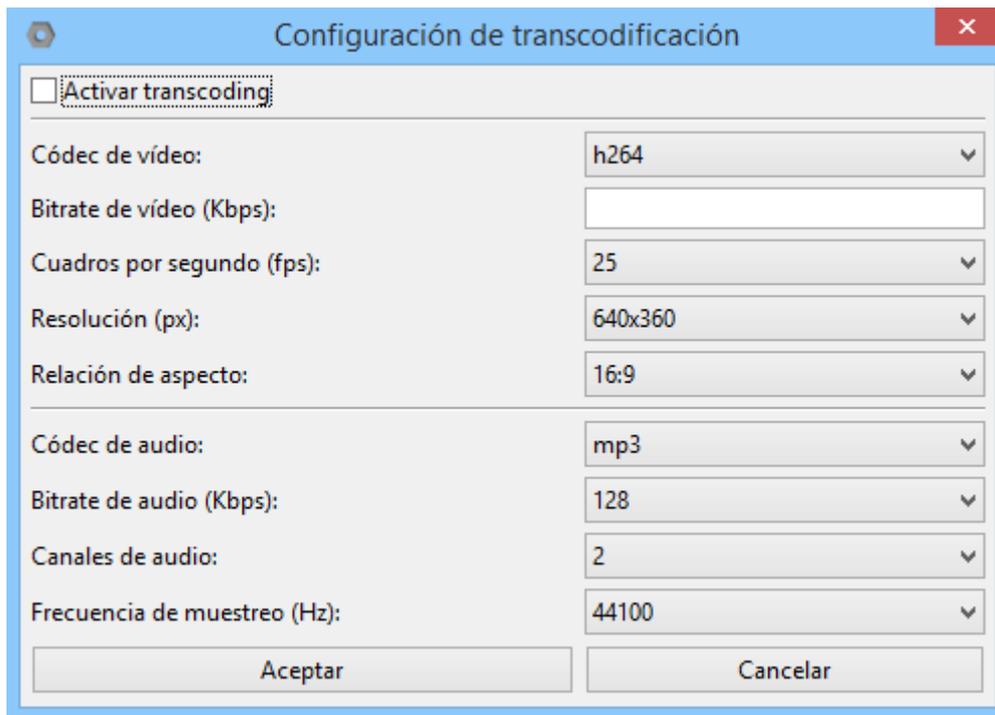


Figura 38 - Ventana de configuración de transcoding para un determinado canal.

- Utilice la primera casilla para activar la codificación en vivo para el canal seleccionado. Tenga en cuenta que el transcoding es opcional. Su uso es recomendable tanto si los medios de origen están en diferentes formatos como si el cliente destino requiere unos parámetros específicos. El transcoding hace uso de gran cantidad de CPU y aumenta notablemente el uso de RAM por parte de VLC.
- Seleccione el valor deseado para cada parámetro disponible. Tenga en cuenta que no todos los codecs funcionan correctamente con todas las combinaciones de protocolos y modelos de multiplexación que ofrece el programa. De forma orientativa, consulte la *tabla 1* para ver las combinaciones que VLC propone (con datos obtenidos de <http://www.videolan.org/doc/streaming-howto/en/ch03.html>):

Encapsulación	Protocolo	Códex de vídeo	Códex de audio
TS	Cualquiera	MPEG 1/2/4, MJPEG, H263, H264, I263, WMV 1/2, Theora	MPEG audio, AAC, a52
Ogg	HTTP	Theora, MPEG 1/2/4, MJPEG, WMV 1/2	vorbis, flac, speex, a52, MPEG audio.
Asf	HTTP	MPEG 4, MJPEG, WMV 1/2	MPEG audio, a52

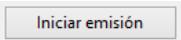
Tabla 1 - Configuraciones de muxer+codecs propuestas por el proyecto VideoLan.

- El único apartado totalmente personalizable es de bitrate de vídeo. Éste debe ser un número entero y viene dado en Kbps.
- Pulse el botón **Aceptar** para efectuar los cambios.
- Tenga en cuenta que puede volver a este panel posteriormente para modificar los campos.

- El botón **Eliminar canal**  sirve para borrar un canal previamente creado. Antes se le preguntará si realmente desea borrar el canal. Tras el borrado se eliminan las asignaciones a este canal de todas las instancias de medios, con la excepción de la instancia de medio relativa a un medio en emisión.

6.6 Inicio de la emisión

Aunque ya estén configuradas las instancias de medios y asignados los canales, la emisión general debe ser iniciada. Por defecto el programa inicia emisión automáticamente, pero usted puede haber cambiado ese comportamiento en el panel preferencias.

Para iniciar la emisión, debe pulsarse el botón **Inicio de la emisión**  del panel *Emisión*. Si todo ha ido correctamente, las etiquetas de la parte inferior de la aplicación habrán adoptado el aspecto mostrado en las figuras 39 y 40, respectivamente.

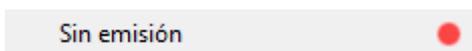


Figura 39 - Mensaje cuando VLC no se encuentra corriendo en segundo plano.

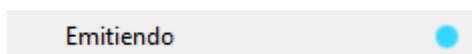


Figura 40 - Mensaje cuando se ha iniciado la instancia VLC (encargada del streaming).

Al iniciarse emisión, el programa llama a una instancia VLC que queda corriendo en segundo plano, sin interfaz gráfica (por lo que resulta un proceso transparente para el usuario). A esta instancia VLC se le carga la configuración de emisión de medios realizada hasta el momento, así como posteriores cambios y configuraciones que sean realizadas (todo de forma automática y transparente para el usuario). VLC actuará como servidor, realizando todas las labores de streaming y transcodificación.

La emisión puede ser iniciada en cualquier momento, no es necesario esperar a tener instancias de medios y canales ya configurados.

A partir de ahora, VLC será testigo de todos los cambios que se realicen (configuración de instancias de medios, creación y asignación de canales, etc.), por lo que el usuario no debe preocuparse de nada más.

La detención de la emisión provoca que muera el proceso relativo a la instancia VLC corriendo en segundo plano. Para ello, presione el botón **Detención de la emisión**  del panel *Emisión*. El botón **Actualización de la emisión**  vuelve a cargar toda la configuración realizada hasta el momento en la instancia VLC. Utilice esta opción únicamente si considera que VLC no ha recibido correctamente la configuración (consulte capítulo de *Monitorización*). Si VLC no pareciera reaccionar a los cambios, pruebe a detener y volver a iniciar la emisión.

6.7 Monitorización

El programa ofrece varios modos de ver la información configurada sobre la instancia VLC que actúa como servidor. Los tres modos principales son:

1. *Visor de listado (figura 41)*. Para abrirlo, presione sobre el botón **Visor de listado** (barra de herramientas 1). Se mostrará un panel con una tabla con las instancias de medios configuradas en orden de emisión, así como un botón para la selección del canal a listar.

Nombre	Ruta	Fecha	Hora Inicio	Hora Final
Big_Fish.avi	C:\Users\Alain...	16/04/2015	12:00:00	14:00:01
bbb_sunflower_1080p_30f...	C:\Users\Alain...	16/04/2015	14:00:01	14:10:35
Black Label Society Concrete Jungle [OFFICIAL VIDEO].webm	C:\Users\Alain...	16/04/2015	14:10:35	14:14:13
bueno.mp4	C:\Users\Alain...	16/04/2015	14:14:13	14:19:05

Figura 41 - Visor de listado. Las instancias de medios se muestran en orden de emisión para un determinado canal.

Este visor posee una pequeña barra de herramientas:

- Botón **Subir** ▲. Sube un puesto el elemento seleccionado (sólo si su hora de inicio es igual a la hora de final del elemento previo).
- Botón **Bajar** ▼. Baja un puesto el elemento seleccionado (sólo si su hora de final es igual a la hora de inicio del elemento próximo).
- Botón **Compactar** ▲▲▲. Elimina cualquier espacio temporal sin emisión existente por encima del elemento seleccionado (éste incluido). Dicho de otro modo, junta todas las instancias de medios para que se emitan una detrás de la otra.
- Botón **Eliminar del canal** 🗑️. Elimina el canal (del conjunto de canales asignados como salida) a la instancia de medio seleccionada.

