

Plataformas de publicación y distribución

PEC2

Nombre	Jesús
Apellidos	García Arranz
Nombre de usuario UOC	Jegaarr

https://github.com/AI21IA/Plataformas_PEC2

Ejercicio 1: familiarización con la aplicación Avidemux. Contenido estático

Pregunta 1.1: Identifica el tipo de codificación de vídeo del clip original

Códec Vídeo: H264, y el fomato de Píxel, 4:2:0 por tanto conn submuestreo de crominancia (Graham & Summersby, 2025) y esto implica “con pérdidas”.

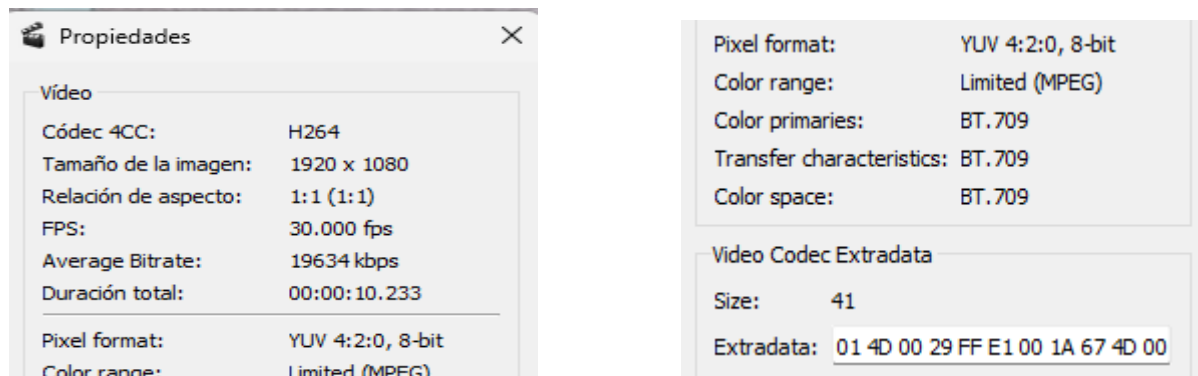


Figura 1.- Codificación asset-01

Pregunta 1.2: Mira las características del vídeo: ¿qué definición en píxeles tiene? Cuántos cuadros/segundo tiene? ¿Cuántos segundos dura? Con ellas, calcula cuál sería el peso en bytes de un fichero de vídeo que tuviera el mismo contenido y duración, pero sin comprimir. Recuerda: un píxel de color son 3 bytes (24 bits) en un vídeo no comprimido..

La definición en píxeles es de 1920 x 1080 (HD)

FPS: 30 fps (cuadros por segundo)

Dura: 10.233 s

Peso sin comprimir: ancho x alto x n° bits = 1920 x 1080 x 8bits/canal x 3 canales x 30 fps x 10.233s (bits)

ancho	alto	bits/canal	nº canales	FPS	Duración (s)	Total Bits	Bits/Byte	Peso(Bytes)
1.920	1.080	8	3	30	10,233	15.277.787.136	8	1.909.723.392

bit (b):

byte (B):

kilobyte (kB):

megabyte (MB):

gigabyte (GB):

Figura 2.- calculadora bytes asset-01

Es decir, **1,78 Gigabytes** (1.909.723.392 bytes)

Pregunta 1.3: Tomando como referencia el fichero de vídeo sin comprimir, ¿cuál es, entonces, el factor de compresión que ya posee asset-01?

Factor de compresión= Peso sin comprimir / Peso imagen comprimida

Peso imagen comprimida= 24.526 Kbytes

kpbs	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
19.634	19.634.000	10,233	200.914.722	25.114.340	24.526

Factor compresión= 76

	1.909.723.392
	25.114.340
Factor	76

Pregunta 1.4: Identifica el tipo de contenedor. Confirma si cuadra o no con la tabla de contenedores del apartado 3.3 del módulo. Si no existiese en el cuadro, haz una búsqueda por Internet

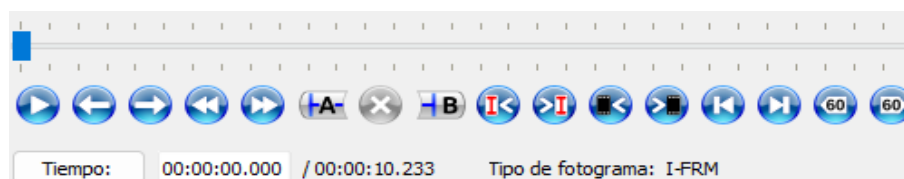
El contenedor es MP4. Sí, si que cuadra con la tabla de contenedores del 3.3

			Formato	Tipo	Licencia	MKV	MP4
			MPEG-H HEVC	Con o sin pérdida	Patente gravada	Sí	Sí
			MPEG-4 AVC	Con o sin pérdida	Patente gravada	Sí	Sí

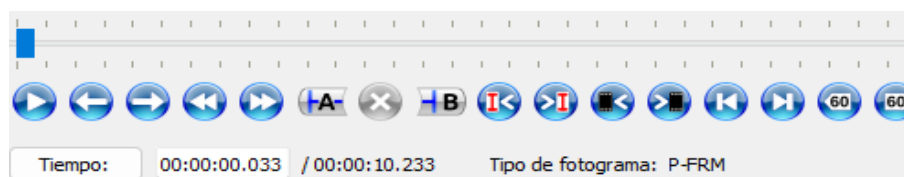
Nombre	Fecha de modificación	Tipo
asset-01.mp4	17/04/2025 19:48	Archivo MP4

Figura 3.- Contenedor, tipo y tabla

Pregunta 1.5: ¿De qué tipo es el primer fotograma? ¿Y el segundo?



Primer fotograma es de tipo: I



Segundo fotograma es: P

Figura 4.- Tipos de Fotograma Asset-01

Pregunta 1.6: Cuántos cuadros de este tipo hay en el clip entero? ¿Qué relación hay entre el contenido de la imagen y la existencia de cuadros tipo I??

Sólo hay 2 fotogramas tipo I , en los instantes 0s y 5,266s



Figura 5.- Recuento de cuadros tipo I

Un fotograma I (Intra-frame) es una imagen completa, como un archivo de imagen JPG. Como se indica en el tema 3 de los apuntes, las trama I se muestran en pantalla en las búsquedas Rápidas al reproducir, porque con pocos datos y cálculos, se descomprime económicamente la imagen estática I, mostrando qué altura de película estamos viendo.

Pregunta 1.7: En resumen, ¿qué tipo de estructuras GOP posee este clip? ¿Y por qué?

La estructura GOP del H264 se basa en tramas I tramas P, y tramas B (Wang, 2024). En este caso particular, tenemos la primera trama I, seguida de 157 tramas P, y luego una nueva trama I, seguida de tramas P. No hay tramas de tipo B en este video. Es decir tenemos una imagen I, que muestra la primera imagen y aporta la correlación espacial, seguida de tramas predictivas P, con la correlación temporal.

Pregunta 1.8: ¿Hay que tener en cuenta algo sobre GOPs cuando se edita un vídeo? Indaga en los fórums de Avidemux para encontrar información sobre qué hace esta aplicación (y cualquier otra aplicación de edición) cuando corta un clip para ensamblarlo a otro.

Como se indica en el foro de avidemux (Avidemux , Forum;, 2021), al ensamblar cortes de vídeos diferentes, se puede producir dos problemas debido a que pueden tener diferentes estructuras GOP, y además habría que hacerlo en las tramas de tipo I. En caso contrario, al hacerlo por ejemplo en tramas de tipo P, la fusión podría mostrar imágenes distorsionadas, transiciones defectuosas y pérdidas de fotogramas, en esos puntos de unión. En otros foros alternativos, se indica que algunos videos producidos desde avidemux como fusión de 2 vídeos, tienen problemas para ser reproducidos desde algunas aplicaciones. Las recomendaciones de otros usuarios eran redecodificar de nuevo el video, con GOP.

