



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Implementación y evaluación de MPEG- DASH bajo las características de redes móviles.

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Miguel Guzmán Marín

Tutor: José Salvador Oliver Gil

Curso Académico 2019/2020

Implementación y evaluación de MPEG-DASH bajo las características de redes móviles.

Resumen

En los últimos años la forma de consumir recursos multimedia está cambiando considerablemente, hemos pasado de estar enganchados a la televisión a ser unos adictos de internet y nuestro teléfono móvil. Y es que las posibilidades que nos ofrecen los smartphones y el resto de dispositivos inteligentes son muy superiores a lo que nos da la televisión tradicional.

Hoy en día no queremos estar esperando todo el día a que se emita el nuevo capítulo de nuestra serie favorita o esa película que llevamos tanto tiempo esperando. Queremos verlo ya, sin que se corte y en la mejor calidad posible. Además, también queremos poder ver contenido en vivo allá donde estemos como si nos encontrásemos en el sofá de nuestra casa. Ahí es donde entra el protocolo MPEG-DASH y su aplicación en el streaming de contenidos multimedia.

En este trabajo se ha hecho un recorrido por la evolución de las redes móviles en la historia reciente, analizado sus características y, posteriormente, se ha implementado una solución que nos permita evaluar el funcionamiento de MPEG-DASH y las prestaciones que nos puede ofrecer en función del ancho de banda disponible.

Palabras clave: Streaming, Redes móviles, DASH, Internet, MPEG

Abstract

In recent years the way to consume multimedia resources is changing considerably, we have gone from being hooked to television to be addicted to the internet and our mobile phone. And the possibilities offered by smartphones and other smart devices are far superior to what traditional television gives us.

Today we do not want to be waiting all day for the new chapter of our favorite series or that movie that we have been waiting for so long. We want to see it now, without being cut and in the best possible quality. In addition, we also want to be able to watch live content wherever we are as if we were on the sofa in our house. That's where the MPEG-DASH protocol and its application in streaming multimedia content comes in.

In this work we have made a journey through the evolution of mobile networks in recent history, analyzed their characteristics and, subsequently, a solution has been implemented that allows us to evaluate the operation of MPEG-DASH and the benefits it can offer us in Bandwidth function available.

Keywords : Streaming, Mobile Networks, DASH, Internet, MPEG



Implementación y evaluación de MPEG-DASH bajo las características de redes móviles.

Tabla de contenidos

1. Introducción	9
1.1 Motivación	9
1.2 Objetivos	9
1.3 Estructura de la memoria	10
2. Redes móviles	11
2.1 Internet móvil	11
2.2 Evolución histórica	11
2.3 Estándares y tecnologías	13
2.3.1 GPRS	13
2.3.2 EDGE	14
2.3.3 3G	14
2.3.4 HSPA	14
2.3.5 4G LTE	15
3. Streaming	17
3.1 Streaming	17
3.2 Evolución histórica	17
3.3 Streaming en vivo y bajo demanda	18
3.4 Protocolos	19
3.4.1 UDP (User Datagram Protocol)	19
3.4.2 TCP (Transmission Control Protocol)	19
3.4.3 HTTP (HyperText Transfer Protocol)	20
3.4.4 Evolución de HTTP	21
3.5 Diferentes streaming en HTTP	24
3.5.1 Descarga Progresiva	24
3.5.2 HTTP Pseudo Streaming	24



3.5.3 HAS (HTTP Adaptive Streaming)	24
3.6 MSS, HLS y HDS	26
3.6.1 MSS (Microsoft Smooth Streaming)	26
3.6.2 HLS (HTTP Live Streaming)	26
3.6.3 HDS (HTTP Dynamic Streaming)	27
3.7 MPEG-DASH	27
3.7.1 MPD (Media Presentation Description)	28
3.7.1.1 Period	30
3.7.1.2 Adaptation Set	31
3.7.1.3 Representation	32
3.7.1.4 Segment	33
3.7.1.5 Tipos de Segment	35
4. Implementación	37
4.1 Escenario	37
4.1.1 Servidor	37
4.1.2 Cliente	44
4.1.3 Router	45
5. Pruebas y evaluación	47
5.1 3G	47
5.1.1 3G segmentos de cuatro segundos	48
5.1.2 3G segmentos de ocho segundos	52
5.1.3 3G segmentos de dieciséis segundos	56
5.2 4G	60
5.2.1 4G segmentos de cuatro segundos	60
5.2.2 4G segmentos de ocho segundos	63
5.2.3 4G segmentos de dieciséis segundos	66
6. Conclusiones	71