Si una instancia de medio tiene varios canales como salida y se intenta cambiar de hora mediante los botones vistos, el programa ofrecerá la posibilidad de duplicar la instancia de medio de forma automática; de esta forma los demás canales no se verán afectados.

Este es un visor *offline*, es decir, no se requiere haber iniciado emisión para ver las instancias de medios configuradas (a diferencia del *visor de escaleta gráfica*).

Este visor se actualiza automáticamente ante cualquier cambio.

2. *Visor de escaleta gráfica (figura 42)*. Para abrirlo, presione el botón **Visor de escaleta gráfica**  (barra de herramientas 1). Se mostrará un panel escalable que dibuja el margen horario que ocupa la emisión de cada medio en determinado canal, seleccionable mediante el botón superior. Dejando apoyado el puntero sobre alguno de los rectángulos, aparecerá una ventana con información relativa a su correspondiente instancia de medio.



Figura 42 - Visor de escaleta gráfica. Se muestran rectángulos que abarcan el margen horario de la emisión de determinado medio.

La anterior imagen muestra la escaleta gráfica correspondiente al canal 'ACChannel' con dos instancias de medios configuradas (2 películas de aproximadamente 2 horas, una después de otra, comenzando a las 12:00:00 h).

Las instancias de medios se dibujan con el color de fondo que tienen configurado. Es editable desde el panel *Configuración*. Este modelo de visor es *online*, es decir, la información sólo aparece una vez se ha iniciado la emisión. Es, junto con el *visor VLM*, la forma correcta y definitiva de comprobar que VLC ha recibido correctamente los cambios realizados por el usuario, ya que se vale del feedback entregado por éste para la rutina de dibujo.

Para centrar en el visor un día diferente, selecciónelo en el calendario gráfico del *panel de configuración*. Este visor se actualiza automáticamente ante cualquier cambio.

3. *Visor VLM (figura 43)*. Actívalo mediante el botón **Visor VLM**  VLM es el código que VLC utiliza para la configuración de emisiones (horarios de emisión, destinos, parámetros de transcodificación, etc.). Este visor ofrece al usuario la oportunidad de ver el código VLM que VLC tiene cargado en ese instante, ya que el visor muestra exactamente el feedback que VLC devuelve.

Figura 43 - Visor de código VLM. Se muestra el código cargado en la instancia VLC-servidor.

Es un visor *online*, ya que sólo mostrará información una vez iniciada la emisión. Utilice el botón **Actualizar información** para actualizar el nuevo feedback de VLC, si ha realizado algún cambio.

Los colores ayudan a diferenciar los elementos tipo 'broadcast' de los de tipo 'schedule'. Para saber más sobre el código VLM y sus opciones consulte la documentación de VLC en <https://www.videolan.org/doc/streaming-howto/en/ch05.html>.

Existe un pequeño visor en el panel *Emisión (figura 44)* cuyo cometido es mostrar qué medio se está emitiendo en ese preciso instante en qué canal. En el siguiente ejemplo puede verse que en el segundo de los 4 canales configurados se está emitiendo la película 'Big Fish':

Figura 44 - Monitorizador de medios en emisión. Muestra el medio emitiéndose por cada canal.

El medio aparece representado por el Alias (nombre que usted le proporcionó a su respectiva instancia dentro del programa) y su nombre original (nombre del archivo, el cual se muestra entre paréntesis).

La instancia relativa a un medio que se halle en emisión posee características especiales en el flujo de trabajo del programa:

- Si se borra un canal por el que se está emitiendo, la emisión no se corta.
- No pueden modificarse sus horarios de inicio y parada.
- Se alerta a usuario si se intenta desconfigurar o borrar la instancia de medio (mediante botones de barra de *herramientas 2*).

6.8 Drag and Drop (D&D)

Dado que puede resultar tedioso configurar manualmente el horario de inicio y parada de cada instancia de medio, así como la asignación individual de canales por los que se emitirá, el programa ofrece funciones de *Drag and Drop* para un flujo de trabajo más óptimo. Los dos posibles modos Drag & Drop se presentan en la *figura 45*.

Como es muy probable que usted desee configurar un medio a continuación de otro, y con los mismos canales destino, puede arrastrar cualquier instancia de medio de la *tabla 1* y depositarla sobre cualquier instancia de medio de la *tabla de instancias configuradas* (flecha 1) o del *visor de listado* (flecha 2).

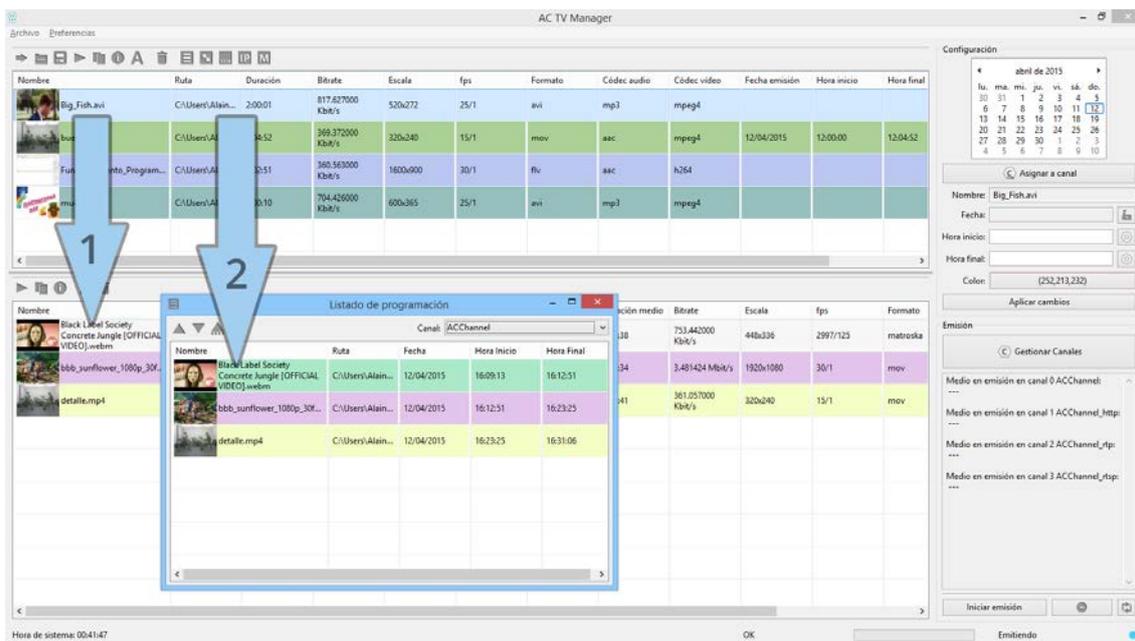


Figura 45 - Dos opciones Drag & Drop; De *tabla 1* a *tabla 2* y de *tabla 1* a *Visor de listado*.

La diferencia entre ambos procedimientos es la siguiente:

- Si se arrastra una instancia de *tabla 1* a una de *tabla 2*, la instancia fuente se configura a continuación de la instancia destino, adquiriendo los mismos canales de salida que ésta última. Esto es útil si se pretende configurar la misma planificación de medios en 2 o más canales simultáneamente.
- Si se arrastra una instancia de *tabla 1* a una del *visor de listado*, la instancia fuente se configura a continuación de la instancia destino, adquiriendo como canal de salida únicamente el canal que se esté visualizando en el visor.

Puede hacer doble clic sobre una instancia de medio del *Visor de listado* para encontrarlo instantáneamente en la *tabla 2* y poder así proceder a su información, desconfiguración, borrado, etc.

6.9 Intersección de medios

Para evitar conflictos de emisión, el programa no permite que dos medios compartan canal de salida al mismo tiempo; es por ello que el programa no permite realizar ninguna configuración horaria o de canal que provoque tal situación.

Puede darse el caso de que el usuario intente realizar una configuración de ese tipo; en ese caso, el programa no aplicará el último cambio y avisará al usuario con mensajes como el de la *figura 46*.