Pregunta 1.9: MediaInfo ¿Qué información coincide con la que da Avidemux? ¿Qué información adicional da? Recuerda obviar todo lo referente al audio..

Coinciden en cuanto a Duración, tamaño, FPS, formato de píxel, códec, Bitrate promedio

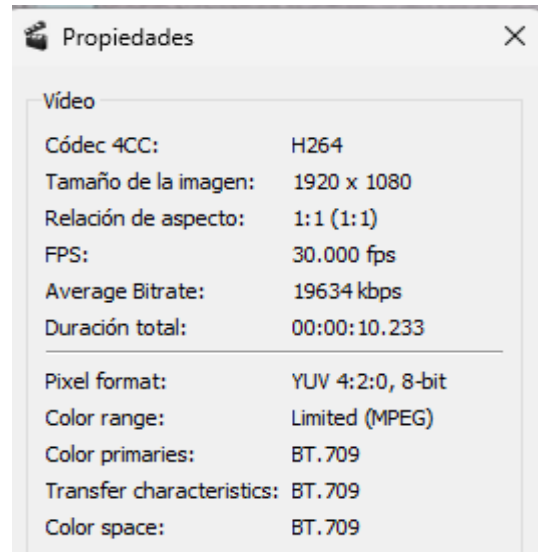
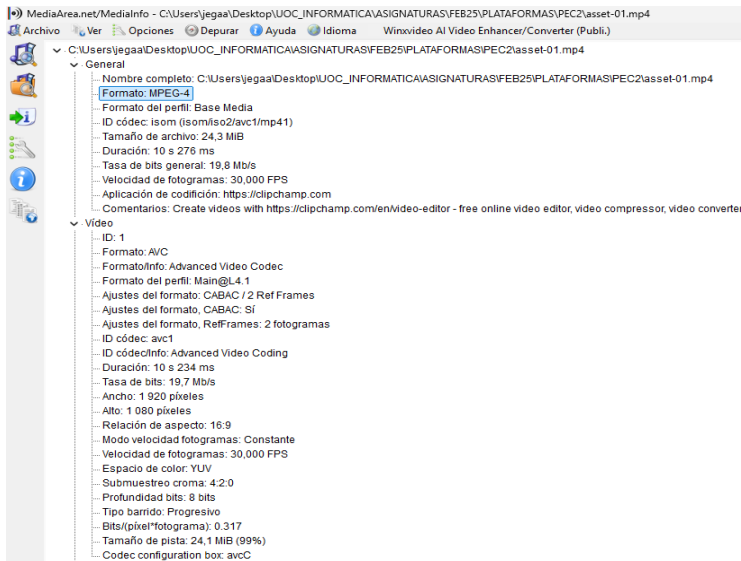


Figura 6.- MedainInfo vs Avidemux

A mayores, MedainInfo muestra metadatos como la aplicación de codificación, comentarios, el ID Codec, la relación e aspecto (16:9), el modo de velocidad de fotogramas (constante), $\text{bits}/(\text{píxel} \times \text{fotograma}) = 0.317$, formato de perfil ([Main@L4.1](#)) y ajustes de formato (Cabac /2), Codec Configuration Box: avcC...

Ejercicio 2: Contenido dinámico

Pregunta 2.1: Mira las características de este nuevo clip y vuelve a responder: ¿qué definición en píxeles tiene? ¿Cuántos cuadros/segundo posee? ¿Cuántos segundos dura? Con ellas, calcula cuál sería el peso en bytes de un fichero de vídeo que tuviese el mismo contenido y duración, pero sin comprimir. Recuerda: un píxel de color son 3 bytes (24 bits) en un vídeo no comprimido.

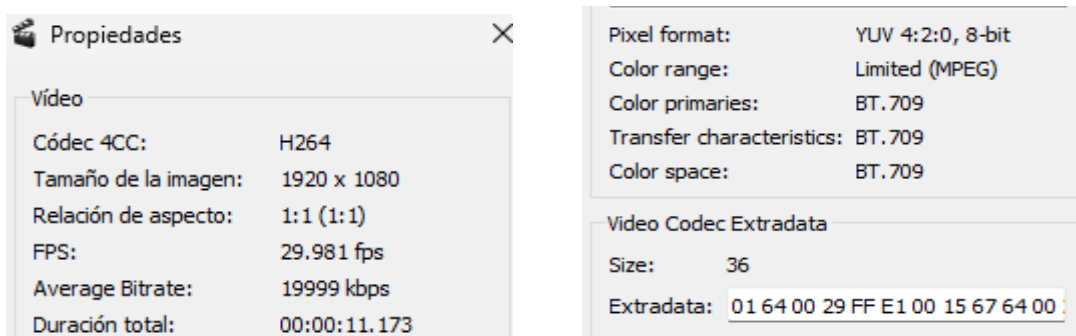


Figura 7.- Codificación asset-02

Píxeles es de 1920 x 1080 (HD), FPS: 29.981 fps (cuadros por segundo), duración: 11.173 s

Peso sin comprimir: $\text{ancho} \times \text{alto} \times \text{n}^\circ \text{ bits} = 1920 \times 1080 \times 8 \text{ bits/canal} \times 3 \text{ canales} \times 29,981 \text{ fps} \times 11,173 \text{ s}$ (bits)

ancho	alto	bits/canal	nº canales	FPS	Duración (s)	Total Bits	Bits/Byte	Peso(Bytes)
1.920	1.080	8	3	29,981	11,173	16.670.634.856	8	2.083.829.357

bit (b):	16670634856
byte (B):	2083829357
kilobyte (kB):	2034989,60645
megabyte (MB):	1987,29454
gigabyte (GB):	1.94072

Es decir, 1,94 Gigabytes (2.083.829.357 bytes)

Figura 8.- calculadora bytes asset-02

Pregunta 2.2: Contrasta esta información con la dada por MediaInfo.

General	Nombre completo: C:\Users\jegaa\Desktop\UOC_INFORMATICA\SIGNATURAS\FEB25\PLATAFORMAS\PEC2\asset-02.mp4 Formato: MPEG-4 Formato del perfil: Base Media / Version 2 ID códec: mp42 (isom/mp42) Tamaño de archivo: 28,2 MiB Duración: 11 s 202 ms Tasa de bits general: 21,1 Mb/s Velocidad de fotogramas: 29,970 FPS Fecha de codificación: 2025-04-18 09:30:19 UTC Fecha de etiquetado: 2025-04-18 09:30:19 UTC Writing operating system: Google Android 13 Writing hardware: Xiaomi 2201117SY
Vídeo	ID: 1 Formato: AVC Formato/Info: Advanced Video Codec Formato del perfil: High@L4.1 Ajustes del formato: CABAC / 1 Ref Frames Ajustes del formato, CABAC: Sí Ajustes del formato, RefFrames: 1 fotograma Ajustes del formato, GOP: M=1, N=30 ID códec: avc1 ID códec/Info: Advanced Video Coding Duración: 11 s 202 ms Duración original: 11 s 174 ms Tasa de bits: 20,0 Mb/s Ancho: 1 920 píxeles Alto: 1 080 píxeles Relación de aspecto: 16:9 Modo velocidad fotogramas: Variable Velocidad de fotogramas: 29,970 (29970/1000) FPS Velocidad fotogramas mínima: 29,060 FPS Velocidad fotogramas máxima: 30,303 FPS Espacio de color: YUV Submuestreo croma: 4:2:0 Profundidad bits: 8 bits Tipo barrido: Progresivo Bits/(píxel*fotograma): 0,323 Tamaño de pista: 26,7 MiB (95%) Cantidad de pistas original: 26,7 MiB (95%) Título: VideoHandle Idioma: Inglés Fecha de codificación: 2025-04-18 09:30:19 UTC Fecha de etiquetado: 2025-04-18 09:30:19 UTC Rango de color: Limited Colores primarios: BT.709 Características transferencia: BT.709 Coeficientes matriz: BT.709 mdhd_Duration: 11202 Codec configuration box: avcC

Figura 9.- Asset-02, información de MediaInfo

La duración No es la misma. 11,173s frente a 11,202s. Tampoco coincide FPS (29,981 vs 29,970). Tenemos ajustes de formato GOP: M=1 N=30 (avidemus no lo ha proporcionado). El espacio de color y el submuestreo coinciden: YUV 4:2:0

Pregunta 2.3: Teniendo como referencia el fichero de vídeo ficticio sin compresión, ¿cuál es, entonces, el factor de compresión que ya posee asset-02?.

