

PROJECTE FINAL
MÀSTER PROGRAMARI LLIURE

**Gravació amb transcodificació al
vol des de càmera IP**

DESENVOLUPAMENT D'APLICACIONS DE
PROGRAMARI LLIURE

Autor:
Héctor CANO VÁZQUEZ

Consultor UOC:
Gregorio ROBLES MARTÍNEZ
Enginyer de Telecomunicacions
per la Universitat Politècnica
de Madrid
Consultor del màster
internacional de Programari
Lliure a la UOC

Consultor Extern:
Ferran FERRANDIS
Enginyer de Telecomunicació
per la Universitat Politècnica
de Catalunya
Director Àrea TIC a
DÈBID

dèbid

Llicència

Aquesta memòria es distribueix sota llicència
Reconeixement-NoComercial-CompartirIgual 3.0 No adaptada
CC BY-NC-SA 3.0



Això és un resum fàcilment llegible de la llicència¹

Sou lliure de:

- copiar, distribuir i comunicar públicament l'obra
- fer-ne obres derivades

Amb les condicions següents:

Reconeixement Heu de reconèixer els crèdits de l'obra de la manera especificada per l'autor o el licenciadador (però no d'una manera que suggereixi que us donen suport o rebeu suport per l'ús que feu l'obra).

No comercial No podeu utilitzar aquesta obra per a finalitats comercials.

Compartir Igual Si altereu o transformeu aquesta obra, o en genereu obres derivades, només podeu distribuir l'obra generada amb una llicència idèntica a aquesta.

Entenent que:

Renuncia Es pot renunciar a alguna d'aquestes condicions si obteniu el permís del titular dels drets d'autor.

Domini Públic Aquesta llicència no afecta a la situació de l'obra o algun dels seus elements quan es trobi en el domini públic, segons la llei vigent aplicable.

Altres drets Els drets següents no queden afectats de cap manera per la llicència:

- Els vostres drets de repartiment just o ús just;
- Els drets morals de l'autor;
- Drets que altres persones poden ostentar sobre l'obra o sobre l'ús que se'n fa, com per exemple drets de publicitat o privacitat.

Avís Quan reutilitzeu o distribuïu l'obra, heu de deixar ben clar els termes de la llicència de l'obra.

¹extret de <http://creativecommons.org/licenses/by-nc-sa/3.0/deed.ca> Podeu trobar el text legal (la llicència complerta), en anglès, a <http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode> i a l'apèndix A

Resum

És proposa realitzar una aplicació, pensada per a ser accedida en forma d'aplicatiu web que permeti gestionar gravacions de llarga durada o fins i tot indefinides d'imatges i so provinents d'una càmera, fent servir el còdec h.264 per al vídeo i emmagatzemant en un contenidor mp4.

L'aplicació haurà de permetre engegar i aturar la gravació, i accedir (reproduir) al vídeo i àudio gravat sense necessitat d'aturar la gravació en curs. S'haurà de poder reproduir el vídeo gravat desplaçant-se amb facilitat al punt desitjat.

S'haurà de poder accedir a l'aplicació mitjançant un navegador web. Es tindrà especialment en compte la possibilitat d'accedir des de dispositius de pantalla tàctil.

Els sistemes de gravació 24x7 que es fan servir en monitorització i vídeo-vigilància, o sistemes de gravació pensats per gravacions de llarga durada (hores, en comptes de minuts) requereixen que la gravació no es realitzi en un únic fitxer, sinó en múltiples fitxers de curta durada, i que aquests fragments es puguin reproduir després de forma continua sense que s'evidenciïn els talls. Aquesta funció és especialment adient si es pretén poder revisar el vídeo gravat sense aturar la gravació en curs.

FFmpeg (ffmpeg.org) es una solució completa i multiplataforma per a gravar, convertir i fer streaming d'àudio i vídeo. Inclou libavcodec —la llibreria líder per a còdecs d'àudio i vídeo—. FFmpeg permet capturar vídeo de diversos orígens, com ara càmeres IP, i gravar-lo en formats molt diversos. Per a l'àmbit d'aquest projecte ressaltaré H.264 sobre mp4. A partir de la versió 0.9 FFmpeg permet realitzar gravacions fraccionades.

Aquest projecte es desenvolupa fent servir càmeres del fabricant Axis (www.axis.com), ja que són càmeres IP amb una API pública que permet accedir a les imatges en temps real. Axis és el fabricant que més ajuda als desenvolupadors, fent pública la seva API i facilitant gran quantitat d'informació i exemples d'ús.

A en Ferran Ferrandis i en Gustau Ballester, per la seva col·laboració, per deixar-me realitzar aquest projecte a Dèbid, i sobre tot per que són amics.

A en Lluís Casas i en Xavi Guilà, de MEbcn, per la oportunitat que em van donar de seguir fent el mateix, el que m'agrada, però fent quelcom completament diferent.

A María José, la meua dona, per la seva paciència i recolzament.

Índex

1	Introducció	1
1.1	Objectius	1
1.2	Àmbit del projecte	2
1.3	Orígens	2
1.4	Estat de l'art	4
1.4.1	FFmpeg	4
1.4.2	H.264/MPEG-4 AVC	6
1.5	Estructura d'aquesta memòria	7
1.6	Llicència del programari	7
2	Gestió del projecte	9
2.1	Introducció	9
2.2	Entorn de treball	9
2.2.1	Instal·lació Ubuntu sobre Virtualbox	10
2.2.2	Llibreries i executables FFMpeg	11
2.2.3	Entorn de desenvolupament	12
2.2.4	Git i Dropbox	16
2.2.5	Gestió de errors i noves funcionalitats	20
2.2.6	Llistes de correu	20
2.3	Entorn de la memòria	21
2.3.1	Estructura de documents	21
2.3.2	Instal·lació de L ^A T _E X	23
2.3.3	Documentació i ajuda de L ^A T _E X	24
2.3.4	Gestió de versions	24
3	Estudi viabilitat	25
3.1	Anàlisi de requeriments	25
3.1.1	Casos d'us	25
3.1.2	Requisits funcionals	27
3.1.3	Requisits no funcionals	27
3.2	Anàlisi de viabilitat	28

3.3	Proposta de desenvolupament	29
3.3.1	Gravació	29
3.3.2	Reproducció	30
4	FFmpeg	31
4.1	Opcions d'instal·lació	31
4.2	Instal·lació des de repositori	32
4.2.1	libav	32
4.2.2	FFmpeg	34
4.3	Obtenció i compilació	37
4.3.1	Abans d'instal·lar	38
4.3.2	x264	40
4.3.3	Obtenció codi font	43
4.3.4	Compilació	44
4.3.5	Instal·lació	45
4.3.6	Inspecció codi font	47
4.4	FFmpeg tests	50
4.4.1	Funcionament bàsic	51
4.4.2	Captura de càmera	55
4.4.3	Format de sortida	57
4.4.4	Sortida mp4 amb H.264	60
4.4.5	Segmentació al vol de la sortida	63
4.4.6	Conclusions	67
5	Gestió de processos	69
5.1	Introducció	69
5.2	<i>system</i>	70
5.3	<i>popen</i>	71
5.4	<i>popenRWE</i> , <i>pstreams</i>	74
5.5	<i>QProcess</i>	74
6	Reproducció vídeo	79
6.1	Reproductors multimèdia	79
6.2	A través de web	80
6.2.1	L'etiqueta <code><video></code>	81
6.2.2	Gestió de llistes	82
6.2.3	Opcions avançades	84
6.3	El mur	85

7	Conclusions	87
7.1	Objectius complets	87
7.2	Objectius no complets	88
7.3	Futurs desenvolupaments	88
7.4	Aprenentatge	89
7.5	Programari Lliure	90
	Bibliografia	91
A	CC BY-NC-SA 3.0	93
B	GPL v2	103
C	FFmpeg - sortida de configure	113
D	Reproductor HTML 5	123

Índex de figures

2.1	Opcions Virtualbox	11
2.2	Plugins de Netbeans	14
2.3	Netbeans, opcions de C/C++	15
2.4	Estructura de documents de la memòria	22
4.1	Nou projecte amb fonts existents	48
4.2	Modalitat de gestió del nou projecte	48
4.3	Configuració del Projecte	49
4.4	Carpeta de projecte separada	50
4.5	Anàlisi amb VLC del vídeo capturat	57
4.6	Anàlisi amb VLC del vídeo transcodificat al vol FLV	59
4.7	Anàlisi amb VLC del vídeo transcodificat al vol WMV8	60
4.8	Captura de pantalla de la zona gravada	63
4.9	Llista de reproducció	65
6.1	Reproducció video amb HTML5 a Chrome i Safari	81

Índex de llistats de codi

5.1	systemtest.cpp	70
5.2	popentest.cpp	71
5.3	popentestffmpeg.cpp	72
5.4	crida popen amb stderr > stdout	73
5.5	MyProcessStarter.h	75
5.6	MyProcessStarter.cpp	75
5.7	main.cpp	76
5.8	MyProcessStarter.cpp amb escriptura	77
6.1	html5video.html	81
6.2	carregar un nou video	82
6.3	Llegir directori	83
6.4	Llista de vídeos	83
6.5	Resposta al click des de la llista	84
6.6	loadVideo(path)	84
6.7	Next Video	84
D.1	myplayer.php	123
D.2	myplayer.js	124

Capítol 1

Introducció

1.1 Objectius

En el resum s'han plasmat de forma bastant detallada els objectius d'aquest projecte. Es poden resumir en una sola frase. Es desitja poder **gravar amb bona qualitat i de forma còmoda i eficient vídeo provinent una càmera IP**, i gravar també l'àudio si la càmera el proveu.

Comoditat vol dir senzillesa en la gestió de l'aplicació. Amb les eines existents sembla que es pot realitzar la tasca (a la secció 1.4 Estat de l'art s'explica com algun dels objectius només des de molt recentment), però es requereix un gran coneixement de les eines i de la interfície d'aplicació de les càmeres. L'aplicació resultant d'aquest projecte ha de permetre a una persona totalment aliena a aquestes qüestions poder gravar i revisar vídeo de manera que es pugui concentrar en les tasques per a les que en primer lloc s'ha decidit gravar el vídeo (temes de seguretat, anàlisi esportiu, ...).

A l'anàlisi dels requeriments (3.1) ho estudiarem amb més detall però com s'ha indicat al resum l'aplicació haurà de permetre engegar i aturar la gravació, i accedir (reproduir) al vídeo i àudio gravat sense necessitat d'aturar la gravació en curs. S'haurà de poder reproduir el vídeo gravat desplaçant-se amb facilitat al punt desitjat.

A més es desitja poder accedir a la gestió de la gravació i a la revisió del vídeo mitjançant un navegador web, i tenint en compte que es pugui accedir des de dispositius de pantalla tàctil.

El tema de la qualitat s'adreçarà fent servir h.264 com a còdec per emmagatzemar les gravacions. Aquesta decisió s'explicarà amb més detall a la secció Estat de l'art (1.4). L'objectiu final del projecte és però que les característiques del vídeo a emmagatzemar siguin un paràmetre, ja que depenent de l'ús que se'n vulgui fer les necessitats seran diferents.

Es desitja que l'aplicació sigui capaç de realitzar captures de llarga durada, hores, o fins i tot gravacions de durada indefinida (per exemple els sistemes de seguretat han de poder estar gravant 24x7), i que es pugui revisar la gravació sense necessitat d'aturar-la.

Com es veurà en l'anàlisi de requeriments (3.1) això comporta una dificultat afegida. En molts formats no és possible revisar el vídeo fins que l'arxiu no està tancat i, encara més, si es talla la gravació sense tancar el fitxer (per causes alienes) es pot corrompre la gravació i en casos extrems perdre's completament. A això hem d'afegir que en casos de molta durada el fitxer generat arribaria a tenir mides impossibles de gestionar.

Per això es considera adient fer la gravació en segments de curta durada. Però a l'hora de revisar el vídeo aquesta segmentació ha de ser transparent per l'usuari.

1.2 Àmbit del projecte

Tot i que als objectius es diu que les característiques del vídeo resultant han de poder-se modificar, i només es parla de càmeres IP com a font de vídeo (i l'objectiu seria que l'aplicació pogués arribar a treballar amb molt diverses marques i models de càmera), per a aquest projecte es limitaran en gran mesura aquestes qüestions per tal de que sigui realitzable en el plaç establert.

En concret només es generarà vídeo en format mp4 amb còdec h.264, i és capturarà únicament de càmeres Axis, amb firmware 4.x i 5.x.

Es considera que aquestes limitacions no minven la validesa del projecte, ja que quedarà demostrada la viabilitat de la aplicació que es desitja, i a més aquesta s'estructurarà de tal manera que permeti incorporar amb facilitat mòduls que permetin accedir a nous models de càmera, i d'altres que permetin generar sortides de vídeo diferents.

1.3 Orígens

L'any 2005 en Ferran Ferrandis¹ em va engrescar per desenvolupar amb ell un programari de gravació de vídeo provinent de càmeres IP orientat a la vídeo vigilància, el que es coneix com a CCTV. Era una època en que el mercat de vídeo vigilància estava encara basat majoritàriament en sistemes analògics, i la lluita era mostrar als responsables d'aquesta mena de sistemes

¹Ferran Ferrandis, com es pot veure en la portada d'aquesta memòria, és Director de l'Àrea TIC a Dèbid, i consultor extern d'aquest projecte

les bondats de fer servir sistemes IP —tasca complicada, tot i que aquestes bondats puguin semblar evidents—.

Panòptic, que és com varem anomenar aquest sistema, és sense cap mena de dubtes l'avi del que aquí s'està intentant desenvolupar. La recepció del vídeo des de les càmeres es feia en MJPEG, i la gravació directament en imatges JPEG, una per frame.

Aquest sistema primitiu tenia les seves raons de ser. En una instal·lació de xarxa local l'ample de banda no és rellevant, i MJPEG requeria un nivell de processament molt inferior a MPEG4. L'emmagatzemament imatge a imatge ocupava molt més espai, però permetia accedir amb facilitat a un instant concret, i gestionar la conservació del vídeo i l'esborrat del vídeo antic per deixar espai. A més en sistemes de CCTV el vídeo no s'ha de moure del gravador, només quan hi ha alguna incidència. En aquests casos *Panòptic* generava un vídeo sota demanda, generalment de pocs minuts.

Diverses vegades es va plantejar la possibilitat de desenvolupar uns mòduls que permetessin fer servir MPEG4. Però hi havia un ball de estàndards entre fabricants de càmeres, i semblava imminent l'adopció d'H.264 —que va trigar una mica més de l'esperat—. Per això es va anar posposant el desenvolupament.

A tall anecdòtic, *Panòptic* va evolucionar en *Promon*, un sistema de monitorització d'instal·lacions tipus CPD², però, com diria Ruyard Kipling, això és una altre història³.

A MEbcn, l'empresa en que treballo actualment, també estic encarregat del desenvolupament d'una eina de gravació, *Robotic Eye*⁴. En aquest cas els requeriments de gestió del vídeo són prou diferents. Es tracta de gravacions limitades però de bastanta durada (una o dues hores), generalment de més resolució que la que es fa servir en vídeo vigilància, que els analistes han de poder moure d'un equip a un altre, i han de poder treballar amb les eines d'anàlisi de vídeo que fan servir. Això fa que la relació qualitat-compressió sigui clau. L'adopció de H.264 com un estàndard de facto permet aquest treball.

El programari que fem servir actualment grava el vídeo en el format que dona la càmera, i *després* en fa la transcodificació. Això implica que l'analista no es pot endur el vídeo fins molta estona després d'acabada la sessió gravada —la relació és aproximadament 1:1, una hora de transcodificació per cada hora de vídeo gravat—.

El desenvolupament que es planteja en aquest projecte pretén servir de

²Centre de Processament de Dades

³més informació a <http://www.debid.cat/index.php/ca/tic/sistemes/promon-ca>

⁴més informació a <http://mebcn.com/what-we-do/what-we-do/solutions>

base per el desenvolupament d'un mòdul útil per ambdues branques de desenvolupament. La transcodificació al vol permetria reduir pràcticament a zero el temps d'espera dels analistes. I l'emmagatzemament codificant amb H.264 permet augmentar enormement la capacitat dels gravadors de vídeo vigilància.

1.4 Estat de l'art

1.4.1 FFmpeg

FFmpeg is the leading multimedia framework, able to decode, encode, transcode, mux, demux, stream, filter and play pretty much anything that humans and machines have created. It supports the most obscure ancient formats up to the cutting edge. No matter if they were designed by some standards committee, the community or a corporation.

Aquest text es correspon a la presentació de FFmpeg a la seva pròpia web⁵, i evidentment no es pot considerar imparcial. Ara bé, un article de The Inquirer de 31 de Juliol de 2006 diu “*The open source ffmpeg converter is awesome, supporting almost every video and audio codec on Earth*”⁶.

Però la manera més adient de convèncer-se de la potència d'aquesta eina és consultar la llista de projectes⁷ que fan servir el programari o les llibreries on es troben alguns dels projectes més forts de processament de vídeo de codi obert, com poden ser VLC⁸, Handbrake⁹ o MPlayer¹⁰. O també fer una ullada a la plana de formats i còdecs suportats¹¹, on es pot veure que si be la presentació pot semblar pretensiosa, no deu estar molt lluny de ser cert.

Molts dels formats i còdecs són suportats directament, i d'altres per llibreries externes —segons la web de FFmpeg, escollint en cada cas la millor opció de codi lliure disponible—. Per exemple, quelcom que afecta directament aquest projecte, H.264 es suporta fent servir x264¹².

⁵FFmpeg <http://ffmpeg.org/about.html>

⁶<http://www.theinquirer.net/inquirer/news/1017528/don-video-converter-trying-freebie>

⁷FFmpeg-Based Projects <http://ffmpeg.org/projects.html>

⁸<http://www.videolan.org/vlc/>

⁹<http://handbrake.fr/>

¹⁰<http://mplayerhq.hu/>

¹¹http://ffmpeg.org/general.html#Supported-File-Formats_002c-Codecs-or-Features

¹²<http://www.videolan.org/developers/x264.html>

clone wars

A principis de 2011 FFmpeg es va separar en dos projectes, FFmpeg (<http://ffmpeg.org>) i Libav (<http://libav.org>). Aquesta mena de separacions o *forks*, són relativament comuns en el món del programari lliure, i s'han de veure com una de les seves forces. Si un grup de desenvolupadors no està d'acord a la gestió d'un desenvolupament en el que participen activament tenen la possibilitat de separar-se i seguir el seu propi camí. A primera vista pot semblar que això pugui portar debilitat i escissions, però en un món tancat aquests desenvolupadors probablement també acabarien abandonant el desenvolupament. El programari lliure permet que facin pública la seva opció, i seran els altres desenvolupadors i els usuaris qui acabaran afavorint unes o altres. Aquest mecanisme també permet rescatar projectes que han caigut en l'abandonament, o que es tem que es puguin acabar deixant morir per interessos comercials.

El que fa particular la separació en aquest cas és que va produir-se primer com una mena de cop d'estat, en que els desenvolupadors descontents van intentar desfer-se del líder, però mantenint el nom, el servidor, les llistes, etc... i el *fork* només va tenir lloc quan aquesta maniobra es va veure frustrada —perquè el desenvolupador original, Fabrice Bellard, que encara posseïa els dominis, es va decantar per l'actual líder Michael Niedermayer—. A l'*about* de Libav¹³ s'explica la història de forma breu i clara, des del punt de vista dels escindits. Seguint els intercanvis de mails a les llistes del projecte original és veu que va ser una situació prou moguda.

No és objectiu d'aquest projecte, ni d'aquesta memòria, fer costat a cap dels dos bàndols. Encara que evidentment ha calgut escollir una de les dues llibreries per al desenvolupament. No es dubta que els escindits tindrien les seves raons, i l'estudi de les llistes de mail i dels logs de desenvolupament dels darrers anys semblen mostrar que l'escissió ha sigut un revulsiu també en la branca original. Ara bé, com ja s'ha comentat, una de les qualitats del programari lliure és que un desenvolupador descontent sempre pot marxar i endur-se amb ell tota la base de codi existent per continuar el desenvolupament com li plagui, i maniobres com la que va tenir lloc a FFmpeg no donen massa bona impressió.

Tot i això per escollir la branca a fer servir s'han tingut més en compte altres consideracions. En el darrer any la branca original (FFmpeg) ha estat incloent (*merge*) en el seu codi totes les millores i noves funcionalitats desenvolupades per Libav. Aquest comportament probablement no sigui sostenible a llarg plaç —generalment després d'una escissió les dues branques d'un projecte tenen tendència a anar separant-se, és el sentit de la escissió—

¹³<http://libav.org/about.html>

i de fet a data de redacció d'aquesta memòria comencen a aparèixer misatges en favor de permetre incompatibilitats, però ara per ara significa que FFmpeg dona totes les funcionalitats de Libav i algunes més.

Una qüestió afegida a ressaltar, ja que ha suposat una dificultat afegida a l'hora de realitzar les proves inicials i l'estudi de la situació, és que el responsable de mantenir els paquets de FFmpeg en les distribucions d'Ubuntu i Dèbian és un dels desenvolupadors que es van afegir a l'escissió —fins aquí tot correcte— i, en el moment d'escriure aquesta memòria, aquests paquets instal·len Libav en comptes de FFmpeg (com es pot veure a 4.2.1). Aquesta maniobra pot dur a error a usuaris i desenvolupadors, que creuen estar fent servir un programari quan en realitat n'estan fent servir un altre.

1.4.2 H.264/MPEG-4 AVC

Segons la Wikipedia¹⁴ “*H.264/MPEG-4 Part 10 or AVC (Advanced Video Coding) is a standard for video compression, and is currently one of the most commonly used formats for the recording, compression, and distribution of high definition video*”. Es tracta en realitat de dos estàndards, el ITU-T H.264 i el ISO/IEC MPEG-4 AVC que es mantenen de forma “sincronitzada”. A la mateixa plana, a l'apartat *Features*, es poden veure els avantatges que presenta quant a menor ample de banda mantenint la qualitat.

Les raons per escollir-lo per aquest projecte són, partint d'una bona qualitat de vídeo, més aviat per extensió d'ús. És un format que ha resultat escollit per diverses marques punteres de càmeres —de fet fins i tot han participat en la definició—, entre elles Axis. També sembla estar guanyant la guerra dels formats en Internet —de l'etiqueta <video> d'HTML5—¹⁵.

Ara, la raó principal per fer-lo servir en aquest projecte és la compatibilitat amb els programes d'edició de vídeo fets servir per els analistes esportius.

La principal raó per no escollir H.264 com a format de transmissió i emmagatzemament seria el fet que no es tracta de programari lliure, i que està subjecte a patents i royalties. De tota manera no hi ha altre elecció, ja que com hem comentat és el còdec que fan servir les càmeres més usades (Axis), i ja el fet d'haver de descodificar aquest vídeo per transcodificar-lo faria que el nostre programari es veies afectat per aquesta problemàtica.

A nivell d'aquest projecte, l'ús d'aquest estàndard es mantindrà a nivell de llibreries externes, que s'hauran d'instal·lar per separat —aproximació que ja fa servir FFmpeg—.

¹⁴<http://en.wikipedia.org/wiki/H.264>

¹⁵Això no sembla tan clar. La compra i posterior alliberació de les patents del còdec VP8 per part de Google és una bona empenta per webm.

1.5 Estructura d'aquesta memòria

Tot i que ens trobem davant un projecte de desenvolupament aquesta memòria s'ha orientat més cap a l'estudi de les problemàtiques associades que al detall del desenvolupament.

A la primera part, Introducció, es defineix el projecte, i s'emmarca a diversos nivells —objectiu, àmbit, motivacions— i en especial se n'estudia l'estat de l'art (1.4).

El següent capítol, Gestió del projecte, es detalla l'entorn de treball i les eines que s'han fet servir, tant per al desenvolupament del projecte com per a la redacció d'aquesta memòria. Es pretén proporcionar informació suficient per que si un lector desitja reproduir les proves i accions que es descriuen a la memòria pugui fer-ho sense dificultat. S'ha dubtat de la idoneïtat de posar aquesta informació en aquest punt o relegar-la a un annexe, ja que sense dubte trenca en certa manera el flux de la memòria. S'ha decidit, però, que el millor era presentar l'entorn abans de submergir-s'hi.

A continuació s'analitzen els requeriments del desenvolupament i se'n fa un estudi de viabilitat. Tot i que en aquest capítol (3) es presenten i analitzen les conclusions, en realitat l'estudi de viabilitat es desglossa al llarg dels següents capítols (4, 5 i 6) i constitueix el cor d'aquesta memòria.

En el capítol 4 s'analitza FFmpeg, amb especial atenció a les diverses opcions d'instal·lació, i realitza les proves necessàries per validar-ne idoneïtat. En el capítol 5 s'estudia la possibilitat de controlar `ffmpeg`, el programa de transcodificació i gravació que proporciona FFmpeg, des de dins d'una aplicació. Finalment el capítol 6 es centra en la reproducció del vídeo gravat.

Com es d'esperar en una memòria d'aquestes característiques, al darrer capítol (7) es presenten les conclusions.

1.6 Llicència del programari

Respecte al tema de la llicència, la idea inicial és alliberar el projecte sota GPLv2 (Annexe B). Això garantiria a més que la llicència és compatible amb les actuals de FFmpeg tal i com es vol compilar (veure 4.3.4), en cas que finalment el projecte inclogui codi derivat.

Capítol 2

Gestió del projecte

2.1 Introducció

En aquest capítol es detallen les eines i l'entorn de treball que s'ha fet servir per el desenvolupament d'aquest projecte. S'inclou tota la informació que es considera necessària per poder dur a terme la instal·lació de l'entorn.

En l'apartat 2.3 es detalla també l'entorn de treball que s'ha fet servir per la elaboració d'aquesta memòria, ja que es considera que pot resultar interessant.

Tot i que aquest projecte beu del Programari Lliure, i pretén retornar a la comunitat alliberant també els resultats, no es considera àmbit del projecte fer “un projecte de Programari Lliure” sinó desenvolupar una eina que permeti cobrir una necessitat amb la que l'autor s'ha trobat (que és una de les raons per les que comencen bona part del projectes de Programari Lliure, per altra banda). No es tracta doncs d'un desenvolupament en comunitat, sinó d'un desenvolupament el resultat del qual és posarà a disposició de la comunitat. És per això que, com es veurà en aquest capítol, no s'allotja en cap servei públic ni cap viver, sinó que s'ha desenvolupat íntegrament sobre recursos propis. No es descarta, però, al tancament del projecte alliberar el codi públicament allotjant el codi a [Sourceforge](#) o [Github](#).