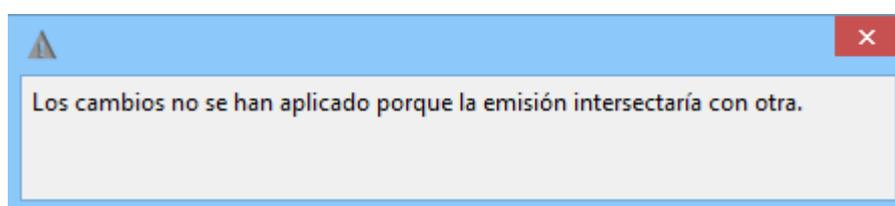


Figura 46 - Mensaje de alerta al intentarse configurar dos instancias de medios en el mismo margen horario.

Tenga en cuenta que el programa ofrece en el inicio y parada de la emisión de medios una resolución de segundos. El mínimo tiempo de transición entre la emisión de dos medios sin provocar intersección se configura del siguiente modo:

- Si el *medio 1* acaba a las 15:33:05 (formato hh:mm:ss), el *medio 2* puede comenzar a las 15:33:05.

6.10 Gestión de la sesión

Se llama sesión al estado en que se encuentra el programa en determinado momento, e incluye:

- Instancias de medios no configuradas.
- Instancias de medios configuradas.
- Instancias VoD.
- Canales (y parámetros de transcoding de cada uno).

Para guardar la sesión hay dos opciones:

1. Botón **Guardado de sesión** . Se abrirá una ventana de explorador de Windows para seleccionar ruta y nombre del fichero a guardar. El archivo resultante tendrá extensión '.acs'. La próxima vez que pulse el botón de guardado, se sobrescribirá el fichero anterior.
2. Opción *Archivo > Guardar sesión* (menú superior de la aplicación). Equivalente a pulsar el atajo *Ctrl+S*. Tiene el mismo comportamiento que el botón de *Guardado de sesión*.

Las opciones *Guardar sesión como...* y *Guardar copia de sesión* sirven para guardar el fichero con otro nombre y directorio, con y sin actualizar la ruta de guardado por defecto (respectivamente), complementando la opción de guardado normal.

Para abrir una sesión previamente guardada presione el botón **Apertura de sesión** (equivalente a *Archivo > Abrir sesión...* o *Ctrl+O*). Se le preguntará si realmente desea abrir una nueva sesión, lo que provoca el cierre de la actual.

El cierre de la sesión actual supone el borrado de medios y canales para dar paso a los de la sesión abierta.

La apertura de una sesión puede llegar a tardar minutos si contiene referencias a un alto número de medios.

6.11 Gestión de preferencias

Para acceder al panel de preferencias (*figura 47*) vaya a *Preferencias > Configurar...* (*Ctrl+P*), en el menú superior de la aplicación. Se abrirá el siguiente asistente:

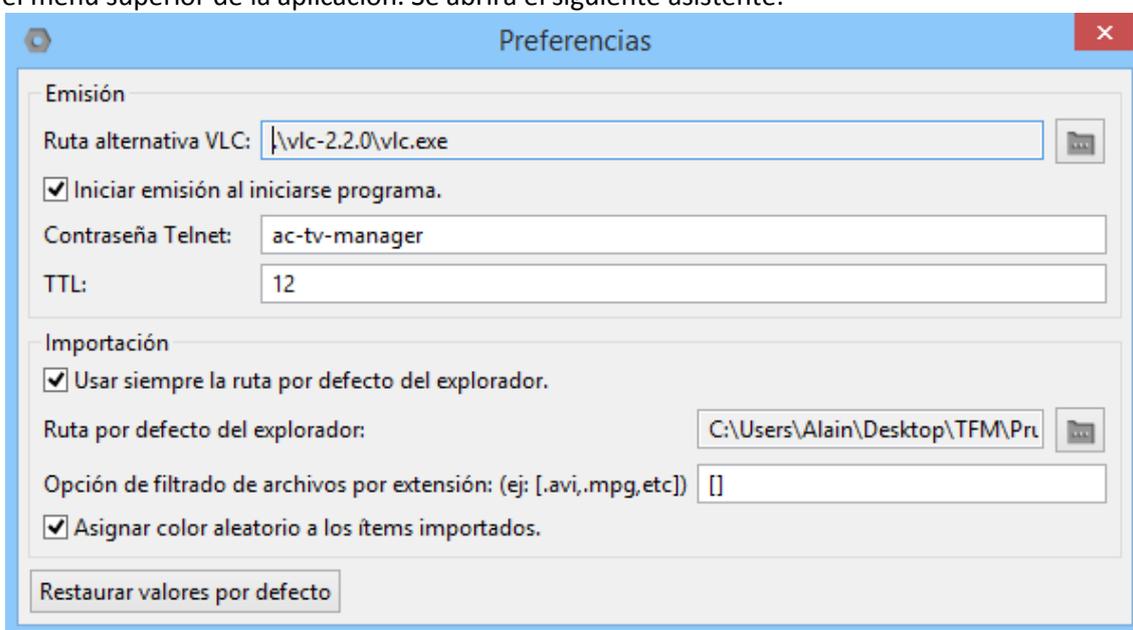


Figura 47 - Ventana de gestión de preferencias.

- **Ruta alternativa VLC.** Por defecto, el programa busca el software VLC que tenga instalado en su PC. Si no lo tiene instalado en un directorio reconocible o simplemente desea que el programa instancie el VLC que usted desea, explore en su sistema mediante el botón de la derecha. La imagen muestra un ejemplo válido de ruta de VLC. Asegúrese de que el ejecutable posee el nombre *vlc.exe*.
- El campo **Iniciar emisión al iniciar programa** obliga a la aplicación a iniciar directamente la instancia de VLC servidor nada más arrancarla.
- El campo de **Contraseña telnet** permite asignar la contraseña Telnet que el programa utilizará para comunicarse con la instancia VLC. Tal comunicación es un proceso transparente para el usuario, no se debe realizar ningún paso más.
- Opción de ajuste de **TTL**. Es el número de routers que cada datagrama IP del stream puede atravesar sin ser eliminado. Por defecto, su valor es 12.

- **Usar siempre la ruta por defecto del explorador** provoca que cada vez que se abra una ventana de exploración de Windows para importar medios se vea el contenido de la ruta configurada en el siguiente campo.
- **Ruta por defecto del explorador** permite seleccionar la ruta por defecto a utilizar en la ventana de importación de medios.
- La opción de **Filtrado** permite filtrar automáticamente ficheros por extensión en la ventana de importación de medios.
- La opción de **Asignación de color aleatorio a ítems importados** ayuda a la diferenciación de éstos en todo el flujo de trabajo del programa. Por defecto viene activada. Desactivado, provoca que los medios tengan fondo blanco (color de tabla).
- Puede volver a los valores predeterminados mediante el botón **Restaurar valores por defecto**.

Salga del asistente cerrando la ventana; los cambios se aplicarán automáticamente.