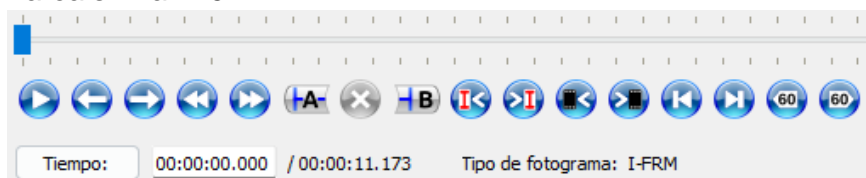
Factor de compresión= Peso sin comprimir / Peso imagen comprimida

Peso imagen comprimida= 27.276 Kbytes

kbps	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
19.999	19.999.000	11,173	223.448.827	27.931.103	27.276
2.083.829.357					
27.931.103					
Factor					75

Pregunta 2.4: Repite las tareas 1.5, 1.6 y 1.7 del ejercicio 1 aplicadas a este segundo clip de vídeo.

Tarea similar 1.5



El primer fotograma es tipo: I



El segundo fotograma es : P

Figura 10.- Tipos de Fotograma Asset-02

Tarea similar 1.6

10 fotogramas tipo I, instantes 0, 3, 4,007s 5,007s, 6,007, 7,007, 8,007, 9,006, 10,006, 11,007s

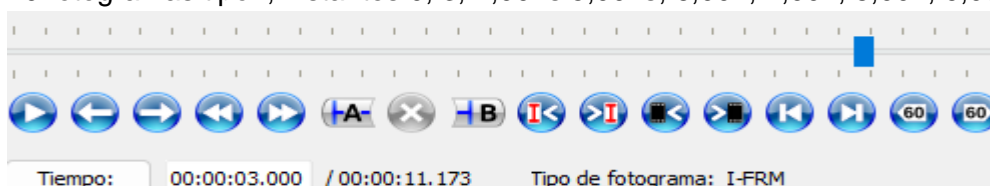


Figura 11.- Recuento de cuadros tipo I

Un fotograma I (Intra-frame) es una imagen completa, como un archivo de imagen JPG.

Tarea similar 1.7

La estructura GOP del H264 se basa en tramas I tramas P, y tramas B (Wang, 2024).

En este caso particular, tenemos la primera trama I, seguida de 89 tramas P, y luego una nueva trama I, seguida de 29 tramas P. Esta estructura I + 29P se mantiene con el movimiento. Es decir cada segundo de vídeo está formado por una trama I y 29 P.

No hay tramas de tipo B en este video.

Es decir tenemos una imagen I, que muestra la primera imagen y aporta la correlación espacial, seguida de tramas predictivas P, con la correlación temporal.

Pregunta 2.5: Comenta diferencias encontradas entre asset-01 y asset-02, y explicación si tienen.

En Asset-01 sólo teníamos 2 tramas I, en 0s y en 5,266s.

En Asset-02 tenemos 10 tramas I, 0, 3, 4,007s 5,007s, 6,007, 7,007, 8,007, 9,006, 10,006, 11,007s

Asset-02, los 2 primeros segundos muestro el nombre y voy despacio, hay 89 tramas P, sin embargo al correr (suave), las tramas I se intercalan cada 1s. Esto es porque la imagen filmada va cambiando y el contenido espacial de la imagen lo aportan las tramas I. Por eso el codificador introduce más I que en Asset-01 que era un video estático y no había variación espacial.

El factor de compresión, no es muy diferente, siendo 76 para asset-01 y 75 para asset-02.

Ejercicio 3: codificación en baja calidad

Pregunta 3.1: ¿Qué formato de codificación de vídeo aceptan los DVD? Si el cliente hubiera pedido Blu-Ray en vez de DVD, ¿cuál habría sido el formato de codificación más adecuada?

DVD usa el códec MPEG-2 (H.262/MPEG-2 Parte 2), la velocidad de datos para películas en DVD varía de 3 a 9,5 Mbit/s, y la velocidad de bits suele ser adaptativa. profundidad de bits de 8 bits por color, codificado como YCbCr con submuestreo de croma 4:2:0

Para Blu Ray pueden ser MPEG-2, pero lo adecuado es H.264. (Marriott, 2023)

Pregunta 3.2: ¿cuál de los dos filtros crees que sería más adecuado si tu clip fuese de 30 fps?

Debería ser “remuestrea los FPS”, así no alteramos la velocidad del vídeo, ni su duración.

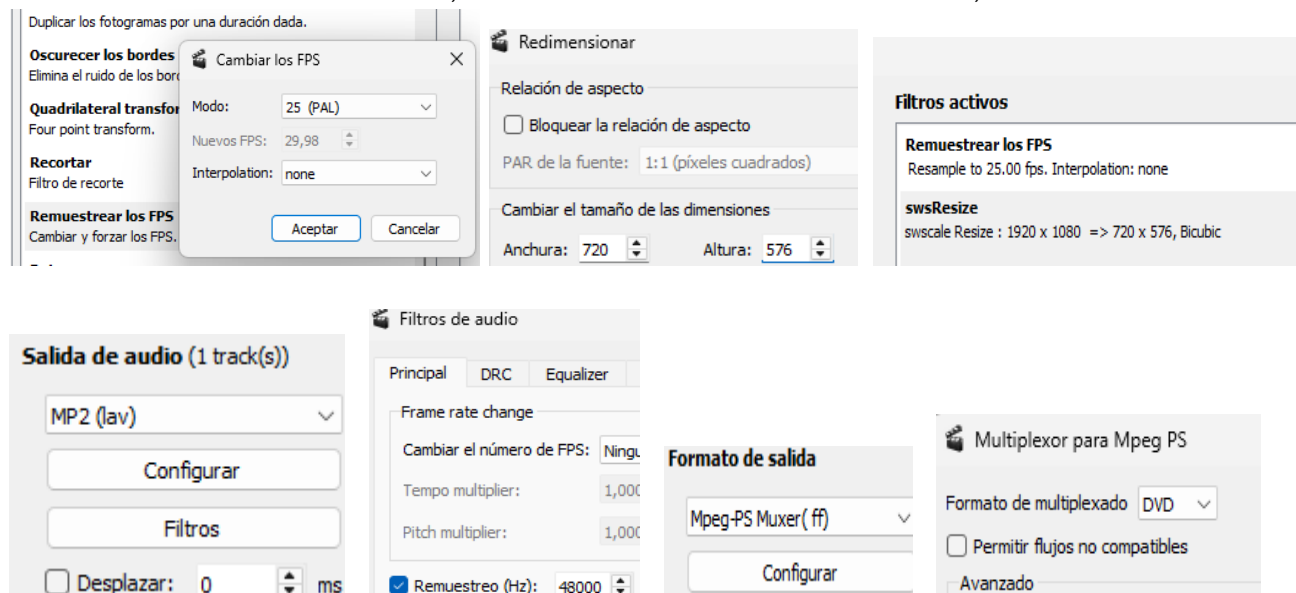


Figura 12.- Remuestreo DVD PAL

Pregunta 3.3: Cuál es el factor de compresión conseguido con el nuevo fichero pec2_dvd respecto al clip ficticio no comprimido de la misma duración y definición calculado en el apartado 2.1? Compáralo al resultante en el apartado 2.3

