

# Digitalización, almacenamiento y transmisión de audio y vídeo

Alex Ribelles García

PID\_00176934



Los textos e imágenes publicados en esta obra están sujetos –excepto que se indique lo contrario– a una licencia de Reconocimiento-NoComercial-SinObraDerivada (BY-NC-ND) v.3.0 España de Creative Commons. Podéis copiarlos, distribuirlos y transmitirlos públicamente siempre que citéis el autor y la fuente (FUOC. Fundació para la Universitat Oberta de Catalunya), no hagáis de ellos un uso comercial y ni obra derivada. La licencia completa se puede consultar en <http://creativecommons.org/licenses/by-nc-nd/3.0/es/legalcode.es>

# Índice

<b>Introducción</b> .....	5
<b>Objetivos</b> .....	6
<b>1. Introducción</b> .....	7
1.1. Contenido del módulo .....	7
<b>2. Codificación y compresión de vídeo y audio digital</b> .....	9
2.1. Captación de audio y vídeo .....	10
2.2. El Formato Intermedio Común de imagen (CIF) .....	12
2.2.1. El espacio de color .....	13
2.2.2. Muestreo de vídeo y submuestreo de color .....	14
2.2.3. Respecto a los tamaños estandarizados en CIF .....	15
2.3. La compresión de vídeo digital .....	16
2.4. Códecs de vídeo digital con pérdidas más habituales .....	22
2.4.1. Almacenamiento .....	22
2.4.2. <i>Streaming</i> .....	23
2.4.3. El caso intermedio, el <i>video on demand</i> .....	23
2.4.4. El receptor .....	24
2.4.5. Ejemplos de estándares .....	24
2.5. Compresión de audio digital .....	25
2.6. Diferenciando contenedor y códec .....	28
<b>3. El estándar MPEG-2</b> .....	31
3.1. Otros estándares relacionados: JPEG/JPEG2000, MPEG-1, H.261 y H.263 .....	31
3.2. ¿Qué es el MPEG-2? .....	32
3.3. Codificación y compresión en MPEG-2 .....	33
3.3.1. Los flujos elementales de datos .....	36
3.3.2. Almacenamiento y transmisión .....	39
3.3.3. Perfiles y niveles de MPEG-2 .....	40
<b>4. El estándar MPEG-4 y el H.264/AVC</b> .....	43
4.1. El estándar y sus patentes .....	44
4.2. H.264 o MPEG-4 Parte 10 .....	45
4.3. Almacenamiento .....	45
4.4. Transmisión .....	46
4.5. Perfiles y niveles .....	46
4.6. Utilización del H.264 en la industria .....	47
<b>5. Transmisión de vídeo en una red IP (<i>streaming</i>)</b> .....	49

---

5.1.	Las redes IP .....	50
5.1.1.	Protocolo de red (Internet Protocol, IP) .....	50
5.1.2.	Protocolos de transporte (UDP, TCP y SCTP) .....	51
5.2.	Técnicas de multidifusión .....	53
5.3.	Protocolos de <i>streaming</i> .....	55
5.3.1.	Protocolos de tiempo real (RTP, RTCP y RTSP) .....	56
5.3.2.	Otros protocolos de <i>streaming</i> .....	57
<b>6.</b>	<b>MPEG-2 TS (Transport Stream)</b> .....	<b>59</b>
6.1.	Paquetes y tablas del Transport Stream .....	59
<b>Resumen</b>	.....	<b>61</b>

## Introducción

En el módulo de introducción, se ha caracterizado la imagen estática y se ha hecho hincapié en el proceso de digitalización para su captura. Si bien se fue un paso más allá explicando el proceso posterior de codificación JPEG, ahora es el momento de centrarse en la captación, codificación y transmisión de contenido de audio y vídeo, informaciones más complejas que la anterior que obligarán a introducir nuevos métodos especializados de compresión.

Los diferentes formatos audiovisuales que se van a presentar se pueden dividir según diferentes criterios. Así, se va a hablar de formatos de compresión con pérdidas o sin pérdidas si conllevan cierto nivel de degradación de la información original y se verá cómo detectar esa degradación. Otra clasificación será en función de su objetivo, ya sea ser transmitido en tiempo real o no, o sencillamente almacenado.

Se explicará la diferenciación entre códec y contenedor, para lo que se ilustrará con ejemplos actuales y se hará especial hincapié en aquellos más extendidos en la actualidad. Para finalizar, vamos a ver dos ejemplos de plataformas de distribución actual de contenido multimedia: Internet y la televisión digital terrestre.

Para la ilustración de estos estándares, tanto en este módulo como en las prácticas asociadas, se utilizarán tanto software propietario (Adobe Photoshop) como herramientas de código libre (MPEG Streamclip y VLC).



## Objetivos

Los objetivos que se pretende alcanzar con el estudio de este módulo son los siguientes:

1. Identificar las características de los códecs más habituales de vídeo.
2. Capacitar para decidir la forma más adecuada de publicar un contenido digital bajo demanda.
3. Identificar y diferenciar los contenedores multimedia más frecuentes.
4. Capacitar para establecer las características oportunas del contenido digital para su almacenamiento y su acceso.
5. Identificar las dificultades de transmisión multimedia en redes IP.
6. Capacitar para decidir la vía de transmisión más adecuada para un contenido digital.

## 1. Introducción

La digitalización de los canales de comunicación y de la propia información ha supuesto la denominada **convergencia digital** y con ella un fácil intercambio de todo tipo de información digital. El vídeo, sin duda debido a sus necesidades específicas de volumen de datos y cadencia, fue de las últimas en integrarse, pero una vez dado el paso está presente en todas partes, en nuestro día a día, como un servicio más. El proceso de integración ha contado, pues, con dos elementos paralelos: los estándares de formato digital de vídeo y la implementación de sistemas digitales de transmisión.

El *streaming*, o transferencia de vídeo y audio digital en tiempo real, ya existía en comunicaciones móviles desde la década de 1970, si entendemos como tales los sistemas dedicados experimentales sin objetivo comercial. En la actualidad, son básicamente servicios de valor añadido cuya calidad está adaptada a la capacidad y tipo de receptor-cliente (teléfono móvil, tabletas, puntos de información) gracias a estándares más o menos abiertos que posibilitan dar ese servicio al vuelo. Sobre éstos trataremos en este módulo.

### 1.1. Contenido del módulo

En los siguientes apartados, vamos a describir los componentes de un sistema de vídeo digital y, haciendo especial hincapié en los estándares, vamos a presentar de forma abreviada la primera fase de un sistema digital, la captura de vídeo y su compresión.

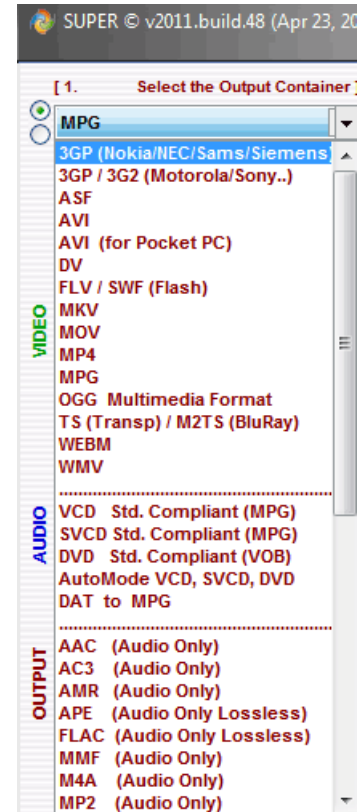
De los estándares actuales, veremos los más aceptados por la industria, de los que destacamos claramente los especificados por el grupo MPEG (Moving Pictures Experts Group) de la ISO (Organización Internacional de Normalización), sin dejar de lado otros de gran alcance como .FLV, .OGG y .3GP. Un vistazo a la lista de opciones de formato de exportación de una aplicación de vídeo cualquiera es un fiel reflejo de los formatos de vídeo más utilizados en la actualidad.

Actualmente los sistemas de vídeo digital, como la televisión digital por cable, terrestre y satelital o el formato DVD utilizan el estándar MPEG-2, pero existe una clara adopción por la industria del estándar MPEG-4 para el despliegue de nuevos sistemas de vídeo digital. En particular, los sistemas de transmisión de datos basados en TCP/IP, como IPTV o 3G, parecen adoptar en su mayoría el estándar H.264/AVC, que es una parte del estándar MPEG-4 estandarizado en conjunto con la ITU-T. Sin embargo, y por herencia de Internet, algunos estándares de sobremesa altamente eficientes como Flash tienen una vida por delante, basando su funcionamiento en el H.264.

El estándar MPEG-2 se presenta en el apartado 3, mientras que el MPEG-4 y derivados se presentan en el apartado 4. Como se verá, la fuerza de los estándares MPEG se encuentra en la clara especificación de cómo codificar y descodificar el contenido multimedia.

Respecto a la transmisión de los datos generados, hay una solución para cada tipo de medio posible que asegure su **distribución** e igualmente hay otra solución para cada medio en el que se desee su **almacenamiento**. Son los dos destinos habituales de esta información audiovisual. Curiosamente, para redes IPTV, ambas soluciones son necesarias.

Dado que no siempre se utilizan los estándares MPEG-2 y MPEG-4 para almacenar y transmitir la información, estos mecanismos se presentan en apartados separados de los de los estándares: el apartado 5 muestra los mecanismos actuales de almacenamiento de vídeo, mientras que el apartado 6 muestra los mecanismos de transmisión en una red IP. Hay un caso particular de transmisión de MPEG, es el Transport Stream, comúnmente referenciado por MPEG-2 TS o MPEG-TS, muy utilizado en las redes actuales de *broadcast* digital y que se presenta en el último apartado.



Los diferentes formatos de vídeo y audio más habituales para almacenamiento o transmisión seleccionables en una aplicación como Super ©



## 2. Codificación y compresión de vídeo y audio digital

Codificación y compresión son conceptos tan ligados que es frecuente su confusión. La **codificación** es el proceso en el que se representa una información de vídeo o audio digital de manera diferente a la original, sin perder ninguna información en el proceso.

La mayoría de codificaciones existentes tienen como objetivo o bien comprimir la información o bien ofuscarla para hacerla más segura (o ambos). En esta asignatura, sólo nos vamos a centrar en el primero.

Así pues, una codificación implica usualmente una **compresión**, una reducción del tamaño final del fichero sin pérdida alguna de información. Este comportamiento se conoce también como **compresión sin pérdidas**. Lamentablemente no tiene ni de lejos buenos resultados con vídeo o audio, por ello necesitamos una técnica complementaria: la **compresión con pérdidas**.

En general, la **compresión** hace referencia a la reducción de la información perdiendo o no parte de ésta, al menos sin reducir excesivamente su calidad.

Hablamos entonces de compresión sin pérdidas o con pérdidas, aunque esta última es la más utilizada, ya que consigue reducir mucho más el tamaño de la información que la compresión sin pérdidas. Si bien un cambio de la codificación por sí solo puede lograr cierta reducción del volumen de datos, es sin duda la compresión el quid de todo sistema de almacenamiento o distribución audiovisual. La combinación de los dos es la técnica habitual de trabajo de todos los estándares audiovisuales actuales.

Todo ello es posible si se trata con información digital: los sistemas de audio y vídeo actuales poseen una primera etapa de conversión analógica a digital, ya que la realidad que nos envuelve es analógica por naturaleza.

Si esta primera etapa se ejecuta con equipos profesionales, la señal involucrada no se comprime, sino que sólo se codifica digitalmente para mantener su pureza en un formato digital crudo (*raw*) y ser procesada luego para su compresión en función de la necesidad.

### El formato zip

Un ejemplo de codificación es el formato zip que, sin perder un solo dato, es capaz de representar la misma información original con el beneficio añadido de reducir la necesidad de almacenamiento de esa información.

### Ved también

En el subapartado 2.1, vamos a ver cómo se lleva a cabo esta conversión analógica a digital.

### Ved también

Algunos de estos formatos para mantener la pureza de la señal se tratan en el subapartado 2.2.

En cambio, en entornos industriales y semiprofesionales, la captura y la compresión vienen juntas, como es el caso del hardware y software que usa el estándar MPEG.

**Ved también**

El estándar MPEG se trata en el subapartado 2.3 como una aproximación general a la compresión de vídeo.

## 2.1. Captación de audio y vídeo

La captación y codificación de audio es relativamente sencilla comparada con la de vídeo. Básicamente, se muestrea la señal de audio a una frecuencia siempre superior (como mínimo) a dos veces la frecuencia máxima que pueda haber en la señal de audio.

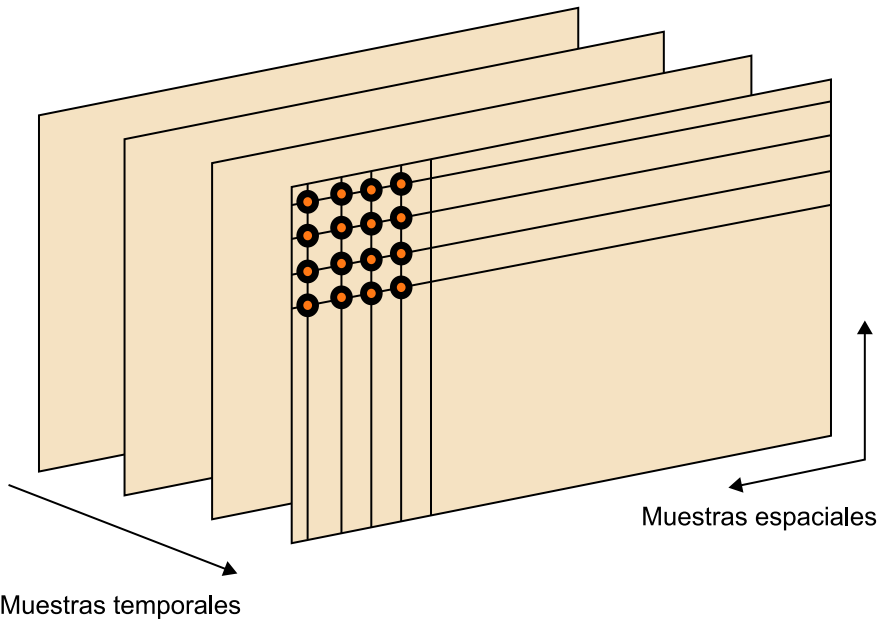
Así, si el oído humano es capaz de captar frecuencias de audio de hasta 16 kilohercios (kHz), es muy habitual que se muestree a 44 o 48 kHz, es decir, 44.000 o 48.000 muestras por segundo.

Cada muestra representa un valor real de la amplitud de la señal de audio, es decir, posee decimales, cosa que dificulta su codificación en bits. Para ello, se convierten estos valores con decimales a números enteros, usualmente en el rango de 0 a 255 (8 bits) o mejor aún de 0 a 65.535 (16 bits). El formato PCM de audio digital es el formato *raw* por excelencia en entornos profesionales: no hay pérdida alguna en esta codificación, a costa de suponer una gran cantidad de bits para almacenar (CD).



Una señal analógica es muestreada para su digitalización.

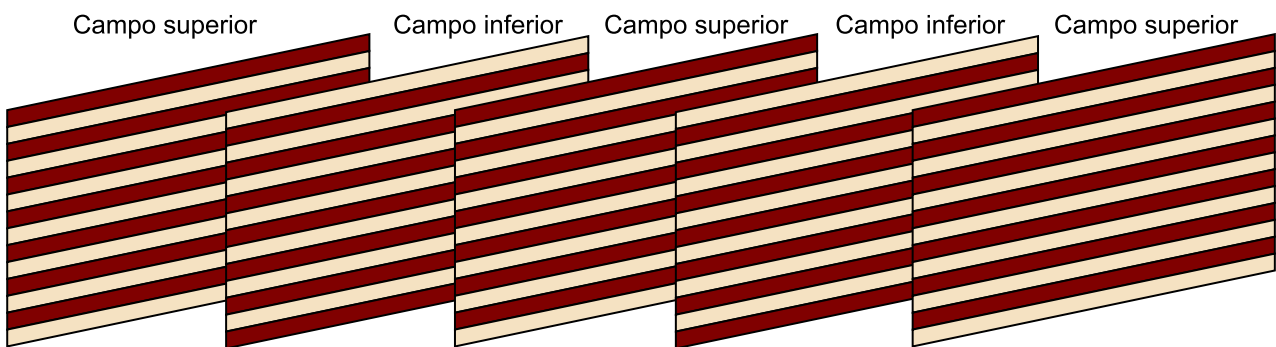
Respecto al vídeo, la representación natural de un vídeo digital es una matriz tridimensional de píxeles. Dos dimensiones (horizontal y vertical) representan un cuadro o imagen y pertenecen al dominio espacial (ancho y alto), la tercera dimensión es el tiempo.



Muestreo en tres dimensiones del vídeo: x, y, t

Esta representación facilita entender que se habrán de tomar muestras de cada cuadro (moverse en el plano espacial, **muestreo espacial**) y cada cierto tiempo (moverse en el plano temporal, **muestreo temporal**). Tal como se plantea, sería una codificación cruda, pero veremos a continuación que por ciertas razones reduciremos siempre el número de muestras por realizar. Por ello, se denomina Common Intermediate Format (CIF) a este formato de codificación, al estar en un punto medio entre la captura cruda y la comprimida.