6.12 Información adicional

- **IP** Botón **Visor de configuración de red**. Abre una ventana (figura 48) que muestra información de configuración de red del sistema (equivalente a `-ipconfig /all` en la shell de windows):

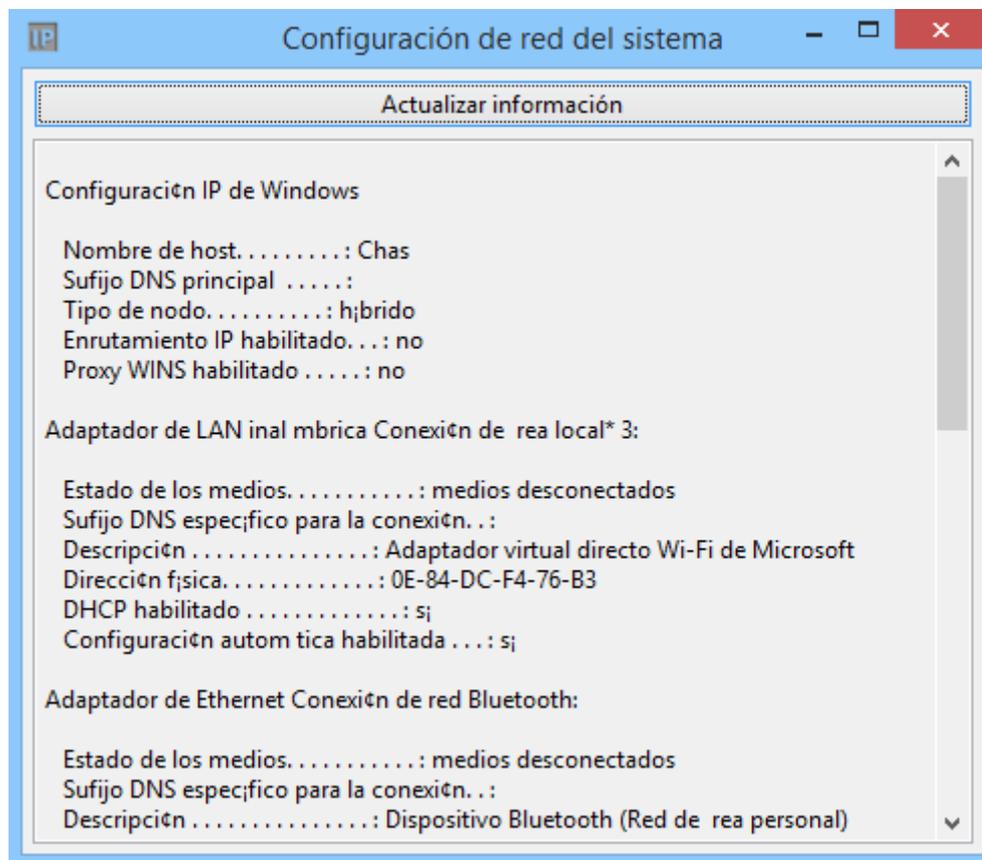


Figura 48 - Ventana de información de red del sistema.

-  Botón **Ver Manual**. Pide al sistema operativo que abra el PDF con el manual de usuario de la aplicación.

6.13 Limpieza de instancias de medios

El programa realiza una limpieza automática de las instancias de medios ya emitidos. El usuario no debe preocuparse de realizar tal labor manualmente.

Si se configura una instancia de medio con una hora de final inferior a la hora actual del sistema sobre el que corre la aplicación, la instancia queda directamente eliminada.

7 Disposición de ficheros

El sistema de gestión *AC TV Manager*, como se ha visto en apartados anteriores, es un sistema modular dependiente de varios softwares para completar sus funciones.

El sistema viene en forma aplicación software ejecutable y portable, cuyo lanzador es el fichero *AC_TV_Manager.exe*. Viene acompañado de una estructura de ficheros y directorios mostrada en la *figura 49*.

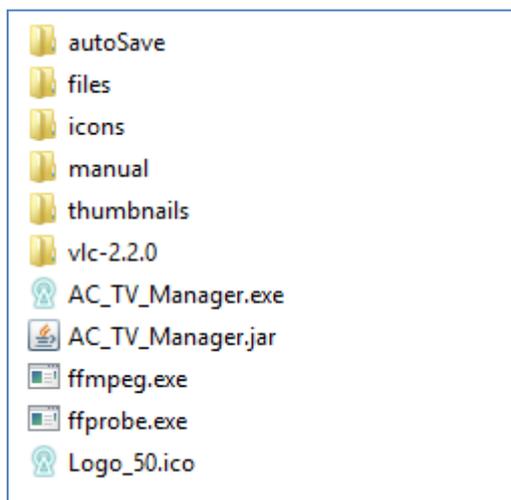


Figura 49 - Estructura raíz de ficheros y carpetas de AC TV Manager.

Para que el sistema cumpla con todas las funcionalidades debe respetarse esta jerarquía de carpetas y archivos, independientemente del directorio donde se hallen.

En relación a la figura 45, a continuación se listan los elementos que componen la estructura.

7.1 autoSave

Es el directorio que contendrá ficheros de auto-guardado de sesión. Éstos ficheros (*figura 50*) los genera automáticamente el programa cada pocos segundos (sólo un fichero por sesión), y mantienen la configuración de medios y canales que se tenga en ese preciso instante.

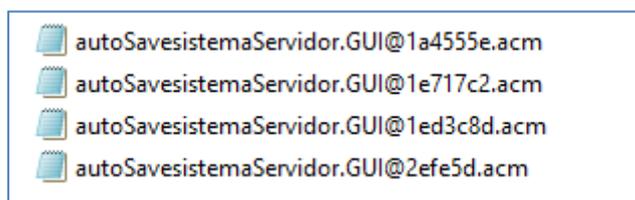


Figura 50 - 4 ficheros de auto-guardado, correspondientes a 4 sesiones diferentes.

7.2 files

Es el directorio que contiene los ficheros *autoVLM.txt* y *replicaVLM.txt* (equivalentes a los *Load.txt* y *Save.txt* planteados en los apartados 4.3.4 y 4.3.5). Son aquellos ficheros con código

VLM en su interior y que son interpretados por la aplicación Java en todo momento. También contiene el fichero *preferencias.txt*, que se sobrescribe cada vez que el usuario las modifica. La *figura 51* presenta el contenido de la carpeta *files*.



Figura 51 - Contenido de la carpeta *files*.

7.3 icons

Es el directorio que contiene todos los iconos (*figura 52*) de los que se hace uso en la aplicación Java.

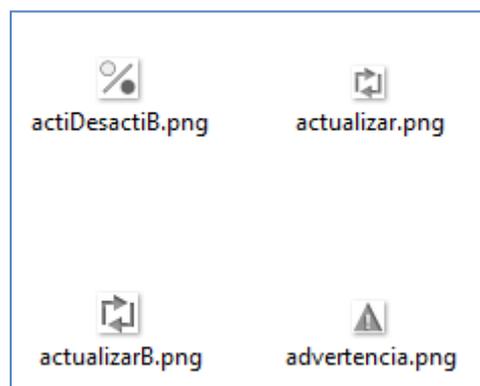


Figura 52 - Algunos de los iconos almacenados dentro del directorio *icons*.

7.4 manual

Este directorio únicamente contiene un fichero PDF (*figura 53*), manual de la aplicación, el cual es accesible desde la interfaz gráfica del programa.

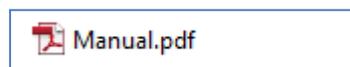


Figura 53 - Fichero PDF almacenado en el directorio *Manual*.

7.5 thumbnails

Este directorio inicialmente sólo contiene una imagen (*figura 54*), la que se aplica por defecto a los medios importados de audio o cuya información de vídeo no es apta.



Figura 54 - Imagen por defecto para los medios sin información de vídeo.

El directorio va adquiriendo más imágenes-miniatura (thumbnails) a medida que se importan medios al programa. Se vacía cada vez que el programa es cerrado. Los thumbnails que se

generan en la importación de medios (uno por medio) son de 60x45 px y rondan los 5 KB de peso.

7.6 vlc

Este directorio es el relativo a una versión portable de VLC. En el momento del desarrollo de la aplicación-servidor fue la 2.2.0 (RinceWind), y el programa apunta por defecto al ejecutable *vlc.exe* de esta versión de VLC que acompaña al sistema gestor.

7.7 AC_TV_Manager.exe

Es el lanzador de la aplicación Java. El programa se abre y es utilizable; no requiere ningún tipo de instalación y/o configuración.

Este lanzador se ha creado a partir de la utilidad *JavaExe* creada por *DevWizard* [18].

7.8 AC_TV_Manager.jar

Es el quid de toda la aplicación. Contiene todas las clases Java necesarias que componen y hacen funcionar al programa.

Se ha obtenido mediante la exportación de un *Runnable Jar File* desde la plataforma Eclipse. *AC_TV_Manager.jar* es lanzado por *AC_TV_Manager.exe*, quien llama al JRE (Java Runtime Environment) instalado en el PC del usuario y hace que corra el *.jar*.