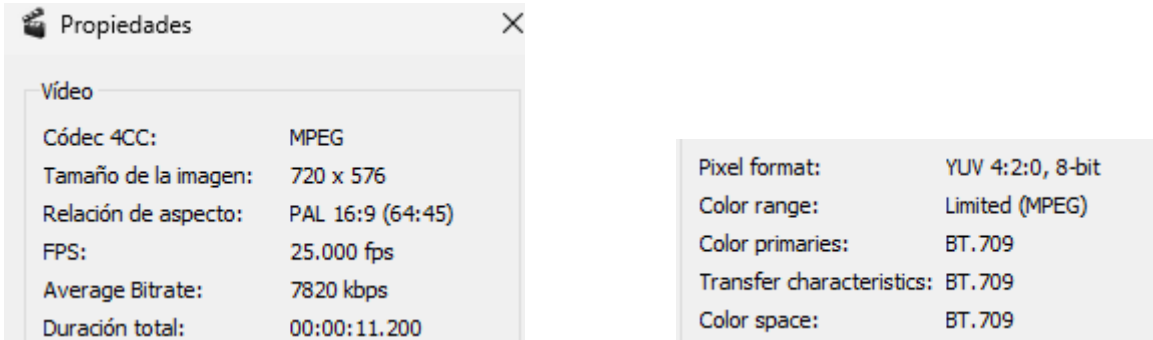


Figura 13.- Codificación pec2_dvd

La definición en píxeles es de 720x 576, FPS: 25 fps (cuadros por segundo), Duración: 11,2 s
Peso sin comprimir: ancho x alto x nº bits = 720 x 576 x 8bits/canal x 3 canales x 25fps x 11,2 s (bits)

ancho	alto	bits/canal	nº canales	FPS	Duración (s)	Total Bits	Bits/Byte	Peso(Bytes)
720	576	8	3	25	11,200	2.786.918.400	8	348.364.800

Peso Real basado en la tasa de bits

kpbs	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
7.820	7.820.000	11,200	87.584.000	10.948.000	10.691
348.364.800					
10.948.000					
Factor		32			

En el apartado 2.3 el factor de compresión era de 75, más del doble que para DVD.
Con VLS se reproduce correctamente

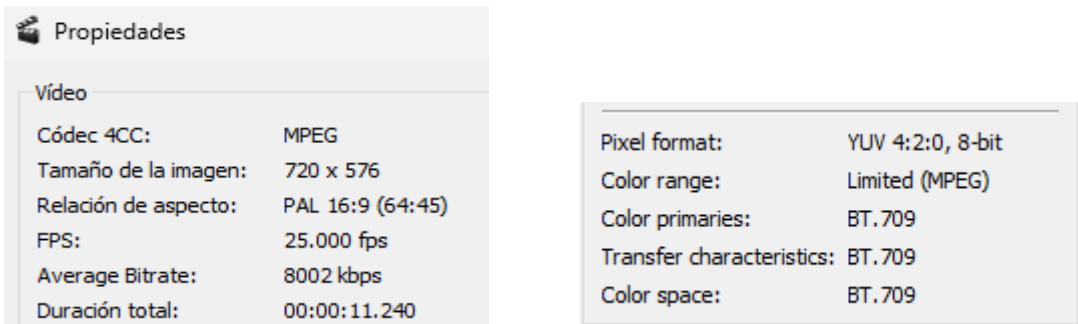


Figura 14.- Fichero pec2_dvd reproducido por VLC

Pregunta 3.4: ¿Qué valores tenemos que poner en los campos Número de fotogramas B y Tamaño del GOP para obtener este formato de GOP?

Para obtener M=2 y N= 8, tendremos que insertar 1 trama B entre cada 2 tramas P o entre I y P. Para obtener N=8, el tamaño del GOP es 8, porque N es GOP size, la distancia entre 2 tramas I.

Pregunta 3.5: ¿Qué factor de compresión obtenemos? ¿Por qué?



Propiedades

Vídeo

Códec 4CC:	MPEG	Pixel format:	YUV 4:2:0, 8-bit
Tamaño de la imagen:	720 x 576	Color range:	Limited (MPEG)
Relación de aspecto:	PAL 16:9 (64:45)	Color primaries:	BT.709
FPS:	25.000 fps	Transfer characteristics:	BT.709
Average Bitrate:	8002 kbps	Color space:	BT.709
Duración total:	00:00:11.240		

Figura 15.- Codificación pec2_dvd_m2_n8

kbps	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
8.000	8.000.000	11,200	89.600.000	11.200.000	10.938
348.364.800					
11.200.000					
Factor	31				

Ha empeorado porque ha aumentado el average Bitrate, y por tanto el peso del nuevo archivo. Ojo la cuenta la hago respecto al peso sin pérdidas pero con FPS=25 (no contra el de 30bps).

Pregunta 3.6: ¿Cómo afecta a los vídeos resultantes el cambio de los parámetros del GOP? ¿Qué cambia entre los vídeos? ¿Por qué?

M=3, N=9, GOP : I B B P B B P B B



Vídeo

Códec 4CC:	MPEG
Tamaño de la imagen:	720 x 576
Relación de aspecto:	PAL 16:9 (64:45)
FPS:	25.000 fps
Average Bitrate:	7949 kbps
Duración total:	00:00:11.240

Figura 16.- pec2_dvd_m3_n9

kbps	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
7.949	7.949.000	11,200	89.028.800	11.128.600	10.868
348.364.800					
11.128.600					
Factor	31				

Aumentamos la distancia entre tramas I, y aumentamos el número de B (2 entre P), esto implica peor calidad, pero nos da una pequeña mejora (muy ligera) en compresión

M=4, N=20, GOP : I B B B P B B B P B B B P B B B P B B B

Vídeo	
Códec 4CC:	MPEG
Tamaño de la imagen:	720 x 576
Relación de aspecto:	PAL 16:9 (64:45)
FPS:	25.000 fps
Average Bitrate:	7844 kbps
Duración total:	00:00:11.200

Figura 17.- pec2_dvd_m4_n20

kbps	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
7.844	7.844.000	11,200	87.852.800	10.981.600	10.724
348.364.800					
10.981.600					
Factor		32			

Aumentamos la distancia entre tramas I, y aumentamos el número de B (3 entre P), esto implica peor calidad, pero nos da una pequeña mejora en compresión

M=1, N=4, GOP : I P P P

Vídeo	
Códec 4CC:	MPEG
Tamaño de la imagen:	720 x 576
Relación de aspecto:	PAL 16:9 (64:45)
FPS:	25.000 fps
Average Bitrate:	8368 kbps
Duración total:	00:00:11.160

Figura 18.- pec2_dvd_m1_n4

kbps	bps	Duración(s)	Total Bits	Peso Bytes	Peso Kbytes
8.368	8.368.000	11,200	93.721.600	11.715.200	11.441
348.364.800					
11.715.200					
Factor		30			

Eliminamos las B, acercamos las I, significa mayor calidad, pero empeoramos el factor de compresión (30).

Ejercicio 4: codificación Blu Ray y codificación 4K

Pregunta 4.1: ¿Qué significa HEVC? ¿Qué novedades incorpora H.265/HEVC respecto al anterior H.264/MPEG-4 AVC? Consulta Internet para encontrar estas informaciones.