La forma natural de realizar las muestras de la señal de vídeo es conocida como **muestreo progresivo**, es decir, tal como plantea la imagen anterior interpretamos el vídeo como una serie de cuadros consecutivos independientes y muestreamos espacialmente uno tras otro. Pero debido a los requerimientos de velocidad de muestreo necesarios, ya que la información de cada cuadro es muy similar a la del anterior, podemos muestrear primero las líneas pares del primer cuadro y luego las impares del mismo, las pares del segundo cuadro, las impares del mismo y así sucesivamente.



Estructura de campos de un vídeo entrelazado

Este tipo de muestreo se denomina **muestreo entrelazado** y con él puede captarse la misma información que con el progresivo pero con un equipo técnicamente más sencillo. Puede parecer drástico, pero conlleva algunos beneficios, por lo que su uso es habitual y los efectos son poco perceptibles en el movimiento de los objetos. Esta opción está implementada en todo software o hardware de captura de vídeo existente. El ojo entrenado sí puede notar los efectos, especialmente en zonas amplias de imagen muy luminosas, pero visualizado en un monitor progresivo como cualquier pantalla plana se camuflan completamente.

En resumen, de cada cuadro original pasamos a **dos campos** con la mitad de resolución vertical (pero la misma resolución horizontal que el original), entendiendo que cada campo es una imagen con la mitad de información que el cuadro original. La velocidad en campos/segundo es el doble que cuadros/segundo, evidentemente, pero el flujo final de datos generados es el mismo.

Este procedimiento no sólo reduce los requerimientos técnicos del equipo de captura, sino que además mejora el efecto de parpadeo que podría haber reproduciendo una señal de 25 cuadros por segundo respecto a una señal de la mitad de definición pero a 50 campos por segundo. No obstante, esta ventaja va desapareciendo en cuanto el receptor posee un monitor progresivo: poseen una memoria buffer que, tras recibir ambos campos de un cuadro, pueden recomponerlo con toda su definición.

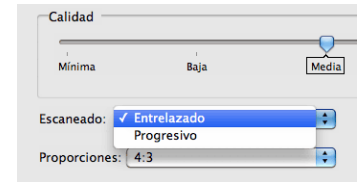
La norma PAL utilizaba este mecanismo en sus cámaras para capturar y transmitir por un canal de frecuencias estrecho. Actualmente, en plena era digital, sigue aplicándose a las señales estándares de la TDT e incluso a las señales HDTV de alta definición 1080i.

## 2.2. El Formato Intermedio Común de imagen (CIF)

Tal como ya se ha indicado, una **codificación cruda** es, en esencia, una secuencia temporal de cuadros o de campos, es decir, de la mitad de las líneas de un cuadro cuando hay entrelazado. Es necesario determinar el formato de cada cuadro y el mecanismo de muestreo para tener una especificación completa del formato CIF que acuerden emisor y receptor.

### Videokonferencia

En una videoconferencia (con Skype, iChat u otros), emisor y receptor deben tener definido claramente el CIF: el tamaño del cuadro de imagen que se presenta, el número de colores que se reproducen y la cadencia de cuadros por segundo, entre otros.

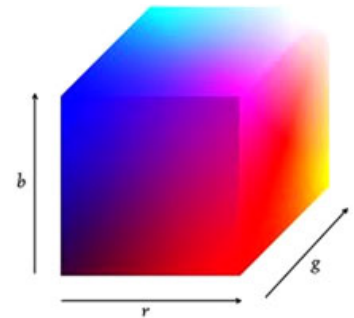


Opciones de captura de vídeo habituales

### 2.2.1. El espacio de color

Un cuadro es una matriz rectangular de píxeles que representa una imagen. El píxel está representado y localizado por sus dos coordenadas X e Y, pero además posee tres valores enteros que representan el color de éste en un espacio de colores. Los espacios de color más conocidos son el RGB, el YCbCr y, especialmente, el YUV.

El RGB representa un nivel de color para los colores básicos rojo (R), verde (G) y azul (B). Es un espacio de color intuitivo en el que cualquier color puede representarse como una combinación de estos tres colores. Dado que la gama es infinita en cualquiera de ellos, usualmente se reduce a 256 valores por color, del 0 al 255. Así,  $R = 0$  es el rojo negro y  $R = 255$ , el rojo más vívido y puro. Hay que especificar, no obstante, que esta codificación es usada por pura conveniencia.



El espacio de color RGB

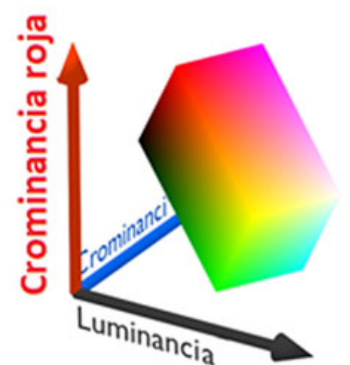
#### Actividad 1

Mediante Adobe Photoshop, abrid una imagen en color. En VENTANA-CANALES se muestran sus componentes R, G y B. Confirmad que allí donde predomina uno de los tres colores su componente posee un valor más claro.



Los otros dos espacios de color, donde se puede definir también cualquiera color con tres valores, son equivalentes al RGB, sólo que plantean cada color como un combinado de blanco y negro y dos componentes de color. Así, en el YCbCr, el valor Y es el valor de luminancia, Cb el valor de crominancia azul y Cr el de crominancia roja. El valor de crominancia verde puede deducirse de los tres anteriores. El espacio YUV también sigue la misma filosofía.

La razón por la que es más habitual utilizar estos dos últimos espacios se debe a las características del ojo, es menos sensible al color que al brillo o luminancia. La comprobación es sencilla: ponemos un hilo de color rojo a cinco metros de un observador. Usualmente detectará la existencia del hilo, pero no su color. El color, parece ser, era poco relevante para la supervivencia de nuestros ancestros, como mínimo menos relevante que la intensidad de luz.



El espacio de color YCbCr

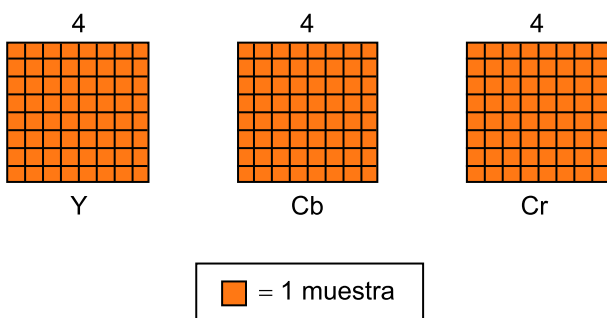
De este modo, conociendo que se puede representar cada cuadro de imagen por sus tres cuadros equivalentes Y, Cb y Cr o Y, U y V y sabiendo que la información visual sobre todo está en el cuadro Y, se podrá comprimir con pocas pérdidas (es decir, una compresión más bien suave) el cuadro Y y comprimir con más pérdidas (mayor compresión) los cuadros Cb y Cr o U y V.

Así ocurre en el estándar JPEG de compresión de imagen fija. La primera etapa de su proceso es convertir la imagen desde el espacio de color RGB (tal como se genera en el sensor de cámara) al espacio YCbCr.

### 2.2.2. Muestreo de vídeo y submuestreo de color

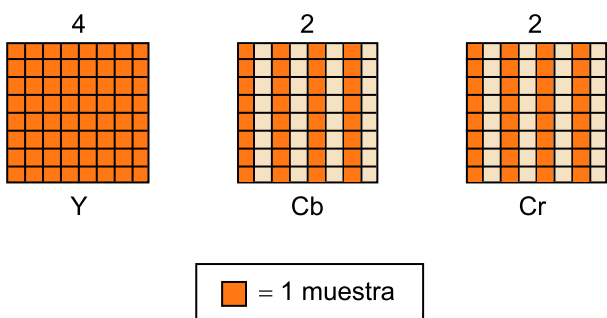
Esta compresión, que se aplica más intensamente al color, comienza en el mero proceso de muestreo de cada cuadro. Se realiza un muestreo más denso en el equivalente Y y menos denso en los equivalentes de color sin que afecte seriamente a la calidad final. La sistemática de muestreo está estandarizada en la familia MPEG, basada en YCbCr o YUV según el caso:

a) **4:4:4**: los tres componentes (Y, Cb y Cr) presentan la misma resolución y existe una muestra de cada componente en cada píxel. Sólo es para entornos profesionales donde no se desea compresión (HDCAM, MPEG-2, H.264).



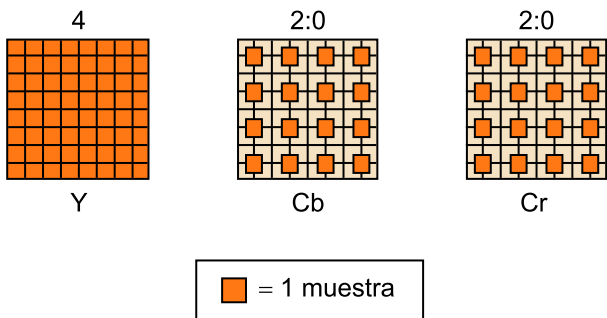
Muestras de cada componente en el caso 4:4:4

b) **4:2:2**: conocido también como YUY2, presenta dos componentes de crominancia por cada cuatro de luminancia (Digital Betacam, DVCPRO 50, MPEG-2, H.264).

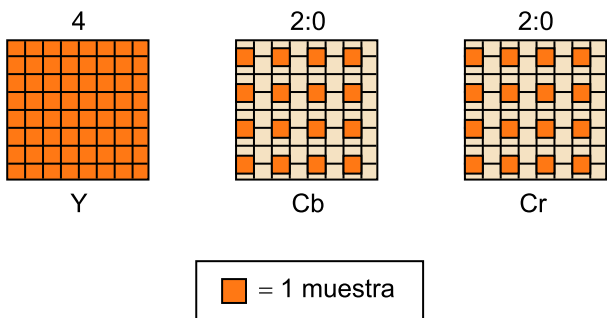


Muestras de cada componente en el caso 4:2:2

c) **4:2:0**: conocido también como YV12, presenta un componente de crominancia por cada cuatro de luminancia (DVD, MPEG-1, MPEG-2, MPEG-4).



Muestreo de cada componente en el caso MPEG-1 (4:2:0)



Muestreo de cada componente en el caso MPEG-2 (4:2:0)

En resumen, los tres patrones tienen una distribución de los píxeles que se muestrean en cada caso bien especificada: en el caso 4:4:4, se muestrean todos y cada uno de los píxeles de los tres componentes de cada cuadro. En el caso 4:2:2, se muestrean todos los píxeles del componente de luminancia Y y la mitad de cada componente de color. Finalmente, en 4:2:0, se muestrean todos los píxeles del componente de luminancia Y y sólo uno de cada cuatro en los componentes de color, que varía según la norma.

En todos ellos, cada muestra conseguida se representa luego con ocho o diez bits según la precisión de la muestra que se desee.

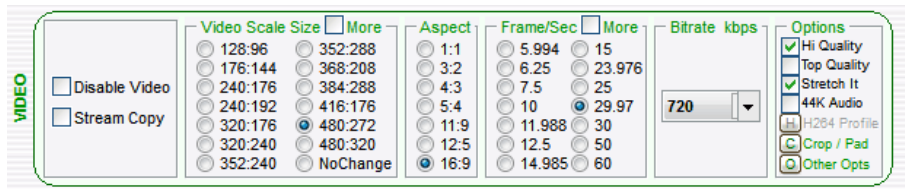
### 2.2.3. Respecto a los tamaños estandarizados en CIF

Definida la sistemática de muestreo, queda por definir el tamaño final que poseerá el vídeo digitalizado resultante. También el formato CIF incluye esta especificación:

Tabla. Tamaños recomendados para el formato CIF

Formato CIF	Resolución de vídeo (ancho x alto)
Sub QCIF	128x96
Quarter CIF (QCIF)	176x144
CIF	352x288
DCIF	528x384

Formato CIF	Resolución de vídeo (ancho x alto)
4CIF	704x480
16CIF	1408x1152



Algunas aplicaciones de codificación de vídeo expresan claramente los formatos de imagen en los que se puede exportar, entre ellos los CIF (en la imagen, Super©).

Un formato muy utilizado en el ambiente de televisión se encuentra especificado en la recomendación ITU-R BT.601-5, que podéis ver en la imagen anterior y que presenta dos variantes: NTSC y PAL.

Tabla. Tamaños recomendados para las variantes NTSC y PAL del formato ITU-R BT.601-5

Parámetro	NTSC 30 Hz	PAL/SE- CAM 25 Hz
Campos por segundo	60	50
Líneas por cuadro completo	525	625
Muestras de Y por línea	858	864
Muestras de Cr y Cn por línea	429	432
Bits por muestra	8	8
Flujo de bits final	216 Mbps	216 Mbps
Líneas activas por cuadro	480	576
Muestras activas por línea (Y)	720	720
Muestras activas por línea (Cr, Cb)	360	360

Observad que se requieren 216 Mbits por segundo para almacenar un vídeo codificado en este estándar o, lo que es lo mismo, un canal de 216 Mbps para transmitirlo. Esto supera ampliamente la disponibilidad actual de las redes de datos (Internet, 3G), por lo tanto es imprescindible aplicar algún mecanismo de compresión de vídeo.

### 2.3. La compresión de vídeo digital

La cantidad de información utilizada en una codificación cruda de vídeo puede exceder la capacidad de almacenamiento y transmisión del sistema de vídeo digital. Por este motivo, se han desarrollado y se han perfeccionado en las últimas décadas gran variedad de técnicas de compresión.

La compresión debe encontrarse completamente especificada tanto en el codificador en origen como en el descodificador del destinatario, así se asegura que la recepción es correcta. En el boom de Internet a finales de la década de 1990, aparecieron innumerables **formatos propietarios** (es decir, exclusivos de cada empresa) que obligaban al uso de programas propios para su visualización, incompatibles con los de la competencia: para RealVideo y RealAudio de Real Networks debía utilizarse la aplicación RealPlayer o el reproductor de



Windows si el fichero estaba en formato Windows Media Video de Microsoft. Hasta bien entrado el siglo XXI, no aparecieron aplicaciones capaces de soportar diferentes formatos, lo que minimizaba el caos que generaba.

El codificador convierte una señal de vídeo previamente capturada en formato intermedio CIF a un formato comprimido, que debe ser conocido por el decodificador y que reconstruye la señal de vídeo intermedia para luego ser presentada en un televisor o monitor. Por lo general, al par codificador/decodificador se le denomina **códec** (codificador/decodificador).

En la especificación de una compresión, o códec, se representa el vídeo original por una codificación. Idealmente, esta codificación debe ser **eficiente** (utilizar la menor cantidad de información posible, es decir, bits) y **eficaz** (debe representar de la forma más fiel posible el vídeo original). Lógicamente, estos dos objetivos se encuentran en conflicto y es necesario un compromiso entre ambos. Habitualmente, un estándar de códec presenta varios **perfiles** o configuraciones con distintos valores de compromiso entre eficacia y eficiencia con el fin de adaptarse a una mayor cantidad de sistemas de vídeo digital.

La principal clasificación de las técnicas de compresión divide los códecs en dos grandes familias: sin pérdidas o con pérdidas.

### 1) Compresión sin pérdidas (*lossless*)

Tras el submuestreo realizado con los componentes de color que ya hemos visto, no se ejecuta ninguna compresión adicional, por lo que lógicamente proporcionan la máxima calidad visual. Lo usual es que el submuestreo realizado sea de 4:4:4 a 8 o 10 bits por componente y se utilizan en entornos de posproducción donde la imagen debe mantenerse a la máxima calidad de origen.

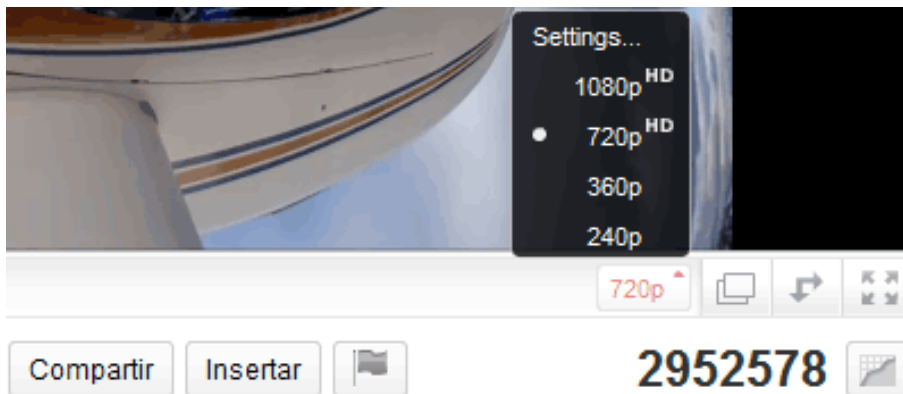
Los códecs Matrox, Aja y Blackmagic se popularizaron en este entorno para vídeo de definición estándar y, con la llegada de la alta definición, siguen utilizándose en su versión HD. El formato DPX de Kodak, basado en el clásico sistema de digitalización Cineon, permite trabajar en HD con vídeo en puro RGB y muestreo 4:4:4 a 10 bits/muestra.

Sin embargo, para reducir las necesidades de espacio y de ancho de banda de trabajo en HD (entre 1 y 1,5 Gbps) se han comenzado a aplicar discretamente códecs de compresión con pérdidas y reducir así los requerimientos de almacenamiento en arrays de discos.

En resumen, las técnicas sin pérdida quedan acotadas a entornos específicos y dedicados, que tienen siempre tasas de compresión muy bajas (por no decir casi inexistentes si se aplican sobre material de vídeo visualmente rico y muy cambiante).