7.9 ffmpeg.exe

Este ejecutable es el encargado de generar los thumbnails en el momento de la importación de medios. Este proceso resulta transparente para el usuario, ya que la aplicación Java lo hace correr en segundo plano, sin interfaz gráfica.

Debe tener el nombre de *ffmpeg.exe* para ser utilizado de forma correcta. Si la aplicación Java no es capaz de enlazar con el ejecutable, asignará la imagen de la figura 6-6 al medio importado.

7.10 ffprobe.exe

Aunque la interfaz gráfica del programa Java es capaz de arrancar y permanecer estable por sí sola, las funciones de importación de medios y emisión de streams requieren tener accesibles las restantes aplicaciones de software: *FFmpeg* y *VLC*. El programa alerta al usuario si alguna de las aplicaciones no es hallada; la *figura 55* muestra un ejemplo.

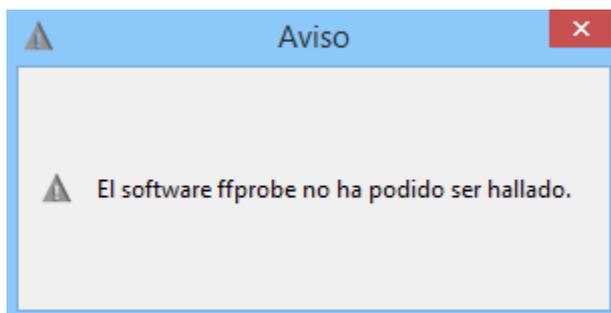


Figura 55 - Mensaje de aviso a usuario si el software FFprobe (parte del proyecto FFmpeg) no está accesible.

ffprobe.exe es parte del proyecto FFmpeg y tiene la característica de hacer un análisis profundo de los streams de determinado medio de vídeo o audio.

Al igual que *ffmpeg.exe*, este ejecutable corre en segundo plano cuando la aplicación Java lo solicita en la importación de medios; su función es la de devolver al programa información detallada sobre el medio importado.

7.11 Logo_50.ico

Es un tipo de fichero de imagen cuyo objetivo es dar estética al ejecutable *AC_TV_Manager.exe*, asignándole el logo del sistema presentado en la *figura 56*.



Figura 56 - Logo del sistema AC TV Manager

8 Sistema cliente

Este proyecto ha centrado sus esfuerzos en crear un sistema servidor versátil y con diferentes tipos de streams de salida, lo que ayuda a que no dependamos de una solución cliente única. La persona que utilice el sistema gestor de IPTV AC TV Manager tiene la posibilidad de escoger qué tipo de software o aplicación acabará siendo cliente, siempre que estos sean capaces de leer los flujos de datos emitidos por VLC.

Las soluciones cliente pueden ir desde una simple instancia VLC recibiendo el stream hasta un televisor en el hogar. Todo dependerá de cómo y dónde se utilice la capacidad del sistema servidor. Imagínese el caso en que la salida stream de VLC es recibida por un hardware multiplexador de vídeo y posteriormente la señal sea modulada y enviada a través de una antena televisiva; el usuario final recibiría la emisión, siempre que fuera correctamente sintonizada, en su televisor.

El caso anterior es una posibilidad real, pero no resulta fácil de imaginar para un usuario que no dispone de dispositivos avanzados, y no es objeto de este proyecto (entre otras cosas por falta de medios) demostrar un caso práctico de tales dimensiones. Es por ello que a continuación se presentan dos soluciones clientes reales realizadas en red local, una basada en el propio VLC y otra basada en reproductor HTML5 corriendo en un navegador Firefox.

8.1 Solución cliente con VLC

Una opción para recibir el/los streams emitidos por el sistema *AC TV Manager* es utilizar el software VLC, capaz de actuar como servidor y como cliente.

Para este ejemplo proponemos dos ordenadores conectados en red. Uno hará de servidor y otro de cliente. La figura 57 muestra la configuración IP equipo servidor desde el propio sistema gestor de IPTV.

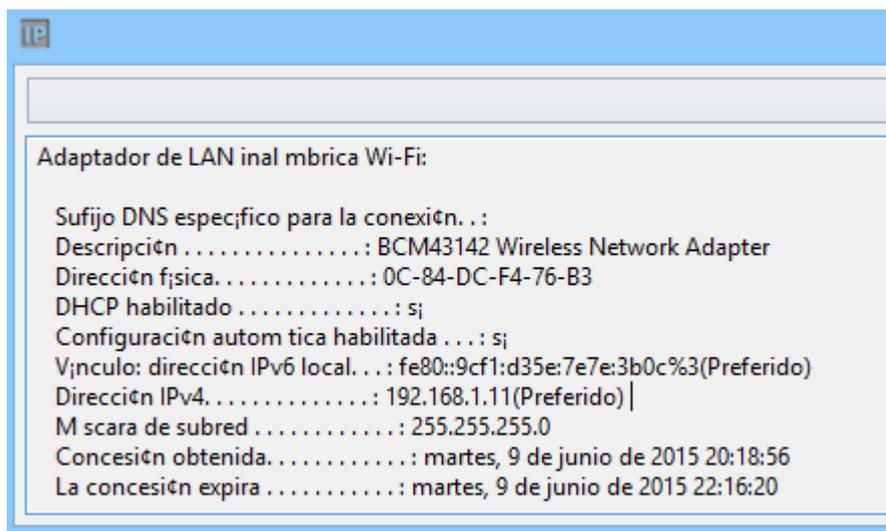


Figura 57 - Equipo servidor: dirección IP 192.168.1.11.

Observamos cuál es la dirección IP del cliente (*figura 58*) para crear un canal de emisión dirigido a él:

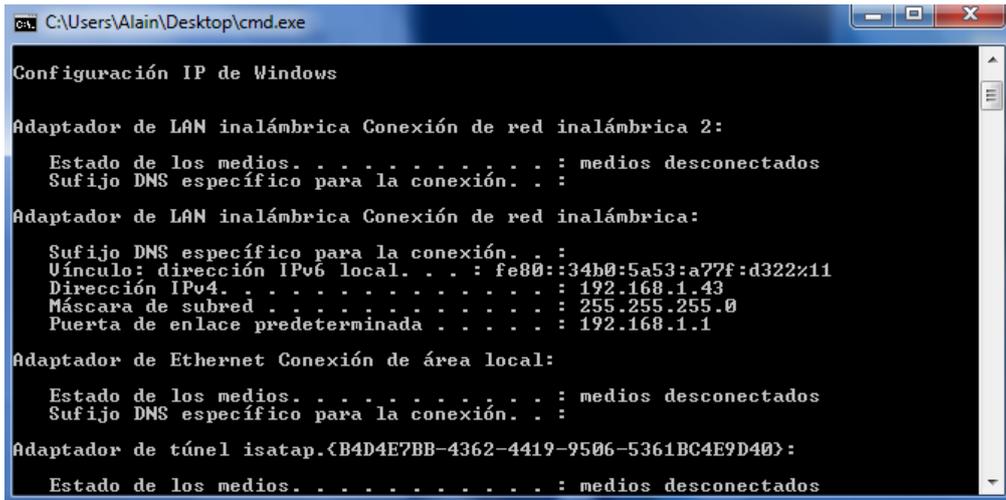


Figura 58 - Equipo cliente: dirección IP 192.168.1.43.

Ahora creamos desde el sistema gestor un canal (de nombre *Emission-a-PC-Cliente*, protocolo UDP y puerto 8000) cuyo stream resultante se dirija a tal dirección IP, tal como muestra la figura 59.

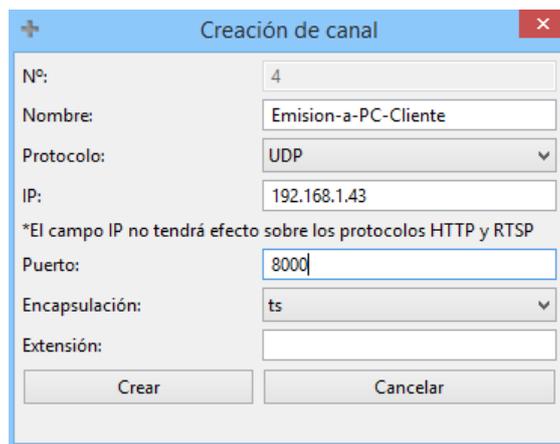


Figura 59 - Creación de canal con destino dirección IP 192.168.1.43.