2.2 Entorn de treball

Per circumstàncies alienes al desenvolupament del màster i a l'objectiu d'aquest projecte, l'ordinador en que s'ha desenvolupat tota la feina tant de desenvolupament com de redacció de la memòria corresponent és un iMac (OSX 10.6.8).

L'objectiu del projecte requeria fer el desenvolupament sobre Gnu/Linux.

Per a poder-ho realitzar s'han fet servir diverses màquines virtuals. Fins i tot en el cas en que la màquina hoste sigui un Gnu/Linux és recomanable fer servir màquines virtuals per aquesta mena de desenvolupaments. Aquestes màquines faciliten enormement les fases de proves, tant a l'hora de decidir quin entorn es vol fer servir com més endavant al provar el programari que s'està desenvolupant. És molt senzill clonar equips, fer instantànies del sistema quan és estable, tornar a punts anteriors si alguna cosa no va be, i permet mantenir net l'equip hoste. Evidentment en casos en que el rendiment és crític caldrà treballar directament sobre l'equip hoste, però com ja he comentat es poden fer servir màquines virtuals per testejar les eines i la seva instal·lació fins que es coneixen be.

Per a fer córrer les màquines virtuals s'ha fet servir Virtualbox. La instal·lació del gestor en Mac és totalment trivial.

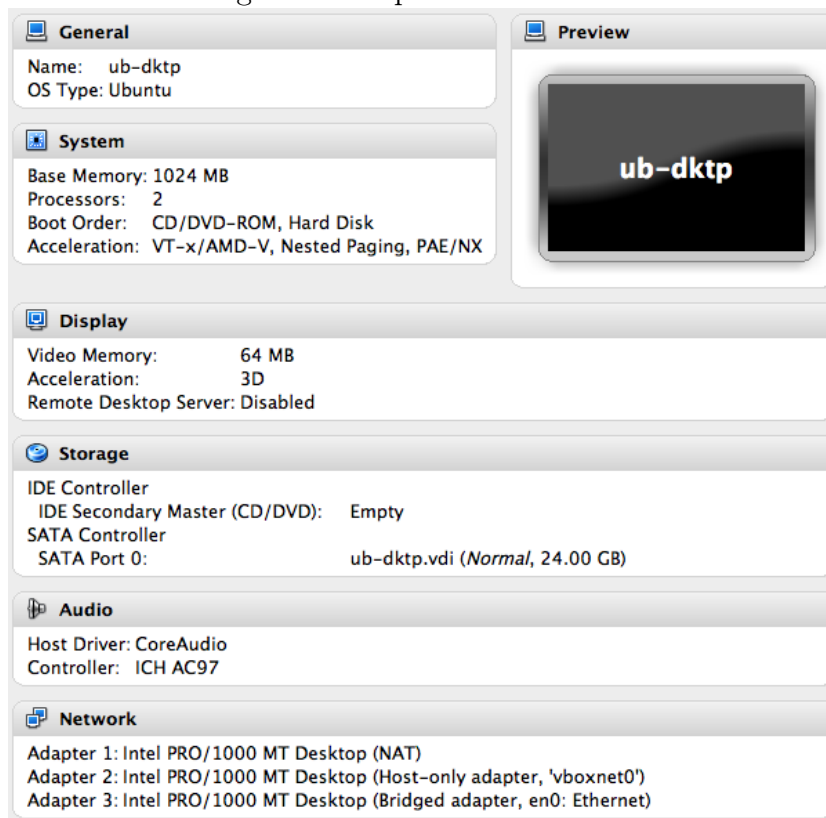
Com a entorn de desenvolupament i proves s'han fet servir dues màquines amb Ubuntu 11.10. Tal i com s'ha explicat a l'apartat sobre l'estat de l'art 1.4, l'ús de Debian o Ubuntu introdueix alguna dificultat, però tot i això s'ha mantingut aquesta decisió per comoditat personal. Com s'anirà explicant als apartats corresponents s'ha fet servir una de les màquines com a entorn de desenvolupament i l'altre per a proves complementaries. En algun cas s'han realitzat també algunes proves de compilació i funcionament sobre l'equip hoste (iMac) ja que, encara que no s'inclou en l'àmbit d'aquest projecte, un dels objectius del desenvolupament és que el programari obtingut sigui *cross-platform* o com a mínim que la portabilitat no sigui excessivament complexa.

2.2.1 Instal·lació Ubuntu sobre Virtualbox

Instal·lar Ubuntu sobre Virtualbox no té cap mena de dificultat afegida a una instal·lació normal, un cop definida la màquina virtual. Per aquest projecte s'ha fet servir Ubuntu 11.10, la versió *desktop* i386. —s'ha optat per una arquitectura de 32 bits per per minimitzar els problemes amb el programari auxiliar— i s'ha fet la instal·lació amb tots els paràmetres per defecte. El fet de ser una màquina que es dedicarà en exclusiva a aquesta funció fa innecessari definir particions més complexes. Únicament s'han deixat sense instal·lar les llibreries propietàries, més per el fet de ser llibreries multimèdia que potencialment podien afectar a les proves i al desenvolupament que no per el fet de ser propietàries.

Quant a la definició de la màquina virtual s'han fet servir els paràmetres que es veuen a la figura 2.1. La majoria d'opcions es corresponen a valors per defecte. Cal ressaltar que s'han definit dos processadors simplement perquè la màquina hoste ho permet, s'ha activat l'acceleració 3D ja que Unity la necessita, i que s'han definit tres interfícies de xarxa.

Figura 2.1: Opcions Virtualbox



La primera de les interfícies de xarxa és fa servir per donar, mitjançant NAT accés a Internet a la màquina virtual a través de la màquina hoste, per a la descàrrega de programari, l'accés als repositoris externs, i qualsevol altre tasca que requereixi accedir a Internet.

La segona, un adaptador *host-only* —es a dir només accessible des del hoste o des d'una altra màquina virtual que corri dins el mateix hoste i que també es connecti a aquesta mateixa xarxa *host-only*— és una costum personal que permet establir una mena de xarxa de gestió interna.

La tercera interfície va connectada directament a una de les targetes de xarxa de la màquina hoste, que està connectada a una xarxa interna, sense sortida a Internet. En aquesta xarxa és on es troben connectades les càmeres que s'han fet servir per a les proves d'adquisició de vídeo.

2.2.2 Llibreries i executables FFmpeg

La instal·lació de les llibreries i executables de FFmpeg, en les seves diferents modalitats i versions, es considera part del projecte i no de l'establiment de

l'entorn, per la qual cosa no s'explica aquí.

2.2.3 Entorn de desenvolupament

A l'hora de decidir-se per un entorn de desenvolupament hi ha diversos factors a tenir en compte. Depenent de la manera de treballar de cada desenvolupador i la feina a desenvolupar és bona idea fer servir un Entorn de Desenvolupament Integrat o IDE per les seves sigles en anglès. Un IDE, entre altres coses, facilita enormement la feina de conèixer un codi que no és propi, ja que permet amb molta senzillesa saltar d'un punt a altre del codi, anar a les definicions, etc...

S'ha de tenir en compte la mena de desenvolupament que es vol realitzar, en quins llenguatges, i sobre quines plataformes.

Les llibreries i el programari associat a FFmpeg estan codificats en C i C++. Si bé Java dona una portabilitat més directa, el fet que les llibreries associades a FFmpeg estiguin desenvolupades en C fa que resultin més fàcils d'integrar fent servir C o C++. Tot i que en aquesta primera versió es fa servir FFmpeg com una eina externa mitjançant crides a l'executable `ffmpeg`, l'objectiu final podria ser integrar les llibreries i no dependre de crides a línia de comandes. Per això s'ha considerat que el més adient per el programari a desenvolupar era C++.

Per la GUI associada a les proves del programari s'ha decidit fer servir les llibreries gràfiques de Qt. Permeten fer un desenvolupament senzill i eficient, estan molt ben documentades, i per arrodonir-ho son *cross-platform*.

Finalment els components web de l'aplicació —com es defineix als objectius (1.1) l'aplicació ha de ser accessible via web— es desenvoluparan fent servir PHP per al *backend* i HTML, CSS i Javascript per al *frontend*.

Tot això fa que els dos IDE lliures més extesos, Eclipse i Netbeans siguin adients.

En una primera instal·lació es va optar per Eclipse, ja que integra el desenvolupament de Qt mitjançant un *plugin*. Però al revisar el codi de FFmpeg en Eclipse apareixen gran quantitats d'errors en el codi, tot i que si es compila resulta que aquests errors no existeixen. Sembla ser que el parsejat de les llibreries en Eclipse té alguna mena de problema que fa que no tingui en compte alguna de les capçaleres.

Es va decidir provar amb Netbeans, i aquests errors no apareixien. La contrapartida és que per a fer el disseny de GUI amb Qt des de Netbeans es necessari instal·lar el QtDesigner de forma independent. De tota manera la instal·lació és senzilla, i un cop configurat és possible cridar el QtDesigner des de dins de Netbeans, sent prou semblant a com es treballa amb el *plugin* des de Eclipse.

Instal·lació

La instal·lació de Netbeans i de QtDesigner en Ubuntu són molt senzilles.

El primer pas per poder desenvolupar en condicions és tenir instal·lades les eines necessàries per a la compilació —compiladors i llibreries—. En Ubuntu s’ha d’instal·lar el paquet `build-essential`. S’hauria de verificar que s’ha instal·lat correctament.

```
# make --version
GNU Make 3.81
Copyright (C) 2006 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE.

This program built for i686-pc-linux-gnu

# gcc --version
gcc (Ubuntu/Linaro 4.6.1-9ubuntu3) 4.6.1
Copyright (C) 2011 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE.

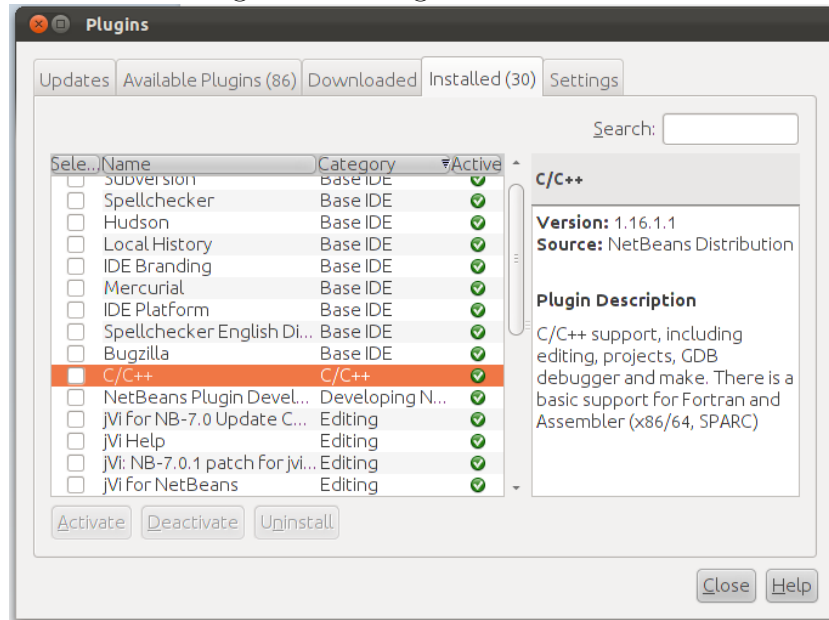
# g++ --version
g++ (Ubuntu/Linaro 4.6.1-9ubuntu3) 4.6.1
Copyright (C) 2011 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE.
```

A continuació es procedirà a instal·lar Netbeans. A la plana de descarregues¹ hi ha diverses opcions en funció de la mena de desenvolupament que és desitgi realitzar. Per exemple una per a desenvolupament en C/C++. De tota manera, tal i com s’explica a la mateixa plana de descarregues, es possible afegir els diversos components a través del *Plugin Manager* a una instal·lació prèvia. Per raons “històriques” es va instal·lar el *bundle Java SE*.

Un cop finalitzada la descàrrega s’ha de donar permís d’execució al fitxer descarregat i executar-lo.

¹<http://netbeans.org/downloads/index.html>

Figura 2.2: Plugins de Netbeans



```
# chmod +x netbeans-7.1.1-ml-javase-linux.sh
# sudo ./netbeans-7.1.1-ml-javase-linux.sh
[sudo] password for knovz:
Configuring the installer...
Searching for JVM on the system...
Extracting installation data...
Running the installer wizard...
```

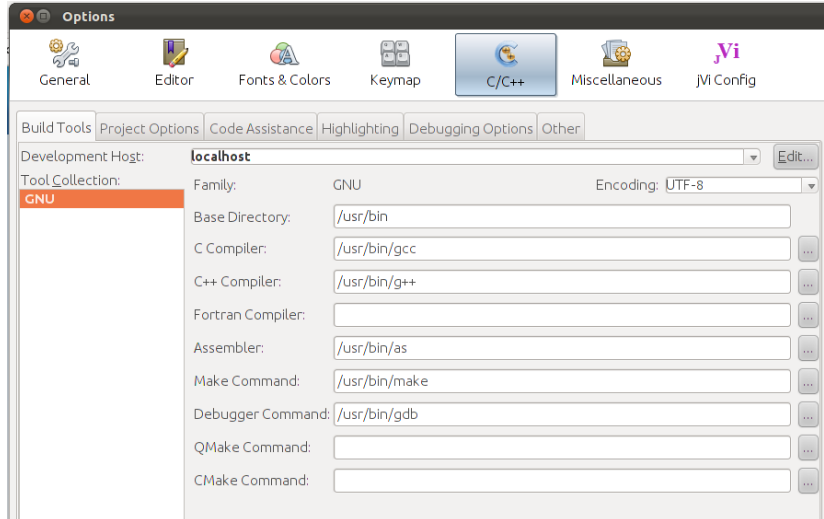
S'obrirà la finestra de l'instal·lador que guiarà l'usuari en la instal·lació.

Un cop instal·lat Netbeans, es pot accedir al *Plugin Manager* per instal·lar els components addicionals. El trobem al menú *Tools* opció *Plugins*. La pestanya *Available Plugins* llista els disponibles i a *Installed* es troben els que ja estan instal·lats. A la figura 2.2 es pot veure el *plugin* de C++ ja instal·lat.

S'ha de comprovar que els *path* de les eines de compilació estan correctament definits a *Tools* opció *Options*, tal i com es pot veure a la figura 2.3.

El pas següent és afegir les eines necessàries per a poder fer servir Qt. A la documentació de Netbeans es pot trobar una guia per desenvolupar apli-

Figura 2.3: Netbeans, opcions de C/C++



cacions amb Qt² on s'indica com configurar-lo. Per instal·lar el compilador, `qmake`, es pot fer servir el paquet adient dels repositoris.

```
# sudo apt-get install libqt4-dev
[...]
# qmake --version
QMake version 2.01a
Using Qt version 4.7.4 in /usr/lib/i386-linux-gnu
# ls /usr/bin/qmake
/usr/bin/qmake
```

Un cop instal·lat s'haurà d'indicar el path a les opcions de Netbeans, com es pot veure a la figura 2.3 —on *QMake Command* encara és en blanc—.

A continuació cal instal·lar el QtSDK. Es descarrega el paquet de la plana de descàrregues³ de la web del Qt Framework. Es recomana fer servir la opció de instal·lació en xarxa, ja que permet configurar quins components s'instal·len, reduint considerablement el volum total de la descàrrega. La plana a la que dirigeix la descàrrega inclou les instruccions d'instal·lació.

```
# chmod u+x QtSdk-online-linux-x86-v1.2.1.run
# sudo ./QtSdk-online-linux-x86-v1.2.1.run
```

²<http://netbeans.org/kb/docs/cnd/qt-applications.html>

³<http://qt.nokia.com/downloads>

S'obre la finestra de l'instal·lador. La opció *Custom* permet definir els components a instal·lar.

Un cop finalitzada la instal·lació és necessari afegir QtDesigner al PATH, per tal que Netbeans el trobi. També es pot fer un enllaç a `/usr/bin` —que ja és al PATH, evidentment—.

```
# cd /usr/bin
# sudo ln -s /opt/QtSDK/QtCreator/bin/designer designer
```

2.2.4 Ús de Git mes Dropbox per control de versions i replica entre equips

Es evident que un projecte de desenvolupament seriós ha de fer servir alguna mena de control de versions, independentment de lo reduït que pugui ser el número de desenvolupadors —fins i tot per equips unipersonals, com el d'aquest projecte—. Eines de control de versions n'hi ha moltes i de prou diverses (CVS, Git, Subversion, Bazaar, Mercurial,...) totes amb els seus defensors i detractors. Pràcticament la única cosa en que la majoria sembla estar d'acord és en que CVS està sent cada cop menys utilitzat en benefici d'altres opcions. Sense anar més lluny a la documentació de l'assignatura [1] es menciona que CVS està sent substituït per Git.

Personalment he treballat sempre amb Subversion, i tot i que n'estic prou content fa temps que volia provar un sistema descentralitzat. En treballar a la feina i a casa en un mateix projecte, o volent avançar estant de viatge s'evidencien els problemes de treballar amb un servidor centralitzat quan no s'hi té accés des de tot arreu. El fet que diversos projectes i desenvolupadors que segueixo afavoreixin Git m'hi predisposava, la lectura de la documentació de l'assignatura m'hi va reafirmar, però sense dubte la gota definitiva és el fet que el codi font del projecte FFmpeg s'hostatja en Git. Per tant el control de versions d'aquest projecte s'ha realitzat fent servir Git.

Git

S'ha optat per treballar amb un repositori base, que per convenció s'ha anomenat *origin*. Aquest repositori conté únicament la base de dades de Git, però no una còpia de treball, ja que no es pretén desenvolupar-hi codi directament, sinó fer-lo servir per clonar des de les diverses màquines en que es desitgi treballar. Es situa el repositori en un directori accessible des de totes aquestes màquines, tal i com s'explica a l'apartat Dropbox. La idea i

el procediment s'han agafat d'una pregunta a stackoverflow on es discuteix aquesta possibilitat *Using git+dropbox together effectively?*⁴.

Per generar l'entorn de treball es poden seguir les següents passes. Primer crear el que serà el directori de treball, i entrar-hi.

```
~# mkdir MPL_PFM_HCV
~# cd MPL_PFM_HCV
~/MPL_PFM_HCV#
```

El següent pas és crear contingut en aquest directori, o copiar-ne el que ja es tingui, per tal de poder començar a controlar les versions.

Un cop hi ha contingut que controlar, es crea el repositori Git en el directori, s'hi afegeix tot el contingut, i es realitza el primer `commit`.

```
~/MPL_PFM_HCV# git init
~/MPL_PFM_HCV# git add .
~/MPL_PFM_HCV# git commit -m "creacio del repositori del projecte"
```

Amb aquestes passes ja està llest el directori de treball. Per tal de fer el directori *origin* és necessari entrar a la ubicació on es desitja tenir, i crear un nou repositori. En aquest cas es farà servir la opció `bare` per tal de crear únicament la base de dades de Git, però indicant que no hi ha d'haver copia de treball local.

```
~/Dropbox# git init --bare MPL_PFM_HCV.git
```

A continuació cal indicar en la copia de treball que es vol fer servir aquesta nova còpia com una còpia remota, i que es considerarà que és la "original" (per convenció, *origin*). Finalment l'ordre `push` actualitzarà al repositori remot tot allò que hi hagi sota control de versions a la còpia de treball.

```
~/MPL_PFM_HCV# git remote add origin ~/Dropbox/MPL_PFM_HCV.git
~/MPL_PFM_HCV# git push -u origin master
```

⁴<http://stackoverflow.com/questions/1960799/using-gitdropbox-together-effectively>

`origin` i `master` són els paràmetres per defecte de `push`, amb la qual cosa cada cop que es desitgi actualitzar la còpia “origen” des de la branca principal (“`master`”) del repositori de treball n’hi haurà prou amb fer un `git push`.

Des de qualsevol altre ordinador en que vulguem treballar, i que tingui visibilitat del directori on hem creat la còpia *origin*, podem clonar el repositori tal i com es fa sempre amb Git per obtenir una còpia de treball d’un repositori remot.

```
# git clone ~/Dropbox/MPL_PFM_HCV.git
```

Dropbox

Dropbox⁵ permet tenir una carpeta o sèrie de carpetes sincronitzades entre diversos ordinadors. Quan el contingut d’aquesta carpeta es modifica en un ordinador s’actualitza en *background* en el servidor, i aquesta modificació es replica automàticament a la resta d’ordinadors. Si un ordinador és fora de línia (apagat o desconnectat d’Internet) l’actualització té lloc la següent vegada que es connecta. S’ha de tenir, però, en compte que es tracta d’una actualització asíncrona, *la rèplica no té perquè ser instantània*.

S’ha fet servir Dropbox per emmagatzemar el repositori *origin*. D’aquesta manera s’ha emulat un repositori en xarxa que pogués agregar els avenços realitzats en els diversos ordinadors (reals i virtuals) que s’han fet servir per la realització d’aquest projecte i aquesta memòria.

Per sincronitzar el contingut d’aquest repositori *origin* entre les màquines virtuals i el hoste es podria haver fet servir la capacitat de tenir-hi carpetes compartides que Virtualbox proporciona. Fer servir Dropbox, però, ha permès incloure en l’equació un ordinador portàtil sense complicar l’escenari —per exemple havent de definir carpetes de xarxa, que a més només estarien disponibles en local—.

Per suposat si es pretén posar aquest repositori *origin* a disposició de més desenvolupadors aquesta no seria ni de bon troç la millor solució, i seria recomanable fer servir servidors tipus Sourceforge o GitHub.

No s’inclou cap explicació de la instal·lació de Dropbox ja que es considera accessori, i a més és totalment trivial.

Instal·lació de Git sobre Ubuntu

Per instal·lar en Ubuntu s’ha decidit instal·lar des dels repositoris, tal i com es veu a l’apartat sobre instal·lació⁶ del llibre “Git Pro” [7]. Però s’ha optat

⁵<http://www.dropbox.com>

⁶<http://git-scm.com/book/en/Getting-Started-Installing-Git>

per instal·lar el paquet `git` en comptes del paquet `git-core` ja que aquest darrer està obsolet, tal i com es pot veure.

```
# sudo apt-cache search git | grep ^git
git - fast, scalable, distributed revision control system
git-core - fast, scalable, distributed revision control
system (obsolete)
[...]
```

Es procedeix a la instal·lació, i un cop instal·lat es verifica. La versió instal·lada per aquest procediment no és la darrera versió —1.7.9.4 al moment d'escriure's aquesta documentació—, però es considera suficient.

```
# sudo apt-get install git
[...]
```

```
# git --version
git version 1.7.5.4
```

Si es desitja poder accedir a GitHub o un altre repositori en xarxa fent servir el protocol `git://`, cal generar i registrar les claus `ssh` necessàries. S'ha de tenir en compte però que l'accés a Github també es pot fer per `https://`, indentificant-se amb usuari i contrasenya.

Instal·lació de Git sobre Mac OSX

Es pot obtenir l'instal·lable des de la plana de descarregues de Git⁷ o des de la plana de Google Code⁸. El fitxer descarregat serà el mateix, un `dmg` amb la darrera versió. S'obre el `dmg` i s'executa el `pkg` que conté. Obrint una consola es pot verificar la instal·lació.

```
# git --version
git version 1.7.9.4
```

Si es desitja treballar amb programes que en facin un ús integrat cal afegir Git al path. En aquest cas tota la feina s'ha fet a través de la consola, per la qual cosa no ha calgut. El mateix `dmg` conté els `shell` necessaris, i a <http://stackoverflow.com/a/4725627> s'explica el procés. Si es desitja poder accedir a Github o un altre repositori en xarxa fent servir el protocol `git://`,

⁷<http://git-scm.com/downloads>

⁸<http://code.google.com/p/git-osx-installer/>

cal generar i registrar les claus ssh necessàries⁹. S'ha de tenir en compte però que l'accés a Github també es pot fer per `https://`, indentificant-se amb usuari i contrasenya.

Configuració de Git

Un cop instal·lat, Git ja està llest per usar, i només demana que es configuri l'identitat del usuari. El mateix Git demana aquestes dades en realitzar el primer commit, i indica com introduir-les.

```
# git config --global user.name "Héctor Cano"
# git config --global user.email hcano@uoc.edu
```

Documentació i ajuda de Git

No és intenció d'aquesta memòria fer cap mena de bibliografia exhaustiva sobre Git, ni tan sols a nivell introductori. Si es mencionen a la bibliografia (pàg. 91) els recursos que s'han fet servir de forma intensiva, com a referència i per la resolució de dubtes, durant la elaboració d'aquesta memòria [7],[8],[9].

2.2.5 Gestió de errors i noves funcionalitats

Per aquesta tasca la documentació de l'assignatura parla de Bugzilla, i és alhora la eina que personalment coneixo millor i he fet servir amb regularitat. A vegades, però, és una mica excessiva, el que en anglès en diuen *bloated*. Sobre tot en començar un nou projecte és massa fàcil perdre's en totes les opcions de fites, versions, etc...

Si tot el desenvolupament el realitza un únic desenvolupador i hi treballa pràcticament a temps complet (com és el cas), és senzill mantenir l'enfoc només amb ajuda d'una llibreta i paper (o millor encara documents de text pla). Aquesta és la opció que s'ha pres, posposant la posada en marxa d'un gestor d'errors al moment d'obrir i publicar el desenvolupament —ja que aleshores a més es podran fer servir les eines que duen integrades tant Github com Subversion—.

2.2.6 Llistes de correu

Evidentment mentre el projecte és en mans d'un desenvolupador únic no te cap sentit crear i mantenir una llista de correu.

⁹<http://help.github.com/mac-set-up-git/>

Si s'ha considerat adient subscriure's a les diverses llistes associades a les llibreries amb que s'ha treballat, i sense cap mena de dubte són molt útils per resoldre dubtes i conèixer més a fons el funcionament i la manera de treballar-hi. En concret *ffmpeg-user.ffmpeg.org*, *libav-user.ffmpeg.org* i *ffmpeg-devel.ffmpeg.org*. A la plana corresponent de FFmpeg¹⁰ es pot trobar informació sobre aquestes llistes —i altres— i accés als arxius.