## 2) Compresión con pérdidas (*lossy*)

Comprimir con pérdidas no es renunciar a una calidad final de trabajo, sino que se define la calidad necesaria para el destinatario final y se comprime hasta llegar a ésta. Un clip informativo colgado en YouTube no necesita ser de máxima calidad si su objetivo es informar y un cortometraje puede necesitar ser HD para presentar la riqueza cromática de sus escenarios a través de la misma plataforma. En ambos casos, contamos en el servidor de YouTube con el fichero de vídeo en cierta calidad inicial y YouTube lo reduce visualmente según la definición requerida por los posibles destinatarios, siempre en sentido descendente desde la calidad original hacia inferiores:



Opciones de visualización de un vídeo 1080p en YouTube

En el caso de vídeo generado en tiempo real (*live streaming*), la compresión con pérdidas es la apuesta necesaria e imprescindible para facilitar la llegada de la imagen y el sonido con la cadencia necesaria para ser entendible. No hay tiempo material para hacer una codificación óptima como se haría en un vídeo almacenado, *obligaciones del directo*. Y aun así no asegura el éxito en este objetivo al haber otros elementos que pueden impedirlo, como veremos más adelante.



Muchos canales de televisión también emiten en tiempo real por Internet (en la imagen, el canal de noticias 3:24).

Las técnicas más comunes de compresión se basan en eliminar la **redundancia espacial y temporal** presentes en el vídeo crudo, características habituales del vídeo:

**a) Redundancia espacial:** se basa en la suposición de que los píxeles de una zona de una imagen son muy parecidos a los de las zonas que lo rodean, salvo donde haya un perfil (la transición de un objeto a otro representado en la imagen). Como vimos en el módulo anterior, el estándar JPEG de imagen se basa en estas similitudes para comprimir la imagen fija y almacena sólo las leves diferencias entre grupos de píxeles y sus vecinos. Veremos que el estándar MPEG también lo aplicará, en este caso a vídeo, sobre algunos de sus cuadros de imagen.

Zona de imagen con **baja** redundancia espacial →



Zona de imagen con **alta** redundancia espacial →

Una imagen con zonas muy diferenciadas de detalle

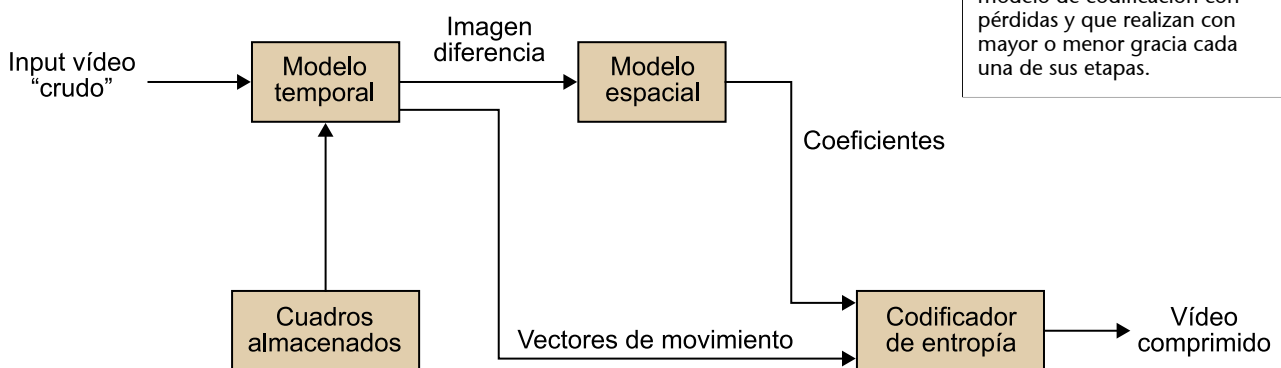
En otras palabras, las zonas homogéneas de la imagen son candidatas a una buena compresión espacial con poco esfuerzo. Sin embargo, el detalle, es decir, la riqueza de la imagen, es lo que se pone en juego. El **compromiso entre compresión y calidad de imagen** diferenciará un buen planteamiento de compresión de otros menos acertados.

b) **Redundancia temporal:** se basa en la suposición de que hay pocas diferencias entre un fotograma y sus inmediatamente anteriores o posteriores.



Un pan o barrido suave de la cámara de izquierda a derecha sobre un escaparate presenta muy pocas diferencias entre dos fotogramas consecutivos del vídeo generado. Sin necesidad de calcular la imagen diferencia, se puede asegurar que hay una alta redundancia temporal.

La manera de detectar estas redundancias y aplicar una compresión específica para cada una se puede representar en el siguiente esquema, que representa un **codificador genérico de vídeo con pérdidas (códec)**:



Modelo básico de un codificador de vídeo con pérdidas

**Ejemplos**

La familia MPEG, H.264, Theora y VC-1 son algunos ejemplos de códecs que utilizan un modelo de codificación con pérdidas y que realizan con mayor o menor gracia cada una de sus etapas.

La primera fase es el **modelo temporal** que, a partir de los cuadros almacenados del vídeo leídos hasta el momento, **prevé el siguiente cuadro** que llegará por la entrada de vídeo crudo. Esta previsión usualmente no está especificada y depende de cada fabricante, lo que condiciona buena parte de la calidad final de todo el proceso. Por ejemplo, un codificador MPEG-2 de la empresa Tandberg utiliza un planteamiento diferente de previsión del siguiente cua-

dro que el codificador MPEG-2 de la empresa Thomson. Sin duda, el funcionamiento del modelo temporal de cada una de estas empresas es un secreto bien guardado.

Una vez que el modelo temporal ha calculado cómo debería ser el siguiente cuadro, recibe el cuadro real de vídeo crudo y ejecuta dos operaciones:

a) Primero, calcula la **imagen diferencia**. En el mejor de los casos, la previsión será tan buena que la imagen diferencia será una serie de ceros, pero la mayoría de veces contiene alguna información visual que plasma el error de esta previsión. Esta imagen error, y no el vídeo real, es lo que seguiremos comprimiendo en la fase siguiente.

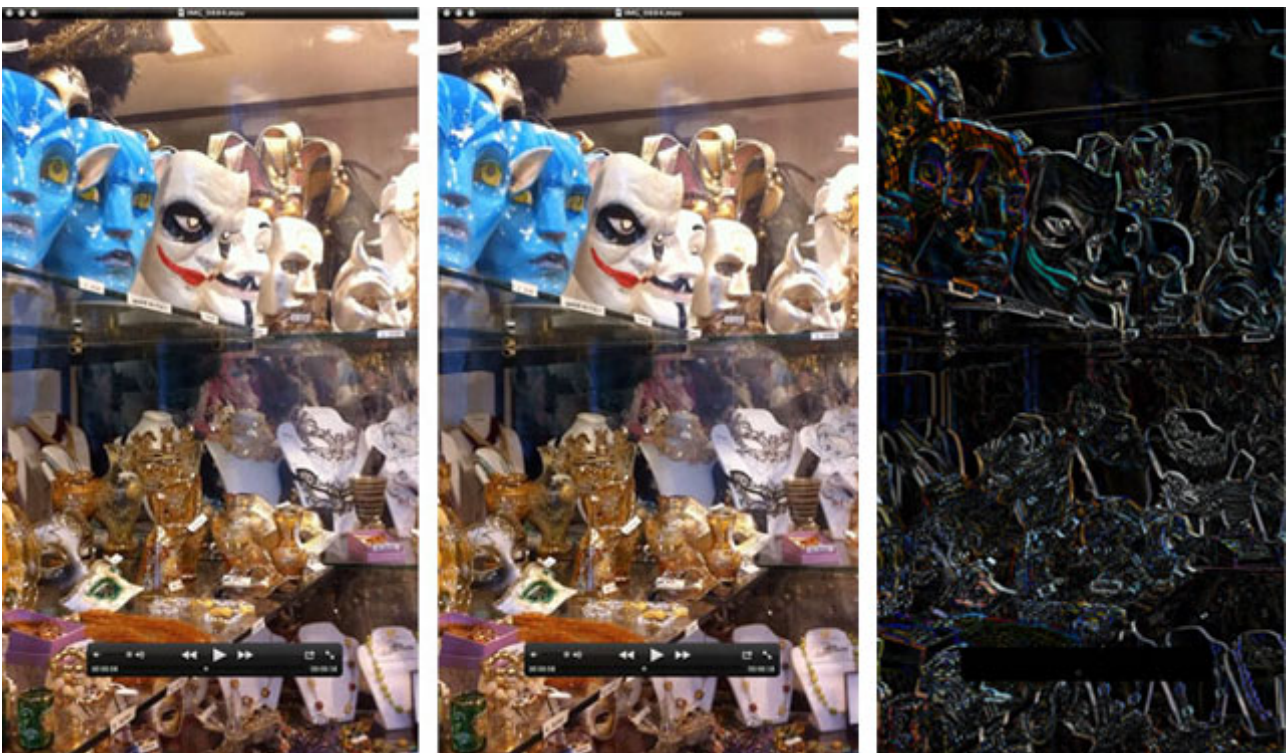


Imagen real (izquierda), imagen prevista (centro) e imagen diferencia (derecha).

b) Segundo, debido a que estamos trabajando con las imágenes de un vídeo donde usualmente hay movimiento, se calcula un conjunto de datos, llamados **vectores de movimiento**, que especifican si se detectó movimiento de partes de la imagen y dónde. Ayudará sobremanera en el momento de descodificar el vídeo.

La siguiente fase la centra el **modelo espacial**. La imagen diferencia se comprime utilizando la similitud de sus píxeles: **comprimir una diferencia** sería básicamente un resumen en tres palabras de todo el proceso. Cada sistema de compresión de vídeo utiliza aquí sus propias técnicas de compresión de imagen, aunque las utilizadas, por ejemplo, por el H.264 no distan mucho de las del estimado JPEG. Además, la mayoría son con pérdidas.

Y manteniendo esa idea en mente, al igual que el JPEG, el resultado es una ristra de valores que también son denominados coeficientes espaciales y que ejecutan el mismo papel: representan la imagen resultante con otro vocabulario matemático.

Por último, en una vuelta de tuerca final, tanto los coeficientes de la imagen como los vectores del movimiento detectado en la misma son empaquetados de forma eficiente en el **codificador de entropía**, denominado así porque analiza la frecuencia de aparición de estos coeficientes y vectores y les asigna menos bits cuanto más frecuentes son. Así, reduce el tamaño final del fichero o del flujo de datos de transmisión al mínimo posible (o al valor más cercano a ese mínimo).

El resultado es una ristra binaria compacta que contiene toda la información necesaria para poder ser descomprimida. Los sistemas de descompresión están claramente definidos, de manera que en este caso no existe diferencia alguna entre descodificadores de una empresa u otra.

## Actividad 2

Realizad el ejercicio 2 indicado en el anexo del módulo 2, donde se presentan los diferentes formatos de vídeo en los que se puede exportar mediante las aplicaciones que se van utilizar en la asignatura.

## 2.4. Códecs de vídeo digital con pérdidas más habituales

Dependiendo de las particularidades de cada sistema de audio y vídeo digital se han desarrollado distintos estándares de códec basados en el modelo anterior. Como ya se ha indicado, es habitual que se especifique completamente el descodificador y se deje a la habilidad de la industria el construir el codificador eficaz, lo que fomenta la competencia y la investigación en nuevos algoritmos de previsión de cuadro. Así, es frecuente que, usando el mismo estándar, se logre mejor calidad y menor ancho de banda con el paso del tiempo.

Los dos entornos habituales de aplicación de los códecs son, por una parte, el almacenamiento y la transmisión de audio/vídeo en tiempo real (*streaming*).

### 2.4.1. Almacenamiento

En los sistemas de codificación dirigidos a almacenamiento, el codificador tiene acceso a todo el material que codificar, por lo que puede hacer previsiones más fiables y conseguir tasas altas de compresión manteniendo la calidad audiovisual. El tiempo que tarde en generar el fichero final no es un factor crítico, por lo que los requerimientos de capacidad de cálculo y necesidad de memoria no son exageradamente elevados. Esto posibilita que puedan existir codificadores de software dignos para cumplir esta misión e incluso funcionales en equipos informáticos domésticos.

#### MPEG-2

Quien recuerde las primeras codificaciones en MPEG-2 a mediados de la década de 1990 y las que podrá generar en este curso asentirá en el gran cambio de calidad de este formato.

## 2.4.2. Streaming

En los sistemas de transmisión en tiempo real (*streaming*) como Internet Radio o IPTV (televisión basada en protocolo IP), el material que se dispone para realizar la codificación va llegando a la misma velocidad que se genera y debe codificarse a esa misma velocidad, por lo que se necesita un equipo de alta disponibilidad, fiabilidad, gran capacidad de memoria y excelente capacidad de cálculo. La mayoría son equipos dedicados de hardware de alto valor ubicados en la empresa que provee el servicio: la mayoría de webs de compañías *broadcast* actuales de radio y televisión son un ejemplo.

En otros servicios de *streaming* menos ambiciosos (como por ejemplo la videoconferencia), donde la calidad visual no es relevante y la cadencia de las imágenes no se asegura, los requisitos del codificador se relajan y pueden existir incluso en software (FaceTime, Messenger).



La mayoría de las empresas de radio y televisión emiten también parte de su programación por Internet en paralelo a la emisión por el medio habitual.

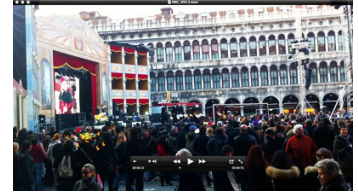
## 2.4.3. El caso intermedio, el video on demand

Acceder a un vídeo de YouTube (*video on demand* o vídeo bajo demanda) parece situarse entre ambos casos, pues consiste en visualizar en tiempo real un material almacenado previamente en un servidor remoto. Sin embargo, lo crucial en este caso es poder asegurar la cadencia de recepción, y eso se ha previsto previamente en la generación del material, al crearlo en un formato audiovisual fácil de transportar y que pueda comenzar a visualizarse desde la llegada de los primeros datos al cliente.

Esta previsión del formato en su almacenamiento hace que clásicamente este caso se incluya en *streaming* y así lo haremos a partir de este momento aunque con puntualizaciones. Por ello, servicios de vídeo como Netflix, Hulu, Google TV o la mayoría de servicios a la carta son *streaming*.

#### 2.4.4. El receptor

Respecto al receptor (tanto de *streaming* como de lectura de ficheros almacenados), posee un descodificador que no necesita grandes prestaciones, ya que los sistemas de codificación son quienes ejecutan el trabajo de análisis y generan la información mínima necesaria para reconstruir el material audiovisual. Así, existe software doméstico capaz tanto de reproducir un Blu-ray en alta definición como de visualizar una conferencia emitida por Internet al otro lado del mundo.



VLC, QuickTime Player, Windows Media Player son ejemplos de reproductores de software de propósito general ejecutables en entornos domésticos de pocas prestaciones.

#### 2.4.5. Ejemplos de estándares

En la actualidad, los dos estándares de códecs más habituales en los sistemas de IPTV son el H.264 (también llamado MPEG-4 Part 10 o AVC), desarrollado en conjunto entre la ITU-T y el MPEG, y el VC-1 (núcleo de Windows Media Audio y Windows Media Video 10), desarrollado por Microsoft. Sin embargo, tras la compra de On2 por parte de Google y las primeras pruebas de Google TV con el códec VP8 en el 2010, el mercado puede alterarse completamente en pocos años.

Para los sistemas que utilizan redes basadas en IP, como Internet y las redes locales, o servicios IP sobre 3G como medio de transporte, las compresiones más utilizadas son las siguientes:

1) **MPEG, H.264/MPEG-4 AVC y 3GPP**: formatos estándares, principalmente utilizados por QuickTime en su línea de servidores Darwin/QuickTime Stream Server y en su reproductor QuickTime. Del mismo modo, Adobe utiliza el H.264 en su Flash Video Server como códec desde la versión 9. cabe destacar que la utilización de los estándares MPEG requiere el pago de licencia tanto en servidores como en reproductor. Por su calidad, el H.264/MPEG-4 se utiliza también en almacenamiento de vídeo en Blu-ray.

2) **Windows Media 10**: codificación propietaria desarrollada por Microsoft, tiene su fortaleza en *streaming* de bajo ancho de banda (como los de Internet). Puede ser servido por la línea de servidores Microsoft utilizando el servicio Windows Media Services y reproducirse con el Windows Media Player. Esta codificación requiere un permiso especial y pago de licencia a Microsoft para poder ser utilizado en otra línea de servidor/reproductores.



3) **Theora**: formato abierto, cuyo principal atractivo es que no requiere el pago de licencia para su utilización. Presenta una gama de estándares en continuo desarrollo que pretenden competir con la línea de estándares MPEG-4. La codificación de vídeo Theora está basada en la codificación On2's VP3 y su contenedor más conocido para el almacenamiento es el formato Ogg.

4) **RealVideo**: codificación propietaria, desarrollada por RealNetwork, empresa pionera en *streaming* de audio y vídeo en Internet actualmente en desuso. Presentaba su fortaleza en *streaming* de bajo ancho de banda. Podía ser servido por la línea de servidores Helix y reproducido por el RealPlayer (o sus antecesores HelixPlayer y PlayerOne) todos de la misma empresa RealNetwork. Esta codificación requería un permiso especial y pago de licencia a RealNetwork para poder ser utilizada en otra línea de servidor/reproductores, por lo que el coste de este códec y la falta de inversión en innovación han supuesto su casi desaparición del mercado actual.