High Efficiency Video Coding, en castellano, Codificación de Vídeo de Alta Eficiencia (Flussonic, 2024).

H265 incorpora más resolución, hasta 8K (8192 x 4320) frente a 2048 x 2048 máxima en H264. H265 tiene mayor eficiencia de compresión, mejora de conficabilidad y mayor capacidad para recuperarse de errores, menor latencia en tiempo real, y mejoras en el tiempo de adquisición de canal reducido, así como latencia de acceso aleatorio.

En codificación, en H264 el macrobloque es de tamaño fijo 16x16 píxeles, mientras en H265 va desde 4x4 hasta 64x64.

Para la misma calidad de imagen, H265 reduce el tamaño del video hasta un 39% más que H264

Resolution	H.264	H.265
480p	25 Mbps	1.6 Mbps
720p	5 Mbps	3.3 Mbps
1080p	8 Mbps	5.3 Mbps
UHD/4K	35-42 Mbps	23-28 Mbps

Figura 19.- Comparativa H264 / H265 (Accsoon, 2024)

Pregunta 4.2: ¿Cuáles son los niveles y los perfiles de H.265/HEVC? Compáralos con los de H.264/MPEG-4 AVC.

En H.264 hay 16 niveles: (ITU-T H.264, 2007)

1 1b 1.1 1.2 1.3 2 2.1 2.2 3 3.1 3.2 4 4.1 4.2 5 5.1

Y 9 perfiles:

Main, High, High 10, High 4:2:2, High 4:4:4 Predictive, High 10 Intra, High 4:2:2 Intra, High 4:4:4 Intra, CAVLC 4:4:4

En H.265/HEVC hay 13 niveles (ITU-T H265, 2021)

1 2 2.1 3 3.1 4 4.1 5 5.1 5.2 6 6.1 6.2

Que se organizan en torno a 2 tiers: Main Tier (≤ 5.2) y High Tier (nivel ≥ 4)

Y 22 perfiles:

Monochrome, Monochrome 10, Monochrome 12, Monochrome 16, Main, Main 10, Main 12, Main 4:2:2 10, Main 4:2:2 12, Main 4:4:4, Main 4:4:4 10, Main 4:4:4 12, Main Intra, Main 10 Intra, Main 12 Intra, Main 4:2:2 10 Intra, Main 4:2:2 12 Intra, Main 4:4:4 Intra, Main 4:4:4 10 Intra, Main 4:4:4 12 Intra, Main 4:4:4 16 Intra, High Throughput 4:4:4 16 Intra profiles

Pregunta 4.3: ¿Dónde se indica el nivel de codificación en esta ventana?

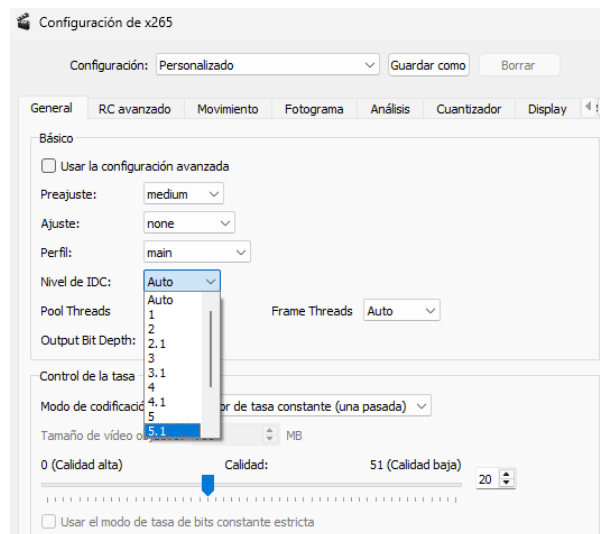
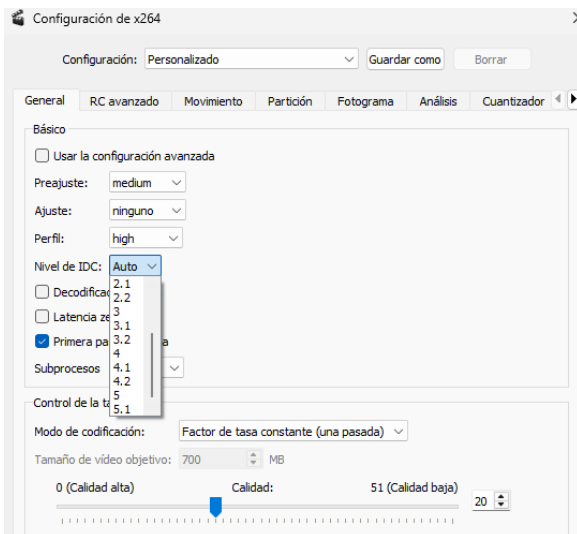


Figura 20.- Nivel IDC o nivel de decodificación

En el Nivel de IDC

Pregunta 4.4: ¿Qué perfiles pueden ser elegidos en Avidemux para H.264 y en H.265?

H.264: Baseline, main, high, high10, high422, high444

H.265: main, mainstillpicture

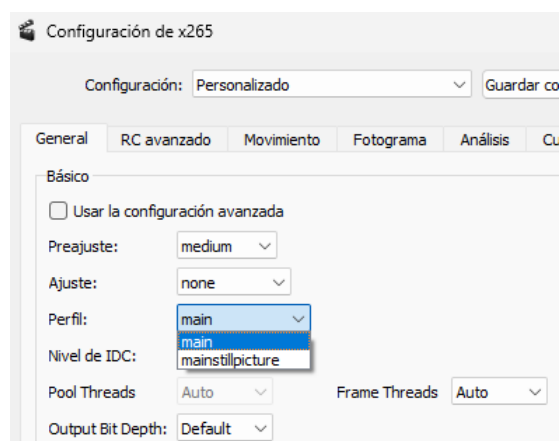
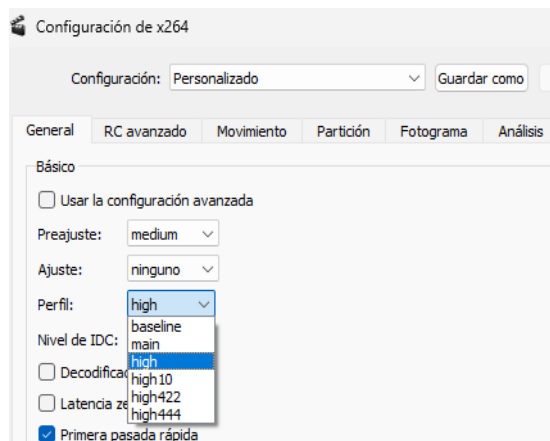


Figura 21.- perfiles en Avidemux para H.264 y en H.265

Pregunta 4.5: Calcula el factor de compresión conseguido respecto el fichero asset-02.