2.3 Entorn de treball per la elaboració d'aquesta memòria

La *Normativa per a la presentació dels Projectes finals del Màster en Programari Lliure* demana que per fer el lliurament de la memòria es faci servir un document digital, preferentment *Open Document Format* o PDF. En aquest cas s'ha optat per realitzar l'entrega en PDF.

Per l'elaboració de la memòria s'ha fet servir L^AT_EX. Les fonts es posen a disposició del tribunal, i es fan públiques, juntament amb el codi font del projecte i també sota llicència lliure (veure pàg. i).

2.3.1 Estructura de documents

Per simplificar la redacció, i sobre tot les revisions i possibles modificacions, de la memòria s'ha evitat fer un únic document —que hauria resultat de llargada considerable— i s'ha fraccionat la memòria en diversos documents tal i com es suggereix a [5]¹¹.

Cada capítol s'ha situat en un document separat. I en el cas dels capítols llargs s'ha fet encara un nivell més de separació, fent servir un document diferent per a cada secció.

Com es pot veure a la figura 2.4 a l'arrel de l'arbre de directoris de la memòria es pot trobar un únic document, el mestre, `pfm_hcv_master.tex`. Aquest document només conté el preamble del document L^AT_EX, i referències a cadascun dels capítols de la memòria mitjançant ordres `\input`.

En paral·lel a aquest document mestre, es troben els directoris `contents` i `figures`. Dins aquest darrer s'hi poden trobar les imatges que es fan servir en el document, i dins el primer hi trobarem els documents que es criden des del document mestre. Tal i com s'ha explicat, per els capítols més extensos es replica l'estructura, amb un directori per capítol.

¹⁰<http://ffmpeg.org/contact.html>

¹¹http://www.eng.cam.ac.uk/help/tpl/textprocessing/latex_advanced/node10.html

Figura 2.4: Estructura de documents de la memòria

```
memoria
├── pfm_hcv_master.tex
├── anexes
│   ├── ccpl-by-nc-sa.tex
│   ├── ffmpegconfigure.tex
│   ├── gplv2.tex
│   └── reproductorhtml5.tex
├── contents
│   ├── abstract.tex
│   ├── agraiments.tex
│   ├── biblio.tex
│   ├── conclusions.tex
│   ├── estudiaviabilitat.tex
│   ├── ffmpeg.tex
│   └── ffmpeg
│       ├── ffmpoggit.tex
│       ├── ffmpogppa.tex
│       ├── ffmpogtest.tex
│       └── libav.tex
├── gestioprojecte.tex
├── gestioprojecte
├── entornmemoria.tex
├── entorntreball.tex
├── introduccio.tex
├── llicencia.tex
├── process.tex
├── reproduccio.tex
├── title.tex
├── figures
│   ├── by-nc-sa.png
│   ├── ffmpegcapturetestflv.png
│   ├── ffmpegcapturetestllista.png
│   ├── ffmpegcapturetestscreecap.png
│   ├── ffmpegcapturetestvlc.png
│   ├── ffmpegcapturetestwmv.png
│   ├── logo_debid.jpg
│   ├── logo_uoc_ca.png
│   ├── netbeanscompilatoroptions.png
│   ├── netbeansplugins.png
│   ├── netbeanspluginscpp.png
│   ├── opcionsvirtualbox.png
│   ├── reproductorHtml5.png
│   ├── treememoria.tex
│   ├── treememoria.txt
│   └── netbeansffmpeg
│       ├── buildtool.png
│       ├── netbeansffmpeg.png
│       ├── newproject.png
│       ├── output.png
│       ├── projectname.png
│       └── selectmode.png
```

2.3.2 Instal·lació de L^AT_EX

La instal·lació de L^AT_EX tant en entorn GnuLinux com en entorn Mac, els dos entorns sobre els que s'ha desenvolupat aquesta memòria, és extremadament simple. S'ha optat per la distribució TeX Live¹², la release 2011, recomanada a la plana del Latex Project¹³.

Per l'edició s'ha fet servir Texmaker¹⁴. Aquesta elecció és més subjectiva, ja que hi ha diversos editors de gran qualitat, com TexStudio¹⁵, Kile¹⁶, ...

En Ubuntu

En Ubuntu existeix un paquet TeX Live als repositoris, però segons s'indica a la documentació de la comunitat¹⁷ aquest paquet està desfasat més de dos anys “*As of October 2011 the texlive package that ships with Ubuntu (TeX Live 2009) is lagging more than two years behind the current TeX Live release (TeX Live 2011)*”. Podem verificar-ho directament:

```
# sudo apt-cache showpkg texlive-full
Package: texlive-full
Versions:
2009-13 ...
```

Es decideix per tant instal·lar directament des de TeX Live. La instal·lació no és gens complicada. S'opta per fer la instal·lació sobre la xarxa, ja que permet escollir quins paquets descarregar i a més garanteix que s'instal·len els paquets més actualitzats.

A la documentació ràpida d'instal·lació¹⁸ s'indica que no es recomana executar l'instal·lable com a root, però si s'executa com a un usuari normal no és possible realitzar la instal·lació a `/usr/local` que és la ubicació per defecte. Es decideix realitzar la instal·lació fent `sudo`.

A l'apartat d'instal·lació de la documentació¹⁹ es detallen els passos per la instal·lació. En el cas que ens ocupa s'ha escollit tots els paquets a excepció del suport per a Windows (per raons obvies), però només s'ha seleccionat anglès, espanyol i francès de la multitud d'idiomes possibles. La mida de la

¹²<http://www.tug.org/texlive/>

¹³<http://www.latex-project.org/ftp.html>

¹⁴<http://www.xmlmath.net/texmaker/>

¹⁵<http://texstudio.sourceforge.net/>

¹⁶<http://kile.sourceforge.net/>

¹⁷<https://help.ubuntu.com/community/LaTeX>

¹⁸<http://www.tug.org/texlive/quickinstall.html>

¹⁹<http://www.tug.org/texlive/doc/texlive-en/texlive-en.html#installation>

descarrega és més que considerable, per la qual cosa es recomana fer proves amb diversos *mirrors* per veure el temps de resposta. En algun cas el temps estimat de descarrega era superior a les 24h, i la instal·lació final va trigar 688 minuts (quasi 12h!).

Un cop finalitzada la instal·lació es va afegir el path a `/etc/environment`:

```
# cat /etc/environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/texlive/2011/bin/i386-linux"
```

Per a instal·lar Texmaker s'ha de descarregar²⁰ el paquet adient per a la distribució i arquitectura que s'estigui fent servir²¹. és un paquet que es pot obrir directament amb el gestor de paquets d'Ubuntu i que s'instal·la de forma transparent.

En Mac

La instal·lació en Mac és planera i sense sorpreses. A MacTex²² es troba el fitxer `MacTeX.mpkg.zip` que només cal descarregar, descomprimir i instal·lar com qualsevol altre paquet similar en Mac. S'ha de tenir, però, en compte que el fitxer és prou gran (1.8GB en el moment d'escriure aquesta memòria).

Per instal·lar el Texmaker també caldrà només descarregar el paquet adient (està molt clarament indicat a la plana de *download*), en aquest cas es un `dmg` i tampoc presenta cap peculiaritat.

2.3.3 Documentació i ajuda de L^AT_EX

No és intenció d'aquesta memòria fer cap mena de bibliografia exhaustiva sobre L^AT_EX, ni tan sols a nivell introductori. Si es mencionen a la bibliografia (pàg. 91) un parell de recursos que s'han fet servir de forma intensiva, com a referència i per la resolució de dubtes, durant la elaboració d'aquesta memòria [4],[5],[6].

2.3.4 Gestió de versions

Per la elaboració de la memòria també s'ha considerat necessari realitzar una gestió de versions robusta, i, encara que per raons diferents, poder treballar des de diversos ordinadors. Per això s'ha inclòs l'arbre de directoris corresponents a la memòria dins el repositori que s'ha descrit a 2.2.4.

²⁰<http://www.xmlmath.net/texmaker/download.html>

²¹en aquest cas `texmaker_ubuntu_11.10_3.3.3_i386.deb`

²²<http://www.tug.org/mactex/2011/>

Capítol 3

Estudi viabilitat

3.1 Anàlisi de requeriments

Per realitzar l'anàlisi de requeriments partirem dels objectius que s'han indicat en la introducció (1.1). L'aplicació ha de poder “*Gravar de forma còmoda i eficient vídeo provinent d'una càmera IP*”, i ha de permetre “*reproduir el vídeo sense necessitat d'aturar la gravació en curs*”. Es demana també que és pugui accedir des d'un navegador.

Ràpidament es veu que només hi ha un actor extern al sistema, l'usuari, si bé algun enfoc podria considerar també les càmeres —proveïdors de vídeo—.

3.1.1 Casos d'ús

A primera vista es troben dos casos d'ús prou senzills, gravació i reproducció del vídeo gravat. Un estudi una mica més detallat complica una mica les coses. La reproducció de vídeo pot ser a posteriori, gravació finalitzada, o en el transcurs de la gravació —sense aturar-la—. I els requeriments demanen que el vídeo pugui ser reproduït via web. Per tant cadascun dels casos tindrà dues versions, directament i a través de web. En el cas de la gravació s'ha de tenir en compte que el sistema pot tenir diverses fonts de vídeo —la flexibilitat és la principal avantatge de treballar amb càmeres IP— i caldrà escollir de quina es vol gravar.

Gravació

1. Usuari selecciona càmera
2. Usuari inicia la gravació
3. Usuari atura la gravació

Entre l'inici de la gravació i l'aturada pot passar un temps indefinit.

Gravació des de navegador web

Aquest cas és idèntic a l'anterior, però la gestió serà per una plana web.

Reproducció amb la gravació finalitzada

Aquest cas no es contemplarà en el desenvolupament de forma específica.

Els analistes esportius no faran servir l'aplicació per reproduir el vídeo, sinó que disposen d'eines avançades per analitzar el vídeo i produir informes¹.

Altres usuaris disposen de multitud d'eines que permeten reproduir vídeo, i es pot assumir amb seguretat que en són usuaris d'alguna. No es descarta com a desenvolupament futur afegir un reproductor integrat a l'aplicació.

Reproducció amb la gravació en marxa

1. Usuari selecciona càmera
2. Usuari inicia la gravació
3. Usuari accedeix al vídeo gravat
 - (a) inici de la reproducció
 - (b) desplaçament per la reproducció
 - (c) finalització de la reproducció
4. Usuari atura la gravació

Aquest cas cal contemplar-lo, tot i que l'accés a la gravació es faci amb eines alienes a aquest desenvolupament, ja que hi ha formats de vídeo que no permeten començar a reproduir el vídeo fins que no s'ha tancat l'arxiu —i per tant s'ha finalitzat o interromput la gravació—. Per a poder complir aquest cas d'ús caldrà segmentar la gravació, i anar generant una llista de gravació que inclogui tots els vídeos que la formen. Aquesta llista permetrà accedir al vídeo que ja s'ha gravat sense afectar a la gravació en curs.

Reproducció amb la gravació en marxa des de navegador web

Aquest cas és idèntic a l'anterior, però la interfície serà una plana web. Cal tenir en compte que la llista de reproducció no és tancada, i s'ha d'anar actualitzant. Aquest cas és el que pot permetre als analistes esportius fer el que s'anomena *instant review*, des del mateix camp mitjançant un equip portàtil connectat en xarxa al sistema de gravació.

¹Si n'esteu interessats a la web de MEbcn <http://www.mebcn.com> podeu trobar més informació. L'eina número ú a nivell mundial en aquests moments és SportsCode http://www.sportstec.com/Products_Sportscode

Reproducció amb la gravació finalitzada des de navegador web

Aquest cas queda inclòs en l'anterior, excepte que cal afegir la possibilitat de escollir entre les gravacions ja realitzades. Aquest cas es deixarà per a futurs desenvolupaments, ja que no és crític.

3.1.2 Requisits funcionals

Per complir els casos d'ús que s'han elaborat i dels requeriments de que s'ha partit l'aplicació a desenvolupar haurà de poder fer les següents tasques.

1. Permetre escollir orígens de vídeo
2. Engregar gravació sota demanda del usuari
3. Aturar gravació sota demanda del usuari
4. Segmentar el vídeo que es grava per permetre l'accés a la gravació sense aturar-la
5. Generar una llista de reproducció a mesura que es fan segments de gravació
6. Transcodificar el vídeo "al vol" generant la sortida en el format desitjat independentment del format de l'entrada
7. Donar accés a la gravació en curs mitjançant una plana web accessible remotament.

Els següents requisits es deixen per a properes versions.

9. Reproduir vídeo (gravació en curs i anteriors) des de l'aplicació
10. Reproduir gravacions anteriors des de la interfície web
11. Gestionar la gravació (engegar, aturar) des de la interfície web

3.1.3 Requisits no funcionals

En els objectius es mencionen dos requisits no funcionals que són claus per a l'aplicació. Senzillesa i qualitat.

Qualitat

El vídeo generat per l'aplicació ha de ser de bona qualitat. Això ho donarà el format i els còdecs que es facin servir, juntament amb una bona parametrització d'aquests. Es farà servir H.264 o MPEG4 sobre mp4. Els paràmetres específics (mida de GOP, resolució, imatges per segon,...) s'ajustaran en funció de l'entrada de vídeo (de la càmera) per tal d'obtenir la millor qualitat possible amb una mida raonable de vídeo. S'ha de tenir en compte que l'ús per anàlisi esportiu requereix poder treballar amb pausa i reproducció a velocitat reduïda, per la qual cosa el GOP haurà de ser més aviat curt.

Senzillesa

L'aplicació està pensada per facilitar la captura de vídeo per a gent que no està familiaritzada amb codificadors i contenidors de vídeo, ni temes de xarxa. En general gent sense un coneixement tècnic especial.

Les opcions han de ser per tant fàcils, entenedores, i presentar al usuari el mínim d'opcions possibles. Hi ha d'haver botons per engegar i aturar la gravació, i selectors per als orígens de vídeo —on l'usuari ha de poder escollir les càmeres pel seu nom, sense complicacions afegides— i per al format de sortida —on es presentaran a l'usuari opcions que li puguin ser entenedores, com ara els dispositius de reproducció o el programari d'anàlisi, sense parlar de còdecs ni contenidors—.

3.2 Anàlisi de viabilitat

Per analitzar la viabilitat del desenvolupament que es demana s'haurà d'analitzar dues vessants. Primer si és factible en el sentit de tècnicament possible. I segon si és realitzable en el sentit econòmic o temporal, que és possible complir els requeriments en l'abast d'aquest projecte.

La primera vessant la podem dividir en dos. Per una banda la gravació, i per l'altre la reproducció.

En el capítol 6 d'aquesta memòria s'analitzarà amb detall tota la problemàtica associada a la reproducció, principalment reproduir vídeo segmentat de forma contínua. Es veurà també que depenent dels còdecs fets servir apareixen altres problemes, sobretot si es desitja accedir al vídeo a través d'un navegador web.

En el capítol 4 s'analitzarà la possibilitat de gravar de forma segmentada i transcodificant al vol fent servir FFmpeg. Es veurà que és una eina complexa però molt versàtil, i que compleix els requeriments. L'ús d'aquesta eina obrirà una nova problemàtica, gestionar un programari existent, una eina de línia de comandes, des de dins una aplicació. I a més fer-ho de manera interactiva. Aquesta nova problemàtica s'analitzarà al capítol 5.

Al llarg d'aquesta memòria es veurà que el projecte és tècnicament viable.

Per enfocar adientment la segona vessant, la temporal/econòmica, caldria un estudi molt més profund dels requeriments complets de l'aplicació. Entre altres, models de càmera que es desitja poder gravar en la primera versió, tipus de sortides de vídeo necessàries, opcions de gestió d'usuaris, seguretat, robustesa, requeriments d'instal·lació i desplegament,... Un desenvolupament complet d'aquestes característiques queda fora de l'abast d'aquest projecte.

Si és viable, però, fer un prototip, i aquest serà l'objectiu d'aquest pro-

jecte: fer l'estudi tècnic i realitzar un prototip funcional. Aquest prototip complirà els requeriments analitzats a 3.1, amb les següents limitacions: gravarà únicament de les càmeres de prova (Axis 210, Axis Q6034E) i generarà vídeo en mp4 amb MPEG4 o H.264 amb la mateixa resolució i imatges per segon que generi la càmera.

3.3 Proposta de desenvolupament

Es proposa per tant, en base al que s'ha analitzat en els dos punts anteriors, el desenvolupament d'un esquelet bàsic d'aplicació que permeti demostrar la viabilitat tècnica, que resolgui les problemàtiques que s'han expressat. Al llarg d'aquesta memòria, i com a part integral i necessària del projecte es durà a terme l'estudi de les solucions per a cadascun dels requisits, tal i com s'ha dit en el punt anterior, i es verificarà que es poden complir o es buscaran alternatives.

Es deixa per futurs desenvolupaments el completar el programari, amb un bon disseny de interfície d'usuari, ampliant models de càmeres, formats de sortida, i afegint noves possibilitats de reproducció de vídeo més elaborades que es puguin integrar.

El programari que es desenvolupa consta de dos mòduls independents. Gravació i reproducció.

3.3.1 Gravació

Es desenvoluparà una petita aplicació basada en el *Qt Framework* que permetrà gravar vídeo provinent d'una càmera IP. Aquesta aplicació ha de fer les funcions d'interfície gràfica simplificada per a l'ús de FFmpeg. La gravació la durà a terme FFmpeg.

La simplicitat de l'aplicació no és deguda a la brevetat del temps de desenvolupament, sinó que és un requisit i qualsevol desenvolupament present o futur l'ha de mantenir. Els potencials usuaris de l'aplicació són persones sense formació tècnica, i han de poder sentir-se còmodes.

L'ús de l'aplicació es separa en dues fases:

Configuració de la gravació

- Selecció origen de gravació.
- Selecció format/qualitat del vídeo.

Aquesta part s'ha de resoldre amb selectors o similar. No es pot demanar cap mena d'entrada escrita al usuari, especialment en el que es correspon

als paràmetres del vídeo a produir. L'aplicació haurà de generar la cadena de paràmetres per a `ffmpeg` a partir de les seleccions de l'usuari, de forma totalment transparent a l'usuari. Als requeriments no s'ha parlat de donar opcions diferents de sortida de vídeo, i de fet en tot moment s'ha parlat d'un format específic, però donades les característiques de l'aplicació seria un error no tenir en compte des del principi la possibilitat de variar el format de la sortida.

Gravació

- Engegar la gravació
- Aturar la gravació

Aquesta part s'implementarà amb un o dos botons. L'estat de l'aplicació, gravant o en repòs, ha de resultar evident. Es proporcionarà a l'usuari indicadors visuals de que la gravació està anant sense problemes (temps transcorregut, espai en disc i/o similar).

3.3.2 Reproducció

El gravador depositarà les gravacions en una ubicació predeterminada. En paral·lel a la gravació generarà llistes de reproducció que permetin accedir al vídeo.

La reproducció del vídeo per part de l'usuari dependrà de l'ús que se'n vulgui donar.

La reproducció a posteriori es realitzarà fent servir eines ja existents (reproductors de vídeo com VLC, QuickTime, WindowsMediaPlayer, ... o eines d'anàlisi). L'aplicació no aporta res en aquest sentit.

La reproducció en temps real (*Instant Video Feedback*) ha de ser accessible amb un navegador web. Es desenvoluparà per tant un petit mòdul de reproducció basat en `html5` i pensat per ser accedit amb un navegador web —simplificant l'accés des de dispositius portàtils—. Aquest mòdul ha de permetre veure les gravacions existents, accedir a la que es desitgi, reproduir-la i moure's amb facilitat per el vídeo.

Es deixa per futurs desenvolupaments completar l'aplicació web que hostatgi aquest reproductor.

Capítol 4

Instal·lació, compil·lació i proves de funcionament amb FFmpeg

4.1 Opcions d'instal·lació

Com amb la gran majoria de programari lliure hi ha diverses maneres d'instal·lar FFmpeg. Existeix l'opció de compilar per un mateix la darrera versió del codi, i també l'opció de buscar paquets precompilats. En Ubuntu, com tots els usuaris saben, hi ha uns repositoris predefinitos que permeten accedir a versions estables i provades del programari. Si no es busquen característiques tan noves que només es trobin a les darreres versions, o es desitja personalitzar la compilació per unes circumstàncies particulars, és bona idea fer servir aquests paquets precompilats, i sobre tot és extremadament senzill.

En el cas de FFmpeg sobre Ubuntu l'usuari es troba, com s'ha explicat a *clone wars* (1.4.1), amb una situació prou peculiar. El programari que instal·la el paquet `ffmpeg` és un de diferent!

De tota manera és possible afegir un repositori PPA per fer-ne la instal·lació, a l'apartat FFmpeg (4.2.2) de la secció Instal·lació a partir de paquets d'Ubuntu s'explicarà el procediment.

Per la naturalesa d'aquest projecte, però, s'ha optat per fer la instal·lació compilant des del codi font. A la secció Obtenció i compilació (4.3) es detalla el procediment.

4.2 Instal·lació a partir de paquets d'Ubuntu

4.2.1 libav

Tal i com s'ha comentat, si s'opta per instal·lar el paquet `ffmpeg` que ve als repositoris de Ubuntu (des de *Natty Narval*), l'usuari es trobarà amb que el que s'instal·la és el *fork* corresponent a Libav¹. Això es pot verificar inspeccionant el paquet.

```
# apt-cache show ffmpeg
Package: ffmpeg
[...]
Source: libav
Version: 4:0.8.1-0ubuntu1
Depends: libav-tools
Filename: pool/main/liba/libav/ffmpeg_0.8.1-0ubuntu1_all.deb
[...]
Description-en: Multimedia player, server, encoder and
transcoder (transitional package)
  Libav is a complete, cross-platform solution to decode,
  encode, record, convert and stream audio and video.
.
This package is only used for transitional purposes and can be
safely removed when no other packages depend on this package.
Homepage: http://libav.org/
[...]
```

Si s'instal·la:

```
# sudo apt-get install ffmpeg
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libav-tools libavcodec53 libavdevice53 libavfilter2
  libavformat53 libavutil51 libdc1394-22 libgsm1 libpostproc52
  libschroedinger-1.0-0 libswscale2 libva1 libvpx1
The following NEW packages will be installed:
  ffmpeg libav-tools libavcodec53 libavdevice53 libavfilter2
  libavformat53 libavutil51 libdc1394-22 libgsm1 libpostproc52
  libschroedinger-1.0-0 libswscale2 libva1 libvpx1
[...]
```

¹<http://libav.org>

i es prova:

```
# ffmpeg
ffmpeg version 0.8.1-4:0.8.1-0ubuntu1, Copyright (c) 2000-2011
the Libav developers built on Mar 22 2012 05:29:10 with gcc 4.6.3
This program is not developed anymore and is only provided for
compatibility. Use avconv instead (see Changelog for the list of
incompatible changes).
Hyper fast Audio and Video encoder
usage: ffmpeg [options] [[infile options] -i infile]... {[outfile
options] outfile}...

Use -h to get full help or, even better, run 'man ffmpeg'
# avconv
avconv version 0.8.1-4:0.8.1-0ubuntu1, Copyright (c) 2000-2011
the Libav developers built on Mar 22 2012 05:29:10 with gcc 4.6.3
Hyper fast Audio and Video encoder
usage: avconv [options] [[infile options] -i infile]... {[outfile
options] outfile}...

Use -h to get full help or, even better, run 'man avconv'
```

es veu, finalment a les clares, que el que s'ha instal·lat és `avconv`, l'alternativa a `ffmpeg` de Libav. Ara be, es diu que `ffmpeg` ja no es desenvolupa...

Es procedeix a desinstal·lar Libav completament. És necessari invocar `autoremove` per que les llibreries instal·lades com a dependències son part del projecte i són diferents en els dos *forks*.

```
# sudo apt-get purge ffmpeg
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no
longer required:
  libswscale2 libavutil51 libdc1394-22 libav-tools libpostproc52
  libavformat53 libschrödinger-1.0-0 libvpx1 libavfilter2
  libgsm1 libavcodec53 libva1 libavdevice53
Use 'apt-get autoremove' to remove them.
The following packages will be REMOVED:
  ffmpeg*
[...]
# sudo apt-get autoremove
Reading package lists... Done
```

```

Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  libav-tools libavcodec53 libavdevice53 libavfilter2 libavformat53
  libavutil51 libdc1394-22 libgsm1 libpostproc52
  libschroedinger-1.0-0 libswscale2 libva1 libvpx1
[...]

```

Com a darrera consideració en aquest punt, el paquet que permet instal·lar directament la versió corresponent a Libav és `libav-tools`.

```

# apt-cache show libav-tools
Package: libav-tools
[...]
Source: libav
Version: 4:0.8.1-0ubuntu1
Replaces: ffmpeg (<< 4:0.8~), libavcodec-extra-53 (<< 4:0.6~), lib
avcodec53 (<< 4:0.8.1-0ubuntu1)
[...]
Conflicts: ffprobe
Breaks: ffmpeg (<< 4:0.8~)
Filename: pool/main/liba/libav/libav-tools_0.8.1-0ubuntu1_i386.deb
[...]
Description-en: Multimedia player, server, encoder and transcoder
  Libav is a complete, cross-platform solution to decode, encode,
  record, convert and stream audio and video.
.
  This package contains the avplay multimedia player, the avserver
  streaming server, the avconv audio and video encoder, and the
  avprobe stream analyzer. They support most existing file formats
  (AVI, MPEG, OGG, Matroska, ASF...) and encoding formats (MPEG,
  DivX, MPEG4, AC3, DV...).
Homepage: http://libav.org/
[...]

```

4.2.2 FFmpeg

A la plana de descàrregues de FFmpeg², a l'apartat de *FFmpeg Linux Builds*, s'indica que per instal·lar FFmpeg a Ubuntu, per les diferents versions i

²<http://ffmpeg.org/download.html>

arquitectures, es pot fer servir un repositori de tipus PPA (*Personal Package Archives*), mantingut per Jon Severinsson³.