Índice de figuras

Figura 1: Esquema de la evolución de las redes móviles	13
Figura 2: Establecimiento de conexión TCP	19
Figura 3: Esquema de métodos de petición HTTP	21
Figura 4: Esquema de la arquitectura de MPEG-DASH	28
Figura 5: Contenido archivo MPD	28
Figura 6: Ejemplo de elemento Segment Template	34
Figura 7: Esquema del escenario a implementar	37
Figura 8: Activar servicio IIS	38
Figura 9: Configuración IIS	39
Figura 10: Configuración FFmpeg	39
Figura 11: FFmpeg en terminal	40
Figura 12: Instalación framework GPAC	40
Figura 13: Resultado codificación con FFmpeg	42
Figura 14: Resultado codificación con mp4box	43
Figura 15: Uso de NetLimiter	45
Figura 16: Configuración NAT router	46
Figura 17: 3G cuatro segundos. Inicio reproducción DASH	48
Figura 18: 3G cuatro segundos. Estabilización de la reproducción DASH	49
Figura 19: 3G cuatro segundos. Reducción del límite de velocidad	50
Figura 20: 3G cuatro segundos. Vuelta al límite de velocidad inicial	51
Figura 21: 3G cuatro segundos. Gráfica calidad de imagen	51
Figura 22: 3G ocho segundos. Inicio reproducción DASH	52
Figura 23: 3G ocho segundos. Estabilización de la reproducción DASH	53
Figura 24: 3G ocho segundos. Reducción del límite de velocidad	54
Figura 25: 3G ocho segundos. Vuelta al límite de velocidad inicial	55
Figura 26: 3G ocho segundos. Gráfica calidad de imagen	55
Figura 27: 3G dieciséis segundos. Inicio reproducción DASH	56

Figura 28: 3G dieciséis segundos. Estabilización de la reproducción DASH	57
Figura 29: 3G dieciséis segundos. Reducción del límite de velocidad	58
Figura 30: 3G dieciséis segundos. Vuelta al límite de velocidad inicial	59
Figura 31: 3G dieciséis segundos. Gráfica calidad de imagen	59
Figura 32: 4G cuatro segundos. Inicio y estabilización de la reproducción DASH ...	60
Figura 33: 4G cuatro segundos. Reducción del límite de velocidad	61
Figura 34: 4G cuatro segundos. Vuelta al límite de velocidad inicial	62
Figura 35: 4G cuatro segundos. Gráfica calidad de imagen	62
Figura 36: 4G ocho segundos. Inicio y estabilización de la reproducción DASH	63
Figura 37: 4G ocho segundos. Reducción del límite de velocidad	64
Figura 38: 4G ocho segundos. Vuelta al límite de velocidad inicial	65
Figura 39: 4G ocho segundos. Gráfica calidad de imagen	65
Figura 40: 4G dieciséis segundos. Inicio y estabiliz. de la reproducción DASH	66
Figura 41: 4G dieciséis segundos. Reducción del límite de velocidad	67
Figura 42: 4G dieciséis segundos. Vuelta al límite de velocidad inicial	68
Figura 43: 4G dieciséis segundos. Gráfica calidad de imagen	68
Figura 44: Ejemplo imagen tasa de bits 250 Kbps	69
Figura 45: Ejemplo imagen tasa de bits 500 Kbps	69
Figura 46: Ejemplo imagen tasa de bits 1000 Kbps	70
Figura 47: Ejemplo imagen tasa de bits 2500 Kbps	70