5) **VP8**: es el códec abierto de vídeo en *streaming* de WebM, una nueva propuesta de formato de Google para ser utilizada en HTML 5. De calidad semejante al H.264, se basa en una librería de software libre denominada x264 bajo licencia GNU. Su contenedor se basa en el contenedor abierto Matroska.

## 2.5. Compresión de audio digital

Las técnicas de compresión de audio son cronológicamente anteriores a las de vídeo. Como en éste, hay técnicas con pérdidas y sin pérdidas, algunas específicas para transmitir en tiempo real y otras para el almacenamiento. Todas ellas se basan en la **reducción de la redundancia** existente en la señal y así minimizan el flujo de datos generado por segundo sin que afecte a la calidad en función de su necesidad.

Del gran abanico de estándares existentes, unos se centran en la codificación de la voz humana para su transmisión en redes digitales. Servicios de transmisión de voz en tiempo real como la telefonía fija, la telefonía móvil o Internet consiguen grandes tasas de compresión al centrarse en las características específicas de esta señal. Es habitual que se basen en técnicas de predicción, codificando, como en imagen y vídeo, las diferencias entre la información predicha de audio y la real.

Como ejemplos ilustrativos tenemos los siguientes:

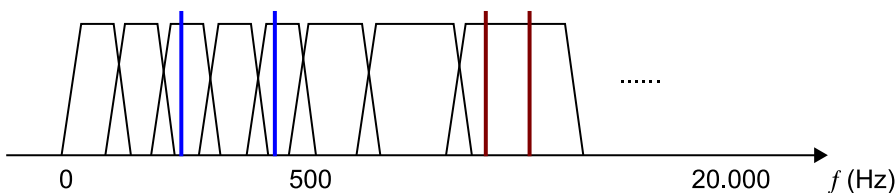
- Las redes **GSM** y **UMTS** codifican el audio en formato AMR, tanto en su emisión como en el almacenamiento de mensajes en el buzón de voz (extensión .amr) y es el formato base del grupo 3GPP para futuras redes móviles de cuarta generación.

- **Skype** se basa en su propio estándar Silk (Super Windeband Audio Codec).

Otro gran grupo, más interesante en nuestro caso, tiene como objetivo conseguir la máxima calidad de sonido con el mínimo número de bits por segundo, incluso posibilita sonido multicanal. Éstas se basan en su mayoría en la **codificación perceptual**, un avance en el mundo de la codificación de audio implementado por primera vez por Philips en su ya desaparecido sistema de audio compacto digital (DCC) que pretendía sustituir la casete de cinta.

La codificación perceptual nace de los estudios sobre cómo el ser humano interpreta el sonido y así aparecen tres patrones que abren la puerta a comprimir el audio sin una pérdida audible de calidad:

1) El oído no detecta todos los sonidos que recibe, se comporta como un **banco de filtros** que más o menos se solapan en frecuencia. Dos sonidos situados en bancos diferentes son diferenciados claramente por el cerebro, pero si ambos están en el mismo banco de frecuencias, no.

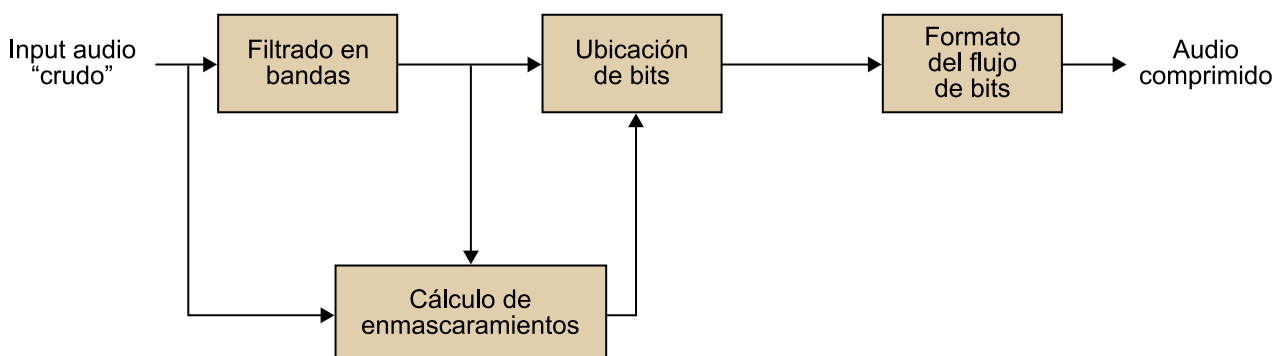


Dos tonos (las dos líneas verticales de la izquierda) situados en frecuencias de diferentes bandas son diferenciados por el oído, mientras que dos situados en la misma banda (las dos líneas verticales de la derecha) son interpretados como uno solo.

2) La sensibilidad del oído ante un sonido cambia si hay más sonidos. De alguna manera, unos sonidos enmascaran otros y los hacen indetectables al oído. Este fenómeno se denomina **enmascaramiento frecuencial**.

3) Más si cabe, si escuchando un sonido débil éste se detiene, se tardará cierto tiempo hasta poder escuchar un sonido débil otra vez. Este segundo fenómeno se denomina **enmascaramiento temporal**.

El modelo general de codificación perceptual puede expresarse gráficamente de la siguiente manera:



Modelo de codificación perceptual

Primero se hace un filtrado en bandas, que rechaza en cada una aquellos tonos que comparten banda pero que no serán detectados. Un calculador de enmascaramiento temporal y frecuencia recibe tales datos y, tomando como referencia la señal original, envía o no datos a ser representados por bits. Finalmente, se le da formato al flujo de bits en función del estándar específico.

El sistema está fundamentado en los datos experimentales recogidos por miles de pruebas en voluntarios, de manera que el modelo psicoacústico puede interpretarse como un comportamiento medio del oído humano. Por lo tanto, hay sujetos capacitados acústicamente mejor que otros que pueden detectar la pérdida de calidad inherente a esta codificación.

De los códecs de audio habituales actualmente, la mayoría siguen este modelo o variaciones del mismo y otros han tendido hacia sistemas de compresión más conservadores:

- El formato de sonido **MP2** (conocido como Musicam) es el habitual en ficheros de audio profesional.
- El estándar MPEG-1 Audio Layer 3, nacido a partir del MP2, es un clásico de la familia de códecs con pérdidas; se definió para almacenamiento de audio digital doméstico. Se conoce como **MP3**.
- **Vorbis** es también un estándar de audio con pérdidas resultado de un proyecto de software libre ofrecido como alternativa al estándar propietario MP3, que ofrece una calidad de audio similar e incluso mejor en bajas velocidades de transmisión o lectura (< 64 Kbps).
- **AAC** (Advances Audio Coding), posterior y de mayor calidad que el MP3, es el que se encuentra en los equipos domésticos portátiles de audio de Apple (iPod, iPhone, iPad), la familia de móviles Android, Sony (PlayStation 3) y Nintendo (Dsi, Wii), entre otros. Soporta audio multicanal (hasta 48 canales) y es una codificación con pérdidas.
- **Dolby AC-3** (conocido como Dolby Digital) es un estándar propietario con pérdidas nacido para el cine, pero su alta eficiencia (320 Kbps para cinco canales de audio de alta fidelidad más un sexto canal de surround) y una buena relación con los fabricantes de equipos lo alzó como estándar de audio para DVD y Blu-ray e incluso para la televisión digital en los Estados Unidos (y, por empatía, también accesible en algunas televisiones europeas).
- Sin embargo, el **Dolby Digital Life** es un estándar de codificación en tiempo real dirigido al mercado multimedia, en especial al sector de los videojuegos.

- **Windows Media Audio (WMA)** es la apuesta de Microsoft por un estándar de codificación. Existe tanto con pérdidas como sin pérdidas (WMA Lossless).
- Y, sólo como comentario, **Real Audio** fue un estándar de compresión con pérdidas específicamente diseñado para *streaming*, muy utilizado en los primeros años de Internet, pero su naturaleza propietaria y la falta de mejoras por parte de la empresa Real Networks lo llevó a la desaparición hace unos años.

## 2.6. Diferenciando contenedor y códec

Un **formato de contenedor** es un formato de archivo digital que almacena un determinado tipo de información codificada con uno o varios códecs estándares.

Es habitual confundir ambos conceptos y mezclar el nombre del códec con el del contenedor. Así, Flash Video es un formato contenedor cuya extensión de fichero es .flv o .f4v y en el que el vídeo puede estar codificado en H.264 o en otro códec como Sorenson Spark o VP-6, mientras que el audio puede estar codificado en MPEG-1 Audio Layer 3.

En algunos casos, el nombre del contenedor acaba sustituyendo al del códec por costumbre o simplificación.

Los contenedores más complejos son capaces de soportar distintos códecs de audio y vídeo, manejar subtítulos, capítulos y metadata (o *tags*). Uno de los papeles más importantes del contenedor es propiciar información temporal para poder sincronizar la reproducción de más de un *stream* de forma simultánea.



Buena parte de las aplicaciones de codificación de audio y vídeo indican claramente los códecs de vídeo y audio posibles de un contenedor seleccionado (en la imagen, Super©).

Algunas características que diferencian a los contenedores entre sí son las siguientes:

- su popularidad (y soporte por parte de las aplicaciones),
- el *overhead* (distintos contenedores presentan distintos tamaños de archivo para el mismo contenido),
- el soporte de códecs (por ejemplo, algunos contenedores como AVI no soportan codificaciones con cuadros B),
- el soporte de subtítulos u otras características avanzadas,



Un ejemplo de asociación pura entre códec de vídeo y contenedor

### MP3

El contenedor para el códec MPEG-1 Audio Layer 3 es comúnmente conocido como MP3 y la extensión de los archivos es .mp3, pero es habitual hablar de codificación MP3.

- que sea apto para los servidores de *streaming* como formato de entrada.

La mayoría de los contenedores presentan la posibilidad de flujos de bits variables tanto para audio como vídeo. Alguna de las excepciones más notorias es por ejemplo el contenedor AVI de Microsoft, que no lo permite. Actualmente, cuando un contenedor no cuenta con alguna característica deseada, los distintos programas realizan extensiones para poder soportarlas, aunque muchas veces estas extensiones son incompatibles entre sí, lo que complica el escenario.

La siguiente tabla resume las características de los contenedores más populares:

Características de los contenedores más populares

Contenedor	Códec de vídeo soportado	Códec de audio soportado	Capacitado para <i>streaming</i>	Capacitado para 3D
.3gp	MPEG-4 Parte II H.264/MPEG-4 AVC	AAC AAC v2	-	-
.avi	La mayoría salvo H.264/AVC	La mayoría	Sí	-
.divx	MPEG-4 Parte II	MPEG-1 Audio AC-3 PCM	Sí	-
(Flash Video) .f4v	H.264/MPEG-4 AVC	AAC MPEG-1 Audio	Sí	.
(Flash Video) .flv	H.264/MPEG-4 AVC Sorenson Spark VP6	AAC MPEG-1 Audio PCM	Sí	-
.(Matroska) mkv .mka .mks .mk3d	La mayoría	La mayoría	Sí	Sí
(MPEG) .mp4	MPEG-2 Parte II H.264/MPEG-4 AVC H.263 VC-1	AC-3 MPEG-2 MPEG-4 Vorbis	Sí	Sí
(MPEG) .mpg .mpeg	MPEG-1 MPEG-2	MPEG-1 Layer I, II, III	Sí	-
(MPEG PRO- GRAM STREAM) .ps	MPEG-1 MPEG-2 MPEG-4 Parte II H.264 VC-1	MPEG-1 Layer I, II, III	Sí	-
(MPEG TRANS- PORT STREAM) .ts	MPEG-1 MPEG-2 MPEG-4 Parte II H.264 VC-1	MPEG-1 Layer I, II, III	Sí	-

Contenedor	Códec de vídeo soportado	Códec de audio soportado	Capacitado para <i>streaming</i>	Capacitado para 3D
(BLU RAY TRANSPORT STREAM) .m2ts	MPEG-2 Parte II H.264 VC-1	Dolby AC-3 Dolby Digital Plus DTS DTS HD	No aplicable	Sí
(AVID) .mxf	La mayoría	La mayoría	No aplicable	-
(XIPH.ORG) .ogg	Theora y muchos otros	Vorbis y muchos otros	Sí	-
(Apple) .qt .mov	MPEG-1 MPEG-2 Parte II MPEG-4 Parte II H.264/MPEG-4 AVC Cinepak	AAC MPEG-1 Audio PCM	Sí	-
(REAL) .rmvb	Real Video	Real Audio AAC Vorbis	Sí	-
(Microsoft) .wma .wmv	La mayoría salvo H.264/AVC	La mayoría	Sí	-
(Google) WebM	VP8	Vorbis	Sí	Sí

### 3. El estándar MPEG-2

El grupo MPEG (Moving Pictures Experts Group) de la ISO (Organización Internacional de Normalización) es el grupo más destacado y aceptado por la industria en lo que respecta a la estandarización del vídeo digital.

Tras el éxito obtenido con el estándar MPEG-1 en 1993, el grupo decidió obtener un estándar más completo y mejorado, dirigido a un mercado potencial como el *broadcasting* de televisión digital, atractivo y de gran escala. Finalizado en 1995, el estándar presenta mejoras en la compresión respecto a su antecesor, codificación entrelazada (además de progresiva) y gran flexibilidad debido a la utilización de **perfiles y niveles**. El MPEG-2 estandariza tanto el vídeo como el audio que lo acompaña.

Los perfiles y niveles son variaciones de calidad de compresión que permiten adaptar el estándar a las diferentes exigencias de los sistemas particulares, además de ofrecer a los fabricantes reglas más claras respecto a la conformidad de sus productos con el estándar. Los perfiles y niveles del MPEG-2 los vamos a ver más adelante.

El estándar MPEG-2 es muy utilizado en la actualidad, aunque va siendo lentamente remplazado por el MPEG-4 Part 10/H.264/AVC. Es el utilizado por la mayoría de los sistemas de *broadcast* de televisión digital (DVB), tanto por cable (DVB-C), satélite (DVB-S) o terrestre (DVB-T), y está presente en el mercado doméstico a través del formato de almacenamiento DVD.

Sin ánimo de entrar en detalles profundamente técnicos, haremos una visión general del estándar a partir de las partes que lo conforman: primero, veremos el apartado más importante, la definición del formato multimedia. Los flujos de datos y la forma de empaquetarlos los veremos muy por encima.

El almacenamiento y la transmisión de flujos MPEG-2 se estudia con más detalle más adelante, aquí haremos sólo una primera aproximación. Una breve reseña del códec estandarizado en MPEG-2 y describir los perfiles y niveles de calidad permitidos cerrarán este apartado.

#### 3.1. Otros estándares relacionados: JPEG/JPEG2000, MPEG-1, H.261 y H.263

Algunos estándares anteriores al MPEG-2 fueron fuente de inspiración para el MPEG-2 y en algún caso son aún vigentes:

1) **JPEG/JPEG2000**. Similar al MPEG, el JPEG (Joint Photographic Experts Group) es un grupo de trabajo de la ISO, en este caso especializado en la compresión de imágenes. En 1992, terminó el estándar JPEG, que sigue siendo utilizado. La compresión espacial de la codificación de vídeo digital se encuentra fuertemente ligada a las técnicas presentadas por el grupo JPEG.

**JPEG**

ISO/IEC 10918-1 / ITU-T Recommendation T.81.

En el 2000, se presentó una versión mejorada del estándar conocida como JPEG2000, que apuesta por un nuevo sistema de codificación más eficiente que además posibilita codificar de manera escalada e incluso con compresión sin pérdidas. No tuvo tanta popularidad al tener una licencia.

**JPEG2000**

ISO/IEC 15444.

2) **MPEG-1**. El primer estándar del grupo MPEG es el MPEG-1. Culminado en 1993, actualmente sigue siendo utilizado. El caso más notorio de éxito es el códec de audio MP3.

**MPEG-1**

ISO/IEC 11172.

Para vídeo, el códec MPEG-1 fue utilizado en el formato Vídeo CD (o VCD) actualmente reproducible en la mayoría de los reproductores DVD y de calidad similar a la de un vídeo VHS doméstico.

**MP3**

Especificado en la norma MPEG-1 Part 3 Audio Layer 3.

3) **H.261 y H.263**. En 1993, la ITU-T estandariza el H.261, un códec para servicios de videoconferencia de baja velocidad (se transmitía por RDSI a 64 kbps o múltiplos de éste). Pensado para redes conmutadas como la de telefonía analógica pero con servicios digitales como RDSI, actualmente es el único contexto donde sigue siendo utilizado en general por razones de compatibilidad hacia atrás.

En 1998, la ITU-T estandariza el H.263, un códec de vídeo de mayor calidad y menor flujo de bits resultante (30 kbps). En este códec, aparece el concepto de distintos perfiles de codificación (variaciones de calidad en la compresión que generaron inicialmente algunos problemas de compatibilidad entre los fabricantes, resueltos cuando en el H.264 se clarifican las especificaciones).

### 3.2. ¿Qué es el MPEG-2?

Entremos en el MPEG-2. Este estándar se encuentra completamente definido en la ISO/IEC 13818. Esta norma se encuentra dividida en partes según el tema. Cada parte se considera un estándar en sí mismo; por lo tanto, de forma más correcta, el MPEG-2 es un conjunto de estándares.