A la plana de presentació del repositori s'explica breument la situació de Libav i FFmpeg, i es poden trobar les instruccions necessàries per afegir el repositori PPA a un Ubuntu i poder procedir a instal·lar FFmpeg “*[the] *actual* FFmpeg, from ffmpeg.org*”. Primer s'ha d'afegir el repositori.

```
# sudo add-apt-repository ppa:jon-severinsson/ffmpeg
You are about to add the following PPA to your system:
  Updated FFmpeg packages, including dependencies.
[...]
```

A continuació s'ha d'actualitzar la llista de programari disponible als repositoris configurats, per tal que sigui efectiu el nou repositori, i un cop actualitzat es pot verificar com ara apareixen dos paquets ffmpeg.

```
# sudo apt-get update
# apt-cache show ffmpeg
Package: ffmpeg
[...]
Version: 4:0.10.2-0ubuntu0jon1
Replaces: libav-tools (<< 4:0.10~), libavcodec-extra-52 (<< 4:0.6~
), libavcodec52 (<< 4:0.6-2~)
[...]
Conflicts: ffmpeg
Breaks: libav-tools (<< 4:0.10~)
Filename: pool/main/f/ffmpeg/ffmpeg_0.10.2-0ubuntu0jon1_i386.deb
[...]
Description: Multimedia player, server, encoder and transcoder
  FFmpeg is a complete, cross-platform solution to decode, encode,
  record, convert and stream audio and video.
.
This package contains the ffmpeg multimedia player, the ffmpeg
streaming server, the ffmpeg audio and video encoder, and the
ffmpeg stream analyzer. They support most existing file formats
(AVI, MPEG, OGG, Matroska, ASF...) and encoding formats (MPEG,
DivX, MPEG4, AC3, DV...).
Original-Maintainer: Debian Multimedia Maintainers <pkg-multimedia
-maintainers@lists.alioth.debian.org>
```

³<https://launchpad.net/~jon-severinsson/+archive/ffmpeg>

```

Package: ffmpeg
[...]
Source: libav
Version: 4:0.8.1-0ubuntu1
Depends: libav-tools
Filename: pool/main/liba/libav/ffmpeg_0.8.1-0ubuntu1_all.deb
[...]
Description-en: Multimedia player, server, encoder and transcoder
(transitional package)
  Libav is a complete, cross-platform solution to decode, encode,
  record, convert and stream audio and video.
.
  This package is only used for transitional purposes and can be
  safely removed when no other packages depend on this package.
Homepage: http://libav.org/
[...]

```

Es veu ràpidament que el primer (4:0.10.2-0ubuntu0jon1) es correspon, per contingut, al libav-tools que s'ha vist abans. Ara es pot instal·lar FFmpeg de forma convencional.

```

# sudo apt-get install ffmpeg
[sudo] password for knovz:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libavcodec53 libavdevice53 libavfilter2 libavformat53 libavutil51
  libdc1394-22 libgsm1 libpostproc52 libschrödinger-1.0-0
  libswresample0 libswscale2 libva1 libvo-aacenc0
  libvo-amrwbenc0 libvpx1
The following NEW packages will be installed:
  ffmpeg libavcodec53 libavdevice53 libavfilter2 libavformat53
  libavutil51 libdc1394-22 libgsm1 libpostproc52
  libschrödinger-1.0-0 libswresample0 libswscale2 libva1
  libvo-aacenc0 libvo-amrwbenc0 libvpx1
[...]

```

I es pot provar, verificant que es té instal·lada la versió que es desitjava.

```

# ffmpeg -version
ffmpeg version 0.10.2-4:0.10.2-0ubuntu0jon1

```

```

built on Mar 18 2012 10:01:10 with gcc 4.6.3
configuration: --extra-version='4:0.10.2-0ubuntu0jon1' --arch=i386
--prefix=/usr --libdir=/usr/lib/i386-linux-gnu --disable-stripping
--enable-vidpau --enable-bzlib --enable-libgsm --enable-libschröedinger
--enable-libspeex --enable-libtheora --enable-libvorbis --enable-
pthread --enable-zlib --enable-libvpx --enable-runtime-cpudetect
--enable-libfreetype --enable-vaapi --enable-frei0r --enable-gpl
--enable-postproc --enable-x11grab --enable-librtmp --enable-libvo-aacenc
--enable-version3 --enable-libvo-amrwbenc --enable-version3
--enable-libdc1394 --shlibdir=/usr/lib/i386-linux-gnu --enable-shared
--disable-static
libavutil      51. 35.100 / 51. 35.100
libavcodec     53. 61.100 / 53. 61.100
libavformat    53. 32.100 / 53. 32.100
libavdevice    53.  4.100 / 53.  4.100
libavfilter     2. 61.100 /  2. 61.100
libswscale     2.  1.100 /  2.  1.100
libswresample  0.  6.100 /  0.  6.100
libpostproc   52.  0.100 / 52.  0.100

```

Aquesta versió de FFmpeg és completament funcional, i es podria fer servir per les proves de viabilitat, i per suposat per a qualsevol ús a nivell d'usuari. La versió instal·lada (0.10.2) és pràcticament la darrera *release* publicada a la data de redacció d'aquesta memòria⁴.

4.3 Obtenció i compilació de la darrera versió estable

Donat que aquesta llibreria és el centre d'aquest projecte, i que aquest es un projecte de desenvolupament, té sentit fer la instal·lació compilant directament el codi font. Això comporta tres avantatges clars respecte instal·lar paquets precompilats. El primer, que garanteix que s'estarà treballant amb la darrera versió disponible, amb les correccions de *bugs* i l'afegit de funcionalitats que pugui haver. El segon, que permet personalitzar la compilació, afegint o no les opcions que es desitgi. El tercer, poder la més important en quant a aquest projecte, és l'accés al codi font.

L'accés al codi font permet estudiar com es realitzen les diverses tasques, verificar-ne el funcionament, i fins i tot modificar el que es consideri escaient. En el cas específic de FFmpeg hi ha un avantatge addicional, per a tothom

⁴En realitat durant el temps de redacció d'aquesta memòria s'ha arribat la 0.11.1

que desitgi fer servir les llibreries de codificació i descodificació que conté, és pot accedir i estudiar el codi font dels executables que acompanyen aquestes llibreries. Aquests executables, a més de ser unes eines de gran potència, són uns casos d'estudi i exemples d'ús de primer nivell.

En cas que les raons expressades no fossin suficient, n'hi ha una d'afegida. En cas de tenir dubtes o problemes amb FFMpeg es pot acudir a les llistes de correu del projecte. Aquestes llistes són molt actives, i proporcionen molta ajuda. Ara bé, la condició indispensable és que es treballi amb la darrera versió, ja que, com ja s'ha mencionat, és la que inclou totes les correccions actualitzades i la única amb que es pot estar segur que el problema no es correspon a un *bug* que ja està resolt.

Per aquest apartat s'ha fet servir com a guia *Compile FFmpeg on Ubuntu* [3]⁵, si be, com es veurà als diversos apartats, s'ha anat personalitzant en funció de l'ús que es desitja.

4.3.1 Abans d'instal·lar

Es recomanable, i no només en aquest cas, tenir el programari de l'ordinador en que es vol realitzar la tasca completament actualitzat.

Neteja

Per evitar problemes d'incompatibilitat, i garantir que es crida i s'executa la versió desitjada del programari que s'està instal·lant, és adient començar per desinstal·lar els paquets que ja pugui haver instal·lats corresponents al programari a instal·lar. Es decideix fer servir `autoremove`, per que es desinstal·lin també els paquets que poguessin estar instal·lats com a dependències (veure els apartats corresponents a la instal·lació a partir de paquets. Es fa servir la opció `-purge` per que s'esborri també la configuració associada.

```
# sudo apt-get autoremove --purge ffmpeg
```

A la guia *Compile FFmpeg on Ubuntu* [3], es desinstal·len també `x264` i `libx264-dev`. En el ordinador que s'ha fet servir aquests paquets no estaven instal·lats.

Dependències

A la guia [3], es presenta un llistat de dependències amb la ordre necessària per instal·lar-les totes amb un simple copiar i enganxar. Si bé és una molt

⁵<https://ffmpeg.org/trac/ffmpeg/wiki/UbuntuCompilationGuide>

bona opció si el que es desitja és tenir FFmpeg instal·lat i usable el més ràpid possible, en aquest cas el que es desitja és entendre el funcionament i les dependències. Per això es considera que el millor és començar amb el mínim necessari, i afegir elements si es necessiten.

S'ha reduït considerablement la llista. Diverses de les llibreries que no s'inclouen es corresponen a còdecs d'àudio i vídeo que es consideren necessaris en una instal·lació normal però en aquest cas només s'han inclòs aquelles que s'ha decidit fer servir en el desenvolupament i les proves.

Git ja és instal·lat. Si no és el cas, cal afegir-lo en aquest pas.

Per el mateix desig d'aprenentatge es recomana no fer servir l'opció `-y`, ja que és més interessant veure quines preguntes apareixen, i tenir opció d'estudiar quins paquets s'instal·len com a dependències.

S'instal·len els `build-essential` (si no són ja instal·lats), ja que inclouen els compiladors i les llibreries necessàries per poder realitzar la compilació. `yasm`⁶ ja que es necessita també per la compilació de FFmpeg. `libSDL1.2-dev` només és necessària si es vol poder compilar FFplay, la resta de components de FFmpeg no fan servir aquesta llibreria —FFplay es l'únic component amb GUI—. `libx11-dev` i `libxfixes-dev` són útils si es vol compilar FFmpeg amb la opció `-enable-x11grab` que, com el seu nom indica, permet capturar d'X11 com a origen de vídeo.

Finalment s'ha inclòs `checkinstall`, ja que a la guia [3] es fa servir en comptes de `make install` alhora de fer la instal·lació del programari que es compila. Com es pot veure a la seva plana web⁷, `checkinstall` executa `make install`, però a més monitoritza els fitxers que genera/modifica la instal·lació. Un cop ha fet això, genera un paquet de la mena que faci servir la distribució en que s'està instal·lant —en el cas que ens ocupa, un `deb`—, l'instal·la, i en deixa una còpia al directori en que es troba el codi font. Això simplifica la desinstal·lació, que es pot fer com si s'hagués instal·lat des de els repositoris. I també permet fer servir aquest paquet que s'ha generat per instal·lar en altres equips, sense necessitat de recompilar en cadascun.

```
# sudo apt-get update
# sudo apt-get install build-essential yasm \
  libSDL1.2-dev \
  libx11-dev libxfixes-dev \
  checkinstall
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

⁶<http://yasm.tortall.net/>

⁷<http://asic-linux.com.mx/~izto/checkinstall/>

```

The following extra packages will be installed:
 dpkg-dev g++ g++-4.6 libalgorithm-diff-perl
 libalgorithm-diff-xs-perl libalgorithm-merge-perl libasound2-dev
 libavahi-client-dev libavahi-common-dev libcaca-dev
 libdbus-1-dev libdpkg-perl libdrm-dev libgl1-mesa-dev
 libglu1-mesa-dev libice-dev libkms1
 libncurses5-dev libpcre3-dev libpcrecpp0 libpng12-dev
 libpthread-stubs0 libpthread-stubs0-dev libpulse-dev
 libslang2-dev libsm-dev libstdc++6-4.6-dev libtimedate-perl
 libtinfo-dev libx11-doc libxau-dev libxcb1-dev libxdmcp-dev
 libxext-dev libxt-dev mesa-common-dev x11proto-core-dev
 x11proto-fixes-dev x11proto-input-dev x11proto-kb-dev
 x11proto-xext-dev xorg-sgml-doctools xtrans-dev zlib1g-dev
Suggested packages:
 gettext debian-keyring g++-multilib g++-4.6-multilib gcc-4.6-doc
 libstdc++6-4.6-dbg libasound2-doc libglu1-mesa-doc ncurses-doc
 libstdc++6-4.6-doc libxcb-doc
The following NEW packages will be installed:
 build-essential checkinstall dpkg-dev g++ g++-4.6
 libalgorithm-diff-perl libalgorithm-diff-xs-perl
 libalgorithm-merge-perl libasound2-dev libavahi-client-dev
 libavahi-common-dev libcaca-dev libdbus-1-dev libdpkg-perl
 libdrm-dev libgl1-mesa-dev libglu1-mesa-dev
 libice-dev libkms1 libncurses5-dev libpcre3-dev libpcrecpp0
 libpng12-dev libpthread-stubs0 libpthread-stubs0-dev libpulse-dev
 libSDL1.2-dev libslang2-dev libsm-dev libstdc++6-4.6-dev
 libtimedate-perl libtinfo-dev libx11-dev libx11-doc libxau-dev
 libxcb1-dev libxdmcp-dev libxext-dev libxfixes-dev libxt-dev
 mesa-common-dev x11proto-core-dev x11proto-fixes-dev
 x11proto-input-dev x11proto-kb-dev x11proto-xext-dev
 xorg-sgml-doctools xtrans-dev yasm zlib1g-dev
0 upgraded, 51 newly installed, 0 to remove and 0 not upgraded.
Need to get 20.9 MB of archives.
After this operation, 70.2 MB of additional disk space will be used

```

4.3.2 x264

Per la codificació amb H.264 FFmpeg fa servir x264. Es tracta d'un projecte de programari lliure hostatjat per Videolan⁸, fet servir per VLC, MEncoder i Handbrake — a més de FFmpeg— entre d'altres.

⁸<http://www.videolan.org/developers/x264.html>

x264 es pot instal·lar a partir dels repositoris d'Ubuntu. En Ubuntu 12.04 això proporciona la versió 0.120.2151 a3f4407. També es pot instal·lar compilant des del codi font, tal i com indica la guia *Compile FFmpeg on Ubuntu* [3], i això és el que s'ha optat per fer.

Per organitzar s'ha decidit crear un directori *sources* per emmagatzemar les diferents fonts a dins del directori *home*.

```
# cd sources/
# git clone git://git.videolan.org/x264
Cloning into 'x264'...
remote: Counting objects: 15575, done.
remote: Compressing objects: 100% (2836/2836), done.
remote: Total 15575 (delta 12756), reused 15481 (delta 12698)
Receiving objects: 100% (15575/15575), 3.85 MiB | 278 KiB/s, done.
Resolving deltas: 100% (12756/12756), done.
# cd x264
# ./configure --enable-static
platform:      X86
system:       LINUX
cli:          yes
libx264:      internal
shared:       no
static:       yes
asm:          yes
interlaced:   yes
avs:          no
lavf:         no
ffms:         no
gpac:         no
gpl:          yes
thread:       posix
filters:      crop select_every
debug:        no
gprof:        no
strip:        no
PIC:          no
visualize:    no
bit depth:    8
chroma format: all

You can run 'make' or 'make fprofiled' now.
# make
[...]
```

Un cop fet el `make` es pot verificar que la compilació ha funcionat correctament —a més del petit detall que si no hagués sigut així hauria aparegut algun missatge d'error— i veure quina és la versió obtinguda.

```
# ./x264 --version
x264 0.124.2197 69a0443
built on May 21 2012, gcc: 4.6.3
configuration: --bit-depth=8 --chroma-format=all
x264 license: GPL version 2 or later
# ./version.sh
#define X264_REV 2197
#define X264_REV_DIFF 0
#define X264_VERSION " r2197 69a0443"
#define X264_POINTVER "0.124.2197 69a0443"
```

A continuació es procedeix a la instal·lació, fent servir `checkinstall`. Es pot veure com entre els paràmetres de `checkinstall` es dona nom al paquet —nom que es podrà fer servir per desinstal·lar de forma convencional— i s'indica la versió fent servir la sortida de `versio.sh`.

```
# sudo checkinstall --pkgname=x264 \
  --pkgversion="3:${./version.sh | \
  awk -F'[" ]' ' /POINT/{print $4"+git"$5}'}" \
  --backup=no --deldoc=yes --fstrans=no --default

checkinstall 1.6.2, Copyright 2009 Felipe Eduardo Sanchez Diaz Duran
      This software is released under the GNU GPL.
[...]
This package will be built according to these values:

0 - Maintainer: [ root@pfm-ub ]
1 - Summary: [ Package created with checkinstall 1.6.2 ]
2 - Name: [ x264 ]
3 - Version: [ 3:0.124.2197+git69a0443 ]
4 - Release: [ 1 ]
5 - License: [ GPL ]
6 - Group: [ checkinstall ]
7 - Architecture: [ i386 ]
8 - Source location: [ x264 ]
9 - Alternate source location: [ ]
10 - Requires: [ ]
11 - Provides: [ x264 ]
```

```
12 - Conflicts: [ ]
13 - Replaces: [ ]

[...]
```

Un cop acabada la instal·lació amb `checkinstall` es pot verificar cridant `x264` sense `path`, i veure la versió instal·lada. Es pot també comprovar com el paquet generat ha sigut donat d'alta en la *cache* local dels repositoris (i es pot trobar juntament amb la versió que hi havia).

```
# x264 --version
x264 0.124.2197 69a0443
built on May 21 2012, gcc: 4.6.3
configuration: --bit-depth=8 --chroma-format=all
x264 license: GPL version 2 or later

# apt-cache show x264
Package: x264
Status: install ok installed
Priority: extra
Section: checkinstall
Installed-Size: 2668
Maintainer: root@pfm-ub
Architecture: i386
Version: 3:0.124.2197+git69a0443-1
Provides: x264
Description: Package created with checkinstall 1.6.2

Package: x264
Priority: optional
[...]
Version: 2:0.120.2151+gita3f4407-2
[...]
```

4.3.3 Obtenció codi font

Una vegada instal·lades les dependències de compilació i el programari auxiliar, es procedirà a instal·lar FFmpeg. Obtenir el codi font és ben senzill, tal i com s'ha vist al obtenir `x264`.

```
# cd sources
# git clone git://source.ffmpeg.org/ffmpeg
```

```
Cloning into 'ffmpeg'...
remote: Counting objects: 225363, done.
remote: Compressing objects: 100% (50940/50940), done.
remote: Total 225363 (delta 177313), reused 221105 (delta 174060)
Receiving objects: 100% (225363/225363), 57.77 MiB | 376 KiB/s,
done.
Resolving deltas: 100% (177313/177313), done.
```

Es pot fer servir l'opció `--depth 1`, que trunca a 1 —la darrera revisió— la història, ja que per a fer la instal·lació no cal més. Això redueix en gran mesura la mida de la descarrega, i el temps de creació. A la màquina de desenvolupament, però, pot ser interessant tenir-ne la història completa.

```
# git clone --depth 1 git://source.ffmpeg.org/ffmpeg ffmpegshallow
Cloning into 'ffmpegshallow'...
remote: Counting objects: 11007, done.
remote: Compressing objects: 100% (7052/7052), done.
remote: Total 11007 (delta 7549), reused 5972 (delta 3750)
Receiving objects: 100% (11007/11007), 12.44 MiB | 268 KiB/s, done.
Resolving deltas: 100% (7549/7549), done.
```

4.3.4 Compilació

Per compilar s'ha de tenir en compte l'ús que se'n vol fer, i els components addicionals que s'han instal·lat (4.3.1). `libx264` per treballar amb `x264` —per codificar i decodificar H.264—, `X11grab` per obtenir captura de vídeo de pantalla, i `GPL` per activar els components que estan sota llicència GPLv2.

```
# cd ffmpeg
# ./configure --enable-gpl --enable-libx264 --enable-x11grab
[...]
# make
```

Pot trigar uns minuts en realitzar el `make`. La sortida de `./configure` no s'ha inclòs aquí per la seva extensió, però és interessant de veure, ja que dona una idea de la potència de FFmpeg com a solució multimedia, per això s'ha inclòs a l'Apèndix C.

Un cop realitzada la compilació es pot provar, i verificar la versió de les llibreries.

```
# ./ffmpeg -version
ffmpeg version N-40928-g307562b
built on May 21 2012 17:49:13 with gcc 4.6.3
configuration: --enable-gpl --enable-libx264 --enable-x11grab
libavutil      51. 53.100 / 51. 53.100
libavcodec     54. 21.101 / 54. 21.101
libavformat    54.  5.101 / 54.  5.101
libavdevice    53.  4.100 / 53.  4.100
libavfilter     2. 75.100 /  2. 75.100
libswscale      2.  1.100 /  2.  1.100
libswresample  0. 15.100 /  0. 15.100
libpostproc   52.  0.100 / 52.  0.100
```

Llicència GPL

Per saber que implica activar o no GPLv2 s'ha de consultar el fitxer LICENSE que es troba a l'arrel de les fonts de FFmpeg. També es pot veure com hi ha diverses opcions de FFmpeg que queden sota altres llicències, fins i tot alguns components externs *nonfree* que si s'utilitzen fan que FFmpeg no es pugui redistribuir. En el que aplica a aquest treball es pot veure que per fer servir X11grab i per treballar amb les llibreries externes de x264 és necessari activar-ho.

Specifically, the GPL parts of FFmpeg are

- *libpostproc*
- *optional x86 optimizations in the files libavcodec/x86/idct_mmx.c*
- *the X11 grabber in libavdevice/x11grab.c*

Some external libraries, e.g. libx264, are under GPL and can be used in conjunction with FFmpeg. They require --enable-gpl to be passed to configure as well.

4.3.5 Instal·lació

Un cop realitzada la compilació es pot instal·lar el programari fent servir checkinstall, tal i com s'ha fet amb x264, i provar-lo.

```
# sudo checkinstall --pkgname=ffmpeg \
--pkgversion="5:$(date +%Y%m%d%H%M)-git" \
--backup=no --deldoc=yes --fstrans=no --default

checkinstall 1.6.2, Copyright 2009 Felipe Eduardo Sanchez Diaz Duran
```

```

                This software is released under the GNU GPL.
[...]
This package will be built according to these values:

0 - Maintainer: [ root@pfm-ub ]
1 - Summary: [ Package created with checkinstall 1.6.2 ]
2 - Name: [ ffmpeg ]
3 - Version: [ 5:201205211804-git ]
4 - Release: [ 1 ]
5 - License: [ GPL ]
6 - Group: [ checkinstall ]
7 - Architecture: [ i386 ]
8 - Source location: [ ffmpeg ]
9 - Alternate source location: [ ]
10 - Requires: [ ]
11 - Provides: [ ffmpeg ]
12 - Conflicts: [ ]
13 - Replaces: [ ]

[...]

# ffmpeg -version
ffmpeg version N-40928-g307562b
built on May 21 2012 17:49:13 with gcc 4.6.3
configuration: --enable-gpl --enable-libx264 --enable-x11grab
libavutil      51. 53.100 / 51. 53.100
libavcodec     54. 21.101 / 54. 21.101
libavformat    54.  5.101 / 54.  5.101
libavdevice    53.  4.100 / 53.  4.100
libavfilter     2. 75.100 /  2. 75.100
libswscale      2.  1.100 /  2.  1.100
libswresample  0. 15.100 /  0. 15.100
libpostproc    52.  0.100 / 52.  0.100

```

És possible verificar com les llibreries i els executables que s'han creat en aquesta secció han quedat adientment instal·lats en el sistema.