El peso del fichero es: $29612032 \text{ bytes} = 29612032 \text{ byte} \cdot 8 \text{ bits/byte} = 236.896.256 \text{ bits}$

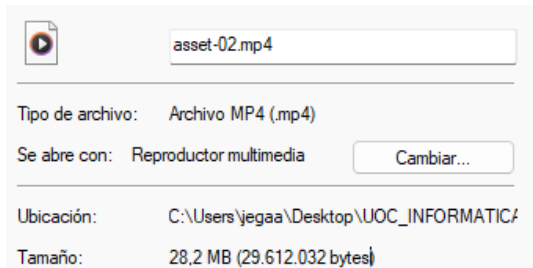


Figura 22.- Peso del asset-02

El bit rate = $\text{Peso} / \text{duración} = 236896256 / 11,173 \text{ s} = 21202564 \text{ bps} = 21202 \text{ Kbps}$

Tasa de bits objetivo = $21202 \text{ Kbps} \cdot 70\% = 14.842 \text{ Kbps}$

El nuevo fichero hevc-70.mkv tiene un peso de: $204210376 \text{ bits} / 11,173 \text{ s} = 18277 \text{ Kbps}$

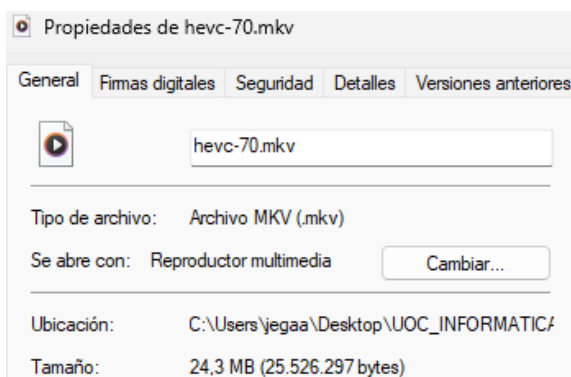


Figura 23.- Peso del hevc-70

Factor de compresión: $\text{Peso asset-02} / \text{peso hevc-70} = 29612032 \text{ bytes} / 25526297 \text{ bytes} = 1,16$

Pregunta 4.6: Calcula también el factor de compresión respecto a un hipotético clip de la misma duración y definición (sin comprimir, evidentemente).

El fichero sin comprimir, misma definición que el asset-02 (1920 x 1080 x 30fps x 3canales x 8bit)

Factor de compresión: 82

Asset-02 sin comprimir	2.083.829.357
hevc-70	25.526.297
Factor	82

Pregunta 4.7: Calcula otra vez el factor de compresión en ambos casos. ¿Ha habido una variación importante de compresión entre los dos clips generados? ¿Y de calidad visual? ¿Por qué crees que ha sido así?

El peso asset-02 sigue siendo: $29612032 \text{ bytes} = 29612032 \text{ byte} \cdot 8 \text{ bits/byte} = 236.896.256 \text{ bits}$

El bit rate = $\text{Peso}/\text{duración} = 236896256/11,173 \text{ s} = 21202564 \text{ bps} = 21202 \text{ Kbps}$

Tasa de bits objetivo = $21202 \text{ Kbps} \cdot 50\% = 10.601 \text{ Kbps}$

El nuevo fichero hevc-40.mkv tiene un bit rate de: $117782288 \text{ bits}/11,173 \text{ s} = 10541 \text{ Kbps}$

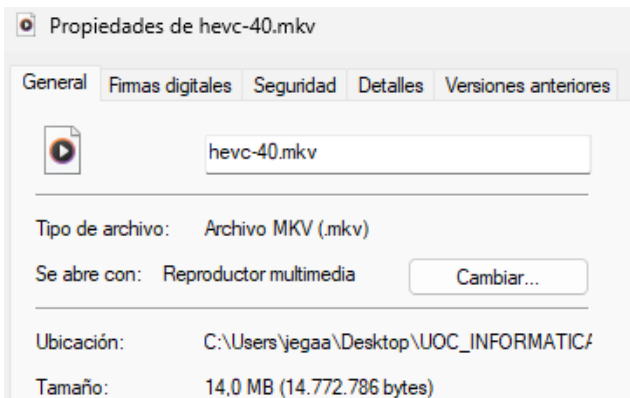


Figura 24.- Peso del hevc-40

Factor de compresión: $\text{Peso asset-02} / \text{peso hevc-40} = 29612032 \text{ bytes} / 14722786 \text{ bytes} = \mathbf{2,011}$

El fichero sin comprimir, misma definición que el asset-02 (1920 x 1080 x 30fps x 3canales x 8bit)

Factor de compresión: 142

Asset-02 sin comprimir	2.083.829.357
hevc-40	14.722.786
Factor	142

Sí, ha habido una variación muy importante en la reducción del tamaño, por tanto en la compresión, pasando del 1,16 al 2,0 teniendo en cuenta pesos de ficheros reales, mientras que si comparamos contra un escenario sin comprimir, se pasa de 82 a 142 ese factor.

No, no he notado pérdida de calidad, al menos significativa.

Creo que ha sido así por una suma de motivos:

Primero porque hemos reducido la tasa de bits objetivo, en el primer caso estaba al 70% en el segundo caso al 50%.

En segundo lugar hemos usado para el caso hevc-70 tasa de bits constante y una sola pasada, mientras que para el segundo hevc-40, hemos dejado tasa variable y dos pasadas.

El análisis de 2 pasadas, usa la primera para analizar la complejidad de cada frame, y aprovecha esa información en la segunda para asignar una tasa de bits óptima según la complejidad de la escena.

La tasa de bits variable ajusta dinámicamente la cantidad de bits para codificar cada frame de video. (Krings, 2025)

La suma de ambas técnicas, permite ahorrar bits en las zonas simples, (mejorar la comprensión) y aumentar recursos en las partes con más detalle y movimiento, para no perder calidad.

La contrapartida, a la hora de codificar el fichero, es que el tiempo ha aumentado bastante.

Pregunta 4.8: Qué efecto visual produce la pérdida de libertad de elección de su medida? ¿Y a la compresión del fichero? ¿Por qué?

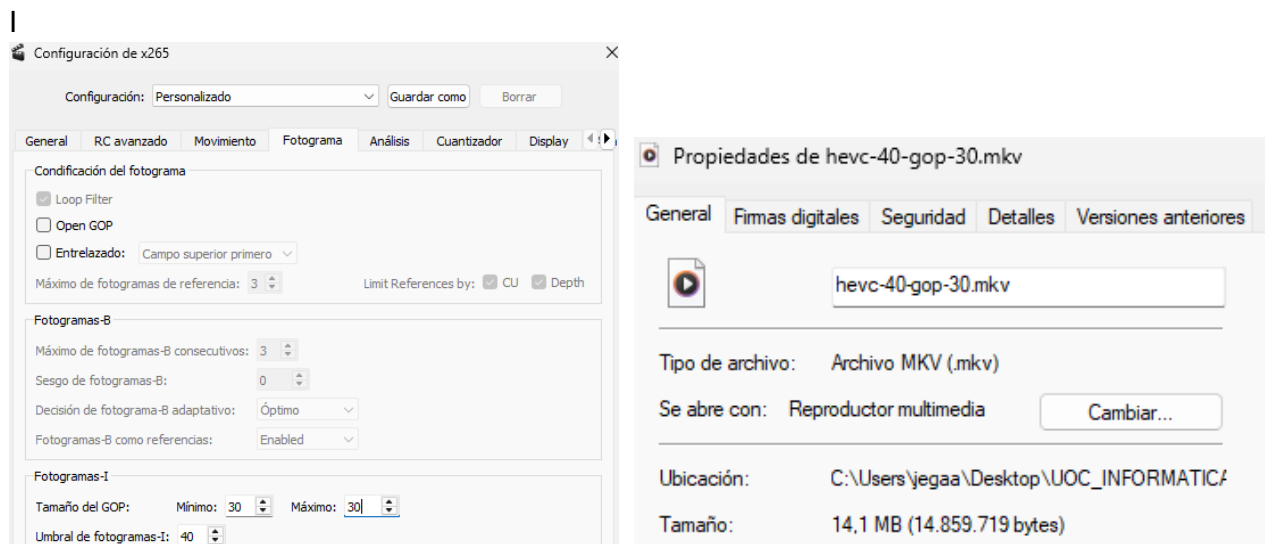


Figura 25.- Peso del hevc-40-gop-30

Visualmente, en los 11s que dura el video, no puedo valorar que ha cambiado, no noto diferencia. En cuanto al tamaño, ha aumentado ligeramente, pasando a 14.859.719 bytes desde los 14.722.786 bytes de la versión anterior.