Las partes que conforman el estándar MPEG-2 son, entre otras, las siguientes:

- El sistema: cómo sincronizar y conjuntar los flujos de vídeo y audio en un solo flujo de datos.
- El códec de vídeo para señales entrelazadas y progresivas.
- El códec de audio, que es una extensión del MPEG-1 audio (también conocido como MP3) capacitado para sonido multicanal.



- El control del flujo por parte del usuario final (para reproducir, pausar, situarse en un punto en particular, estructurar en escenas).

La propiedad intelectual del estándar MPEG-2 es compartida por múltiples corporaciones. Más de 500 patentes forman el estándar. Los vendedores de productos y servicios basados en el estándar MPEG-2 deben pagar por la explotación de la licencia. La institución beneficiaria es MPEG-LA (<http://www.mpegla.com/avc/>), que administra el conjunto de patentes MPEG.

### 3.3. Codificación y compresión en MPEG-2

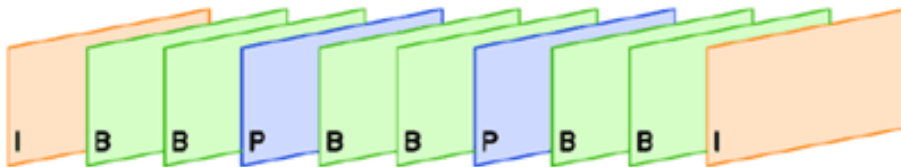
Cuando nos sentamos cómodamente ante nuestro equipo para disfrutar de un DVD en el reproductor o de un fichero en formato MPEG-2 en el equipo multimedia u ordenador y deseamos ir a un punto en particular de éste, el avance rápido o el retroceso del mando a distancia tienen un comportamiento diferente al que tenían estas funciones en el viejo vídeo VHS.

Antes, cuando avanzábamos o retrocedíamos en una cinta de vídeo, podíamos ver con más o menos claridad la película acelerada, pero ahora la imagen va a trompicones, dando saltos de segundos a segundos, presentando instantáneas de la película.

¿Dónde están el resto de fotogramas? ¿Por qué no vemos a los actores moviéndose de forma acelerada? La culpa es del GOP (el grupo de imágenes).

#### 1) MPEG y GOP

La imagen en movimiento puede comprimirse con más pérdidas que la fija, pues entre una imagen y la siguiente usualmente hay pocas diferencias (salvo un cambio de plano o una transición rápida). La intención es, de todas ellas, mantener intactas algunas (es decir, con una compresión similar al JPEG, denominada *intraframe*) que servirán de referencia al resto, que se calculará como diferencia o desplazamiento de éstas. Las imágenes de referencia son denominadas I.



### Ejemplo de GOP de $M = 9$ y $N = 3$ (orden de visualización) Imágenes I

La imagen I (*intracode*) no tiene ninguna referencia con las otras, se comprime espacialmente (à la JPEG, es decir, se divide en bloques de 8x8 píxeles y se transforma, se cuantifica, se codifica, lo que queramos). Se codifican ellas solas, por ello se denominan *intracode*.

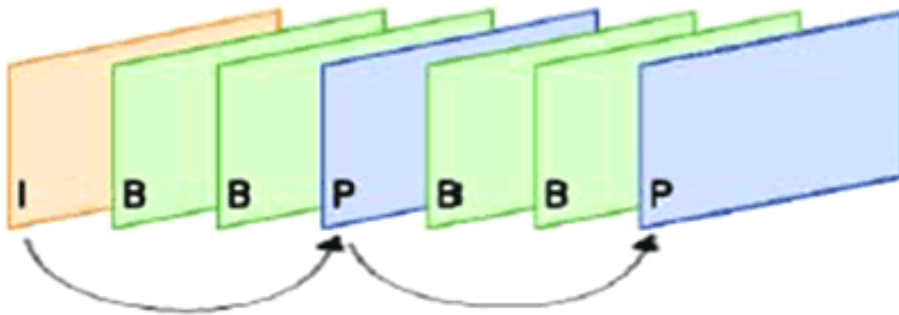
Éstas son las que nos aparecen en pantalla cuando realizamos una búsqueda rápida hacia adelante o atrás en el reproductor de DVD o software: el reproductor de 39,95 € no necesita de otras imágenes (esto implica memoria, muy cara en estos días), sólo unos pocos cálculos para poder descomprimir la imagen I. Es una manera económica de presentarnos el punto aproximado de la película donde estamos.

Si codificásemos todas las imágenes de una secuencia como imágenes I, no tendríamos una gran tasa de compresión y la película no cabría en un DVD o el fichero sería de decenas de gigabytes. Hay que buscar un método más agresivo de compresión para el resto de imágenes y tenemos la oportunidad, pues el ojo humano ante objetos en movimiento no discierne con claridad, este método se denomina **estimación de movimiento**.

### 2) Imágenes P y la estimación de movimiento

Teniendo los pilares I de una secuencia de imagen, las imágenes que existen entre dos imágenes I pueden codificarse mediante estimación de movimiento de dos maneras: imágenes P y B.

Las imágenes P (predictivas) se calculan a partir de la imagen I o la imagen P inmediatamente anterior (no quiere decir que sea la anterior, sino la última I o P que se haya codificado). Son también de gran calidad aunque menor que la I, ya que se calculan mediante una estimación del movimiento existente en la imagen (se divide la imagen en macrobloques de 16x16 píxeles y se intenta ver cuánto se han movido respecto la imagen que toman de referencia, sea I o P).

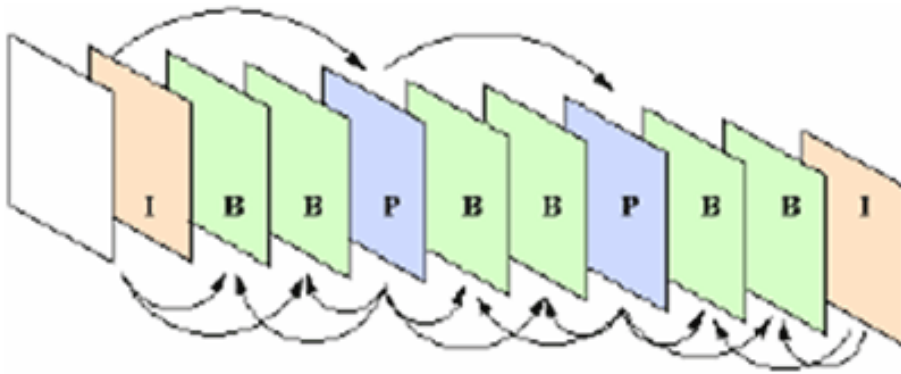


La intención es que de esta imagen no guardemos más que unos cuantos píxeles y muchos vectores de movimiento que hagan referencia a cuánto se han movido los bloques de píxeles de la imagen I o P de referencia. Este proceso provoca un ahorro de bits impresionante. Aquí pues el problema es calcular los vectores de movimiento, es decir, analizar las diferencias entre la imagen P y la de referencia (I o P), compararlas, ver si existen bloques de píxeles similares y detectar en cuánto se han desplazado en horizontal o vertical.

En secuencias estáticas (la cámara quieta y los objetos de la imagen en reposo), las imágenes P son sencillamente nulas. En secuencias en movimiento (y no digamos una persecución policíaca), las imágenes P engordan al tener que almacenar el movimiento de los píxeles de la imagen de referencia y, si encima aparecen nuevos objetos, han de añadirlos.

### 3) Imágenes B

Las imágenes B (bidireccional predictiva) son el caso más extremo de compresión. A partir de dos imágenes de tipo I o P, se deduce la imagen B intermedia, con lo que contiene más información de estimación de movimiento que aumenta la compresión, pero de una calidad relativamente baja. Ved que, si bien las P usaban estimación de movimiento en un solo sentido (desde la I o P anterior hasta ellas), las B usan los dos sentidos (la anterior y la posterior I o P).



### 4) Propagación de errores (calidad visual) y compresión, el eterno dilema

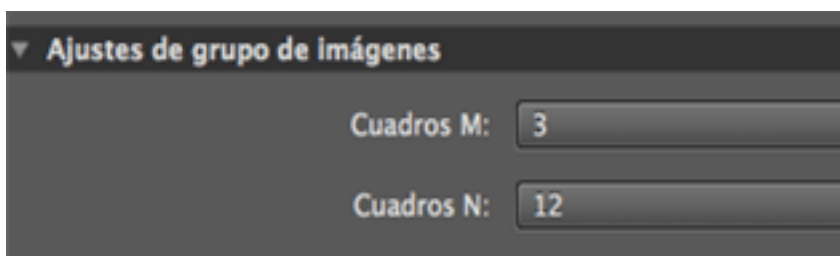
Las imágenes I contienen errores, ya sean propios de la codificación estilo JPEG (una codificación con pérdidas), ya sean externos debidos a su transmisión (por ejemplo, una antena mal instalada para la TDT) o a procesos de grabación (un DVD en malas condiciones). Ahora bien, si toda la ristra de imágenes P y B depende de ellas, es imaginable pensar que esos errores de base van a propagarse y a afectarlas.

También las P pueden contener errores, ya sea heredados de la I o P de la que dependan, ya sea por un error en su transmisión o grabación. La B, sencillamente, son las víctimas últimas de todos los errores que se hayan producido de toda clase.

El valor N es la distancia entre dos imágenes I. Repasando el gráfico que representa un GOP un par de páginas antes, se deduce que el GOP debe poseer un tamaño de N imágenes, ya que comienza con una imagen I y acaba justo antes de la siguiente imagen I. Un valor de 12 es un equilibrio entre una gran distancia entre dos imágenes de referencia I y una compresión interesante con una propagación de errores aceptable.

Aumentar esta distancia supondría dejar en manos de imágenes P y B la calidad de la secuencia y arriesgar que pequeños errores de las imágenes I se amplificasen en todo el GOP. Reducirla sería mejorar la calidad, pero aumentando el peso de la secuencia y reduciendo la compresión.

El valor M es la distancia entre una imagen I y la siguiente imagen de tipo I o P que haya. En el ejemplo de antes, es 3. Cuanto mayor es el valor de M, más errores contendrá la estimación de movimiento pero más alta será la compresión, eterno dilema. En una exportación de proyecto a MPEG-2, por ejemplo Adobe Premiere deja al usuario su elección:



Ejemplo de selección de los valores N y M en una codificación MPEG-2 (caso Adobe Premiere)

### Ejemplos

- DVD: al realizar una búsqueda de imagen con el mando a distancia en un DVD, el vídeo no fluye dinámicamente en pantalla, sino a saltos. Esto se debe a que va saltando de imagen I a imagen I.
- Al codificar una película en MPEG-2, cada cambio de plano fuerza el inicio de un GOP. Así, la primera I contiene los nuevos objetos.
- Vídeo *streaming* por Internet, un GOP de  $N = 12$  y  $M = 3$  es habitual
- Edición de vídeo no lineal: cuantas más imágenes I haya, mejor. Incluso existe el estándar M-JPEG, que está formado exclusivamente por imágenes I.

### 3.3.1. Los flujos elementales de datos

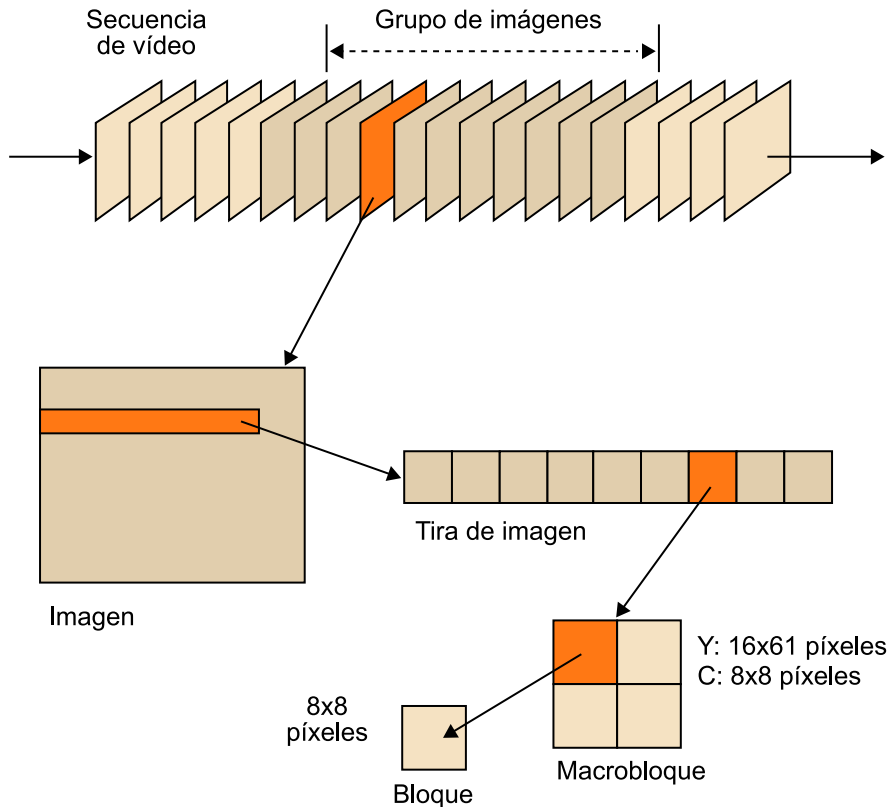
Una vez generadas las imágenes I, P y B, a partir de ellas hay que construir un flujo final de bits que almacenar o transmitir y este proceso tiene varias etapas.

Comencemos definiendo el **flujo elemental** (ES de Elementary Stream), que es sencillamente la salida del codificador, que contiene toda la información necesaria para que un descodificador pueda crear una aproximación lo más certera posible al vídeo original.

Existen dos tipos diferentes de ES, los de audio (Audio Elementary Streams, AES) y los de vídeo (Video Elementary Streams, VES), a continuación se describen cada uno de ellos:

#### 1) Flujo elemental de vídeo (VES)

Un VES es una secuencia de vídeo; habitualmente la salida del codificador ya se encuentra en formato VES. Como se ve en la siguiente figura, el flujo no contiene cada imagen por separado, sino que se estructura en grupos de imágenes (GOP), la unidad de trabajo de cualquier reproductor que quiera presentar este flujo (un reproductor de DVD o un programa de reproducción, por ejemplo).



Esquema del funcionamiento por grupo de imágenes

## 2) Construcción del VES

Cada imagen se convierte en tres matrices rectangulares de valores, cada una con los valores de Y, Cb y Cr, respectivamente. En general, la matriz Y es cuatro veces más grande que las otras dos, pues como ya hemos visto es más relevante la luminancia (Y) que los componentes de color (Cb, Cr).

En función del contenido y de la comparativa que realiza entre imágenes consecutivas, el codificador toma la decisión de definir una de las imágenes como de tipo I, es decir, codificable de manera independiente y pilar básico para la codificación de las inmediatas. Es habitual que, por ejemplo, la primera imagen tras un cambio de plano se defina como I, ya que no posee relación con las anteriores, pero el criterio es mucho más complejo y cada fabricante aplica el que cree más conveniente (el algoritmo no está definido en el estándar).

A partir de la imagen I, se calculan las imágenes P posteriores y las B tal como se explicó anteriormente. Es importante destacar que dentro de un GOP siempre se mantiene la misma secuencia de imágenes, una vez definido el M y N de la codificación.

Es importante recalcar que el MPEG no especifica en sus estándares cómo debe ser un codificador, sino que especifica la salida del codificador para así asegurarse compatibilidad con el decodificador. De esta forma, codificadores y decodificadores de distintos fabricantes pueden interoperar, además de permitir a la industria investigar y mejorar sus técnicas de codificación permaneciendo dentro del estándar.

En cambio, el estándar sí especifica un mecanismo de decodificación (un decodificador), pero debe ser tomado como referencia, dejando también libre a la industria cambiarlo siempre que logre el mismo resultado que el estándar (en este aspecto se piensa en mejoras de *performance* y adaptación del método a hardware específico).

En MPEG-2, existen múltiples implementaciones de codificadores y decodificadores, algunas por hardware y muchas por software. La calidad de codificación MPEG-2 ha mejorado sensiblemente desde sus inicios y su calidad es alta incluso con flujos de menos de 2 Mbps. Es un estándar maduro, pero amenazado por el más actualizado MPEG-4, que lo supera en todos los aspectos, salvo el económico.

Habitualmente, las soluciones de IPTV utilizan equipamiento dedicado y costoso para realizar la codificación. La decodificación se realiza en el equipo del cliente, en su mayoría en software (navegadores de Internet), pero existen algunas excepciones en hardware (televisión por cable).

### 3) Flujo elemental de audio (AES)

Un AES es la salida del codificador de audio. Este caso es muy diferente al de compresión vídeo, aquí no existen diferentes tipos de cuadros, los cuadros de audio se codifican todos de la misma manera, por lo que todos tienen el mismo tamaño.

### 4) Flujo elemental empaquetado (PES)

A la hora de agrupar el *stream* de vídeo (VES) y el de audio (AES) en un solo *stream*, hay que tener cuidado de no generar retardos entre ambos (enviar demasiado vídeo y poco audio supondría no tener suficiente audio o tenerlo retrasado respecto al vídeo). Hay que dividir los dos *streams* en paquetes de un tamaño adecuado antes de ser puestos secuencialmente uno tras otro. Este proceso se denomina empaquetado y supone indicar en cada paquete su tipo y unos identificadores para que el receptor pueda reconstruirlos correctamen-

te, así como datos adicionales para asegurar la sincronización (qué fragmento de audio va con qué fragmento de vídeo). El flujo total es el PES (*packetized elementary stream*).