```

# cd /usr/local/lib/
# ll
total 72468
drwxr-xr-x  4 root root      4096 May 21 18:05 ./
drwxr-xr-x 10 root root      4096 May 21 16:20 ../
-rw-r--r--  1 root root 51154530 May 21 18:04 libavcodec.a

```

```

-rw-r--r--  1 root root    684306 May 21 18:04 libavdevice.a
-rw-r--r--  1 root root    3599208 May 21 18:04 libavfilter.a
-rw-r--r--  1 root root   14995850 May 21 18:04 libavformat.a
-rw-r--r--  1 root root    222312 May 21 18:04 libavresample.a
-rw-r--r--  1 root root    568776 May 21 18:04 libavutil.a
-rw-r--r--  1 root root    209192 May 21 18:04 libpostproc.a
-rw-r--r--  1 root root    204516 May 21 18:04 libswresample.a
-rw-r--r--  1 root root   1176382 May 21 18:04 libswscale.a
-rw-r--r--  1 root root   1355686 May 21 16:20 libx264.a
drwxr-xr-x  2 root root      4096 May 21 18:05 pkgconfig/
drwxrwsr-x  4 root staff      4096 May 21 13:13 python2.7/
# cd /usr/local/bin
# ll
total 43552
drwxr-xr-x  2 root root      4096 May 21 18:05 ./
drwxr-xr-x 10 root root      4096 May 21 16:20 ../
-rwxr-xr-x  1 root root 10978944 May 21 18:04 ffmpeg*
-rwxr-xr-x  1 root root 10926016 May 21 18:04 ffplay*
-rwxr-xr-x  1 root root 10917536 May 21 18:04 ffprobe*
-rwxr-xr-x  1 root root 10543840 May 21 18:04 ffmpegserver*
-rwxr-xr-x  1 root root 1209184 May 21 16:20 x264*

```

4.3.6 Inspecció codi font

Com s'ha explicat al principi d'aquesta secció (4.3), el més interessant d'instal·lar des del codi font és poder-lo investigar. Els fitxers de codi es poden obrir amb l'editor que es desitgi, però per gestionar un projecte del volum de FFmpeg els avantatges de fer servir un IDE són innegables. Al parlar de l'entorn de desenvolupament (2.2.3), s'ha comentat que s'ha escollit Netbeans, i perquè. A continuació es veurà com crear-ne un projecte que faciliti la inspecció del codi. Es seguiran les indicacions corresponents del *Quick Start Tutorial* de Netbeans⁹.

Es tenen dues opcions a l'hora de crear el projecte. Si s'ha fet la instal·lació amb checkinstall i la configuració des de la línia de comandes, com s'ha vist en els apartats anteriors, no interessa que l'IDE faci res de compilació, ni neteja ni similar, només veure el codi. Per altra banda, si es desitja fer alguna modificació en el codi i tornar a compilar, es pot decidir fer-ho tot des de Netbeans.

Primer de tot cal crear un nou projecte. Ha de ser un projecte de C/C++, amb fonts existents, tal i com es veu a la figura 4.1.

⁹<http://netbeans.org/kb/docs/cnd/quickstart.html#makefileprojects>

Figura 4.1: Nou projecte amb fonts existents

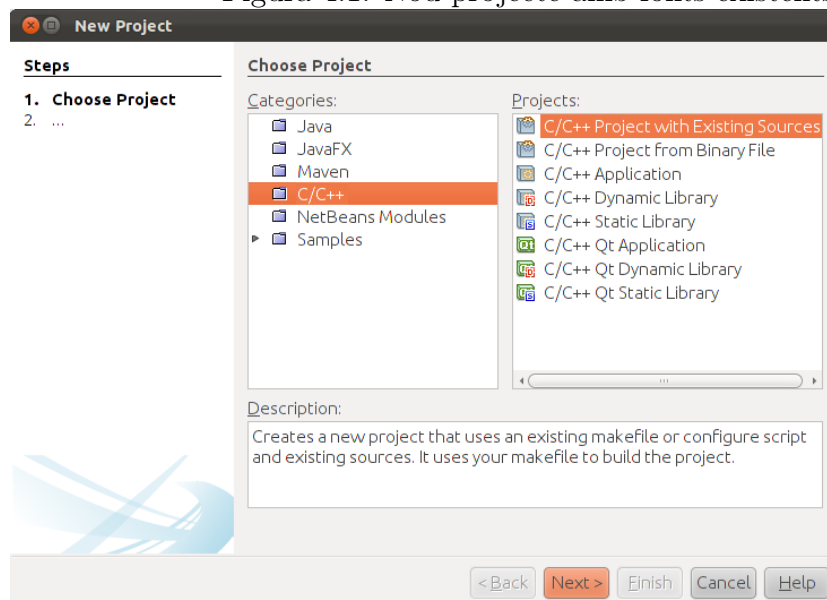


Figura 4.2: Modalitat de gestió del nou projecte

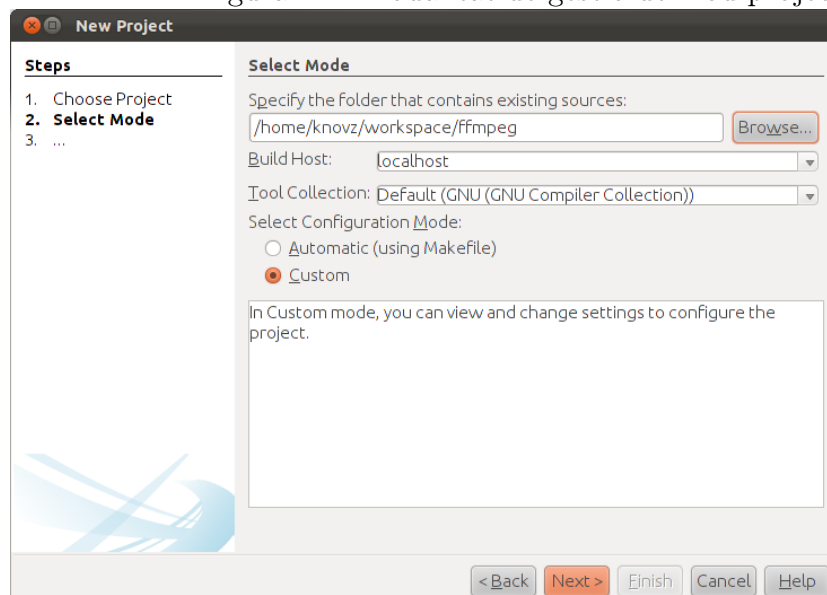
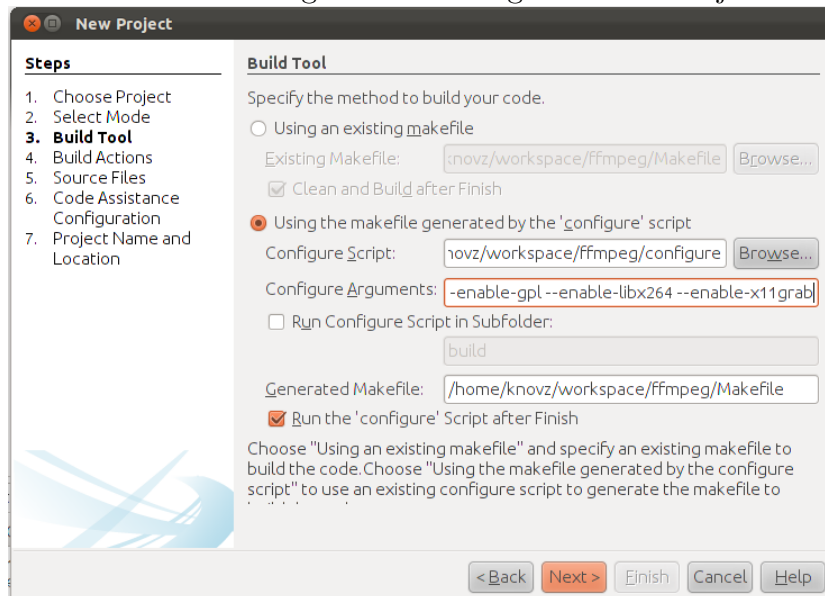


Figura 4.3: Configuració del Projecte



La següent pantalla (figura 4.2) és la pantalla clau. Si es manté *Automatic (using makefile)* seleccionat, Netbeans interpretarà que n'ha de fer la gestió, fent servir el *Makefile* existent, i farà un *clean* i un *build* al crear el projecte *ELIMINANT ELS BINARIS QUE PUGUIN EXISTIR!*. S'hauria de seleccionar *Custom*, ni que fos per poder revisar la configuració.

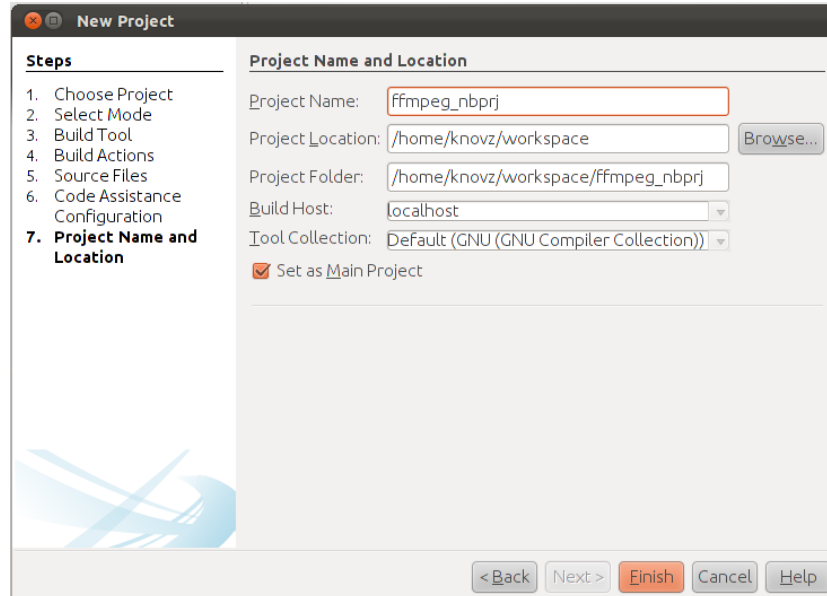
A continuació es pot indicar la configuració que es desitja del projecte. Es pot indicar que es faci servir un *makefile* existent, o un de generat per un *script* de configuració (que és el cas que ens ocupa).

Com es pot veure a la figura 4.3, a arguments de configuració s'ha de posar els arguments que es desitja passar per a la configuració. Aquí es té la opció de no marcar *Run the 'configure' Script after Finish*, per evitar que Netbeans cridi a l'*script* de configuració i faci *make clean* i *make*, *ELIMINANT ELS BINARIS QUE PUGUIN EXISTIR!*.

Si només es desitja inspeccionar el codi, però es vol realitzar tota la gestió des de línia de comandes, s'ha de desmarcar. Si es desitja gestionar la compilació des de dins l'IDE, s'ha de deixar marcat com a la figura 4.3.

Es pot anar donant a next, revisant les diverses opcions com es desitgi, fins arribar a *Project Name and Location*, com es veu a la figura 4.4. Aquí es adient posar al projecte un nom diferent del del directori on hi ha les fonts, o indicar una ubicació diferent. D'aquesta manera s'aconsegueix mantenir fora del directori on hi ha el codi font els fitxers propis de la gestió del projecte per part de Netbeans. En aquesta plana s'escull també, cas que n'hi hagi

Figura 4.4: Carpeta de projecte separada



diversos instal·lats, el *tool chain* que s'ha de fer servir amb el projecte.

Finalment, prement *Finish*, es crearà el projecte. Si s'ha decidit gestionar el projecte des de dins de Netbeans, en crear-se el projecte s'executarà l'*script* de configuració, i a continuació es farà automàticament un **make clean** i un **make**. Com a darrer pas Netbeans escanejarà els directoris que formen el projecte, interpretant el codi. S'ha de tenir en compte que tant l'execució del **make** com el *parse* del codi font per part de Netbeans pot trigar uns minuts, donada la mida del projecte.

4.4 Verificació i proves de FFmpeg

Un cop instal·lat FFmpeg, és necessari verificar que funciona correctament, i que és l'eina adient per a les tasques que es desitja realitzar. Per això s'han fet les següents verificacions:

- Que funciona ⇒ 4.4.1
- Que permet capturar desde càmera ⇒ 4.4.2
- Que permet definir el format de sortida al vol ⇒ 4.4.3
- Que es pot treballar en H.264 ⇒ 4.4.4
- Que permet segmentar la sortida ⇒ 4.4.5

4.4.1 Funcionament bàsic

La primera prova de funcionament ja s'ha fet al finalitzar la instal·lació a 4.3. Es tracta, com ja s'ha vist, de comprovar que s'executa, i verificar-ne la versió. FFmpeg conté, a més de les llibreries i del propi ffmpeg, altres eines. `ffprobe` n'és una que pot resultar de molta utilitat en un projecte com aquest, ja que serveix per analitzar vídeos.

```
# ffprobe -version
ffprobe version N-40962-g0da28d6
built on May 23 2012 16:23:18 with gcc 4.6.1
configuration: --enable-gpl --enable-libx264 --enable-x11grab
libavutil      51. 53.100 / 51. 53.100
libavcodec     54. 23.100 / 54. 23.100
libavformat    54.  6.100 / 54.  6.100
libavdevice    53.  4.100 / 53.  4.100
libavfilter    2. 76.100 / 2. 76.100
libswscale     2.  1.100 / 2.  1.100
libswresample  0. 15.100 / 0. 15.100
libpostproc   52.  0.100 / 52.  0.100
```

Es pot veure que la versió no es correspon amb la instal·lada a 4.3, ja que en aquest cas les proves s'han fet en un equip diferent i s'ha actualitzat la instal·lació abans de realitzar-les. Per provar `ffprobe` s'ha fet servir un vídeo provinent d'una càmera Axis Q6034, gravat amb el propi control de Axis.

```
# ffprobe testPanoramica.asf
ffprobe version N-40962-g0da28d6 Copyright (c) 2007-2012
the FFmpeg developers
  built on May 23 2012 16:23:18 with gcc 4.6.1
  configuration: --enable-gpl --enable-libx264 --enable-x11grab
  libavutil      51. 53.100 / 51. 53.100
  libavcodec     54. 23.100 / 54. 23.100
  libavformat    54.  6.100 / 54.  6.100
  libavdevice    53.  4.100 / 53.  4.100
  libavfilter    2. 76.100 / 2. 76.100
  libswscale     2.  1.100 / 2.  1.100
  libswresample  0. 15.100 / 0. 15.100
  libpostproc   52.  0.100 / 52.  0.100
Input #0, asf, from 'testPanoramica.asf':
  Metadata:
```

```

WMFSDKVersion   : 12.0.7600.16385
WMFSDKNeeded    : 0.0.0.0000
IsVBR           : 0
Duration: 00:00:39.91, bitrate: 4719 kb/s
Stream #0:0(spa): Video: h264 (Baseline) (H264 / 0x34363248),
yuvj420p, 1280x720 [SAR 1:1 DAR 16:9], 100000 kb/s, 25 tbr,
1k tbn, 2k tbc

```

Es pot veure que les eines de FFmpeg donen la informació de compilació, configuració, i versió en cada execució. Per alleugerir aquesta memòria i permetre concentrar-se en la informació més rellevant, a partir d'aquest moment s'elidirà aquesta informació ([...]).

Per seguir amb les proves bàsiques de funcionament, s'ha modificat el format contenidor d'aquest mateix vídeo amb `ffmpeg` però mantenint el còdec original, i se n'ha verificat el resultat amb `ffprobe`.

```

# ffmpeg -i testPanoramica.asf -f mp4 -c:v copy testPanoramica.mp4
[...]
Input #0, asf, from 'testPanoramica.asf':
  Metadata:
    WMFSDKVersion   : 12.0.7600.16385
    WMFSDKNeeded    : 0.0.0.0000
    IsVBR           : 0
  Duration: 00:00:39.91, bitrate: 4719 kb/s
  Stream #0:0(spa): Video: h264 (Baseline) (H264 / 0x34363248),
  yuvj420p, 1280x720 [SAR 1:1 DAR 16:9], 100000 kb/s, 25 tbr,
  1k tbn, 2k tbc
Output #0, mp4, to 'testPanoramica.mp4':
  Metadata:
    WMFSDKVersion   : 12.0.7600.16385
    WMFSDKNeeded    : 0.0.0.0000
    IsVBR           : 0
    encoder         : Lavf54.6.100
  Stream #0:0(spa): Video: h264 (![0][0][0] / 0x0021), yuvj420p,
  1280x720 [SAR 1:1 DAR 16:9], q=2-31, 100000 kb/s, 1k tbn,
  1k tbc
Stream mapping:
  Stream #0:0 -> #0:0 (copy)
Press [q] to stop, [?] for help
frame= 998 fps=0.0 q=-1.0 Lsize= 22884kB time=00:00:39.88
bitrate=4700.5kbits/s

```

```
video:22878kB audio:0kB global headers:0kB
muxing overhead 0.026824%
```

```
# ffprobe testPanoramica.mp4
[...]
[mov,mp4,m4a,3gp,3g2,mj2 @ 0xaca84a0]
multiple edit list entries, a/v desync might occur, patch welcome
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'testPanoramica.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2avc1mp41
    encoder          : Lavf54.6.100
  Duration: 00:00:39.88, start: 0.001000, bitrate: 4700 kb/s
  Stream #0:0(spa): Video: h264 (Baseline) (avc1 / 0x31637661),
  yuvj420p, 1280x720 [SAR 1:1 DAR 16:9], 4699 kb/s, 25.03 fps,
  25 tbr, 1k tbn, 2k tbc
  Metadata:
    handler_name     : VideoHandler
```

Es pot veure que el contenidor és diferent en ambdós casos, però que l'*stream* contingut és igual (amb lleugeres diferències). En concret marca explícitament que són 25 fps, i que la durada i el *bitrate* són lleugerament diferents. Aquests són valors calculats, i tenint en compte que són promitjos —és un vídeo de *bitrate* variable—, la diferència és tolerable.

Comparant les dades del `Input` donades per `ffprobe` i `ffmpeg`, es pot veure que, com seria d'esperar, són idèntiques. De fet es podria fer servir `ffmpeg -i` per analitzar els vídeos, i el resultat seria el mateix, però amb un missatge d'error al final indicant que manca el paràmetre de sortida. `ffprobe` pot proporcionar, amb les opcions adients, una informació més detallada o estructurada, però per aquestes proves no és necessari.

Inspeccionant el vídeo visualment —valgui la redundància— es pot verificar que realment el vídeo obtingut es correspon al desitjat.

Amb això s'ha comprovat la capacitat de `FFmpeg` de modificar el contenidor, a continuació es comprovarà la de transcodificació.

```
# ffmpeg -i testPanoramica.asf -f mp4 -b:v 4700k -c mpeg4 \
  testPanoramica2.mp4
[...]
Input #0, asf, from 'testPanoramica.asf':
  Metadata:
```

```

    WMFSDKVersion   : 12.0.7600.16385
    WMFSDKNeeded    : 0.0.0.0000
    IsVBR           : 0
Duration: 00:00:39.91, bitrate: 4719 kb/s
  Stream #0:0(spa): Video: h264 (Baseline) (H264 / 0x34363248),
    yuvj420p, 1280x720 [SAR 1:1 DAR 16:9], 100000 kb/s, 25 tbr,
    1k tbn, 2k tbc
w:1280 h:720 pixfmt:yuvj420p tb:1/1000 sar:1/1 sws_param:flags=2
[buffersink @ 0x9cdfbe0] No opaque field provided
[format @ 0x9cdfce0] auto-inserting filter 'auto-inserted scaler 0'
between the filter 'src' and the filter 'format'
[scale @ 0x95b2a00] w:1280 h:720 fmt:yuvj420p sar:1/1 -> w:1280
h:720 fmt:yuv420p sar:1/1 flags:0x4
Output #0, mp4, to 'testPanoramica2.mp4':
  Metadata:
    WMFSDKVersion   : 12.0.7600.16385
    WMFSDKNeeded    : 0.0.0.0000
    IsVBR           : 0
    encoder         : Lavf54.6.100
  Stream #0:0(spa): Video: mpeg4 ( [0][0][0] / 0x0020), yuv420p,
    1280x720 [SAR 1:1 DAR 16:9], q=2-31, 4700 kb/s, 25 tbn, 25 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (h264 -> mpeg4)
Press [q] to stop, [?] for help
frame= 998 fps= 86 q=3.4 Lsize= 22990kB time=00:00:39.92
bitrate=4717.8kb/s
video:22985kB audio:0kB global headers:0kB
muxing overhead 0.023066%

```

```

# ffprobe testPanoramica2.mp4
[...]
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'testPanoramica2.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2mp41
    encoder         : Lavf54.6.100
Duration: 00:00:39.92, start: 0.000000, bitrate: 4717 kb/s
  Stream #0:0(spa): Video: mpeg4 (Simple Profile)
    (mp4v / 0x7634706D), yuv420p, 1280x720 [SAR 1:1 DAR 16:9],
    4716 kb/s, 25 fps, 25 tbr, 25 tbn, 25 tbc
  Metadata:

```

```
handler_name      : VideoHandler
```

En aquest darrer cas s'ha forçat un *bitrate*, ja que el valor per defecte deixa un vídeo de baixíssima qualitat.

Novament convindria revisar el vídeo, per veure que els resultats són els esperats. Amb això es pot donar per verificat el funcionament bàsic de FFmpeg. Com es veurà a continuació FFmpeg és una eina amb multitud d'opcions, i fins ara només s'ha vist una petita part.

4.4.2 Captura de càmera

A les proves de funcionament bàsic (4.4.1) s'ha vist com FFmpeg permet processar un arxiu de vídeo. Per poder assolir els objectius d'aquest projecte cal, però, que es pugui fer servir com entrada un **stream** de vídeo que provingui directament d'una càmera IP.

Per a fer aquestes proves farem servir una càmera Axis 210A, amb *firmware* 4.47.1 —el darrer disponible en la data de realització d'aquesta pràctica, i probablement el darrer ja que és obsoleta—.

Les càmeres amb aquest *firmware* (4.x.x) segueixen la API VAPIX[®] *Version 2*, les que tenen *firmware* 5.x.x segueixen la *Version 3*. Ambues API són públiques i es poden trobar a la web d'Axis¹⁰.

Segons aquesta API es pot accedir al stream de la càmera per RTSP, amb la següent url:

```
rtsp://<IP>:<PORT>/mpeg4/1/media.amp
```

Si el port és 554 —port per defecte per RTSP— es pot obviar. En teoria es pot fer servir aquesta url com a entrada per a FFmpeg.

Tot i que fent servir l'aplicació per consola es pot aturar manualment —prement q— és convenient saber que **ffmpeg** admet el paràmetre **-t** que permet especificar d'avançada el número de segons de l'entrada que es desitja processar. Aquesta opció és ben útil treballant amb streams, que —com en aquest cas— poden ser indefinits.

Com a altres paràmetres únicament s'ha fixat el format del contenidor sortida, i s'ha indicat que es mantingués la codificació del *stream* com a l'entrada.

```
# ffmpeg -i "rtsp://192.168.1.222/mpeg4/1/media.amp" -f mp4 \
  -c:v copy -t 30 capturetest.mp4
```

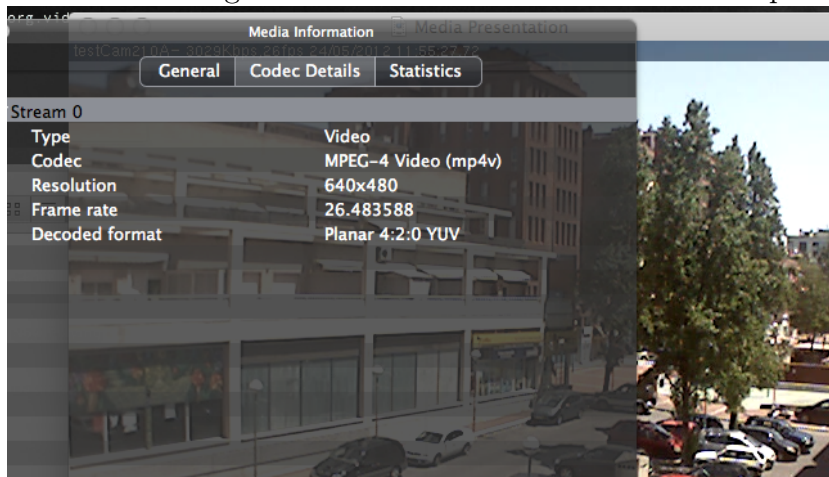
¹⁰http://www.axis.com/techsup/cam_servers/dev/cam_http_api_index.php

```
[...]
[rtsp @ 0xae343c0] Estimating duration from bitrate, this may be
inaccurate
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
  Metadata:
    title           : Media Presentation
  Duration: N/A, start: 0.023356, bitrate: N/A
    Stream #0:0: Video: mpeg4 (Advanced Simple Profile), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], 100 tbr, 90k tbn, 100 tbc
Output #0, mp4, to 'capturetest.mp4':
  Metadata:
    title           : Media Presentation
    encoder         : Lavf54.6.100
    Stream #0:0: Video: mpeg4 ( [0][0][0] / 0x0020), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], q=2-31, 90k tbn, 90k tbc
Stream mapping:
  Stream #0:0 -> #0:0 (copy)
Press [q] to stop, [?] for help
frame= 795 fps= 26 q=-1.0 Lsize= 11414kB time=00:00:29.98
bitrate=3118.8kb/s
video:11405kB audio:0kB global headers:0kB
muxing overhead 0.078269%
```

```
# ffprobe capturetest.mp4
[...]
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'capturetest.mp4':
  Metadata:
    major_brand     : isom
    minor_version   : 512
    compatible_brands: isomiso2mp41
    title           : Media Presentation
    encoder         : Lavf54.6.100
  Duration: 00:00:29.99, start: 0.000000, bitrate: 3117 kb/s
  Stream #0:0(und): Video: mpeg4 (Advanced Simple Profile)
  (mp4v / 0x7634706D), yuv420p, 640x480 [SAR 1:1 DAR 4:3],
  3115 kb/s, 26.51 fps, 100 tbr, 90k tbn, 100 tbc
  Metadata:
    handler_name    : VideoHandler
```

Inspeccionant el vídeo visualment es pot verificar que la qualitat és l'esperada (com la del stream visualitzat en viu). Per arrodonir la inspecció, tenint en compte que aquest vídeo s'acaba de crear i podria no semblar adient quedar-se amb la inspecció realitzada per la mateixa eina que el crea,

Figura 4.5: Anàlisi amb VLC del vídeo capturat



s'ha inspeccionat el vídeo amb un VLC instal·lat en un altre ordinador com es pot veure a la figura 4.5.

4.4.3 Format de sortida

En el cas de les càmeres d'Axis amb *firmware* 5.x.x l'*stream* generat pot ser directament codificat amb H.264, i per tant la sortida de `ffmpeg` amb `-c:v` `copy` ja compliria les especificacions d'aquest projecte, sempre que es pogués fragmentar (4.4.5).

Interessa doncs provar que es pot transcodificar al vol, generant un *stream* de sortida amb codificació diferent a la del de entrada. Tot i que en concret interessa la sortida en H.264 es separarà aquesta prova en dos, transcodificació al vol en aquest apartat i treball amb H.264 en 4.4.4, ja que per a treballar amb aquest format —com amb altres— FFmpeg fa servir llibreries externes (x.264). A més, d'aquesta manera, es valida que la solució és vàlida per altre mena de necessitats de sortida.

Per exemple podem fer servir FLV.