Pregunta 4.9: Comparando los tres casos, ¿cómo afecta a la calidad visual la variación del GOP? ¿Y a la compresión del fichero? ¿Por qué?

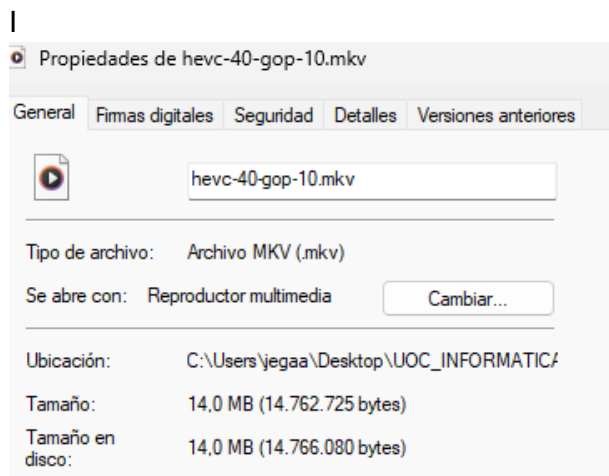


Figura 26.- Peso del hevc-40-gop-10

Ahora el fichero está algo más comprimido, pero tampoco es significativo, y realmente no he visto en esos 11s, una diferencia visual apreciable.

“hevc-40” estructura de GOP

IBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBPBBBBP
BPBBPBPPPPBPP

|

“hevc-40-gop-30” estructura GOP

IBBBPBBBBPBBBBPBBBBPBBBBPBBBBP (grupo de 30 correcto)

“hevc-40-gop-10” estructura GOP

IBBBPBBBBP (grupo de 10 correcto)

En este caso el fichero pesa menos los 2 anteriores, muy ligeramente, pero no noto diferencias visuales entre los 3 ficheros.

Ejercicio 5: codificación de vídeo mediante un CDN

1.- CloudFront (CloudFront, 2024), es un servicio en la nube, de contenido web **estático y dinámico**, que acerca los contenidos al usuario, usando una red global de AWS que incluye centros de datos en el borde. Al ser solicitado contenido distribuido por CloudFront, la petición se envía a un centro de datos de borde, aquel que en ese momento y para ese usuario es capaz de ofrecer mínima latencia y mayor rendimiento.

Si ese centro de datos tiene contenido, lo entrega directamente, en caso contrario lo recupera de la fuente origen (servidor web o bucket S3 por ejemplo) y lo envía al usuario. Estas operaciones son transparentes al usuario, que lo que logra es una mejor calidad de servicio y mejor experiencia de usuario.

Como la información puede estar replicada en las cachés de varios centros de datos de borde, se tiene mayor confiabilidad y mejor disponibilidad.

2.- Si además de html e imágenes, hay **contenido en vídeo**, ¿cómo lo entregará? (Qué es , 2025)
El reproductor de vídeo solicita el vídeo a través de la URL de la web, petición reenviada por CloudFront a la mejor ubicación de borde, que suministrará el vídeo original a partir de su caché o lo buscará en el bucket de S3 si aún no se encuentra en la caché.

Pero, ¿ese vídeo original se adecua al reproductor del usuario?

Posiblemente no, y malgastará el ancho de banda de bajada, pudiendo saturar las redes (hay miles o millones de reproducciones de vídeos en la red).

Elemental MediaConvert es una solución que automáticamente transcodifica el vídeo fuente, creando versiones comprimidas de tamaño reducido, en diferentes formatos compatibles con distintos estándares y reproductores.

Estos se almacenarán y estarán disponibles en la red de entrega de contenido (CDN).

Y cuando el usuario lo solicite, se entregará como explicado en el punto 1.

¿Esto es lo más eficiente para vídeo?

No, porque de esa manera el vídeo se descargará siempre por completo, aunque el usuario decida dejar de verlo, y lo peor es que se pagará por todo (aunque luego no se reproduzca).

La solución es transmitirlo por **streaming** (Streaming, 2023)

hay que distinguir 2 situaciones, dependiendo de la fuente de vídeo:

Vídeo almacenado y streaming en directo

3.- Streaming de Vídeo almacenado, que puede haber sido ya tratado

HTTP Live Streaming de Apple (HLS), es capaz de transmitir los vídeos en segmentos cortos, que irán siendo servidos a medida que avanza la reproducción (unos segundos antes de necesitarse) y de este modo garantizar que no haya cortes, que el proceso sea transparente al usuario. Para tener una estructura documentada de esos trocitos de vídeos que componen el todo, se emplea un manifiesto. Desde CloudFront ya no se apunta al archivo original de vídeo, sino al manifiesto, y esa URL se le pasa al reproductor.

Existen protocolos alternativos Dynamic Adaptive Streaming over HTTP (DASH), Smooth Streaming de Microsoft (MSS) y HTTP Dynamic Streaming de Adobe (HDS), todos al igual que HLS mejoran la experiencia del usuario porque la reproducción es inmediata permiten usar los controles de avance, retceso tradicionales y la experiencia de usuario mejora.

Y lo mejor es que no se descarga todo el contenido, sólo se factura lo descargado, no se satura innecesariamente la red.

Para convertir a HLS y alternativos se usa MediaConvert, y un ejemplo de URL de manifiesto para reproducir: <http://videos.cloudfront.net/mivideo1/lista.m3u8>

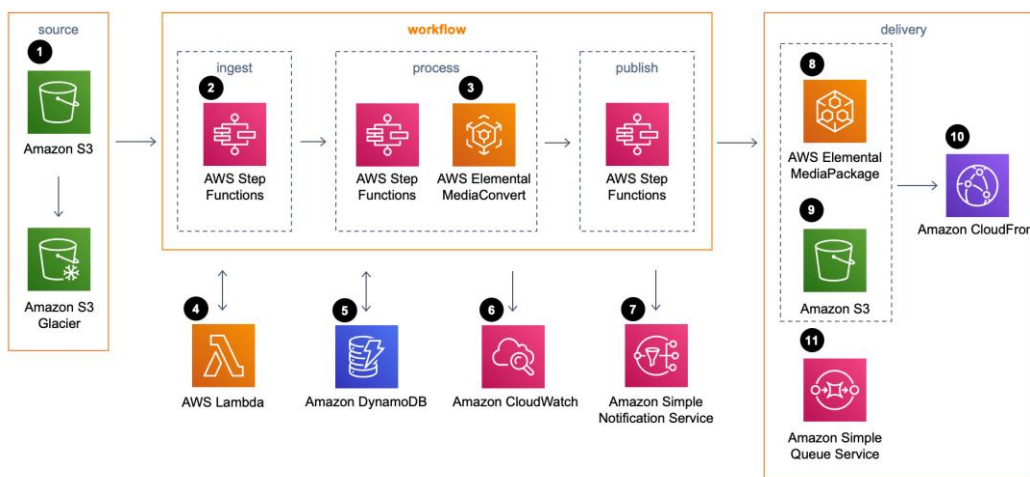


Figura 27.- Video (bajo demanda, 2022)

4.- Streaming en directo

¿Cómo podemos transmitir en directo un telediario o un partido de hockey de las cadetes del Zaratán?

Lo primero es procesar el video que procede de la fuente en directo, con **AWS Elemental MediaLive**, para comprimirlo en versiones más pequeñas. Si además queremos “adaptarlo” a diferentes formatos usaremos Elemental Media Package, que además permite control de reproducción y administración de derechos digitales. Si la fuente ya teiene adaptado el contenido, se usará **Elemental MediaStore** como origen de alto rendimiento y escalable. Tanto mediaStore como MediaPackage, entregan a CloudFront el contenido, para que se direcciona a la audiencia, de manera escalable y adaptable.