### 3.3.2. Almacenamiento y transmisión

Por lo general, es necesario combinar varios PES (al menos un audio y un vídeo) para crear un contenido multimedia que será reproducido con posterioridad. Dos posibilidades de tratamiento surgen entonces: almacenar el contenido para su posterior reproducción o transmitirlo (en nuestro caso por una red IP).

#### 1) Almacenamiento

El estándar de MPEG-2 no especifica ningún formato de contenedor de archivo. Esto también sucede con el MPEG-1, el H.263 y otros, por lo que muchos contenedores pueden ser utilizados, como el MP3 para audio o el MOV para vídeo, entre otros.

#### 2) Transmisión

El estándar del sistema MPEG-2 (ISO/IEC 13818-1) define dos métodos de combinación de los datos de audio, vídeo y contenido asociados para la transmisión en un único flujo de datos:

- **Program Streams** (conocido como MPEG-2 PS o simplemente MPEG-PS)
- **Transport Streams** (conocido como MPEG-2 TS o simplemente MPEG-TS)

Un canal (como un canal de TDT) se encuentra formado al menos por un PES de vídeo y un PES de audio. El proceso de unir varios PES que serán reproducidos de forma conjunta se llama **multiplexado**. Para lograr una reproducción conjunta sincronizada (sin desfase entre audio y vídeo), es necesario enviar información de reloj en el multiplexado.

Mientras que el MPEG-PS permite transportar un único canal, el MPEG-TS permite enviar varios simultáneamente (cada uno con una sincronización de reloj independiente). El MPEG-TS incorpora además mecanismos de detección y corrección de fallos en la transmisión, tan habituales en las redes públicas como Internet o el 3G.

Respecto a la transmisión por red, se define un conjunto de protocolos para la transmisión de contenido multimedia por una red IP que veremos más adelante. Cabe destacar, entre ellos y para sistemas de vídeo digital de tiempo real, el protocolo Real-Time Protocol (RTP).

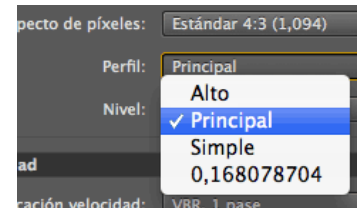
### 3.3.3. Perfiles y niveles de MPEG-2

El MPEG-2 fue diseñado para abarcar la mayor cantidad de sistemas de vídeo digital presentes en el momento de realización del estándar. Esto implica disponer de distintos grados de calidad para cada aplicación: generalmente determinado por la tasa de bits y la resolución de la codificación. Los grados de calidad están definidos en el estándar en perfiles (*profiles*) y niveles (*levels*).

#### 1) Perfiles

El perfil define la resolución del espacio de colores y la escalabilidad del flujo de bits. Las opciones más habituales son las siguientes:

- **Perfil alto (high profile, HP):** destinado a codificar señales de alta definición. En realidad, nunca se utiliza, ya que el MPEG-4 es mucho más efectivo en este caso.
- **Perfil principal (main profile, MP):** el más habitual, dirigido a conseguir una calidad estándar de vídeo. Activa el uso de imágenes I, P y B en la codificación y posibilita codificar señales de vídeo entrelazadas.
- **Perfil simple (simple profile, SP):** necesario cuando el *stream* generado va dirigido a equipos de bajas prestaciones técnicas (como telefonía móvil, PDA o tabletas). No utiliza imágenes de tipo B, lo que simplifica el cálculo de la decodificación y la necesidad de memorias buffer.



Perfiles de exportación (Adobe Premiere)

#### 2) Niveles

El nivel define la resolución de imagen, la cantidad de muestras de luminancia (Y) por segundo, la cantidad de capas de audio y vídeo y el máximo *bit rate* por perfil.

- Nivel alto (high level, HL): 1920x1152, 80 Mbps. Para alta definición, nunca utilizado.
- Nivel alto 1440 (high-1440, H-14): 1440x1152, 60 Mbps. También para alta definición, nunca utilizado.
- Nivel principal (main level, ML): 720x576, 15 Mbps. Es el habitual para imágenes de calidad estándar.
- Nivel bajo (low level, LL): 288x352, 4 Mbps.



Niveles de exportación (Adobe Premiere)

#### 3) Combinaciones de perfiles y niveles

No se utilizan todas las combinaciones de perfiles y niveles. La notación utilizada es perfil@nivel. Las más habituales son las siguientes:



Tabla. Combinaciones de perfiles y niveles

Perfil @ nivel de resolución	Hz	Muestreo	Mbps	Ejemplo de aplicación
SP@LL	176×144	15 4:2:0	0.096	tablets, móviles
SP@ML	352×288 320×240	15 4:2:0 24	0.384	PDA
MP@LL	352×288	30 4:2:0	4	descodificadores
MP@ML	720×480 720×576	30 4:2:0 25	15 (DVD: 9.8)	DVD, TDT, televisión por cable y satélite
422P@ML	720×480 720×576	30 4:2:2 25	50	Sony IMX
422P@H-14	1440×1080 1280×720	30 4:2:2 60	80	reservado
422P@HL	1920×1080 1280×720	30 4:2:2 60	300	reservado

#### 4) El MPEG-2 en la industria

El MPEG-2 es muy utilizado en la actualidad tanto en almacenamiento como en transmisión de audio y vídeo. Algunos de los sistemas que utilizan MPEG-2 son los DVD y los estándares de TDT, satélite y cable en Europa (DVB) y América (ATSC). En cada caso, se implementa sólo la porción necesaria del estándar.

Algunos ejemplos reales meramente ilustrativos de MPEG-2 para almacenamiento son los siguientes:

##### a) DVD (Digital Video Disc)

- Contenedor de vídeo: MPEG-2 Program Stream.
- Codificación de vídeo: MPEG-2.
- Codificación de audio: PCM, MP2 (Musicam), Dolby Digital o DTS en Europa y Dolby Digital o DTS en los Estados Unidos.

Resolución (px):

- NTSC: 720x480, 704x480, 352x480, 352x240.
- PAL: 720x576, 704x576, 352x576, 352x288.
- Tasa de bits de audio + vídeo: Pico 15 Mbps, mínimo 300 Kbit/s.
- Relación de aspecto: 4:3, 16:9, 2.21:1.
- Tasa de cuadros: 29.97 cuadros/seg (NTSC), 25 cuadros/seg (PAL).
- YUV 4:2:0.

##### b) DVB (Digital Video Broadcast)

Incluye la televisión digital terrestre, por satélite y por cable.

### Resoluciones posibles:

- SDTV: 720, 640, 544, 480 o 352×480 píxeles, 24/1.001, 24, 30/1.001 o 30 cuadros/s.
- 720, 704, 544, 480 o 352×576 píxeles, 25 cuadros/seg.
- 352×240 píxeles, 24/1.001, 24, 30/1.001 o 30 cuadros/seg.
- 352×288 píxeles, 25 cuadros/s.

## 4. El estándar MPEG-4 y el H.264/AVC

Tras el MPEG-2, finalizado en 1995, el Moving Picture Experts Group (MPEG) prosigue su trabajo de estandarización. En 1993 (antes de terminar el MPEG-2), comienza el trabajo en el estándar MPEG-4. Este MPEG-4 abarca muchos más aspectos de los sistemas de vídeo digital que sus predecesores MPEG-1 y MPEG-2.

La parte 2 del estándar MPEG-4 (ISO/IEC 14496-2), conocida como MPEG-4 Visual, especifica la codificación y decodificación de vídeo digital. El MPEG-4 Visual se terminó de estandarizar en 1999.

El estándar H.264 se inició por parte del grupo de trabajo Video Coding Experts Group (VCEG) de la International Telecommunication Union (ITU-T). Las últimas etapas del trabajo las llevó a cabo el Joint Video Team (JVT), un grupo conformado por los grupos VCEG y MPEG. El estándar final fue publicado en conjunto en el 2003. Por lo tanto, debe entenderse que el MPEG-4 y el H.264 no son sinónimos, el H.264 es una parte del estándar MPEG-4.

### Nota

La publicación conjunta del estándar en el 2003 fue por parte de MPEG como la parte 10 del estándar MPEG-4 (ISO/IEC 14496-10) y por la ITU-T como H.264.

El H.264 también se llama H.264 Advanced Video Coding (AVC), por lo tanto las formas correctas de llamar a este estándar son MPEG-4 Part 10 y H.264/AVC.

El MPEG-4 Visual y el H.264, aunque contemporáneos, son muy diferentes en su concepción: el MPEG-4 Visual apunta a la flexibilidad abarcando la mayor cantidad de sistemas de vídeo digitales posibles, mientras que el H.264 apunta a la eficacia en la compresión y la confiabilidad de la transmisión.

La codificación utilizada en la mayoría de los sistemas de IPTV es H.264, por eso vamos a dedicar este apartado en exclusiva a esta parte del estándar MPEG-4. Para su mejor comprensión, la estructura de esta sección se presenta de forma muy similar a la del MPEG-2, comenzando por el detalle de los estándares que la componen, siguiendo por algunos detalles del formato, los mecanismos de transmisión y almacenamiento, los perfiles y niveles definidos y culminando con la aplicación actual en la industria.

El contenedor de este códec está bien especificado, se basa en un formato estándar que curiosamente se inspiró en el del QuickTime de Apple: el ISO Media File Format. La extensión de los archivos MPEG-4 es .mp4.

#### 4.1. El estándar y sus patentes

El MPEG-4 se encuentra estandarizado en la norma ISO/IEC 14496. Al igual que para el MPEG-2, se compone de varias partes, entre ellas:

- El sistema: describe la sincronización y la transmisión simultánea de audio y vídeo.
- El vídeo: el códec de compresión para elementos visuales (vídeo, texturas, imágenes sintéticas, entre otros). Uno de los muchos perfiles definidos en la parte 2 es el Advanced Simple Profile (ASP).
- El audio: el conjunto de códecs de compresión para la codificación de flujos de audio; incluyen variantes de Advanced Audio Coding (AAC), así como herramientas de codificación de audio y habla.
- Transporte sobre redes IP: especifica un método para transportar contenido MPEG-4 sobre redes IP.
- H.264 - Advanced Vídeo Coding (AVC): códec estandarizado de señales de vídeo.
- Ingeniería de aplicación y descripción de escenas (BIFS): para contenido interactivo 2D y 3D.
- Formato para medios audiovisuales basado en la ISO: un formato de archivos para almacenar contenido multimedia.
- Extensiones para el manejo y protección de la propiedad intelectual (IPMP).
- El formato de archivo MPEG-4: el formato de archivo de contenedor designado para contenidos MPEG-4.
- El formato de archivo AVC: para el almacenamiento de vídeo.

Es importante resaltar la parte 10 del estándar que especifica el H.264 y la parte 8, que especifica el transporte de MPEG-4 sobre una red IP.

Al igual que el MPEG-2 partes 1 y 2, para la utilización y explotación de este estándar es necesario pagar por una licencia a la institución MPEG-LA.

## 4.2. H.264 o MPEG-4 Parte 10

En filosofía, la codificación H.264 no difiere del MPEG-2 en gran forma. Se especifica el formato de codificación y cómo descodificarlo (nuevamente se deja libre a la industria introducir codificadores eficientes).

También utiliza la nomenclatura de perfiles y niveles para definir variantes de resolución y calidad de vídeo. En la actualidad, relativamente pocos perfiles se encuentran definidos por completo en el estándar, en sus comienzos sólo tres perfiles fueron definidos: Baseline, Main y Extended; sin embargo, en la actualidad parece perfilarse con mayor aceptación en el mercado el perfil High como ya veremos.

Una mejora respecto al estándar MPEG-2 es que se separa la codificación de la transferencia, lo que facilita su implementación.

El formato de salida del codificador se llama Video Coding Layer (VCL) y básicamente es una secuencia de bits que representa vídeo codificado. La jerarquía de codificación de vídeo MPEG-2 (GOP, cuadros, bloques) permanece prácticamente inalterada.

Un problema habitual en el MPEG-2 es que, si hay diferentes subflujos o servicios contenidos en el mismo flujo de bits, al conmutar de uno a otro, el decodificador perdía calidad unos instantes. Contra eso, en los perfiles más altos del H.264 aparece la posibilidad de utilizar dos tipos extra de cuadros (además de los I, P, B):

- **SP (Switching P)** para facilitar cambiar de flujo de codificación (contiene macrobloques I o P)
- **SI (Switching I)** para facilitar cambiar de flujo de codificación (contiene macrobloques I).

## 4.3. Almacenamiento

El estándar MPEG-4 realiza una especificación de almacenamiento en disco, como ya se ha comentado. Se basa en el estándar ISO Media File Format, que se inspira en el formato QuickTime de Apple. La extensión de los archivos MPEG-4 es .mp4.

Muchos otros formatos contenedores poseen esta codificación, como por ejemplo 3GPP, MOV o Flash Video.

## 4.4. Transmisión

Tanto el MPEG-4 como el H.264 no definen un sistema específico de transporte para la transmisión. Con la enmienda 3 del sistema MPEG-2 (ISO/IEC 13818-1), es posible transportar MPEG-4 y en particular H.264 de la siguiente manera:

- Program Streams (conocido como MPEG-2 PS o simplemente MPEG-PS).
- Transport Streams (conocido como MPEG-2 TS o simplemente MPEG-TS), que permite hacer convivir canales TDT en H.264/AVC en HD con los MPEG-2 en SD.

Respecto a la transmisión por red IP, se ha actualizado el protocolo de tiempo real RTP (Real-Time Protocol) para la transmisión de MPEG-4 y H.264. Es relativamente sencillo transmitir contenido multimedia en una red de paquetes IP, incluso limitándonos al contexto de IPTV no sólo se usa RTP. En las redes actuales para transmisión de vídeo digital por Internet, existe una clara preferencia por protocolos basados en TCP y en particular aquellos basados en HTTP, debido a su compatibilidad con las configuraciones frecuentes de *firewalls* y *proxies*.

## 4.5. Perfiles y niveles

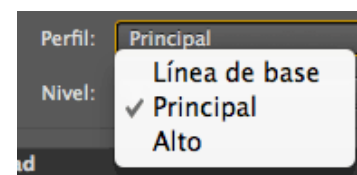
Del mismo modo que el MPEG-2, el estándar H.264/AVC presenta diferentes grados de calidad. Los grados de calidad están definidos en el estándar en perfiles (*profiles*) y niveles (*levels*).

### 1) Perfiles

Existen siete perfiles en el estándar, de los que nos vamos a centrar en los tres habituales, indicados en el Adobe Premiere:

- **Línea de base (baseline profile, BP):** usado para dispositivos sencillos como reproductores portátiles y telefonía móvil, así como para *streaming* de vídeo por red IP.
- **Principal (main profile, MP):** dirigido a *broadcast* (TDT HD) y almacenamiento (Blu-ray), provee alta definición.
- **Alto (high Profile, HP):** para aplicaciones profesionales que requieren alta definición de crominancia y luminancia (4:4:4).

### 2) Niveles



Perfiles de exportación (Adobe Premiere)

Representados con un número del 1.0 al 5.1, son configuraciones para velocidades de transmisión crecientes (desde 64 Kbps en 1.0 hasta 240 Mbps en 5.1).

#### 4.6. Utilización del H.264 en la industria

El H.264 es siempre el elegido en sistema de vídeo digital de alta resolución, como las redes de IPTV y la televisión en HD por terrestre, cable o satélite. Es de esperar que con el paso del tiempo los distintos sistemas de transmisión de vídeo migren a este formato y sustituyan los tradicionales H.262/MPEG-2, tal como ha ocurrido en Francia al apostar desde el inicio de sus emisiones TDT por este formato.

Algunos de los sistemas que utilizan o plantean utilizar el H.264 son los que se mencionan a continuación.

##### 1) Para almacenamiento

Blu-ray Disc de la Blu-ray Disc Association (BDA) (utiliza un perfil alto, HP). Como características más conocidas de Blu-ray tenemos las siguientes:

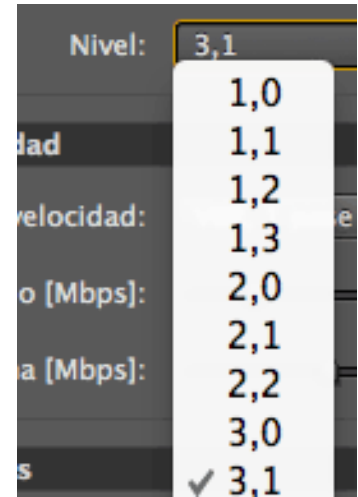
- Contenedor de vídeo: BDAV MPEG-2 Transport Stream (ficheros de extensión .bdav), con lo que la grabación de una emisión de TDT en HD puede almacenarse directamente sin alterar el formato en absoluto.
- Codificación de vídeo: H.264/MPEG-4 AVC (o VC-1 de Microsoft).
- Codificación de audio: Dolby Digital, DTS, Dolby Digital Plus, DTS-HD, PCM.

Combinaciones de resolución y tasa de cuadros en modos progresivos:

- **1080p**: 1920x1080 píxeles a 24/23,97 cuadros/segundo (Estados Unidos/Europa).
- **720p**: 1280x720 píxeles a 50/59,94 cuadros/segundo (Estados Unidos/Europa).
- Otras combinaciones progresivas (1440x1080, 1280x720) a 24 cuadros/segundo (modo cine).