```
# ffmpeg -i "rtsp://192.168.1.222/mpeg4/1/media.amp" -f flv \
  -same_quant -c:v flv -t 30 capturetest.flv
[...]
[rtsp @ 0xad903c0] Estimating duration from bitrate, this may be
inaccurate
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
  Metadata:
    title           : Media Presentation
```

```

Duration: N/A, start: 0.023389, bitrate: N/A
  Stream #0:0: Video: mpeg4 (Advanced Simple Profile), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], 100 tbr, 90k tbn, 100 tbc
[buffer @ 0xad3320] w:640 h:480 pixfmt:yuv420p tb:1/90000 sar:1/1
sws_param:flags=2
[buffersink @ 0xad36a0] No opaque field provided
Output #0, flv, to 'capturetest.flv':
  Metadata:
    title           : Media Presentation
    encoder         : Lavf54.6.100
  Stream #0:0: Video: flv1 ([2][0][0][0] / 0x0002), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], q=2-31, 200 kb/s, 1k tbn, 100 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (mpeg4 -> flv)
Press [q] to stop, [?] for help
frame= 795 fps= 26 q=0.0 Lsize= 22284kB time=00:00:29.99
bitrate=6087.0kbits/s
video:22271kB audio:0kB global headers:0kB
muxing overhead 0.056823%

```

```

# ffprobe capturetest.flv
[...]
Input #0, flv, from 'capturetest.flv':
  Metadata:
    title           : Media Presentation
    encoder         : Lavf54.6.100
  Duration: 00:00:29.98, start: 0.000000, bitrate: 6087 kb/s
    Stream #0:0: Video: flv1, yuv420p, 640x480, 200 kb/s, 100 tbr,
    1k tbn, 1k tbc

```

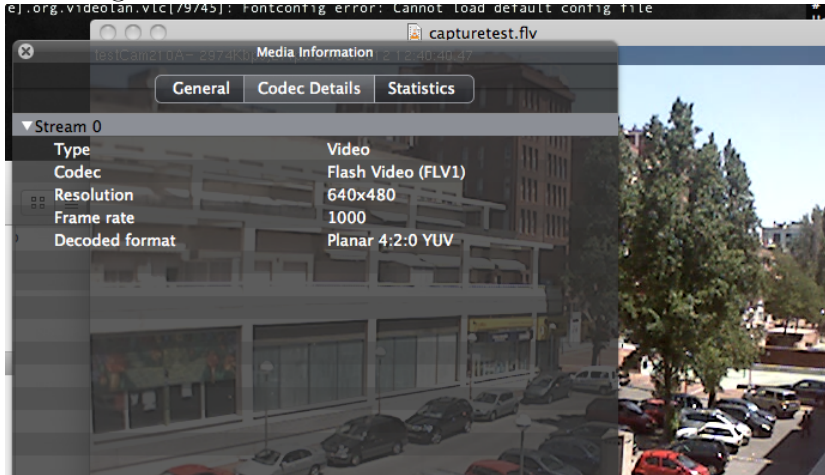
O wmv2 —que es correspon al Windows Media Video 8—

```

# ffmpeg -i "rtsp://192.168.1.222/mpeg4/1/media.amp" -f asf \
  -same_quant -c:v wmv2 -t 30 capturetest.wmv
[...]
[rtsp @ 0x94113c0] Estimating duration from bitrate, this may be
inaccurate
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
  Metadata:
    title           : Media Presentation
  Duration: N/A, start: 0.023333, bitrate: N/A
    Stream #0:0: Video: mpeg4 (Advanced Simple Profile), yuv420p,

```

Figura 4.6: Anàlisi amb VLC del vídeo transcodificat al vol FLV



```

640x480 [SAR 1:1 DAR 4:3], 100 tbr, 90k tbn, 100 tbc
[buffer @ 0x9444300] w:640 h:480 pixfmt:yuv420p tb:1/90000 sar:1/1
sws_param:flags=2
[buffersink @ 0x94446a0] No opaque field provided
Output #0, asf, to 'capturetest.wmv':
  Metadata:
    title           : Media Presentation
    WM/EncodingSettings: Lavf54.6.100
    Stream #0:0: Video: wmv2 (WMV2 / 0x32564D57), yuv420p, 640x480
    [SAR 1:1 DAR 4:3], q=2-31, 200 kb/s, 1k tbn, 100 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (mpeg4 -> wmv2)
Press [q] to stop, [?] for help
frame= 2999 fps= 99 q=0.0 Lsize=   50454kB time=00:00:29.99
bitrate=13781.9kbits/s dup=2201 drop=0
video:49989kB audio:0kB global headers:0kB
muxing overhead 0.931149%

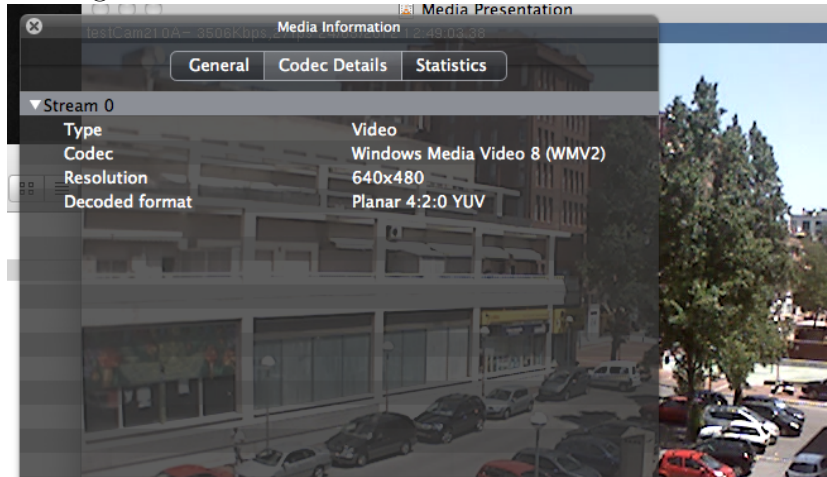
```

```

# ffprobe capturetest.wmv
[...]
Input #0, asf, from 'capturetest.wmv':
  Metadata:
    title           : Media Presentation
    encoder         : Lavf54.6.100
  Duration: 00:00:29.99, start: 0.000000, bitrate: 13781 kb/s
  Stream #0:0: Video: wmv2 (WMV2 / 0x32564D57), yuv420p,

```

Figura 4.7: Anàlisi amb VLC del vídeo transcodificat al vol WMV8



```
640x480, 100 tbr, 1k tbn, 1k tbc
```

Es interessant comparar les mides dels fitxers generats, de qualitats similars, tot i que no és vàlid de cap manera per a treure conclusions ja que en cap cas s'ha buscat cap mena d'optimització.

```
22M 2012-05-24 13:40 capturetest.flv
12M 2012-05-24 12:55 capturetest.mp4
50M 2012-05-24 13:49 capturetest.wmv
```

4.4.4 Sortida mp4 amb H.264

El següent pas és verificar que es pot obtenir una sortida codificada amb H.264 partint d'una entrada que no ho està. Es tracta de fer una captura transcodificant al vol. Com a còdec es farà servir libx264 i com a format del contenidor mp4.

En aquest cas s'ha de fer servir algun paràmetre més. S'ha de definir el perfil de codificació i els paràmetres a passar a x264 (es pot fer mitjançant *presets*).

```
# ffmpeg -i "rtsp://192.168.1.222/mpeg4/1/media.amp" -f mp4 \
-c:v libx264 -profile:v baseline -preset fast -t 30 \
capturetesth264.mp4
[...]
[rtsp @ 0x9ef53c0] Estimating duration from bitrate, this may be
```

```

inaccurate
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
  Metadata:
    title           : Media Presentation
  Duration: N/A, start: 0.023356, bitrate: N/A
    Stream #0:0: Video: mpeg4 (Advanced Simple Profile), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], 100 tbr, 90k tbn, 100 tbc
[buffer @ 0x9ef4120] w:640 h:480 pixfmt:yuv420p tb:1/90000 sar:1/1
sws_param:flags=2
[buffersink @ 0x9ef44c0] No opaque field provided
[libx264 @ 0x9f27ea0] using SAR=1/1
[libx264 @ 0x9f27ea0] using cpu capabilities: MMX2 SSE2Fast SSSE3
FastShuffle Cache64
[libx264 @ 0x9f27ea0] profile Constrained Baseline, level 3.2
[libx264 @ 0x9f27ea0] 264 - core 125 r2200 999b753 -
H.264/MPEG-4 AVC codec - Copyleft 2003-2012 -
http://www.videolan.org/x264.html - options: cabac=0 ref=2
deblock=1:0:0 analyse=0x1:0x111 me=hex subme=6 psy=1
psy_rd=1.00:0.00 mixed_ref=1 me_range=16 chroma_me=1 trellis=1
8x8dct=0 cqm=0 deadzone=21,11 fast_pskip=1 chroma_qp_offset=-2
threads=3 lookahead_threads=1 sliced_threads=0 nr=0 decimate=1
interlaced=0 bluray_compat=0 constrained_intra=0 bframes=0
weightp=0 keyint=250 keyint_min=25 scenecut=40 intra_refresh=0
rc_lookahead=30 rc=crf mbtree=1 crf=23.0 qcomp=0.60 qpmin=0
qpmax=69 qpstep=4 ip_ratio=1.40 aq=1:1.00
Output #0, mp4, to 'capturetesth264.mp4':
  Metadata:
    title           : Media Presentation
    encoder         : Lavf54.6.100
    Stream #0:0: Video: h264 (![0][0][0] / 0x0021), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], q=-1--1, 100 tbn, 100 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (mpeg4 -> libx264)
Press [q] to stop, [?] for help
frame= 2999 fps= 99 q=-7088997.0 Lsize=      1965kB time=00:00:29.99
bitrate= 536.7kbits/s dup=2204 drop=0
video:1952kB audio:0kB global headers:0kB muxing overhead 0.644190%
[libx264 @ 0x9f27ea0] frame I:12   Avg QP:24.01   size: 67480
[libx264 @ 0x9f27ea0] frame P:2987 Avg QP:26.74   size:   398
[libx264 @ 0x9f27ea0] mb I  I16..4: 10.7%  0.0% 89.3%
[libx264 @ 0x9f27ea0] mb P  I16..4:  0.1%  0.0%  0.0%
P16..4:  8.5%  0.4%  0.2%  0.0%  0.0%   skip:90.8%
[libx264 @ 0x9f27ea0] coded y,uvDC,uvAC intra: 72.3% 89.4% 69.1%

```

```

inter: 1.2% 2.2% 0.2%
[libx264 @ 0x9f27ea0] i16 v,h,dc,p: 18% 53% 21% 8%
[libx264 @ 0x9f27ea0] i4 v,h,dc,ddl,ddr,vr,hd,vl,hu: 19% 33% 10%
5% 5% 4% 10% 5% 9%
[libx264 @ 0x9f27ea0] i8c dc,h,v,p: 52% 27% 14% 8%
[libx264 @ 0x9f27ea0] ref P L0: 43.0% 57.0%
[libx264 @ 0x9f27ea0] kb/s:533.09

```

Es pot veure com `ffmpeg` crida `libx264` per realitzar la codificació.

Es molt interessant, tot i que segueix sense ser vàlid, comparar un cop més, les mides dels vídeos resultants.

```

23M 2012-05-24 16:36 capturetest.flv
2.0M 2012-05-25 12:23 capturetesth264.mp4
12M 2012-05-24 16:36 capturetest.mp4

```

Ara bé, en realitat la diferència de mida amb `mpeg4` en aquest cas no és deguda a una millor compressió sinó a la mida del `gop` fet servir. La càmera està configurada per generar una sortida amb una longitud de `gop` de 16, mentre que el `gop` per defecte de `libx264` és 250. Si es força a 16, la diferència s'esvaeix.

```

# ffmpeg -i "rtsp://192.168.1.222/mpeg4/1/media.amp" -f mp4 \
-c:v libx264 -profile:v baseline -preset fast -g 16 -t 30 \
capturetesth264.mp4
[...]

# ll
23M 2012-05-24 16:36 capturetest.flv
12M 2012-05-25 12:09 capturetesth264_b.mp4
2.0M 2012-05-25 12:23 capturetesth264.mp4
12M 2012-05-24 16:36 capturetest.mp4

```

Finalment es vol ressaltar que les captures que s'estan fent servir afavoreixen molt la compressió de H.264, ja que bona part de la imatge és estàtica. A la figura 4.8 es ressalten les zones on hi ha moviment. Això sí, el moviment que hi ha es veu fidelment reflexat i amb molt bona qualitat.

Figura 4.8: Captura de pantalla de la zona gravada



4.4.5 Segmentació al vol de la sortida

La darrera prova que resta per fer de cara a validar la utilitat de FFmpeg com a base per a la realització d'aquest projecte és la segmentació de la sortida. Es podria pensar que una solució podria ser aturar el procés cada tants segons i rellançar-lo. Ja s'ha vist que es pot cridar a `ffmpeg` amb un paràmetre que limiti la captura, i es podria incloure la crida en un *script* o similar que cridés a `ffmpeg` de forma repetida —mitjançant un bucle—. Aquesta aproximació produiria una sortida tallada. Per ràpid que sigui el temps d'arrancada de `ffmpeg`, no és nul. I les característiques dels *streams* MPEG4 o H.264 requereixen a més un cert nivell de buffer a l'inici que agreujarien aquests talls.

Des de la versió 0.9, FFmpeg inclou la possibilitat de segmentar la sortida segons es va codificant. De primeres sembla sorprenentment que aquesta opció no està implementada com una opció, sinó com un format de sortida `segment`. No és tant sorprenent si s'estudia la forma de funcionar de FFmpeg, com a components modulars que es van enllaçant un darrera l'altre. La sortida segmentada no és sinó un altre format de contenidor, que entre les seves característiques té que es fragmenta la sortida en fitxers consecutius.

Aquest format de sortida incorpora noves opcions, una de les quals és la durada dels segments en que es vol dividir la sortida. El nom del fitxer de sortida és un patró que es fa servir per generar els noms de la sèrie de fitxers

generats. I si es vol forçar el format de sortida —en oposició a que el marqui l'extensió de l'arxiu de sortida— es fa servir la opció `-segment_format`.

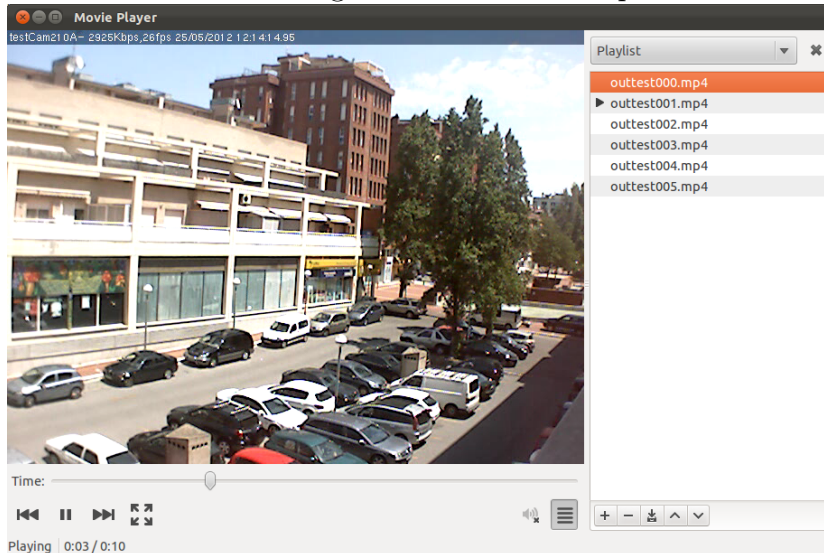
```
# ffmpeg -i "rtsp://192.168.1.222/mpeg4/1/media.amp" \
-c:v copy -map 0 -t 60 -f segment -segment_time 10 \
-segment_format mp4 outtest%03d.mp4
[...]
[rtsp @ 0xa55d3c0] Estimating duration from bitrate, this may be
inaccurate
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
  Metadata:
    title           : Media Presentation
  Duration: N/A, start: 0.023356, bitrate: N/A
    Stream #0:0: Video: mpeg4 (Advanced Simple Profile), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], 100 tbr, 90k tbn, 100 tbc
Output #0, segment, to 'outtest%03d.mp4':
  Metadata:
    title           : Media Presentation
    encoder         : Lavf54.6.100
    Stream #0:0: Video: mpeg4 ( [0][0][0] / 0x0020), yuv420p,
    640x480 [SAR 1:1 DAR 4:3], q=2-31, 100 tbn, 100 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (copy)
Press [q] to stop, [?] for help
frame= 1593 fps= 27 q=-1.0 Lsize=          0kB time=00:01:00.00
bitrate=   0.0kbits/s
video:22294kB audio:0kB global headers:0kB
muxing overhead -100.000000%
```

La opció `map` serveix per mapejar els streams continguts a l'entrada amb els de la sortida —no s'ha d'oblidar que un contenidor multimèdia pot portar diversos *streams* d'àudio i de vídeo—. Tot i que s'està treballant amb un únic *stream* a l'entrada i a la sortida, `segment` requereix que s'expliciti el mapejat.

Com era d'esperar 60 segons a talls de 10 segons, s'han generat 6 fitxers en comptes d'un.

```
3.8M 2012-05-25 13:14 outtest000.mp4
3.8M 2012-05-25 13:14 outtest001.mp4
3.6M 2012-05-25 13:14 outtest002.mp4
3.7M 2012-05-25 13:14 outtest003.mp4
3.5M 2012-05-25 13:14 outtest004.mp4
3.7M 2012-05-25 13:14 outtest005.mp4
```


Figura 4.9: Llista de reproducció



És possible posar tots sis vídeos com una llista de reproducció en un reproductor de vídeo, i es reproduiran un darrera l'altre (figura). La sensació visual és de continuïtat, amb un salt mínim entre vídeos.

Es interessant analitzar els vídeos resultants. En concret, observant al informació de duració i temps d'inici (**duration** i **start**) es pot veure que els diferents vídeos es consideren part d'un tot, ja que els vídeos tenen un temps d'inici que es correspon al moment en que començarien si s'estigués treballant amb un vídeo únic. També es pot avaluar el salt entre vídeos (diferència entre la duració del 0 i l'inici del 1, i així consecutivament) i es veu que està en 2 centèsimes de segon. Tenint en compte que a 25 fps la distància entre dos *frames* consecutius és de 4 centèsimes de segon, es pot considerar que hi ha la continuïtat desitjada entre vídeos.

```
# ffprobe outtest000.mp4
[...]
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'outtest000.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2mp41
    encoder          : Lavf54.6.100
  Duration: 00:00:10.22, start: 0.000000, bitrate: 3105 kb/s
  Stream #0:0(und): Video: mpeg4 (Advanced Simple Profile)
    (mp4v / 0x7634706D), yuv420p, 640x480 [SAR 1:1 DAR 4:3],
```

```
3102 kb/s, 26.61 fps, 100 tbr, 100 tbn, 100 tbc
Metadata:
  handler_name      : VideoHandler

# ffprobe outtest001.mp4
[...]
[mov,mp4,m4a,3gp,3g2,mj2 @ 0xa22e4a0] multiple edit list entries,
a/v desync might occur, patch welcome
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'outtest001.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2mp41
    encoder          : Lavf54.6.100
  Duration: 00:00:10.22, start: 10.240000, bitrate: 3048 kb/s
  Stream #0:0(und): Video: mpeg4 (Advanced Simple Profile)
    (mp4v / 0x7634706D), yuv420p, 640x480 [SAR 1:1 DAR 4:3],
    3046 kb/s, 26.61 fps, 100 tbr, 100 tbn, 100 tbc
  Metadata:
    handler_name      : VideoHandler

# ffprobe outtest002.mp4
[...]
[mov,mp4,m4a,3gp,3g2,mj2 @ 0xa5784a0] multiple edit list entries,
a/v desync might occur, patch welcome
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'outtest002.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2mp41
    encoder          : Lavf54.6.100
  Duration: 00:00:09.58, start: 20.480000, bitrate: 3071 kb/s
  Stream #0:0(und): Video: mpeg4 (Advanced Simple Profile)
    (mp4v / 0x7634706D), yuv420p, 640x480 [SAR 1:1 DAR 4:3],
    3068 kb/s, 26.72 fps, 100 tbr, 100 tbn, 100 tbc
  Metadata:
    handler_name      : VideoHandler

# ffprobe outtest003.mp4
[...]
[mov,mp4,m4a,3gp,3g2,mj2 @ 0xaab24a0] multiple edit list
entries, a/v desync might occur, patch welcome
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'outtest003.mp4':
```

```

Metadata:
  major_brand      : isom
  minor_version    : 512
  compatible_brands: isomiso2mp41
  encoder          : Lavf54.6.100
Duration: 00:00:10.25, start: 30.080000, bitrate: 3025 kb/s
Stream #0:0(und): Video: mpeg4 (Advanced Simple Profile)
(mpeg4 / 0x7634706D), yuv420p, 640x480 [SAR 1:1 DAR 4:3],
3023 kb/s, 26.54 fps, 100 tbr, 100 tbn, 100 tbc
Metadata:
  handler_name     : VideoHandler

```

Per altre banda s'ha observat que aquesta informació d'*start* diferent de 0 causa un comportament erroni en alguns reproductors —p.e. VLC 2.01 sobre Mac OSX 10.6.8—, generant un temps d'espera igual al valor d'*start* entre que s'inicia la reproducció i que apareixen les imatges en pantalla. Aquest comportament probablement no sigui tant un error com un excés de zel, ja que formalment un contenidor podria contenir diversos *streams* que comencessin en instants diferents.

Afegint la opció `-segment_list` es pot fer que `ffmpeg` generi directament una llista de reproducció que es podrà fer servir per reproduir tota la seqüència d'una tirada. En la versió amb que s'han realitzat les proves hi havia un comportament peculiar —*bug* o *feature*— i és que la longitud per defecte de la llista era 1, amb la qual cosa només el darrer vídeo de la sèrie apareixia a la llista. Això es pot arreglar posant `-segment_list_size N` on *N* haurà de ser un sencer suficientment gran.

4.4.6 Conclusions

Amb el resultat de les proves realitzades es pot concloure que FFmpeg és l'eina adient per la tasca que es vol dur a terme en aquest projecte —si més no, *una* eina adient—. Manca però definir amb més detall el format escaient per la sortida de vídeo, en funció de l'ús que se li vol donar, i afinar els paràmetres necessaris per obtenir-lo. Certament FFmpeg és una gran eina, però no és una eina senzilla —com no podia ser d'altre manera, per una eina que és coneguda com la navalla suïssa del processament de vídeo, un camp prou complex—.

Capítol 5

Gestió i monitorització de processos externs al programa

5.1 Introducció

Al realitzar les proves de FFmpeg (4.4) s'ha arribat a la conclusió que era l'eina que es faria servir per a desenvolupar aquest projecte. Un desenvolupament més en profunditat probablement requeriria fer servir directament les llibreries que aquest programari proveu, com ara `libavcodec`, i integrar-les en el codi. Ara be, això comporta haver de fer directament tota una serie de feines que ja estan resoltes en `ffmpeg`, com ara gestió d'errors, valors per defecte més comuns, i altres, i treballar a un nivell molt més baix (decodificació i codificació frame a frame, i fins i tot més).

L'abast d'aquest projecte es pot assolir perfectament, i de forma molt eficient, realitzant crides a `ffmpeg`. Això facilita a més la gestió de llicències de còdecs, formats i llibreries. L'eina que s'està desenvolupant requerirà que FFmpeg estigui instal·lat en el maquinari en que es vulgui fer servir, i tindrà més o menys capacitats en funció dels còdecs i llibreries que hi hagi instal·lats (com ja passa amb FFmpeg).

Aquesta estratègia porta que el següent repte d'aquest projecte sigui poder controlar una eina de línia de comandes (CLI), com és `ffmpeg`, des de dins el programari.

5.2 *system*

Si es desitja executar una ordre de línia de comandes des de dins un programa fet en c++ es pot fer servir *system*¹. A la descripció que se'n fa ja es pot veure que difícilment serà adient per al que aquí es desitja fer "*Once the command execution has terminated, the processor gives the control back to the program*".

El programa queda bloquejat mentre s'executa la ordre i a més no es té accés al resultat d'aquesta execució —només s'obté el retorn—, excepte que se'n redirigeixi la sortida a un fitxer i després s'obri aquest fitxer. Es modifica lleugerament l'exemple, per llençar `ffmpeg -version`, i se'n prova l'execució.

Llistat 5.1: `systemtest.cpp`

```
#include <stdio.h>
#include <stdlib.h>

int main ()
{
    int i;
    printf ("Checking if processor is available...");
    if (system(NULL)) puts ("Ok");
    else exit (1);
    printf ("Executing command...\n");
    i=system ("ffmpeg -version");
    printf ("The value returned was: %d.\n",i);
    return 0;
}
```

```
Checking if processor is available...Ok
Executing command...
ffmpeg version N-40962-g0da28d6
built on May 23 2012 16:23:18 with gcc 4.6.1
[...]
The value returned was: 0.
```

La sortida per consola del programa de prova inclou la sortida per consola de `ffmpeg -version`, però aquesta no ha sigut processada per el programa.

Com s'ha dit, *system* serveix per *executar* una ordre de línia de comandes, i serà adient si es vol llençar un procés breu —o no és rellevant que el procés

¹<http://www.cplusplus.com/reference/cstdlib/system/>

que el llença estigui bloquejat durant l'execució— i només es té interès en el resultat. El que es desitja en aquest apartat és trobar la manera adient de llençar `ffmpeg`, i *controlar-ne* l'execució.

5.3 *popen*

Es desitjaria poder rebre la sortida de `ffmpeg` segons es va produint, en lloc d'esperar al final de l'execució —si més no per què l'execució pot trigar hores!—. És possible treballar amb *pipes*, per aconseguir-ho. Això es pot fer *the hard way* com es pot veure en una qüestió² a *stackoverflow* —inclou un exemple—, però també hi ha capes d'abstracció que ho simplifiquen.

Una d'elles és *popen*³. Fent servir un petit exemple tret de la mateixa qüestió anterior de *stackoverflow*, es verifica que es té accés a la sortida de l'execució segons es produeix, i que es pot processar.

Llistat 5.2: `popentest.cpp`

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main(int argc, char** argv) {
    FILE *pipe= popen("ffmpeg -version", "r");
    if (!pipe) {
        cout << "ERROR";
        return 1;
    }
    char buffer[128]; string result = "";
    while(!feof(pipe)){
        if(fgets(buffer, 128, pipe) != NULL)
            result+= buffer;
        cout << "-X-";
    }
    pclose(pipe);
    cout << "pipe closed" << endl;
    cout << result;
    return 0;
}
```

²<http://stackoverflow.com/questions/478898/how-to-execute-a-command-and-get-output-of-command-within-c>

³<http://pubs.opengroup.org/onlinepubs/009696799/functions/popen.html>

```
-X--X--X--X--X--X--X--X--X--X--X--X-pipe closed
ffmpeg version N-40962-g0da28d6
built on May 23 2012 16:23:18 with gcc 4.6.1
[...]
```

El pas següent seria llençar una execució una mica més complexa, i veure'n el resultat. Ja que el que es desitja és gravar de càmera, es pot fer la prova llençant `ffmpeg` amb els paràmetres necessaris per capturar des de la càmera, tal i com es va fer a les proves de gravació.