Programamos un medio a determinada hora desde el sistema gestor, como se muestra en la figura 60.

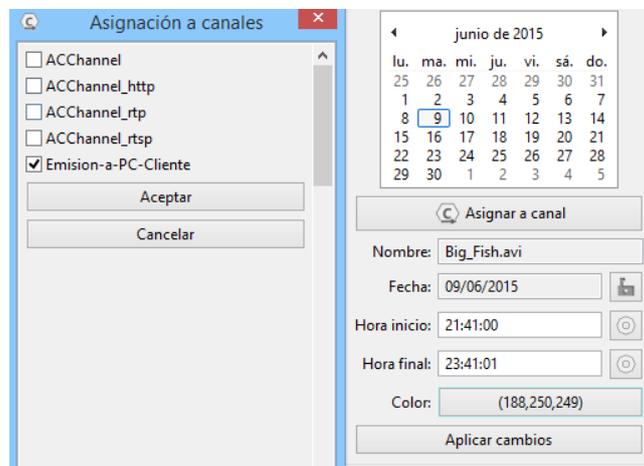


Figura 60 - Emisión de la película *Big Fish* a las 21:41:00 el día 09/06/2015 por el canal *Emission-a-PC-Cliente*.

Desde el cliente, en este ejemplo, lo único que debería hacerse para comenzar a recibir el stream es abrir un volcado de red tal como muestra la figura 61 (recepción con protocolo UDP y puerto 8000).

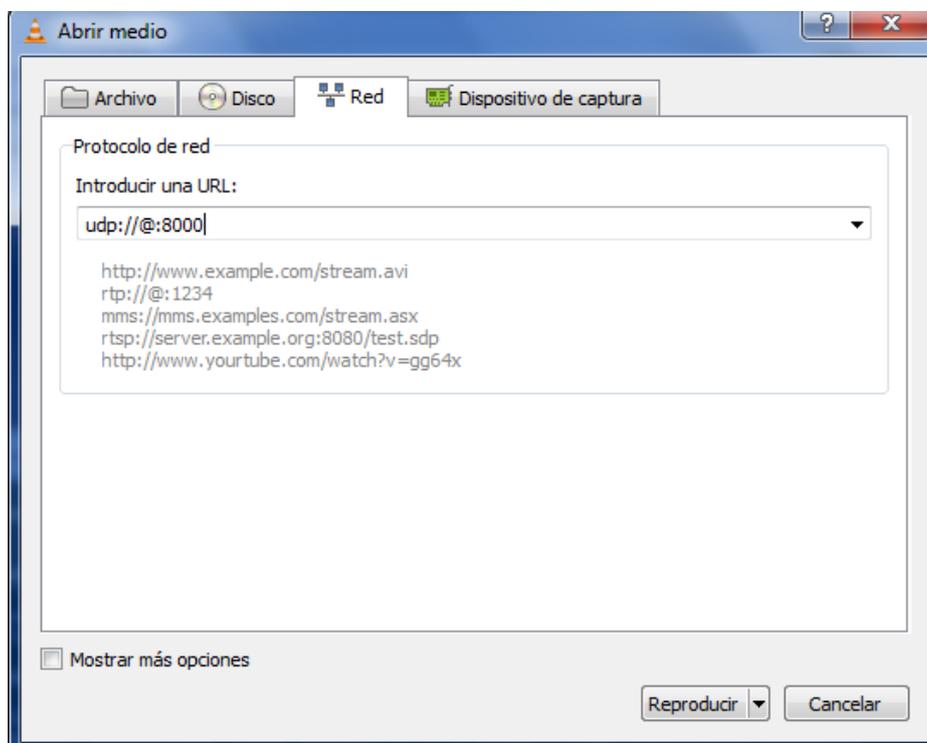


Figura 61 - Recepción de stream en puerto 8000, protocolo UDP.

El resultado sería la visualización (figura 62) del medio programado y más medios posteriores, si fueran programados desde el sistema gestor.



Figura 62 - Visualización del stream enviado por el canal Emision-a-PC-Cliente.

8.2 Solución cliente con reproductor HTML5

Pongamos ahora un caso diferente, en el que queremos que el sistema gestor de IPTV envíe un stream a todo usuario que lo solicite. Configuraremos un canal HTTP en AC TV Manager tal como muestra la figura 63.

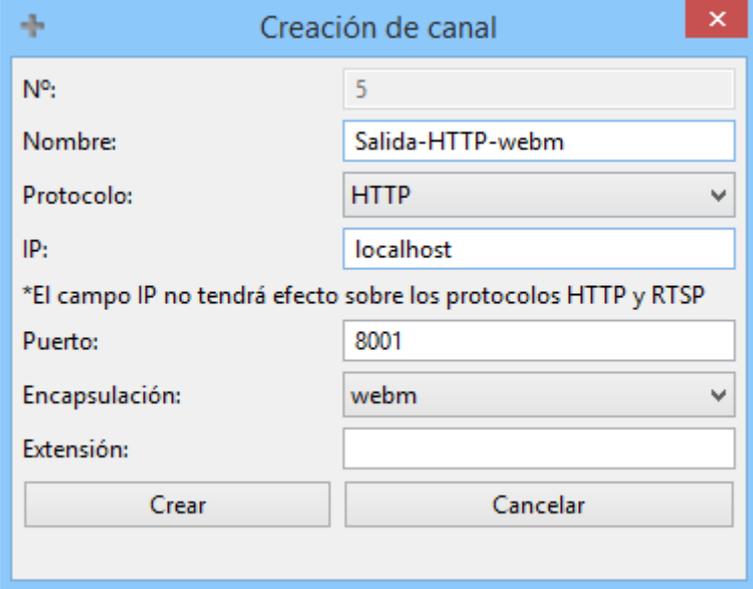


Figura 63 - Canal configurado para envío de streams válidos para reproductores HTML5.

La configuración del canal posee las siguientes características:

- Nombre *Salida-HTTP-webm*.
- Protocolo HTTP, para actuar como servidor bajo demanda (detallado en el apartado 4.2 de la memoria).
- La IP siempre será la del equipo que corre el programa si se seleccionan los protocolos HTTP o RTSP.
- Puerto 8001. Es el puerto al que los demás equipos en red que quieran recibir el stream deberán enviar su solicitud.
- Debido a que el estándar HTML5 no permite cualquier formato de streaming, para este caso se ha seleccionado una encapsulación válida: webm. Más adelante se muestra la solución relativa a códecs de vídeo y audio.
- Dejamos la extensión en blanco para este ejemplo.

Para crear un flujo de datos interpretable por el reproductor HTML5 necesitaremos crear la combinación adecuada de códecs y formato. La encapsulación asignada al canal ha sido webm pero resulta necesario que los códecs sean:

- VP8 para vídeo.
- Vorbis para audio.

Existen dos opciones; o bien codificamos con esos códecs todos los medios antes de ser emitidos o bien aprovechamos la característica de transcoding en tiempo real del sistema. La figura 64 muestra la configuración de transcodificación adoptada para este caso (*VP80* y *vorb* se corresponden con VP8 y Vorbis, cambia el nombre debido a la nomenclatura que VLC le proporciona).