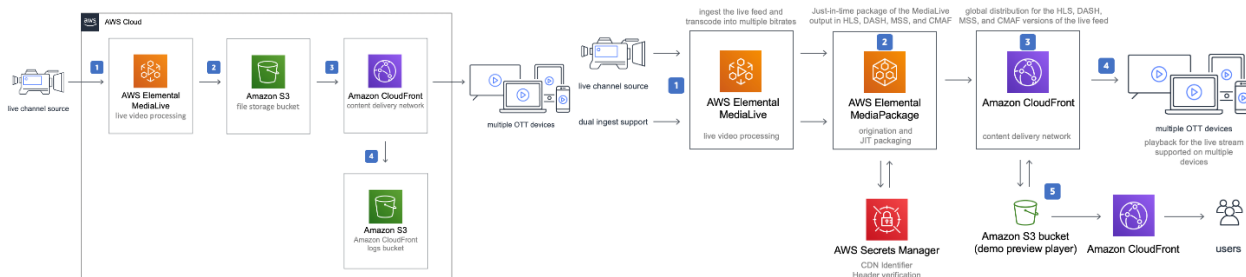


Figura 28.- LiveStreaming S3 vs Live Streaming (Live Streaming , 2020)

5.- Precios

En el caso de emplear como origen de datos Amazon S3, la transferencia S3->CloudFront es gratuita. Sin embargo sí que se factura cada dato saliente desde CloudFront al usuario peticionario, por ejemplo vía HTTP, HTTPS.

En el nivel gratuito se incluye:

- Transferencia de 1 TB de datos a Internet por mes
- 10 000 000 de solicitudes HTTP o HTTPS por mes
- 2 000 000 de invocaciones de CloudFront Function por mes
- 2 000 000 de lecturas de KeyValueStore de CloudFront al mes
- Certificados SSL gratuitos
- Sin limitaciones, todas las herramientas disponibles

Por encima de la capa gratuita, los precios mensuales en Europa son los siguientes:

Transferencia saliente de datos regionales a Internet (por GB)	Europa, Israel y Turquía
Primer 1 TB	Gratis
Próximos 9 TB	0,085 USD
Siguientes 40 TB	0,080 USD
Siguientes 100 TB	0,060 USD
Siguientes 350 TB	0,040 USD
Siguientes 524 TB	0,030 USD
Siguientes 4 PB	0,025 USD
Más de 5 PB	0,020 USD

Transferencia saliente de datos regional al origen (por GB)	Europa, Israel y Turquía
Transferencia de todos los datos	0,020 USD

Es decir, en torno a los 80 dólares por cada TeraByte saliente y 20 por los entrantes.

Todo es negociable, para clientes preferentes ofrecen soluciones personalizables en función de consumos, tipos de laencia según zonas, paquetes ahorro...

En cualquier caso, Amazon proporciona para todos sus servicios una calculadora el línea:

<https://calculator.aws/#/addService>

En resumen, AWS CloudFront, permite transmitir streaming con baja latencia, mejorar la fiabilidad y la seguridad, transmitir video bajo demanda y streaming en directo y en diferido, incluso distribuir parches y actualizaciones, con precios personalizados.

Referencias

- Accsoon ;. (25 de 6 de 2024). *H.264 Vs H.265 – Which Should You Use?* Obtenido de <https://accsoon.com/explore/h264-vs-h265-which-should-you-use/>
- Avidemux , Forum;. (19 de 11 de 2021). Obtenido de <https://avidemux.org/smif/index.php/topic,19710.msg92887.html#msg92887>
- bajo demanda. (2022). *Video bajo demanda en AWS*. Obtenido de <https://aws.amazon.com/es/solutions/implementations/video-on-demand-on-aws/>
- CloudFront. (2024). *Amazon CloudFront*. Obtenido de https://aws.amazon.com/es/cloudfront/?nc2=h_ql_sol_use_cd
- Flussonic;. (22 de 1 de 2024). *H.264 y H.265 - AVC y HEVC - ¿Cuál es la diferencia?* Obtenido de <https://flussonic.com/es/blog/news/h264-vs-h265/>
- Graham, M., & Summersby, A. (1 de Abril de 2025). *Conocer los ajustes y formatos de vídeo*. Obtenido de <https://www.canon.es/pro/infobank/video-formats/>
- ITU-T H.264;. (2007). *ITU-T Rec. H.264 (11/2007) Advanced video coding for generic audiovisual services*. Obtenido de <https://eclass.uoa.gr/modules/document/file.php/D221/%CE%A3%CE%B7%CE%BC%CE%B5%CE%B9%CF%8E%CF%83%CE%B5%CE%B9%CF%82/H.264.pdf>
- ITU-T H265;. (2021). *Recommendation ITU-T H.265*. Obtenido de https://e2e.ti.com/cfs-file/__key/communityserver-discussions-components-files/791/T_2D00_REC_2D00_H.265_2D00_202108_2D00_S_21002100_PDF_2D00_E.pdf
- Krings, E. (27 de 3 de 2025). *CBR vs. VBR: la diferencia entre velocidad de bits constante y velocidad de bits variable*. Obtenido de <https://www.dacast.com/es/blog-es/cbr-vs-vbr/>
- Live Streaming . (2020). *Live Streaming en AWS*. Obtenido de <https://aws.amazon.com/es/solutions/implementations/live-streaming-on-aws/>
- Marriott , A. (17 de 1 de 2023). *Cuáles son los formatos de vídeo y sus aplicaciones*. Obtenido de <https://massive.io/es/transferencia-de-archivos/formatos-de-video/>
- Qué es . (2025). *Qué es Amazon CloudFront*. Obtenido de https://docs.aws.amazon.com/es_es/AmazonCloudFront/latest/DeveloperGuide/Introduction.html
- Streaming. (2023). *Tutoriales sobre streaming de contenido multimedia de Amazon CloudFront*. Obtenido de <https://aws.amazon.com/es/cloudfront/streaming/>
- Wang, F. (2024). *Introducción al estándar de compresión de video H.264*. Obtenido de <https://www.engeniustech.com/span/introduction-to-h-264-video-compression-standard/>

Tabla de Figuras

Figura 1.- Codificación asset-01	2
Figura 2.- calculadora bytes asset-01	2
Figura 3.- Contenedor, tipo y tabla	3
Figura 4.- Tipos de Fotograma Asset-01	3
Figura 5.- Recuento de cuadros tipo I	4
Figura 6.- MedainInfo vs Avidemux	5
Figura 7.- Codificación asset-02	5
Figura 8.- calculadora bytes asset-02	6
Figura 9.- Asset-02, información de MedianInfo	6
Figura 10.- Tipos de Fotograma Asset-02	7
Figura 11.- Recuento de cuadros tipo I	7
Figura 12.- Remuestreo DVD PAL	8
Figura 13.- Codificación pec2_dvd	9
Figura 14.- Fichero pec2_dvd reproducido por VLC	9
Figura 15.- Codificación pec2_dvd_m2_n8	10
Figura 16.- pec2_dvd_m3_n9	10
Figura 17.- pec2_dvd_m4_n20	11
Figura 18.- pec2_dvd_m1_n4	11
Figura 19.- Comparativa H264 / H265 (Accsoon ;, 2024)	12
Figura 20.- Nivel IDC o nivel de decodificación	13
Figura 21.- perfiles en Avidemux para H.264 y en H.265	13
Figura 22.- Peso del asset-02	14
Figura 23.- Peso del hevc-70	14
Figura 24.- Peso del hevc-40	15
Figura 25.- Peso del hevc-40-gop-30	16
Figura 26.- Peso del hevc-40-gop-10	17
Figura 27.- Video (bajo demanda, 2022)	19
Figura 28.- LiveStreaming S3 vs Live Streaming (Live Streaming , 2020)	19