Llistat 5.3: `popentestffmpeg.cpp`

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main(int argc, char** argv) {
    FILE *pipe= popen("ffmpeg -i
        \"rtsp://192.168.1.222/mpeg4/1/media.amp\" -f mp4
        -c:v copy -t 10 capturetest.mp4\", \"r\");
    if (!pipe) {
        cout << "ERROR";
        return 1;
    }
    char buffer[128]; string result = "";
    while (!feof(pipe)){
        if (fgets(buffer, 128, pipe) != NULL)
            result+= buffer;
        cout << "-X-";
    }
    pclose(pipe);
    cout << "pipe closed" << endl;
    cout << result;
    return 0;
}
```

```
ffmpeg version N-40962-g0da28d6
  Copyright (c) 2000-2012 the FFmpeg developers
[...]
```

Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':

```
[...]
```

Output #0, mp4, to 'capturetest.mp4':


```
[...]
Stream mapping:
  Stream #0:0 -> #0:0 (copy)
Press [q] to stop, [?] for help
[...]
video:8296kB audio:0kB global headers:0kB muxing overhead 0.030430%
-X-pipe closed
```

Tot i que la sortida de `ffmpeg` s'ha barrejat amb la sortida per consola de l'aplicació de prova, es pot veure que no ha sigut processada per aquesta. El codi de control que s'ha fet servir, `-X-`, juntament amb el text *pipe closed* permeten veure que l'execució de `ffmpeg` per gravar un petit fragment *no ha tret res per la sortida estàndard!*⁴.

Tal i com s'ha fet servir *popen* només ha obert un pipe des de la sortida estàndard del procés llençat fins al programa des que es llença. És una opció millor que *system*, però encara no és suficient.

És pot provar de redirigir la sortida d'error cap a la sortida estàndard en fer la crida.

Llistat 5.4: crida `popen` amb `stderr > stdout`

```
FILE *pipe= popen("ffmpeg -i
  \\"rtsp://192.168.1.222/mpeg4/1/media.amp\\" -f mp4
  -c:v copy -t 10 capturetest.mp4 2>&1", "r");
```

(No es copia el programa font ni la sortida complerta de nou per repetitiu)

```
-X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X-
-X--X--X--X--X--X--X--X--X--X--X--X--X--X--X--X-pipe closed
ffmpeg version N-40962-g0da28d6
Copyright (c) 2000-2012 the FFmpeg developers
  built on May 23 2012 16:23:18 with gcc 4.6.1
[...]
```

Es pot veure com ara si que s'ha capturat la sortida complerta en el programa de prova. Tot i això aquesta solució presenta dos problemes. El primer, que s'està barrejant ambdues sortides de `ffmpeg`. Tot i que no sembla greu per l'ús que se'n fa d'aquestes sortides a `ffmpeg`, no és una solució

⁴El fet que `ffmpeg` faci servir la sortida d'error en comptes de la sortida estàndard és degut a que la sortida d'error acostuma a ser directa mentre la estàndard sol passar per un *buffer*

gaire neta. A més, s'estaria fent *buffer* de la sortida d'error. I segon, encara manca poder accedir a l'entrada estàndard del procés llençat. `ffmpeg` permet aturar l'execució netament amb una `q` a l'entrada estàndard, i el programa desenvolupat hauria de poder fer-ne us.

5.4 *popenRWE, pstreams*

Seria desitjable per tant poder obrir tres *pipes*, una per la sortida estàndard, una altre per la sortida d'error, i una darrera per a connectat amb la entrada estàndard. D'aquesta manera es podria obtenir tota la sortida i interactuar amb el procés llençat mitjançant l'entrada.

Això evidentment és pot fer, com ja s'ha dit, *the hard way*, ja que *popen* és un *wrapper* per treballar amb *fork,exec* i *pipe*. Hi ha altres opcions, com ara *popenRWE*⁵ i *pstreams*⁶ que permeten també abstroure aquest nivell, proporcionant accés a tots tres streams.

No s'entra en detall a estudiar aquestes solucions, ja que s'ha optat per una altre, fer servir el *QProcess* que proporciona el *framework* de *Qt*, ja que a més de permetre accedir a aquestes mateixes informacions en facilita l'ús.

5.5 *QProcess*

A més de les llibreries per a interfície gràfica d'usuari el *Qt Framework*⁷ proporciona altres API. En concret la classe *QProcess*⁸ es pot fer servir *to start external programs and to communicate with them*. A més de permetre comunicar-se amb el procés permet fer-ho fent servir el sistema de *signals* i *slots*⁹ propi de *Qt*, amb la qual cosa el programa pot estar atent a l'usuari (o al que calgui), i rebrà avís cada cop que hi hagi quelcom per llegir.

Fer servir el *Qt Framework* té l'avantatge afegit que, com a mínim sobre el paper, és *cross-platform*, mentre que els sistemes que s'han vist a base de *pipes* requereixen sistemes *POSIX*.

Caldria provar que efectivament s'aconsegueix controlar l'execució de `ffmpeg` mitjançant una classe *QProcess*. Novament a una qüestió a *stackoverflow*, es pot trobar un exemple¹⁰ que es pot adaptar. Es verificarà que es

⁵<http://www.jukie.net/bart/blog/popenRWE>

⁶<http://pstreams.sourceforge.net/>

⁷<http://qt.nokia.com/>

⁸<http://doc.qt.nokia.com/4.7-snapshot/qprocess.html>

⁹<http://doc.qt.nokia.com/4.7/signalsandslots.html>

¹⁰<http://stackoverflow.com/questions/6600973/asynchronously-run-console-output-and-gui-in-qt>

te accés a les sortides, a mida que es van produint, mitjançant les *signals*. En aquest cas el codi del programa es una mica més extens, i es treballarà amb objectes.

Primer es definirà la classe `MyProcessStarter`, en el fitxer de capçalera corresponent. Es pot veure com es crida la macro `QOBJECT`, com es defineix el membre privat de tipus `QProcess`, i com es defineixen els *slots* per a rebre tant la sortida estàndard com la d'error.

Llistat 5.5: MyProcessStarter.h

```

#ifndef MYPROCESSSTARTER_H
#define MYPROCESSSTARTER_H
#include <QObject>
#include <QProcess>
#include <QDebug>
class MyProcessStarter : public QObject {
    Q_OBJECT
public:
    MyProcessStarter () : QObject () {};
    void StartProcess ();
private slots:
    void readStandardOutput ();
    void readStandardError ();
private:
    QProcess *myProcess;
};
#endif /* MYPROCESSSTARTER_H */

```

A continuació s'implementa el comportament de les funcions definides. Els *slots* que reben les sortides del procés executat només escriuen el que reben, fent servir `QDebug()`, però ja es pot veure com permetrien processar-ho. Es fa servir una cadena "STD: " per distingir la sortida estàndard.

La funció `StartProcess()` prepara la crida al programa extern, instancia la classe `QProcess` per controlar-lo, connecta les *signals* que emetrà el procés als *slots* que es desitja que en manipulin el resultat, i llença (`start()`) el procés. Finalment espera a que acabi l'execució —aquesta darrera part només es necessària perquè el programa de prova no fa res més i no es vol que surti, matant el procés, abans de temps—.

Llistat 5.6: MyProcessStarter.cpp

```

#include "MyProcessStarter.h"
void MyProcessStarter::StartProcess () {
    QString program = "ffmpeg";

```

```

QStringList arguments;
arguments << "-y";
arguments << "-i" <<
    "rtsp://192.168.1.222/mpeg4/1/media.amp";
arguments << "-f" << "mp4";
arguments << "-c:v" << "copy";
arguments << "-t" << "10";
arguments << "capturetest.mp4";

myProcess = new QProcess(this);

connect(myProcess, SIGNAL(readyReadStandardOutput()),
        this, SLOT(readStandardOutput()));
connect(myProcess, SIGNAL(readyReadStandardError()),
        this, SLOT(readStandardError()));

myProcess->start(program, arguments);
while(!myProcess->waitForFinished());
}
void MyProcessStarter::readStandardOutput() {
    QByteArray processOutput;
    processOutput = myProcess->readAllStandardOutput();
    qDebug() << "STD: " << QString(processOutput);
}
void MyProcessStarter::readStandardError() {
    QByteArray processOutput;
    processOutput = myProcess->readAllStandardError();
    qDebug() << QString(processOutput);
}

```

El main només instancia un objecte `MyProcessStarter` i crida al mètode `StartProcess()`.

Llistat 5.7: main.cpp

```

#include "MyProcessStarter.h"

int main(int argc, char *argv[])
{
    MyProcessStarter s;
    s.StartProcess();
}

```

Es prova l'execució, i es pot veure com es rep i es processa la sortida de `ffmpeg`. L'absència de cap "STD: " ens permet tornar a comprovar com

ffmpeg escriu el *log* per la sortida d'error.

```
ffmpeg version N-40962-g0da28d6
[...]
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
[...]
Output #0, mp4, to 'capturetest.mp4':
[...]
Stream mapping:
  Stream #0:0 -> #0:0 (copy)
Press [q] to stop, [?] for help
[...]
video:5978kB audio:0kB global headers:0kB muxing overhead 0.029096%
```

Per comprovar la idoneïtat de la solució manca verificar que es pot aturar l'execució ordenadament abans no acabi la captura programada —abans no passin `-t` segons—. Per això es modificarà la funció `StartProcess()`, posant el paràmetre `-t` a 30 segons i canviant les dues darreres línies:

```
myProcess->start (program , arguments) ;
while (!myProcess->waitForFinished ()) ;
```

Per:

Llistat 5.8: `MyProcessStarter.cpp` amb escriptura

```
myProcess->start (program , arguments) ;
if (!myProcess->waitForStarted ()) return ;
qDebug () << "Started!";

myProcess->waitForFinished (10000) ;
qDebug () << "10 seconds!";

myProcess->write ("q") ;
while (!myProcess->waitForFinished ()) ;
```

`waitForStarted` esperarà a que arrenqui el procés, i retornarà `true`. Si en 30 segons no hagués arrencat tornarà fals i s'avortarà l'execució.

`waitForFinished(10000)` esperarà 10 segons o fins que acabi l'execució. Com ara `-t` és 30 segons, els 10 segons s'esgotaran mentre el procés segueix en marxa.

Aleshores s'escriu la `q` a l'entrada estàndard de `ffmpeg`, i l'execució hauria d'acabar tancant correctament el vídeo. Si no respon l'execució continuarà fins que passin els 30 segons. Es pot veure que la prova ha sigut un èxit.

```

Started!
ffmpeg version N-40962-g0da28d6
[...]
Input #0, rtsp, from 'rtsp://192.168.1.222/mpeg4/1/media.amp':
[...]
Output #0, mp4, to 'capturetest.mp4':
[...]
Stream mapping:
  Stream #0:0 -> #0:0 (copy)
Press [q] to stop, [?] for help
[...]
frame=  99 fps= 10 q=-1.0 size=    5876kB time=00:00:09.71
  bitrate=4956.6kbits/s
10 seconds!
frame= 100 fps= 10 q=-1.0 Lsize=    5935kB time=00:00:09.81
  bitrate=4955.5kbits/s
video:5933kB audio:0kB global headers:0kB muxing overhead 0.028918%

```

Per suposat es pot obrir el vídeo generat, verificar que s'ha tallat als deu segons i que es pot visionar correctament. I també verificar-ho amb `ffprobe`.

```

# ffprobe capturetest.mp4
[...]
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'capturetest.mp4':
  Metadata:
    major_brand      : isom
    minor_version    : 512
    compatible_brands: isomiso2mp41
    title           : Media Presentation
    encoder         : Lavf54.6.100
  Duration: 00:00:09.82, start: 0.000000, bitrate: 4950 kb/s
  Stream #0:0(und): Video: mpeg4 (Advanced Simple Profile)
  (mp4v / 0x7634706D), yuv420p, 640x480 [SAR 1:1 DAR 4:3],
  4949 kb/s, 10.18 fps, 100 tbr, 90k tbn, 100 tbc
  Metadata:
    handler_name     : VideoHandler

```

Capítol 6

Reproducció del vídeo gravat

La reproducció del vídeo gravat s'ha separat en dos supòsits, directa i a través de web. El primer es realitza amb eines (reproductors multimèdia o eines d'anàlisi de vídeo) existents al mercat, i per el segon s'ha de proporcionar una petita aplicació web.

No es té en compte la separació en reproducció a posteriori i reproducció mentre s'està realitzant la gravació, ja que es considera que el primer cas queda automàticament resolt si el segon ho està. Per a poder reproduir el vídeo mentre s'està gravant es fa la gravació en segments com s'ha vist a 3.1.1. A partir d'aquí el problema és reproduir aquests segments. Si la gravació està en marxa es seguiran generant segments, però no afecta. Si la gravació està aturada no se n'afegiran, i encara afecta menys.

6.1 Reproducció en reproductors multimèdia

Com s'ha vist al fer les proves de gravació segmentada 4.4.5, `ffmpeg` permet generar una llista de reproducció amb els talls (segments) que s'estan generant. Aquesta llista es va omplint a mesura que es tanquen els nous segments, i per tant obrint-la en qualsevol moment es tindrà una llista que inclourà la gravació complerta fins aquell moment. Aquesta mena de llistes es poden obrir en la majoria de reproductors multimèdia —*tots seria massa agosarat*—.

També és possible accedir directament al tall que es desitgi, si es té clar el moment en que ha succeït el que es vol revisar, ja que el nom dels segments indica l'ordre i la data de creació el moment en que s'han gravat.

Els segments gravats presenten però certa problemàtica en alguns reproductors multimèdia com ja s'havia descobert al fer les proves de segmentació (4.4.5). En concret dels reproductors que s'han provat VLC —en Mac i

Ubuntu— i QuickTime donen problemes (en el cas de QuickTime ni tan sols es reproduueix), mentre que MPlayer OSX extended funciona quasi perfecte i Gnome Mplayer i Totem —aquests dos darrers provats sobre Ubuntu— funcionen a la perfecció.

A 6.3 es tracta amb detall aquesta problemàtica —que podria ser deguda a un *bug* en FFmpeg—. En aquest punt del desenvolupament no es considera bloquejant, ja que hi ha reproductors que si permeten veure el vídeo, però s'hauria de buscar una solució, un *workaround* o replantejar la manera d'aconseguir la reproducció amb la gravació en marxa.

Com a informació addicional, el problema és dependent del format utilitzat (mp4), i per exemple no apareix fent servir mpegts —un format orientat a l'streaming—.

6.2 Reproducció del vídeo a través de web

Una de les novetats que va introduir HTML 5 és una etiqueta `<video>` pensada per a la reproducció de vídeo amb la màxima senzillesa, sense necessitat d'afegir **plugins** ni reproductors incrustats. Aquesta etiqueta ja és implementada per els principals navegadors, simplificant en gran mesura aquest punt.

Es presenta però un problema de formats. L'especificació d'HTML5 — com és natural— no explicita la implementació, ni tan sols els formats que s'han de poder reproduir —això potser ja no és tan natural, ja que merma en gran mesura la compatibilitat—. Firefox no implementa la reproducció de formats privatis, en concret mp4, amb la qual cosa quedaria descartat per aquest projecte. Chrome per ara suporta H.264 sobre mp4, però no està gens clar quant de temps ho seguirà fent —al blog de chromium¹ diu un parell de mesos, però en fa cinc i el format encara hi és suportat. És probable que el fet que els dispositius amb iOS5 el suportin, i la seva extensió d'ús, faci que no sigui un format tan fàcil de deixar de banda—. A la Wikipedia² es pot trobar una taula de compatibilitat entre navegadors —o millor dit de la seva absència—.

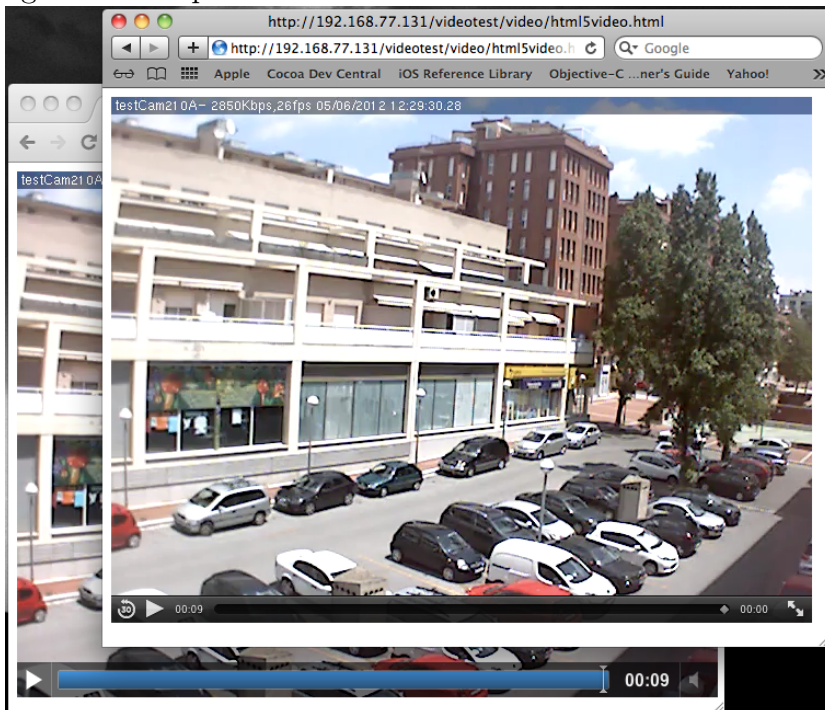
En la data de realització d'aquest projecte, però, el format mp4 és un requeriment, ja que (juntament amb mov) és el més adient per algunes de les eines d'anàlisi més usades en l'esport professional³. Es decideix doncs fer les proves amb Safari i Chrome.

¹<http://blog.chromium.org/2011/01/html-video-codec-support-in-chrome.html>

²http://en.wikipedia.org/wiki/HTML5_video

³La gama SportsCode de Sportstec

Figura 6.1: Reproducció video amb HTML5 a Chrome i Safari



6.2.1 L'etiqueta <video>

L'etiqueta <video> permet reproduir un vídeo en una plana web amb tanta senzillesa com es tractaven les imatges fixes fins ara.

Llistat 6.1: html5video.html

```
<!DOCTYPE html>
<html>
<body>
  <video width="640" height="480" controls="controls">
    <source src="test001.mp4" type="video/mp4" />
    Your browser does not support the video tag.
  </video>
</body>
</html>
```

A la figura 6.1 es pot veure el resultat. S'ha capturat la sortida a Chrome i Safari, per ressaltar que els *controls* són diferents, però en tots dos casos permeten desplaçar-se fàcilment per el vídeo, pausar i reanudar la reproducció.

Aquesta etiqueta no permet la gestió de llistes de vídeo, però com es veurà a 6.2.2 això es pot assolir sense massa complicació fent servir *javascript*.

En aquest punt del desenvolupament torna a aparèixer el problema de reproducció que ja s'ha comentat. Els reproductors de vídeo que implementen tant Safari com Chrome presenten el problema, i el presenten (com QuickTime) en la seva vessant més greu: ni tan sols es reproduïx el vídeo. Tot i que abans s'ha descartat que fos un problema bloquejant per la presència d'alternatives, en aquest punt s'ha de re-avaluar això, ja que la reproducció via web era un requisit imprescindible.

Es considera que no es pot complir aquest requisit, i es posposa per a futures versions la cerca d'una solució. En les conclusions (7) s'aprofundeix en aquest aspecte. Tot i això es continua el desenvolupament del reproductor, ja que com es comenta en 6.3 sembla ser que el problema és un *bug* de *FFmpeg* i no es pot descartar la seva solució.

6.2.2 Gestió de llistes

Abans de fer cap moviment per perfeccionar el reproductor de vídeo, cal tenir en compte que existeixen diverses opcions de reproductors, alguns d'ells gratuïts i de codi obert, que es poden incrustar a la plana i donen moltes opcions de personalització⁴.

De tota manera, com l'objectiu principal d'aquest projecte és fer una prova de concepte i aprendre el màxim sobre el tema, es continuarà treballant directament sobre l'etiqueta `<video>` i les múltiples opcions que proporciona.

Per gestionar una llista de reproducció ens enfrontem a dos problemes principals. El primer generar una llista des del contingut del directori de vídeos⁵ i el segon gestionar la reproducció dels vídeos d'aquesta llista.

A l'apartat anterior s'ha vist que `<video>` té un atribut `src` que apunta al vídeo que es reproduirà. Tal i com es fa amb les imatges es pot fer servir *javascript* per modificar el contingut d'aquest atribut. Però no hi ha prou amb això. Cercant a *stackoverflow*⁶ es troba que s'ha d'indicar expressament que es desitja carregar i reproduir el vídeo.

Llistat 6.2: carregar un nou video

```
videoTag.src = "newVideo";
```

⁴alguns d'aquests reproductors inclouen un fallback a flash quan el navegador no pot gestionar el format del vídeo, la qual cosa solucionaria les limitacions del mp4 i poder fins i tot el *bug* de *FFmpeg*... però això subverteix totalment el sentit de l'etiqueta vídeo.

⁵Evidentment es pot fer servir la llista generada en la gravació.

⁶<http://stackoverflow.com/questions/2772409/change-video-being-played-in-html5-video>

```
videoTag.load();
videoTag.play();
```

Per tant es pot canviar el vídeo tenint una llista de vídeos que cridi a una funció que implementi aquest llistat cada cop que es seleccioni un. Això ens porta a l'altre problema, generar la llista. Aquesta és una tasca per a PHP.

Cercant a Internet s'ha trobat, al blog Bionicworks un petit exemple⁷ gairebé funcional que no només implementa la generació de la llista sinó també passa automàticament al següent quan el vídeo que s'està reproduint arriba al final —tot i que l'exemple té un *bug* i només funciona si es comença a reproduir per el primer de la llista—. S'ha agafat aquest exemple com a base del desenvolupament de la prova de concepte del reproductor, si bé s'ha modificat substancialment.

Una diferència fonamental amb l'exemple és que en aquest cas l'ordre dels vídeos és rellevant. Per tant la lectura del directori dels vídeos s'ha de fer amb una funció que en garanteixi l'ordre. Es fa servir *scandir*⁸.

Llistat 6.3: LLegir directori

```
$dir = "video";
$files = scandir($dir);
```

I es pot generar una llista HTML. Es fa servir una llista desordenada, fent que cada vídeo sigui un link —tot i que se n'anul·la la resposta al *click*—.

Llistat 6.4: Llista de vídeos

```
$j=0;
foreach($files as $file) {
    if($file != '.' && $file != '..'){
        echo "<li>";
        echo "<a id='track_" . $j . "'
            href='". $dir . "/" . $file . "' onclick='return
            false;'>";
        echo $file . "</a>";
        echo "</li>";
        $j++;
    }
}
```

Ara s'ha d'afegir la resposta als click, que carregui el vídeo a que apunta el link a `<video>`. Això es farà en Javascript directament al client, fent servir *jQuery*, ja que facilita molt fer una feina neta.

⁷<http://www.bionicworks.com/php/generate-a-playlist-for-html5-video>

⁸<http://www.php.net/manual/en/function.scandir.php>

Llistat 6.5: Resposta al click des de la llista

```
$(document).ready(function(){
    $('#list > li a').live('click', function(e) {
        var path = $(this).attr('href');
        i=$(this).attr('id').split("_")[1];
        loadVideo(path);
    });
});
```

La línia `i=$(this).attr('id').split("_")[1];` permet mantenir un punter al vídeo que s'està reproduint en cada moment. I `loadVideo(path)` hauria de resultar evident.

Llistat 6.6: `loadVideo(path)`

```
function loadVideo(path) {
    var player =
        document.getElementsByTagName('video')[0];
    player.src = path;
    player.load();
    player.play();
}
```

L'etiqueta `<video>` en HTML 5 té diversos *events*. En concret quan s'acaba el vídeo que s'està reproduint genera un `ended`. És possible per tant associar una funció a aquest event.

Llistat 6.7: Next Video