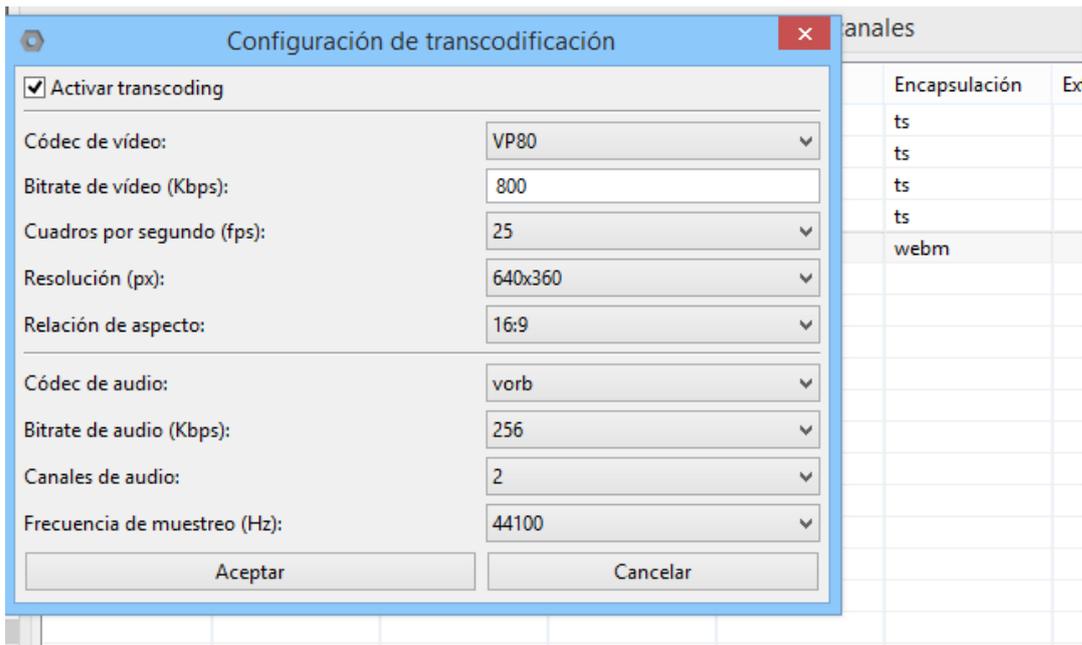


Figura 64 - Configuración de la transcodificación para el canal Salida-HTTP-webm.

Ya tenemos el canal listo. Al igual que hemos hecho en el ejemplo anterior ponemos medios a emitir. El próximo paso es 'sintonizarlos' desde el/los navegadores cliente.

Como reproductor HTML5 nos servirá el siguiente fragmento de código guardado en un fichero con extensión *.html*:

```

<video width="356" height="200" controls>
<source src="http://192.168.1.11:8001/" type="video/webm"/>
<em>Sorry, your browser doesn't support HTML5 video.</em>
</video>

```

Cabe destacar que la fuente del reproductor es la dirección IP del equipo que actúa como servidor junto con el puerto 8001, el que configuramos para el canal.

Este fichero puede ser abierto desde diferentes equipos de la misma red al mismo tiempo y todos recibirán una copia del stream que VLC esté procesando en ese instante; es decir, cada reproductor HTML5 actúa como una solicitud de stream para el sistema servidor en este caso. Las figuras 65 y 66 muestran dos reproductores HTML5 corriendo sobre el navegador Chrome en dos equipos diferentes de la misma red.



Figura 65 - Visualización del stream en reproductor HTML5 en navegador Chrome (cliente 1).

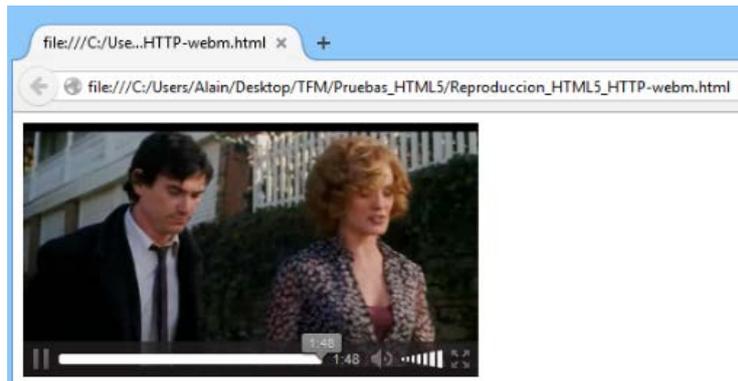


Figura 66 - Visualización del stream en reproductor HTML5 en navegador Chrome (cliente 2).

9 Conclusiones

9.1 Cumplimiento del objetivo

Los objetivos se han llevado a cabo con éxito, cumpliéndose las siguientes pautas:

- Se ha conseguido obtener un sistema servidor IPTV estable y funcional (*figura 67*) capaz de mantener continuidad televisiva de hasta N canales simultáneamente.
- El sistema ha sido desarrollado para Windows (en vez de Linux) por ser el sistema operativo más utilizado y así poder ser utilizado por un número mayor de usuarios.
- Se ha logrado que todo el sistema funcione en colaboración con soluciones de software libre (uso de FFmpeg y VLC), nunca privativas.

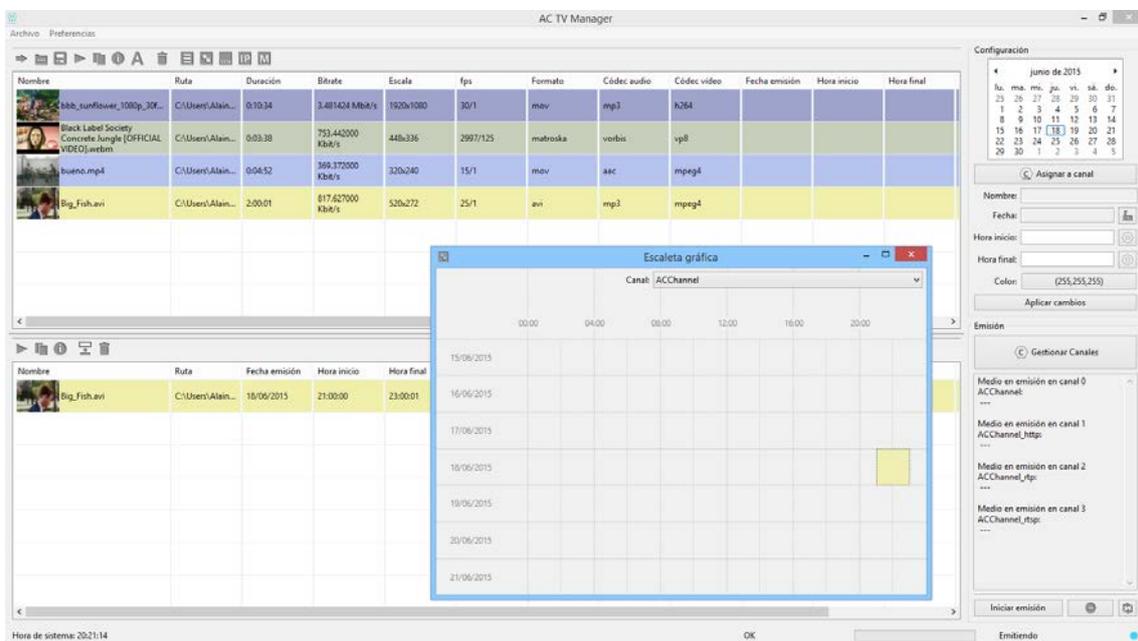


Figura 67 - Versión actual del programa (para Windows)

9.2 Conclusiones sobre el proyecto

9.2.1 Conclusiones generales

Se ha logrado desarrollar un sistema de gestión de IPTV basado en software libre, que funciona y es compatible con las últimas versiones de los proyectos VLC y FFmpeg (2.2.0 *WeatherWax* y *ffmpeg-20150627-git-7728d23-static*, respectivamente, tanto con las versiones de 32 bits como con las de 64). Se ha cumplido, por tanto, el objetivo de crear una solución software no propietaria.

Este sistema permite gestionar la emisión de hasta N canales IPTV simultáneamente, permitiendo manejar los horarios de emisión del contenido de vídeo, así como las

diferentes salidas vía streaming. El sistema posee, además, la opción de transcodificación en vivo para cada canal creado, lo que abre un gran abanico de posibilidades ya que permite adaptar el stream a una gran variedad de formatos.

Se ha desarrollado una aplicación en Java a modo de interfaz de usuario, que además cumple con las funciones de centralización y procesado de la información enviada y recibida por los diferentes programas (VLC, FFmpeg y consola del sistema). Se ha hecho hincapié en la experiencia de usuario, haciendo que la aplicación genere un thumbnail representativo de cada medio al ser importado. Aún más importante resulta la posibilidad de "Drag & Drop" para la configuración horaria de los medios a emitir, lo que provoca un gran aumento de la comodidad en la gestión. También es importante mencionar la nueva distribución de múltiples ventanas, que permite optimizar el espacio de trabajo a gusto del usuario.