Combinaciones de resolución y tasa de cuadros en modos entrelazados:

- **1080i**: 1920x1080 píxeles a 25/29,97 cuadros/segundo (Estados Unidos/Europa) o 1440x1080 píxeles a 25/29,97 cuadros/segundo (Estados Unidos/Europa).
- **720i**: 1280x720 píxeles a 25/29,97 cuadros/segundo (Estados Unidos/Europa).



Niveles de exportación

Relación de aspecto 16:9 (aunque hay dos modos 4:3 poco utilizados). La tasa de bits de audio + vídeo es de pico 36 Mbps y el YUV, de 4:2:0.

El ya desaparecido formato de almacenamiento HD-DVD de Toshiba poseía características similares.

## **2) Para *broadcast***

El consorcio europeo DVB apostó desde el 2004 por aportar (si no reemplazar) a la codificación H.262/MPEG-2 actual este nuevo estándar, tanto en terrestre (DVB-T) como satélite (DVB-S) y cable (DVB-C). El nuevo estándar DVB-T2, por implantar a partir del 2015 en Europa, optimizará el aprovechamiento de las frecuencias de canal para dar cabida a más canales, lo que podría facilitar la migración al H.264.

El comité estadounidense ATSC especifica H.264 y VC-1 como estándares para la transmisión terrestre de televisión. Igualmente, la asociación ARIB japonesa incorpora el H.264/AVC en su sistema de TDT (ISDB-T).

En satélite, está presente en la mayoría de plataformas (Digital+, BBC HD y Euro1080, por ejemplo).

## **3) Para red de telefonía celular móvil**

The 3rd Generation Partnership Project (3GPP) incluye el H.264/AVC como opcional. En la actualidad (año 2011), es infrecuente poder recibir este servicio debido al ancho de banda necesario.



## 5. Transmisión de vídeo en una red IP (*streaming*)

Las técnicas para la transmisión de vídeo en las redes IP se conocen de forma habitual como mecanismos de *streaming*. El *streaming* se podría definir como la transmisión en tiempo real (en vivo) de audio y vídeo sobre una red.

Antes de la aparición de las técnicas de *streaming*, las aplicaciones multimedia usaban Internet únicamente para realizar transferencias de archivos. Así, una vez se descargaban los contenidos por completo podían ser reproducidos. Actualmente, utilizando el *streaming*, es posible ir visualizando un contenido multimedia a medida que éste es transferido.

Como hemos visto en los estándares digitales anteriores, los flujos de datos de audio y vídeo se empaquetan y forman cada uno un flujo elemental (ES), que además guarda información de sincronía entre ellos. A continuación, se combinan (al menos uno de vídeo y otro de audio) en un flujo elemental empaquetado (PES), lo que permite que se combinen más.

A partir de ese momento, podemos realizar dos cosas:

- almacenar este contenido en un contenedor para su posterior reproducción,
- transmitirlo (en nuestro caso por una red IP).

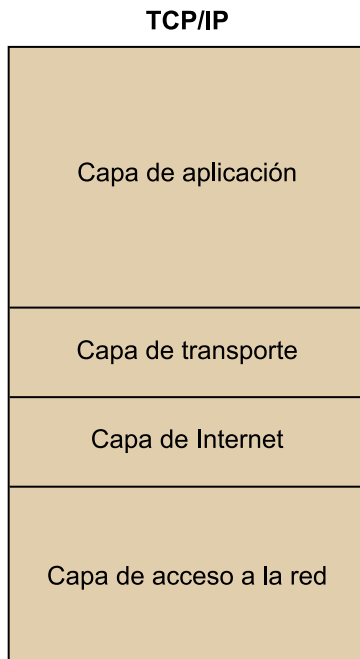
En apartados anteriores, se ha visto por encima cómo se acondicionan los diferentes *streams* para ser almacenados, ahora se va a ver cómo enviarlos por la red.

Existen múltiples técnicas de *streaming* según la realidad particular del sistema de vídeo digital. En este apartado, se van a ver las más importantes en el contexto de las redes basadas en IP, como 3G, IPTV e Internet. Se realizará una introducción a los protocolos de las redes IP (sin profundizar, pues no es objetivo de esta asignatura) y a los protocolos de *streaming* luego.

## 5.1. Las redes IP

Una red puede analizarse desde diferentes puntos de vista. Por ejemplo, se podría estudiar en el ámbito eléctrico, midiendo las tensiones de las señales que circulan y los tipos de conectores necesarios. Este punto de vista es el denominado nivel físico o de acceso a la red, más propio de investigadores e instaladores.

### Esquema



Arquitectura de red TCP/IP

Un segundo punto de vista sería el nivel de enlace, que estudia la identificación de las tarjetas de red y el protocolo de intercambio de bits entre ellas. También es más propia de desarrolladores y programadores que nuestra, al estar situados en un nivel de uso más abstracto.

Los niveles que nos interesan son los dos siguientes: el nivel de red, en el que hablamos de datos circulando por ésta desde un equipo con una dirección a otro u otros, y el nivel de transporte, mecanismos para asegurar que los datos llegan correctamente y, si no lo hacen así, se pide o no reenvío de los datos. Sobre esta cuarta y última capa ya tendríamos las aplicaciones informáticas que usan la red. En esta sección, se hace mención exclusivamente a estas dos últimas capas (red y transporte) de las redes IP con el fin de entender los mecanismos de *streaming* presentados a continuación.

### 5.1.1. Protocolo de red (Internet Protocol, IP)

Las redes IP son un territorio inestable y siempre cambiante de ruta de un punto a otro. Para que un paquete de datos llegue a su destino, ha de tener indicado dicho destino en cada uno de los paquetes que conforman los datos

que enviar. El protocolo IP es un protocolo robusto, que no espera que haya un camino asegurado y estable entre origen y destino, sino que los datos se dividen en paquetes que se envían de manera consecutiva y cada uno navega en una red llena de paquetes hasta alcanzar el destino que tienen marcado.

Dicho técnicamente, el protocolo IP es **no orientado a conexión para la comunicación a través de una red de paquetes conmutados**. Es el protocolo de Internet y de los sistemas de IPTV.

Estos paquetes se envían con información extra para poder realizar acciones relativas a garantizar su propia integridad, pero no proporciona ningún tipo de seguridad de que los datos realmente lleguen al destino (de eso se encargará el protocolo superior, el protocolo de transporte).

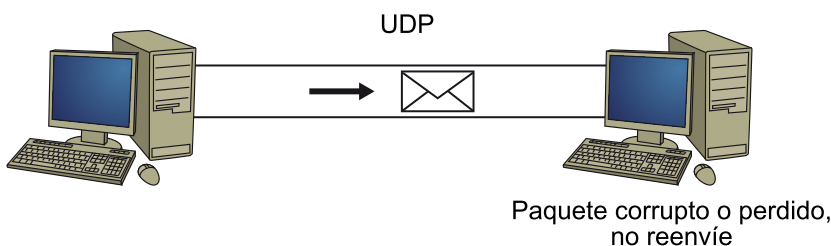
### 5.1.2. Protocolos de transporte (UDP, TCP y SCTP)

Por encima de este cartero que es el protocolo IP, siempre hay un protocolo de transporte cuya responsabilidad es asegurar la transferencia sin errores de datos entre el emisor y el receptor. Sin embargo, ha de ser ágil o su propia acción puede ralentizar la celeridad con la que se reciben tales datos. Los bits de este protocolo van dentro del paquete de red.

Dependiendo del tipo de servicio que queramos, podemos elegir uno de los tres protocolos de transporte siguientes:

#### 1) UDP (User Datagram Protocol)

Protocolo ligero que no da garantía sobre la pérdida de paquetes ni la recepción por duplicado de paquetes. Sólo cuenta con mecanismos de chequeo de datos, si hay que gestionar los errores en la transmisión se deberán hacer en el lado del destinatario (en la aplicación que los reciba). Es el protocolo de transporte natural para *streaming* de vídeo y audio.



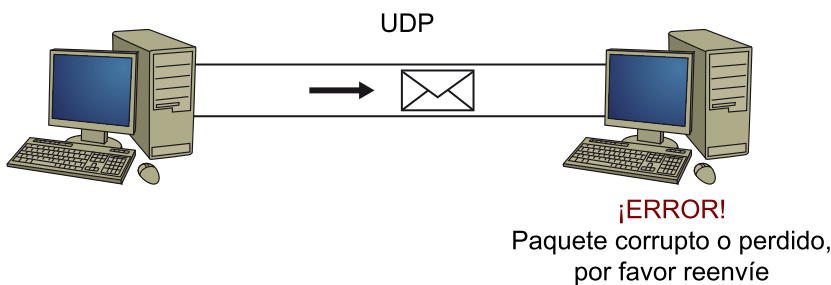
En UDP, una pérdida, error o duplicación de paquete no genera ninguna petición de reenvío por parte del destinatario.

La razón por la que toda videoconferencia que se precie, toda sesión de Spotify o toda web pirata de series de televisión no estrenadas en Europa utilice este protocolo es que consume pocos bits del total de bits de datos y es rápido al inicializarse (pues no hace mucha gestión). Es cierto que es menos seguro que

otros, pero en *streaming* se considera inútil una retransmisión de la información pedida, puesto que ésta llegaría a destiempo (el *streaming* presenta restricciones de tiempo real).

## 2) TCP (Transmission Control Protocol)

Este protocolo nos asegura que los paquetes lleguen y lo hagan en orden. Esto se consigue mediante un mecanismo en el que, cuando no se recibe por parte del destinatario la confirmación del paquete durante cierto tiempo, este paquete se reenvía. Debido a este comportamiento, es posible detectar paquetes perdidos y pedir retransmisión de los mismos. En el caso de transmisión de *stream*, cuando se pierden paquetes la retransmisión aumenta el retardo y el consumo del ancho de banda, lo que puede provocar que se vacíe el buffer de reproductor (y por consiguiente la interrupción de la reproducción del *stream*).



En cambio, en TCP se asegura la recepción correcta de todos y cada uno de los paquetes.

En redes donde no existen mecanismos para asegurar la calidad de servicio, como Internet, en la actualidad se opta por realizar el *streaming* sobre TCP debido a que la existencia de pérdidas esporádicas de paquetes repercuten en la calidad percibida de forma mucho menos sensible en TCP que en UDP. Por el contrario, en redes de IPTV, donde existe calidad de servicio y no hay congestión, el mecanismo de *streaming* sigue siendo el tradicional basado en UDP.

## 3) SCTP (Stream Control Transmission Protocol)

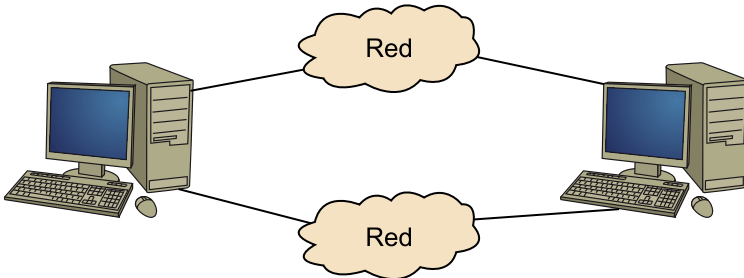
Este protocolo de transporte es una solución de compromiso entre los tradicionales UDP y TCP. A pesar de estar completamente estandarizado, ha tenido una lenta difusión. Tiene una potencial utilidad en el *streaming* de vídeo en redes inalámbricas y otros contextos con grandes pérdidas puntuales de datos.

### Mecanismo NAT

La lenta difusión se debe especialmente por su incompatibilidad con el mecanismo NAT (Network Address Translation o traducción de la dirección de red). Este mecanismo es el que, por ejemplo, posibilita que una red local de una empresa con sus equipos con direcciones IP locales pueda comunicarse con el exterior con equipos remotos con direcciones IP completamente diferentes. El *router* de la empresa es quien se encarga del NAT y hace posible el acceso a Internet de todos ellos.

Este protocolo se basa en datagramas pero, a diferencia del UDP, tiene registro de números de secuencia con los que realiza controles de paquetes fuera de orden y pérdidas que serán retransmitidas. A diferencia del TCP, este protocolo

permite, dentro de una misma conexión (origen SCTP y destino SCTP), diferentes direcciones IP. Es decir, cada uno de los extremos de conexión puede tener diferentes IP (manteniendo el puerto de conexión) y los paquetes serán enviados indistintamente por cualquier IP y a cualquier IP correspondiente cada extremo SCTP.



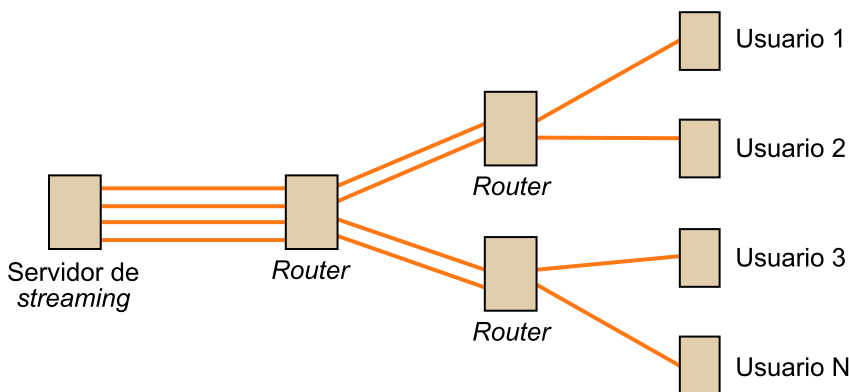
El SCTP posibilita diferentes direcciones IP por un mismo puerto.

También soporta el tráfico sobre la misma conexión de diferentes tipos de paquetes manteniendo número de secuencia para cada uno de ellos.

## 5.2. Técnicas de multidifusión

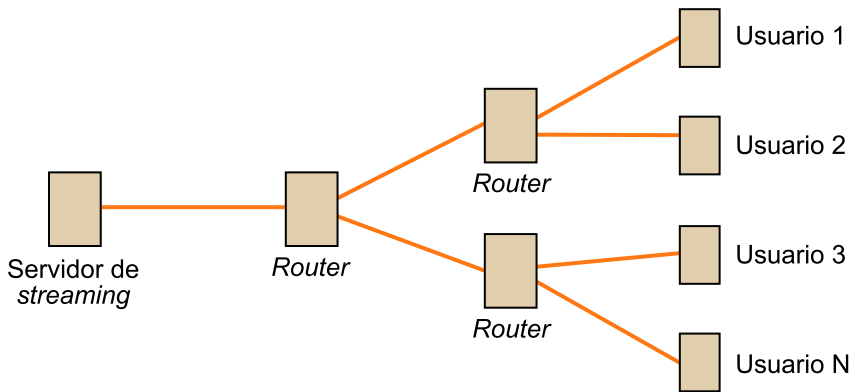
En las redes IP, existen distintas técnicas para la difusión de la información que pasamos a enumerar.

1) **Unicast**: el envío de un punto a otro en una red se denomina unidifusión. El mecanismo habitualmente utilizado por todas las aplicaciones tiene un número N máximo de usuarios limitado.



N usuarios visualizando simultáneamente un programa de televisión por Internet. En *unicast*, el servidor ha de enviar el *stream* para cada usuario.

2) **Multicast**: el envío de la información a múltiples destinos simultáneamente con la ayuda de *routers* preparados que, si están enviando el *stream* a un usuario, al recibir otra petición duplican los datos del *stream*. El servidor queda mucho más descargado, pero obliga a *routers* capacitados para este protocolo.



En *multicast*, el servidor envía inicialmente un *stream* al primer usuario. Los demás reciben el suyo a través de reenvíos de los routers, que se percatan de poseer más de un usuario con la misma petición.

3) **Broadcast**: el envío a todos los nodos en una red se denomina difusión amplia. Lógicamente aplicable sólo en contextos reducidos, por ejemplo para el descubrimiento de recursos en una red local (LAN).

La técnica de *multicast* es de fácil descripción y de complicada implementación. El *multicast* debe usar la estrategia más eficiente para el envío de los mensajes sobre cada usuario de la red (con el fin de disminuir el consumo de ancho de banda), utiliza cada enlace a lo sumo una vez para que cada paquete sea difundido y crea copias en los routers cuando los enlaces en los destinos se dividen. Habitualmente esto se logra mediante la utilización de **árboles de cubrimiento** y surge un protocolo para mantener dicho árbol a través de los routers.

Desde el punto de vista del emisor y los receptores, el proceso es sencillo: el emisor envía la información una sola vez a una dirección IP especial (de forma idéntica a una difusión *unicast*); los receptores se suscriben al grupo de *multicast* asociado a esa IP especial y reciben una copia de todo lo enviado por el emisor. Por lo tanto, antes del envío de la información en multidifusión, deben establecerse una serie de parámetros.

Para poder recibirla, es necesario establecer lo que se denomina grupo *multicast*. En IPv4, las IP especiales pertenecen al rango 224.0.0.0 a 239.255.255.255 y el protocolo utilizado para manejar y asociar a los grupos de *multicast* se llama Internet Group Management Protocol (IGMP).

Dependiendo de la información, se elige el mecanismo ideal para su difusión. En IPTV, la transmisión de canales en vivo habitualmente se realiza mediante *multicast*, puesto que se espera que sean observados por una numerosa audiencia (su envío individual a cada espectador *unicast* redundaría en un dimensionamiento inadmisibles de la red).