```
var player = document.getElementsByTagName('video')[0];
player.addEventListener('ended', nextVideo, false);
[...]
function nextVideo() {
    loadVideo($('#list > li a')[++i%listlength].href);
}
```

El llistat complet del reproductor es pot trobar a l'annex D.

6.2.3 Opcions avançades amb `<video>`

L'etiqueta `<video>` admet un control molt fi de la reproducció. Genera diversos *events*, i permet modificar diversos paràmetres de la reproducció. A la web del W3C⁹ es pot trobar un exemple¹⁰ que permet jugar amb els

⁹World Wide Web Consortium (W3C), <http://www.w3.org>

¹⁰<http://www.w3.org/2010/05/video/mediaevents.html>

paràmetres, i observar els *events* que es generen.

En el codi del reproductor que es pot trobar a l'annex D s'han afegit dos petits elements a tall d'exemple. Un comptador que presenta l'instant que s'està reproduint del vídeo, aprofitant l'*event timeupdate*. I una mena de botó que fa que el vídeo es reproduïxi marxa enrera, jugant amb l'atribut *playbackrate*. S'ha de tenir en compte, però, que la reproducció marxa enrera pot no anar massa be, o fins i tot no funcionar¹¹.

6.3 El mur

Com s'ha explicat en els apartats anteriors, en tractar de resoldre la reproducció dels vídeos resultants de la gravació segmentada s'ha trobat un *bug* de *FFmpeg*. Desgraciadament aquest *bug* fa que aquest projecte, tal i com s'ha concebut, resulti impossible de completar en aquests moments. S'ha topat amb un mur infranquejable... per ara.

Com es comenta en les conclusions, hi ha diverses possibilitats per franquejar aquest entrebanc, però queden totes per a futurs desenvolupaments.

S'ha considerat prou interessant, i relacionat amb el funcionament de les comunitats de programari lliure, el procés de detecció d'un problema en un projecte existent, la interacció amb els desenvolupadors i la apertura d'un *ticket*. Per això s'ha decidit comentar-ho aquí, ni que sigui de forma breu.

En trobar-se el problema la cerca per Internet no va donar fruit. El més proper que es va trobar era un ticket al gestor de *issues* de *FFmpeg*¹², però que estava tancat com a invàlid. En no trobar més informació, es va optar per consultar en la llista de distribució per a usuaris de *FFmpeg* a veure si algun altre usuari se n'havia trobat o algun desenvolupador podia donar alguna idea d'on era el problema. En aquesta fase es considerava que podia ser un problema de configuració de la gravació.

Es pot seguir el intercanvi de mails en els arxius de la llista de distribució¹³. Bàsicament es va tractar de realitzar una sèrie de proves per ajustar el problema, que va dur al desenvolupador a determinar que era un defecte i calia obrir un *ticket*.

S'ha obert per tant, el *ticket* #1425¹⁴

¹¹<http://stackoverflow.com/questions/5277293/is-html5-video-able-to-be-played-in-reverse>

¹²<https://ffmpeg.org/trac/ffmpeg/ticket/910>

¹³<http://ffmpeg.org/pipermail/ffmpeg-user/2012-June/006968.html>

¹⁴<https://ffmpeg.org/trac/ffmpeg/ticket/1425>

Capítol 7

Conclusions

Per donar per finalitzada la feina he de dir que tot i l'esforç que m'ha representat he fruit molt amb aquest projecte. No ha sortit ni de bon tros com me l'imaginava al plantejar-lo, la feina de codificació pura i dura ha sigut menor, i la feina d'investigació, aprenentatge i proves ha sigut molt més profunda i complexa. I probablement això tingui molt a veure amb la satisfacció que em produeix.

I puc dir sense por a equivocar-me que les coses que he après en aquest projecte —i en el màster en general— afecten de forma definitiva la meua manera de treballar, i marquen en bona mida els desenvolupaments que faré en els propers mesos —com a mínim!—.

7.1 Objectius complerts

Revisant els objectius (1.1) o encara millor la proposta de desenvolupament (3.3) resulta evident que no ha donat temps a tancar el 100% del que es volia.

Si s'ha aconseguit realitzar la gravació des de les càmeres definides —i de tal manera que afegir més càmeres serà trivial si l'API d'accés al stream de vídeo és pública—. S'ha aconseguit també obtenir, de forma suficientment independent de la font de vídeo, els formats de sortida desitjats —i amb la qualitat desitjada—, i fent la transcodificació al vol —complint l'objectiu principal de reduir (a 0!!) el temps d'espera—.

La senzillesa d'ús només els usuaris podran valorar si s'ha aconseguit suficientment, però sense cap dubte passar de tenir que introduir manualment els paràmetres —i pitjor encara: tenir que *conèixer-los!*— a gestionar una GUI que només presenta un parell de desplegable i un parell de botons és una bona millora.

Només haver aconseguit la transcodificació al vol, reduint el temps d'es-

pera del programari que fem servir actualment justifica per si mateix l'èxit del projecte, com a mínim des d'un punt de vista comercial.

7.2 Objectius no complerts

Desgraciadament s'ha trobat un *bug* en FFmpeg que fa que en el format desitjat la segmentació de la sortida no sigui compatible amb molts reproductors, entre ells l'etiqueta `<video>` de HTML5 que es pensava fer servir per l'*Instant Review* via web. Aquest és el forat més gros que deixa aquest projecte, i al que s'haurà de dedicar futurs esforços.

Com a primera mesura, s'haurà de buscar una manera de poder donar la funcionalitat, sigui canviant de reproductor, o buscant formats alternatius, o fins i tot fent dues gravacions en paral·lel —una per la web en un format tipus webm, l'altre en mp4 per a l'anàlisi a posteriori—.

7.3 Futurs desenvolupaments

Al final, com ja he comentat, el projecte s'ha convertit més en l'estudi de viabilitat, anàlisi tècnica i desenvolupament d'un prototipus o prova de concepte que no pas un desenvolupament complert i adient per a un usuari final. Per tant el primer “futur desenvolupament” que es pot plantejar —millor dit, que em plantejo— és una nova versió —no goso dir *2.0*— que completi el desenvolupament.

En aquest sentit cal millorar el disseny de les interfícies d'usuari, i per suposat realitzar una extensa fase de proves per garantir-ne la robustesa.

Una altre possibilitat evident —i que també s'ha comentat ja— és completar la separació client-servidor i fer la gestió complerta de l'aplicació des de una interfície web. No només la revisió del vídeo gravat, sinó també engegar i aturar la gravació, i qualsevol funcionalitat futura.

Evidentment això es pot plantejar com a alternativa, però donat que no hi ha res que impedeixi accedir via web a un servei des de la mateixa màquina que l'hostatja —sempre que hi tinguem instal·lat un navegador, és clar!—, una vegada es fa l'esforç a fer una interfície d'usuari adient via web no té massa sentit dividir esforços i mantenir també una GUI “local”.

Un altre camp de millora és la gestió avançada de les càmeres IP. Aquesta mena de càmeres —com a mínim les del fabricant Axis— donen moltes opcions de control i de configuració que són accessibles a través de la API pública. El primer element seria afegir la visió d'imatges en viu. Però el més necessari i agraït pels usuaris seria afegir la gestió de moviment. Moltes

d'aquestes càmeres són PTZ¹ i s'haurien de controlar des de l'aplicació.

Entre les opcions més interessants de configuració cal destacar alguns paràmetres d'ajust d'imatge que, de poder ser controlats des de l'aplicació —i no és complex—, donarien molta millor qualitat de vídeo, sobre tot si es grava en exterior. Per exemple la compensació de contrallum (*backlight compensation*), el marge dinàmic ample (*Wide Dinamic Range*), o la compensació de la vibració de la càmera.

Finalment, però no menys important, un desenvolupament que podria sorgir d'aquest projecte és la resolució del Ticket #1425² que s'ha hagut d'obrir a FFmpeg. Sense dubte és un desenvolupament només apte per a valents, però possiblement sigui el més enriquidor a nivell de desenvolupament personal de tots els que es plantegen.

7.4 Aprentatge

Aquest Màster, i com a punt culminant aquest projecte, m'ha permès tocar —i aprendre— temes nous que m'han resultat molt interessants i que, com he dit al inici de les conclusions, han influït en la meva forma de treballar.

Evidentment fa molts anys que faig servir control de versions en els meus desenvolupaments³ —si no fos així tampoc no gosaria confessar-ho—. Però feia servir Subversion. Fa temps que li donava voltes a Git, però no havia trobat el moment. I aquest desenvolupament ha sigut la oportunitat perfecta. Tot i que estava prou satisfet amb Subversion, he de remarcar que Git m'ha semblat molt còmode d'usar, i molt àgil. I el fet de ser descentralitzat aporta una facilitat impressionant per gestionar el desenvolupament entre diversos ordinadors.

Abans de l'assignatura de programació no tocava el C des de la carrera, i el C++ apenes. I el *Qt framework* ha sigut una experiència totalment nova... i sorprenentment fàcil.

Quant al tema dels formats i contenidors de vídeo, tot i que fa anys que he tocat aquests temes —si més no de forma tangencial— en aquests mesos de proves, ajustos i maldecaps n'he après molt més. Sobre tot, parafrasejant al filòsof, que apenes en se res.

I no vull acabar sense parlar de L^AT_EX. Fa uns anys havia fet algun petit intent, però sense continuïtat. El redactat d'aquesta memòria m'ha donat l'espai necessari per aprofundir-hi —molt lleugerament—, i per fer-me preguntar que he estat fent tots aquests anys amb gestors de documents

¹*Pan* o moviment horitzontal, *Tilt* o moviment vertical i *Zoom*, que no cal traduir

²<https://ffmpeg.org/trac/ffmpeg/ticket/1425>

³Tinc proves!

WYSIWYG⁴. No dic que sigui la única solució bona⁵, ni tan sols que sigui la millor —no la conec prou i les alternatives tampoc—, però és una molt bona eina, i combinada amb un gestor de versions és simplement fantàstica. I per fer una bona feina cal fer servir l'eina adient.

7.5 Programari Lliure

Aquest Màster és de Programari Lliure. I si em vaig decidir a fer-lo, va ser perquè és un món que en certa mesura ja coneixia i m'atreia. Però el Màster m'ha permès aprofundir molt més, i conèixer projectes i comunitats més a fons. Al final, d'aquí uns mesos, el que em quedarà en el dia a dia —a més d'unes quantes tècniques de treball— són les diverses llistes de distribució a que m'he anat subscriuint⁶.

I és quelcom que recomano a tothom. No només per conèixer millor un programari o un projecte determinats, sinó per conèixer les persones que el formen i com interactuen. Fins i tot estant només d'oient, és una experiència enriquidora.

⁴*What You See Is What You Get*, tot i que s'hauria d'afegir “suposadament”

⁵alternatives a \LaTeX de renom serien DocBook o Texinfo, o Markdown tot i que sembla orientat a objectius més senzills

⁶Les darreres les de FFmpeg, però no les úniques

Bibliografia

Documentació de l'assignatura

- [1] Juan José Amor Iglesias, Israel Herraiz Tabernero, Gregorio Robles Martínez, *Desenvolupament de projectes de programari lliure*.
Fundació per a la Universitat Oberta de Catalunya, febrer 2006

FFmpeg

- [2] *FFmpeg home page*
<http://ffmpeg.org/>
- [3] *Compile FFmpeg on Ubuntu*
<https://ffmpeg.org/trac/ffmpeg/wiki/UbuntuCompilationGuide>

L^AT_EX

- [4] Wikibooks Community, *L^AT_EX*
<http://en.wikibooks.org/wiki/LaTeX/>
- [5] Tim Love, *Advanced L^AT_EX*.
University Of Cambridge, Engineering Department, August 30, 2006.
http://www.eng.cam.ac.uk/help/tpl/textprocessing/latex_advanced/latex_advanced.html
- [6] *Tex Stackexchange*.
A free, community driven Q&A for users of TeX, LaTeX, ConTeXt, and related typesetting systems. It is a part of the Stack Exchange network of Q&A websites, and it was created through the open democratic process defined at Stack Exchange Area 51.
<http://tex.stackexchange.com>

Git

- [7] Scott Chacon, *Pro Git*, Apress.
<http://git-scm.com/book>
- [8] Scott Chacon, *Everyday GIT With 20 Commands Or So*.
<http://schacon.github.com/git/everyday.html>
- [9] GitHub Team, *Git Reference*.
<http://gitref.org/>

Apèndix A

Text complet de la Llicència CC BY-NC-SA 3.0

Aquesta memòria es distribueix sota llicència
Reconeixement-NoComercial-CompartirIgual 3.0 No adaptada
CC BY-NC-SA 3.0



Es presenta per el aquí per comoditat el text legal complet de la llicència, copiat de bona fe des de la web. Tot i això per qualsevol referència legal s'hauria de consultar directament l'original que es pot trobar a <http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode>

Creative Commons **Attribution-NonCommercial-ShareAlike 3.0 Unported**

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE (“CCPL” OR “LICENSE”). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE,

YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

- (a) **“Adaptation”** means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image (“synching”) will be considered an Adaptation for the purpose of this License.
- (b) **“Collection”** means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(g) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.
- (c) **“Distribute”** means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
- (d) **“License Elements”** means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, Noncommercial, ShareAlike.
- (e) **“Licensor”** means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.

- (f) **“Original Author”** means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
- (g) **“Work”** means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
- (h) **“You”** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
- (i) **“Publicly Perform”** means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the

communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.

- (j) **“Reproduce”** means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. **Fair Dealing Rights.** Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. **License Grant.** Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- (a) to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
- (b) to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked “The original work was translated from English to Spanish,” or a modification could indicate “The original work has been modified.”;
- (c) to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
- (d) to Distribute and Publicly Perform Adaptations.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights described in Section 4(e).

4. **Restrictions.** The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- (a) You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(d), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(d), as requested.
- (b) You may Distribute or Publicly Perform an Adaptation only under: (i) the terms of this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-NonCommercial-ShareAlike 3.0 US) (“Applicable License”). You must include a copy of, or the URI, for Applicable License with every copy of each Adaptation You Distribute or Publicly Perform. You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License. You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly

Perform. When You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.

- (c) You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.
- (d) If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution (“Attribution Parties”) in Licensor’s copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and, (iv) consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., “French translation of the Work by Original Author,” or “Screenplay based on original Work by Original Author”). The credit required by this Section 4(d) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the

other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.

- (e) For the avoidance of doubt:
 - i. **Non-waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
 - ii. **Waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License if Your exercise of such rights is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(c) and otherwise waives the right to collect royalties through any statutory or compulsory licensing scheme; and,
 - iii. **Voluntary License Schemes.** The Licensor reserves the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License that is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(c).
- (f) Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan),

in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING AND TO THE FULLEST EXTENT PERMITTED BY APPLICABLE LAW, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THIS EXCLUSION MAY NOT APPLY TO YOU.

- 6. Limitation on Liability.** EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

- (a) This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

- (b) Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

- (a) Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
- (b) Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
- (c) If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
- (d) No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
- (e) This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
- (f) The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as

amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Apèndix B

Text de la llicència GNU General Public License, version 2

El programari desenvolupat en aquest projecte es distribueix sota llicència
GNU General Public License, version 2 (GPLv2)
i com a tal s'indica en el mateix programari.

Es presenta per el aquí per comoditat el text legal complet de la llicència, copiat de bona fe des de la web. Tot i això per qualsevol referència legal s'hauria de consultar directament l'original que es pot trobar a <http://www.gnu.org/licenses/gpl-2.0.html>

GNU GENERAL PUBLIC LICENSE Version 2, June 1991

Copyright © 1989, 1991 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License

applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

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

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

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

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

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

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

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

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program"

means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term “modification”.) Each licensee is addressed as “you”.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

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

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

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

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

not normally print such an announcement, your work based on the Program is not required to print an announcement.)

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

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

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

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

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

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

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.
6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions

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

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

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

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

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions

either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

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

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

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

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

one line to give the program’s name and a brief idea of what it does.

Copyright (C) yyyy name of author

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

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

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

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

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

Gnomovision version 69, Copyright (C) yyyy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for
details type ‘show w’.

This is free software, and you are welcome to redistribute it under
certain conditions; type ‘show c’ for details.

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, the commands you use

may be called something other than `show w` and `show c`; they could even be mouse-clicks or menu items—whatever suits your program.

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

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

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

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

Apêndix C

FFmpeg - sortida de configure

```
$ ./configure --enable-gpl \  
> --enable-libx264 \  
> --enable-x11grab  
install prefix           /usr/local  
source path              .  
C compiler               gcc  
ARCH                     x86 (generic)  
big-endian               no  
runtime cpu detection    no  
yasm                     yes  
MMX enabled              yes  
MMX2 enabled             yes  
3DNow! enabled           yes  
3DNow! extended enabled  yes  
SSE enabled              yes  
SSSE3 enabled            yes  
AVX enabled              yes  
CMOV enabled             no  
CMOV is fast             no  
EBX available            yes  
EBP available            yes  
debug symbols            yes  
strip symbols            yes  
optimize for size        no  
optimizations            yes  
static                   yes  
shared                   no  
postprocessing support   yes
```

C. FFMPEG - SORTIDA DE CONFIGURE

new filter support	yes	
network support	yes	
threading support	threads	
safe bitstream reader	yes	
SDL support	yes	
libdxva2 enabled	no	
libva enabled	no	
libvdpau enabled	no	
AVISynth enabled	no	
frei0r enabled	no	
gnutls enabled	no	
libaacplus enabled	no	
libass enabled	no	
libcdio support	no	
libcelt enabled	no	
libdc1394 support	no	
libfaac enabled	no	
libgsm enabled	no	
libmodplug enabled	no	
libmp3lame enabled	no	
libnut enabled	no	
libopencore-amrnb support	no	
libopencore-amrwb support	no	
libopenmv support	no	
libopenjpeg enabled	no	
libpulse enabled	no	
librtmp enabled	no	
libschrödinger enabled	no	
libspeex enabled	no	
libstagefright-h264 enabled	no	no
libtheora enabled	no	
libutvideo enabled	no	
libv4l2 enabled	no	
libvo-aacenc support	no	
libvo-amrwbenc support	no	
libvorbis enabled	no	
libvpx enabled	no	
libx264 enabled	yes	
libxavs enabled	no	
libxvid enabled	no	
openal enabled	no	

C. FFMPEG - SORTIDA DE CONFIGURE

openssl enabled no
zlib enabled yes
bzip enabled no

Enabled decoders:

aac cinepak kmvc
aac_latm cljr lagarith
aasc cook loco
ac3 cscd mace3
adpcm_4xm cyuv mace6
adpcm_adx dca mdec
adpcm_ct dfa microdvd
adpcm_ea dirac mimic
adpcm_ea_maxis_xa dnxhd mjpeg
adpcm_ea_r1 dpx mjpegb
adpcm_ea_r2 dsicinaudio mlp
adpcm_ea_r3 dsicinvideo mmvideo
adpcm_ea_xas dvbsub motionpixels
adpcm_g722 dvdsup mp1
adpcm_g726 dvvideo mp1float
adpcm_ima_amv dxa mp2
adpcm_ima_apc dxtory mp2float
adpcm_ima_dk3 eac3 mp3
adpcm_ima_dk4 eacmv mp3adu
adpcm_ima_ea_eacs eamad mp3adufloat
adpcm_ima_ea_sead eatgq mp3float
adpcm_ima_iss eatgv mp3on4
adpcm_ima_qt eatqi mp3on4float
adpcm_ima_smjpeg eightbps mpc7
adpcm_ima_wav eightsvx_exp mpc8
adpcm_ima_ws eightsvx_fib mpeg1video
adpcm_ms escape124 mpeg2video
adpcm_sbpro_2 escape130 mpeg4
adpcm_sbpro_3 exr mpegvideo
adpcm_sbpro_4 ffv1 msmpeg4v1
adpcm_swf ffvhuff msmpeg4v2
adpcm_thp ffwavesynth msmpeg4v3
adpcm_xa flac msrle
adpcm_yamaha flashsv msvideo1
alac flashsv2 mszh
als flic mxpeg

```
amrnb flv nellymoser
amrwb fourxm nuv
amv fraps pam
anm frwu pbm
ansi g723_1 pcm_alaw
ape g729 pcm_bluray
ass gif pcm_dvd
asv1 gsm pcm_f32be
asv2 gsm_ms pcm_f32le
atrac1 h261 pcm_f64be
atrac3 h263 pcm_f64le
aura h263i pcm_lxf
aura2 h264 pcm_mulaw
avrp huffyuv pcm_s16be
avs idcin pcm_s16le
avui idf pcm_s16le_planar
ayuv iff_byterun1 pcm_s24be
bethsoftvid iff_ilbm pcm_s24daud
bfi imc pcm_s24le
bink indeo2 pcm_s32be
binkaudio_dct indeo3 pcm_s32le
binkaudio_rdft indeo4 pcm_s8
bintext indeo5 pcm_s8_planar
bmp interplay_dpcm pcm_u16be
bmv_audio interplay_video pcm_u16le
bmv_video jacosub pcm_u24be
c93 jpeg2000 pcm_u24le
cavs jpegls pcm_u32be
cdgraphics jv pcm_u32le
cdxl kgv1 pcm_u8
pcm_zork smc vmdvideo
pcx snow vmnc
pgm sol_dpcm vorbis
pgmyuv sonic vp3
pgssub sp5x vp5
pictor srt vp6
png sunrast vp6a
ppm svq1 vp6f
prores svq3 vp8
prores_lgpl targa vqa
ptx theora wavpack
```

C. FFMPEG - SORTIDA DE CONFIGURE

```
qcelp thp wmalossless
qdm2 tiertexseqvideo wmapro
qdraw tiff wmv1
qpeg tmv wmv2
qtrle truehd wmavoice
r10k truemotion1 wmv1
r210 truemotion2 wmv2
ra_144 truespeech wmv3
ra_288 tscc wmv3image
ralf tta wmv1
rawvideo twinvq ws_snd1
rl2 txd xan_dpcm
roq ulti xan_wc3
roq_dpcm utvideo xan_wc4
rpza v210 xbin
rv10 v210x xbm
rv20 v308 xl
rv30 v408 xsub
rv40 v410 xwd
s302m vb y41p
sgi vble yop
shorten vc1 yuv4
sipr vclimage zerocodec
smackaud vcr1 zlib
smacker vmdaudio zmbv
```

Enabled encoders:

```
a64multi huffyuv pgmyuv
a64multi5 jpeg2000 png
aac jpegls ppm
ac3 libx264 prores
ac3_fixed libx264rgb prores_anatoliy
adpcm_adx ljpeg prores_kostya
adpcm_g722 mjpeg qtrle
adpcm_g726 mp2 r10k
adpcm_ima_qt mpeg1video r210
adpcm_ima_wav mpeg2video ra_144
adpcm_ms mpeg4 rawvideo
adpcm_swf msmpeg4v2 roq
adpcm_yamaha msmpeg4v3 roq_dpcm
alac msvideo1 rv10
```

```
amv nellymoser rv20
ass pam sgi
asv1 pbm snow
asv2 pcm_alaw sonic
avrp pcm_f32be sonic_ls
ayuv pcm_f32le srt
bmp pcm_f64be sunrast
cljr pcm_f64le svq1
dca pcm_mulaw targa
dnhhd pcm_s16be tiff
dpx pcm_s16le v210
dvbsub pcm_s24be v308
dvdsup pcm_s24daud v408
dvvideo pcm_s24le v410
eac3 pcm_s32be vorbis
ffv1 pcm_s32le wmv1
ffvhuff pcm_s8 wmv2
flac pcm_u16be wmv1
flashsv pcm_u16le wmv2
flashsv2 pcm_u24be xbm
flv pcm_u24le xsub
g723_1 pcm_u32be xwd
gif pcm_u32le y41p
h261 pcm_u8 yuv4
h263 pcx zlib
h263p pgm zmbv
```

Enabled hwaccels:

```
Enabled parsers:
aac dvdsup mpegvideo
aac_latm flac png
ac3 gsm pnm
adx h261 rv30
cavsvideo h263 rv40
cook h264 vc1
dca mjpeg vorbis
dirac mlp vp3
dnhhd mpeg4video vp8
dvbsub mpegaudio
```

C. FFMPEG - SORTIDA DE CONFIGURE

Enabled demuxers:
aac iff pcm_u24be
ac3 image2 pcm_u24le
act image2pipe pcm_u32be
adf ingenient pcm_u32le
adx ipmovie pcm_u8
aea iss pmp
aiff iv8 pva
amr ivf qcp
anm jacosub r3d
apc jv rawvideo
ape latm rl2
asf lmlm4 rm
ass loas roq
au lxf rpl
avi m4v rso
avs matroska rtp
bethsoftvid mgsts rtsp
bfi microdvd sap
bink mjpeg sbg
bintext mlp sdp
bit mm segafilm
bmv mmf shorten
c93 mov siff
caf mp3 smacker
cavsvideo mpc smjpeg
cdg mpc8 sol
cdxl mpegps sox
daud mpegts spdif
dfa mpegtsraw srt
dirac mpegvideo str
dnhhd msnwc_tcp swf
dsicin mtv thp
dts mvi tiertexseq
dv mxf tmv
dxa mxg truehd
ea nc tta
ea_cdata nsv tty
eac3 nut txd
ffm nuv vc1
ffmetadata ogg vc1t

```
filmstrip oma vmd
flac pcm_alaw voc
flic pcm_f32be vqf
flv pcm_f32le w64
fourxm pcm_f64be wav
g722 pcm_f64le wc3
g723_1 pcm_mulaw wsaud
g729 pcm_s16be wsvqa
gsm pcm_s16le wtv
gxf pcm_s24be wv
h261 pcm_s24le xa
h263 pcm_s32be xbin
h264 pcm_s32le xmv
hls pcm_s8 xwma
ico pcm_u16be yop
idcin pcm_u16le yuv4mpegpipe
idf
```

Enabled muxers:

```
a64 ipod pcm_mulaw
ac3 ismv pcm_s16be
adts ivf pcm_s16le
adx jacosub pcm_s24be
aiff latm pcm_s24le
amr m4v pcm_s32be
asf matroska pcm_s32le
asf_stream matroska_audio pcm_s8
ass md5 pcm_u16be
au microdvd pcm_u16le
avi mjpeg pcm_u24be
avm2 mkvtimestamp_v2 pcm_u24le
bit mlp pcm_u32be
caf mmf pcm_u32le
cavsvideo mov pcm_u8
crc mp2 psp
daud mp3 rawvideo
dirac mp4 rm
dnxhd mpeg1system roq
dts mpeg1vcd rso
dv mpeg1video rtp
eac3 mpeg2dvd rtsp
```


C. FFMPEG - SORTIDA DE CONFIGURE

```
ffm mpeg2svcd sap
ffmetadata mpeg2video segment
filmstrip mpeg2vob smjpeg
flac mpegts sox
flv mpjpeg spdif
framecrc mxf srt
framemd5 mxf_d10 swf
g722 null tg2
g723_1 nut tgp
gif ogg truehd
gxf oma vc1t
h261 pcm_alaw voc
h263 pcm_f32be wav
h264 pcm_f32le webm
image2 pcm_f64be wtv
image2pipe pcm_f64le yuv4mpegpipe
```

Enabled protocols:

```
applehttp hls pipe
cache http rtmp
concat httpproxy rtp
crypto md5 tcp
file mmsh udp
gopher mmst
```

Enabled filters:

```
abuffersink drawbox pan
aconvert earwax pixdesctest
aevalsrc fade removelogo
aformat fieldorder resample
amerge fifo rgbtestsrc
amovie format select
anull fps setdar
anullsink gradfun setfield
anullsrc hflip setpts
aresample hqdn3d setsar
ashowinfo idet settb
asplit life showinfo
astreamsync lut silencedetect
asyncts lutrgb slicify
bbox lutyuv split
```

C. FFMPEG - SORTIDA DE CONFIGURE

```
blackdetect mandelbrot super2xsai
blackframe movie swapuv
boxblur mp testsrc
buffersink mptestsrc thumbnail
cellauto negate tile
color noformat tinterlace
colormatrix null transpose
copy nullsink unsharp
crop nullsrc vflip
cropdetect overlay volume
delogo pad yadif
deshake
```

Enabled bsfs:

```
aac_adtstoasc mjpeg2jpeg mp3_header_decompress
chomp mjpega_dump_header noise
dump_extradata mov2textsub remove_extradata
h264_mp4toannexb mp3_header_compress text2movsub
imx_dump_header
```

Enabled indevs:

```
alsa lavfi v4l2
dv1394 oss x11_grab_device
fbdev
```

Enabled outdevs:

```
alsa oss sdl
```

```
License: GPL version 2 or later
Creating config.mak and config.h...
```

Apèndix D

Reproductor de vídeo amb llista de reproducció

Llistat D.1: myplayer.php

```
?php
$dir = "video";
$files = scandir($dir);
?>
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width" />
  <title>Video Player</title>
  <script type="text/javascript" src="jquery.js"></script>
  <script type="text/javascript" src="myplayer.js"></script>
</head>
<body>
  <header id="header" class="vid">
    <h1>Video Player Proof of Concept</h1>
  </header>
  <video
    controls="controls"
    id="player"
    width="640"
    height="480"
  >
    Your browser does not support the video tag
  </video>
  <hgroup>
    <h2>Playlist</h2>
    <div id="divlist">
      <ul id="list">
```

```

<?php
    $j=0;
    foreach($files as $file) {
        if($file != '.' && $file != '..'){
            echo "<li>";
            echo "<a id='track_" . $j . "'
                href='". $dir . "/" . $file . "' onclick='return
                false;'>";
            echo $file . "</a>";
            echo "</li>";
            $j++;
        }
    }
?>
</ul>
</div>
</hgroup>
<div id="output">xxx</div>
<a onclick="javascript:enrera()">Marxa enrera</a>
</body>
</html>

```

Llistat D.2: myplayer.js

```

var i = 0;
var listlength = 0;
function nextVideo() {
    loadVideo($('#list > li a')[++i%listlength].href);
}
function timeUpdated() {
    var player = document.getElementsByTagName('video')[0];
    var time = player.currentTime;
    var out = document.getElementById('output');
    out.innerHTML = time;
}

function loadVideo(path) {
    var player = document.getElementsByTagName('video')[0];
    player.src = path;
    player.load();
    player.play();
}

function enrera(){
    var player = document.getElementsByTagName('video')[0];
    player.playbackRate = -1;
}

$(document).ready(function(){

```

```
var player = document.getElementsByTagName('video')[0];
player.addEventListener('ended', nextVideo, false);
player.addEventListener('timeupdate', timeUpdated, false);

$('#list > li a').live('click', function(e) {
    var path = $(this).attr('href');
    i=$(this).attr('id').split("_")[1];
    loadVideo(path);
});
listlength = $('#list > li a').length;
});
```