El lado cliente, aunque se presentan dos sencillas soluciones para la recepción de los streams emitidos (mediante cliente VLC y mediante reproductor HTML5), se presenta como una opción abierta para el usuario, ya que puede ir tan lejos como desee: podría lograr que el usuario final fuera un receptor de IPTV específico si se dispusiera de hardware y/o medios necesarios (como routers multicast) para el reparto de los streams.

El sistema servidor puede ser utilizado, además de como gestor IPTV, como herramienta para estudios relacionados a redes de datos, congestión, etc. ya que pueden controlarse las emisiones a realizar y saber qué flujo de datos se está transmitiendo en cada momento.

El hecho de que sea una solución open-source y gratuita permite que cualquier persona interesada que disponga de un sistema Windows y el JRE (Java Runtime Environment) de 32 bits (probado a partir de la versión 1.7.0) instalado corra el programa. Por otra parte, cualquiera puede acceder al código e implementar modificaciones específicas en función de sus necesidades.

9.2.2 Mejoras y cambios incorporados en la nueva versión

De cara a la comparación con respecto a la versión anterior del sistema de gestión, se presentan a continuación los cambios significativos y las mejoras que se han añadido:

- Software desarrollado para Windows en vez de para Linux.
- Sistema de varias ventanas para optimización y personalización del espacio de trabajo.
- Sistema de thumbnails (imágenes pequeñas representativas) para los medios importados.
- Visor de la escaleta mejorado: siendo una ventana independiente, puede hacerse más grande (y por tanto, abarcar mayor resolución horaria) que su antecesora.
- Interfaz gráfica de usuario (GUI) con más opciones: ahora se dispone de botones para facilitar la copia de instancias de medios, apertura de paneles, cambio de nombre de los medios, etc.
- Posibilidad de nombrar los medios importados dentro de la aplicación.
- Opción de transducción del stream: se da la posibilidad de realizar una conversión de códec de vídeo y audio en "tiempo real", con parámetros configurables.
- Continuidad de la emisión tras caída y recuperación de ésta. Si la emisión se detiene y se vuelve a iniciar, se retoma el punto en el que quedó el/los medio(s) en emisión.
- Limpieza automática y transparente para el usuario de los medios ya emitidos.

- Gestión de hasta N canales simultáneos, con transcodificación independiente para cada canal.
- Sistema Drag & Drop que permite una fácil y rápida configuración horaria de los medios, para una mejor experiencia de usuario (UX).
- Notificador auxiliar de medios en emisión.
- Visualizador de programa (para monitorizar un determinado canal como usuario).
- Capacidad de superposición de subtítulos en los streams.

9.3 Problemas encontrados y cómo se han solucionado

La mayor dificultad ha residido en hacer que el sistema permitiera gestionar todos los canales que el usuario quiera, ya que la versión anterior estaba pensada para gestionar un único canal. Esto ha provocado la necesidad de un subsistema (tanto a nivel de programación como a nivel de interfaz gráfica) de gestión de canales, con la implementación de todas las funciones necesarias para la creación, almacenamiento y característica de transcoding de éstos.

Por otra parte, la adaptación del software al sistema operativo Windows ha requerido más cambios de los estimados debido a diferencias de comunicación con la consola de comandos en comparación con Linux.

9.4 Aportaciones personales

La realización de este proyecto me ha permitido terminar de asentar algunos conocimientos de programación con el lenguaje Java, así como mayor entendimiento acerca de las técnicas, características y posibilidades del streaming de vídeo y audio.

Por otra parte, la satisfacción de haber logrado un software estable y funcional que se podría empezar a distribuir para su prueba y uso en diferentes ámbitos.

9.5 Líneas futuras

La interfaz VLM de VLC permite crear "medias" del tipo VoD (Vídeo bajo demanda). Sería interesante dotar al sistema de la característica de servidor VoD. Se realizaron pruebas durante la realización de este proyecto, sin resultados positivos debido a limitaciones de VLC con esta característica. Es posible que en posteriores versiones se mejore por parte del proyecto VideoLan.

También se podría intentar convertir el proyecto en un sistema servidor colaborativo. Es decir, que el sistema pudiera realizar cambios ante peticiones por parte de diferentes usuarios desde diferentes dispositivos.

Por otra parte, aunque con menor importancia, podrían aprovecharse la funcionalidad del software FFmpeg o incluso de VLC para incorporar un conversor de formatos a la interfaz gráfica.

Finalmente, se podría pensar en incorporar algún mecanismo para monitorizar el streaming en la propia GUI, en vez de tener que hacer uso de instancias VLC externas, que tienen el único

inconveniente de que no permiten visualizar streams unicast directos (aquellos realizados mediante protocolo UDP o RTP desde nuestra solución servidor).

10 Bibliografía y referencias

- [1] Wes Simpson, Howard Greenfield. 2012. *IPTV and Internet Video: Expanding the Reach of Television Broadcasting*. NAB Executive Technology Briefings.
- [2] D. Banodkar, K. K. Ramakrishnan, S. Kalyanaraman, A. Gerber, O. Spatscheck, "Multicast Instant Channel Change in IPTV Systems", En *Communication Systems Software and Middleware and Workshops*, 2008. COMSWARE 2008. 3rd International Conference on, IEEE, pp. 370-379, January 2008.
- [3] Miguel Garcia, Jaime Lloret, Miguel Edo, Raquel Lacuesta, IPTV Distribution Network Access System Using WiMAX and WLAN Technologies, Proceedings of the 4th edition of the UPGRADE-CN workshop on Use of P2P, GRID and agents for the development of content networks, Munich (Alemania), 11-13 de Junio 2009.
- [4] W. Poon, K. Lo, J. Feng, "Performance Study for Streaming Layered Encoded Videos in Broadcast Environment", En *Information Technology and Applications*, 2005. ICITA 2005. Third International Conference on, vol. 2, IEEE, pp. 722-727, July 2005.
- [5] A. Lopez-Herreros, A. Canovas, J. M. Jimenez, J. Lloret. "A New IP Video Delivery System for Heterogeneous Networks using HTML5". CSSMA-I02: Multimedia applications and security from Thu, June 11, 2015.
- [6] K. Imran, M. Mellia, M. Meo, " Measurements of Multicast Television over IP", En *Local & Metropolitan Area Networks*, 2007. LANMAN 2007. 15th IEEE Workshop on, IEEE, pp. 170-175, June 2007.
- [7] Chas, Alain. 2013. *Desarrollo de un sistema de gestión de una parrilla de contenidos de vídeo y audio en red*. Pamplona: Universidad Pública de Navarra.
- [8] Eckel , Bruce. 2007. *Piensa en Java: cuarta edición*. Prentice Hall.
- [9] Deployed IPTV for Global Communications. En:
http://www.cisco.com/web/about/ciscoitwork/downloads/ciscoitwork/pdf/cisco_it_case_study_ip_tv.pdf
- [10] AvediaServer - Centralized IPTV Management System. En:
<http://www.exterity.com/products/avediaserver.html>
- [11] IPTV Management Server. En:
<http://www.antiktech.com/en/streamingbripte-headend/management/iptv-management-server>
- [12] IPTV Middleware system.En:
<http://iptvmiddleware.com/iptv-middleware-system/>
- [13] IPTV Content Management System. En:
http://www.snap.tv/technology/IPTV_cms.html
- [14] IPTV Manager. En:

<http://www.encodedmedia.com/iptv-manager/>

- [15] Documentación interfaz VLM del proyecto VLC. En:
https://wiki.videolan.org/Documentation:Streaming_HowTo/VLM/
- [16] Documentación librería SWT del proyecto Eclipse. En:
www.eclipse.org/swt
- [17] Documentación FFmpeg project. En:
<https://www.ffmpeg.org/documentation.html>
- [18] JavaExe. En:
<http://devwizard.free.fr/html/en/JavaExe.html>