En cambio, el **vídeo bajo demanda (VoD)**, por naturaleza es punto a punto y por lo tanto se suele utilizar *unicast*. Pero esto no es una regla, incluso en el contexto de IPTV, por ejemplo en los casos siguientes:

- Existen técnicas para reducir el tiempo de *zapping* entre los canales en vivo (enviados por *multicast*) que implican el envío *unicast* de una ráfaga inicial de gran ancho de banda para llenar rápido el buffer del receptor y reproducir rápidamente tras el cambio de canal.
- Por otro lado, en los sistemas de VoD, cuando un contenido es muy solicitado bajo demanda, puede ser reproducido en forma de nVoD (**near VoD**), que implica el envío por *multicast* y una espera inicial de algunos minutos por el cliente en espera de la próxima transmisión *multicast*. Una técnica un poco más sofisticada implica el envío de una buena porción del VoD por *unicast* y el almacenamiento en buffer de la transmisión *multicast* antigua más cercana en el tiempo. Esto evita la espera inicial del sistema nVoD, pero exige a los receptores poseer un espacio de memoria mucho mayor para almacenar. Este tipo de mecanismo mixto aún ha de investigarse más.

Debe concluirse que en los sistemas de vídeo digital especializados como IPTV la utilización eficiente de las técnicas de difusión redundará en una mejor calidad de experiencia y aprovechamiento de los recursos de la red.

La situación en el contexto de Internet es muy distinta. En los comienzos de Internet, el *multicast* no se encontraba estandarizado y difícilmente soportado por los sistemas operativos y hardware dedicado de la época. Hoy, por razones de índole comercial, se mantiene esta restricción sobre Internet, por lo tanto no se soporta *multicast* (y por razones lógicas de escala tampoco el *broadcast*). Una esperanza para el soporte de *multicast* en Internet es con el protocolo IPv6, la nueva versión del protocolo IP, que nativamente lo soporta.

Si Internet soportara *multicast*, los sistemas dedicados de IPTV presentarían una fuerte competencia, que sería una oferta de contenido global para un público global (por este motivo no se ve atractivo para los proveedores de conectividad ofrecer tales beneficios a los proveedores de servicios de Internet). Los actuales sistemas de vídeo en Internet realizan, según sus posibilidades, un *multicast* en capa de aplicación mediante el despliegue global de *data-centers* y la réplica inteligente de la información.

A pesar de ser técnicas de *multicast* más eficientes, el problema subsiste en el tramo más comprometido: la red de acceso, donde estas redes sólo pueden hacer distribución *unicast* (como YouTube, Google Video y Netflix).

### 5.3. Protocolos de *streaming*

Existe una tercera categoría de protocolos diseñados específicamente para *streaming*, conocidos como protocolos de tiempo real, que intentan asegurar el envío de flujos de datos con una cadencia asegurada, tal como necesita el material audiovisual.



Tanto *multicast* como *broadcast* están restringidos actualmente en Internet.

### 5.3.1. Protocolos de tiempo real (RTP, RTCP y RTSP)

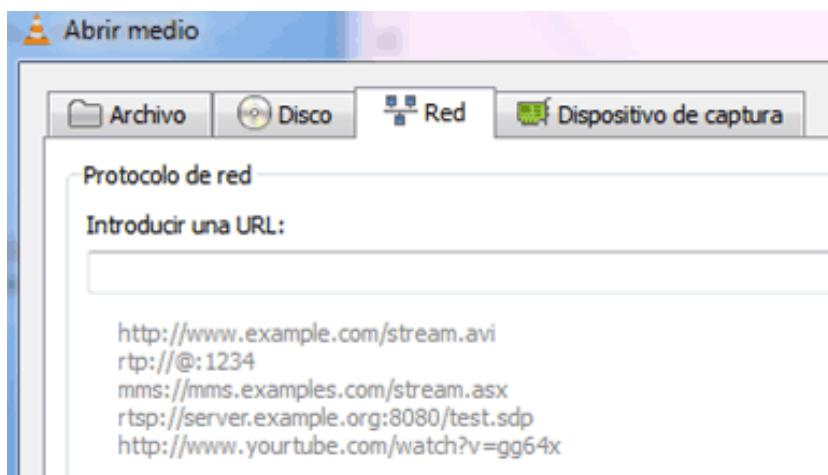
Los protocolos de tiempo real son los siguientes:

1) **RTP (Real-time Transport Protocol)**. Es un protocolo de transporte desarrollado para *streaming*, usualmente montado sobre UDP. El transporte de RTP sobre TCP también se encuentra definido, pero es menos utilizado por presentar una latencia y sobrecarga mayor que UDP, lo que es una desventaja para aplicaciones de tiempo real. Incluye datos extras no presentes en TCP, como marcación de tiempo y número de secuencia por si llega desordenado o si se pierde, lo que contribuye a facilitar el transporte en forma continua. También hay datos de control que permite al servidor realizar el *streaming* a una tasa de flujo definida y correcta.

#### Marcación de tiempo

Gracias a la marcación de tiempo, guarda en cada paquete la hora en la que se transmite a partir de un reloj de referencia, lo que facilita la sincronización.

Lamentablemente, el RTP no cuenta con información de sincronización ni con mecanismos para la recuperación de fallas nativos. Una serie de códecs se encuentran estandarizados para poder ser enviados sobre el RTP, varios dentro de la línea MPEG-2.



Al indicar en la aplicación la dirección de la fuente de *streaming*, hay que indicar el protocolo de tiempo real en el que se está transmitiendo (en la ilustración, el programa Videolan).

2) **RTCP (Real Time Transport Control Protocol)**. Es usado en conjunción con el RTP para la recepción de informes estadísticos. Permite, por ejemplo, la detección de fallos en el árbol de distribución de clientes *multicast*, del número de paquetes perdidos y estadísticas de fluctuaciones (*jitter*).

Existen informes enviados por el emisor e informes enviados por el receptor. Por ejemplo, los del emisor son habitualmente el volumen de cuadros enviados o cantidad de bytes y los informes del receptor presentan los cuadros perdidos o la tasa de cuadros entregados.

Una característica particular que tiene este protocolo es que, para amortizar la sobrecarga que supone consumir bits para el protocolo en vez que para datos (*overhead*), se pueden reunir varios mensajes RTCP y enviados en un mensaje RTCP compuesto. Los paquetes deben estar compuestos al menos por un men-



saje de receptor o emisor y el nombre del participante, y deben ser enviados periódicamente sin llegar a consumir el 5% del ancho de banda de la sesión. Los paquetes RTCP se transportan sobre UDP.

El RTCP no permite encriptación, autenticación y autorización. Una actualización con estas funcionalidades es el estándar Secure Real-time Transport Protocol (SRTCP).

**3) RTSP (Real-Time Streaming Protocol).** Realiza control sobre datos multimedia de tiempo real, lo que brinda la posibilidad de interactividad con el reproductor, de forma similar a un vídeo reproductor doméstico. El RTSP permite reproducir, pausar, adelantar y más. También puede reaccionar a congestiones en la red y reducir el ancho de banda.

Los comandos estandarizados en el RTSP son los siguientes:

- **Describe:** con este comando se recibe una descripción del recurso, una lista de los *streams* de datos de núcleo necesarios.
- **Setup:** se indican los parámetros de configuración del *stream*.
- **Play:** realiza la reproducción del *stream* especificado.
- **Pause:** detiene la reproducción del *stream* indicado, reiniciable con un Play.
- **Teardown:** detiene y libera los recursos utilizados, así finaliza el *stream*.

El RTSP se inspiró en el protocolo web (HTTP 1.1), pero con la mejora por la que se puede mantener el estado de la conexión (el HTTP no mantiene estado) y que ambos (cliente y servidor) pueden realizar pedidos. El RTSP soporta el RTP como protocolo de transporte.

Una de sus utilidades es brindar una forma inicial de escoger el canal de distribución óptimo hacia el cliente.

### 5.3.2. Otros protocolos de *streaming*

Aún nos queda enumerar unos cuantos protocolos de tiempo real más, que son los siguientes:

**1) RDT (Real Data Transport).** Es un protocolo propietario para la transmisión de audio/vídeo desarrollado por RealNetworks en 1995. Al igual que el RTP, trabajaba en conjunto con el RTSP para el control del streaming y, al no presentar mayores ventajas que su equivalente estándar RTP, su continuidad está en entredicho.

**2) HTTP (Hypertext Transfer Protocol).** A pesar de ser diseñado con fines completamente distintos (es el protocolo por el que podemos navegar por web), el HTTP es utilizado para el *streaming* de audio y vídeo por distintas razones. El principal motivo es que la mayoría de las conexiones a Internet se

#### Escoger el canal

Algunos clientes pueden tener filtrados en el cortafuegos los paquetes UDP, por lo que el servidor de *streaming* debería proveer la posibilidad de escoger entre diferentes protocolos de transporte como UDP o TCP o UDP *multicast*.

encuentran protegidas por un cortafuegos o un proxy, por lo que la forma más sencilla de hacer *streaming* de un contenido en ese contexto es hacerlo usando HTTP.

Lógicamente tiene mayor sobrecarga (*overhead*) que la familia de protocolos de tiempo real (RTP, RTCP y RTSP), por lo que no está bien considerado en muchos contextos, por ejemplo por el grupo MPEG. Sin embargo, como ya se indicó antes, es una opción válida para Internet, debido a la existencia de cortafuegos/proxies y a presentar una forma simple de afrontar los errores por pérdidas en la red (mediante retransmisión).

El *streaming* sobre HTTP no se encuentra muy estandarizado, a pesar de ser lo más utilizado por las radios en Internet (<http://www.shoutcast.com/>).

Actualmente sobre HTTP se pueden transmitir varios tipos de formatos de vídeo. Además de flujos de datos de vídeo crudos (como los aceptados por el RTP), sobre el HTTP se puede enviar MPEG-TS, MPEG-PS, OGG, ASF y otros.

**3) MMS (Microsoft Media Services protocol).** Es un protocolo propietario para la transmisión de audio/vídeo. Desarrollado por Microsoft, fue abandonado oficialmente en el 2003 y dejó de dar soporte en el 2008.

Operaba sobre TCP, UDP o HTTP, negociando cliente y servidor cuál elegir de los tres, o unívocamente el servidor según el estado de la red. Si el cliente no podía negociar una buena conexión utilizando MSS sobre UDP (abreviada MMSU) entonces intentaba con MMS sobre TCP (abreviada MMST). Si fallaba, utilizaba una versión modificada del HTTP para establecer la conexión (abreviada MMSH). En cada paso, iba creciendo el *overhead*, lo que lo volvía ineficiente.

## 6. MPEG-2 TS (Transport Stream)

El MPEG TS es un protocolo que nos brinda un mecanismo para **multiplexar**, es decir, combinar en un solo *stream* los *streams* de audio, vídeo y datos para así transmitirlos por una red. Es muy utilizado en los sistemas de vídeo digital de televisión por cable y TDT.

Como ya hemos dicho, un Elementary Stream (ES) es básicamente la salida del codificador. Así, existen ES de dos tipos: los *streams* de vídeo (VES) y los de audio (AES). Para manejar los diferentes ES, se dividen en paquetes de diferente tamaño, según las características de la aplicación y del decodificador. Al proceso de partición en paquetes del ES se denomina **empaquetado** y un ES empaquetado es un Packetized Elementary Stream (PES).

Cuando se utiliza el MPEG-TS, varios PES pueden ser transmitidos de forma conjunta en un mismo *stream* TS, a este proceso se le llama multiplexado. En la mayoría de las aplicaciones, se necesita audio y vídeo, por lo tanto se requiere el multiplexado de al menos un PES de audio y un PES de vídeo. Los PES transmitidos dentro de un TS son agrupados en programas. Habitualmente un *stream* TS transmite varios canales de televisión en simultáneo, cada uno es un programa formado por al menos un PES de vídeo y un PES de audio. Además de los PES, un *stream* TS transmite información de control en formato de tablas.

El MPEG-TS no brinda simplemente una forma adecuada de realizar el multiplexado de los diferentes ES, sino que también ataca el problema de recrear el reloj de la fuente en cada uno de los receptores para lograr así una correcta decodificación y sincronismo del audio y del vídeo.

### 6.1. Paquetes y tablas del Transport Stream

Los paquetes TS tienen un tamaño fijo de 188 bytes. Un flujo TS está compuesto por uno o más programas; para identificar qué PES pertenece a qué programa, el TS envía periódicamente tablas con esa información.

Dentro de la cabecera de los paquetes TS, se encuentra un campo llamado Packet Identifier (PID), que identifica el contenido específico de ese paquete TS. El contenido de un paquete TS puede ser:

- parte de un PES de audio/vídeo,

#### Tablas del flujo *Transport stream*

En estas tablas se puede, por ejemplo, enviar información de la programación presente en el *stream*.

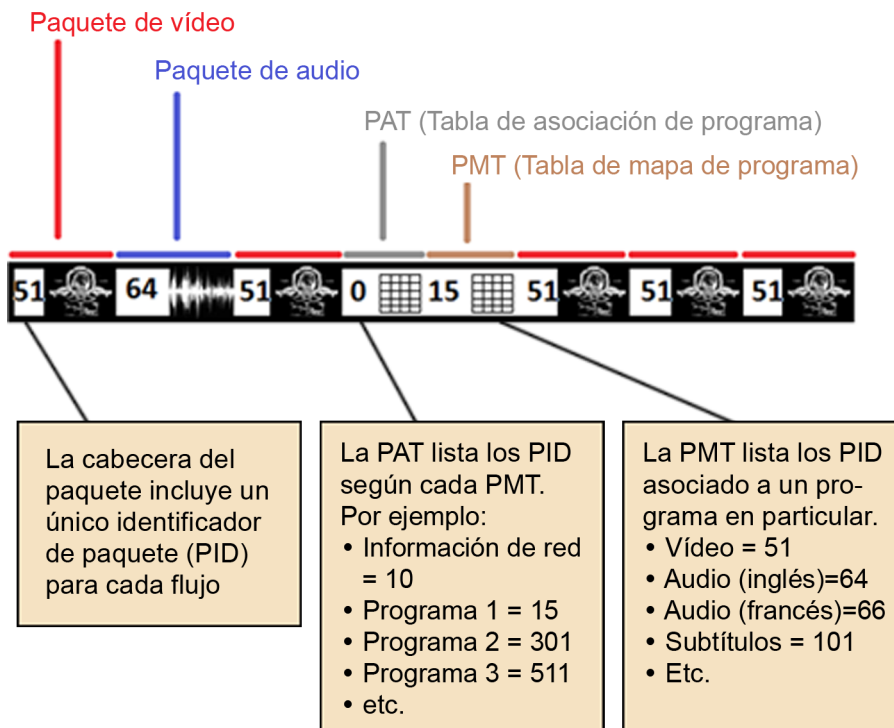
- información de control, conocido como Program Specific Information (PSI).

La PSI está compuesta por cuatro tablas diferentes:

- la Program Association Table (PAT),
- la Program Map Table (PMT),
- la Conditional Access Table (CAT),
- la Network Information Table (NIT).

Dentro de estas tablas, se encuentra toda la información necesaria para extraer de forma correcta cada *stream* de vídeo, audio y datos y seleccionar así los diferentes programas en el receptor. Las tablas PAT y PMT siempre están presentes dentro del *stream*, mientras que las otras dos pueden depender del *stream*, que se está transmitiendo.

En la tabla PAT (identificadas con PID = 0 x 0000), se asocia el número de programa con el PID en el que vendrán la PMT para dicho programa. Por ejemplo, en la figura siguiente se muestra que en los paquetes con PID número 15 van a ser transmitidos los paquetes correspondientes a la PMT del programa 1. A su vez, dentro de la tabla PMT, y entre otras cosas, se encuentra qué PID y, por lo tanto, qué ES están asociados a qué programas; en este ejemplo, se define que el PID número 64 es un ES de audio correspondiente al programa número 1.



## Resumen

La captación de audio y vídeo presenta un nivel de dificultad mayor que la captura de imagen: la naturaleza entrelazada del muestreo y los tipos de muestreo de luminancia y crominancia en función de la densidad de información deseada por píxel. Pero de mayor dificultad, si cabe, es la gestión del volumen de datos digitales generados en la captura, que obliga a aplicar técnicas de compresión.

Dos grandes familias de compresión de audio y vídeo son definidas, las que generan y las que no generan pérdidas de información en la compresión. El estudio de la respuesta humana ante la redundancia espacial y temporal hace posible encontrar técnicas de compresión con pérdidas que minimizan la información visual perdida, por lo que actualmente son las más utilizadas. En el caso del audio, el modelo psicoacústico hace posible el mismo criterio.

Los estándares de compresión de audio y vídeo se conjuntan en un contenedor, un formato de almacenamiento de fichero o de *stream* de datos que facilita su intercambio e interoperabilidad entre diferentes usuarios. De entre éstos estudiamos con detalle el MPEG-2, caracterizado por una estructura de grupos de imágenes, combinación de tres tipos de cuadros (P, B e I) y un sistema definido de perfiles y niveles. Las técnicas de compresión espacial y detección de movimiento utilizadas son un compromiso entre nivel de compresión y la propagación de errores visuales. Basándose en el anterior, el MPEG-4 AVC/H.264 posee mejoras sustanciales que le brindan mayor calidad a menores velocidades de transmisión.

Para finalizar, se enumeran los problemas encontrados en la transmisión en vivo de datos audiovisuales por redes de datos, donde destacan los protocolos de transporte y de tiempo real seleccionables para asegurar la cadencia de la recepción en situaciones de pérdida, error o duplicación de datos y se analizan los mecanismos de transmisión de datos entre los *routers* de la red para reducir la carga inherente sobre la infraestructura de comunicaciones y, por ende, su coste.