1. Introducción

1.1 Motivación

El consumo de recursos multimedia, ya sea bajo demanda o en vivo, es cada vez mayor. La cantidad y diversidad de dispositivos que se conectan a internet crece constantemente, esto combinado con los diferentes tipos de redes que existen puede suponer un problema debido a las características de cada dispositivo y la forma y condiciones con la que se conectan, ya que si se ofrece el mismo recurso sin tener en cuenta las posibilidades de cada uno el consumo de éste puede no cumplir con las expectativas esperadas. Nadie quiere ver un video sin ser capaz de distinguir las imágenes que se le muestran debido a la baja calidad o escuchar una canción con cortes cada pocos segundos. En este TFG nos centraremos en el consumo de este tipo de contenidos vía redes móviles como el 3G o el 4G y como MPEG-DASH nos puede proporcionar la mejor experiencia posible en función de las capacidades de nuestra conexión.

1.2 Objetivos

El objetivo de este trabajo es implementar una solución, utilizando el protocolo MPEG-DASH, de streaming bajo demanda. Con este propósito, se creará un escenario que simule las condiciones con las que se puede llevar a cabo este tipo de consumo de recursos multimedia. Además, se añadirán a las diferentes simulaciones las características de las redes móviles 3G, 4G, haciendo hincapié en las variaciones del ancho de banda y la estabilidad de dichas conexiones. Teniendo el mismo contenido multimedia con diferentes resoluciones se estudiará como, gracias a la adaptabilidad de MPEG-DASH, se estará retransmitiendo nuestro contenido siempre aprovechando el máximo del ancho de banda disponible y como varía la calidad del streaming antes los cambios bruscos o cortes del propio ancho de banda.

1.3 Estructura de la memoria

En este apartado se van a enumerar los diferentes capítulos a continuación en la memoria, así como una pequeña introducción a lo que se va a tratar en cada uno.

Tras acabar la introducción, en el segundo capítulo comentaremos el concepto de red móvil. Haremos un repaso a la evolución histórica de este tipo de tecnologías y por último nombraremos los principales estándares en las redes móviles, hablando de las características de cada uno.

El tercero será el capítulo más extenso de la memoria. Se comentará el origen del streaming, así como las diferencias entre sus principales variantes. Serán expuestos los diferentes protocolos que lo hacen posible. Introduciremos el concepto de streaming en HTTP y streaming adaptativo. Para cerrar el capítulo, se hará un análisis en profundidad del protocolo MPEG-DASH.

En el cuarto capítulo de la memoria se hablará de todo lo relacionado con la implementación que vamos a llevar a cabo, es decir, explicaremos paso a paso el entorno que vamos a preparar, así como todas las configuraciones necesarias para su correcto funcionamiento.

Partiendo del escenario previamente montado y configurado en el capítulo cuarto, en el quinto capítulo llevaremos a cabo pruebas sobre el streaming de un mismo vídeo, pero con diferentes codificaciones.

El sexto será el último capítulo de la memoria. Tomando como referencia los resultados obtenidos en las diferentes pruebas realizadas y toda la información expuesta previamente, haremos un análisis de lo que es el protocolo MPEG-DASH y el porqué de su aparición, así como una resolución de los resultados obtenidos en las pruebas.

2. Redes móviles

En este apartado hablaremos de que son las redes móviles y su evolución a través de la historia reciente centrándonos en el Internet móvil, además de comentar los principales protocolos y estándares utilizados.

2.1 Internet móvil

El Internet móvil es la aplicación mediante la cual un usuario puede acceder a la información de cualquier servicio de internet desde cualquier lugar e independientemente del dispositivo que utilice para ello [1].

Es una tecnología que actualmente se encuentra presente en prácticamente todas las zonas del mundo. Entre otras cosas nos ha permitido: llevara cabo operaciones que normalmente realizaríamos con un equipo de sobremesa, poder conectarnos a Internet sin la necesidad de tener un PC, hacer que Internet y toda la información a la que se puede acceder mediante su uso esté presente en lugares donde un cable no puede llegar.

2.2 Evolución histórica

Aunque debido a las limitaciones que presentaban las redes móviles en sus inicios no podemos hablar del uso del streaming hasta la tercera generación, es conveniente conocer el origen de las mismas.

La primera generación o 1G surgió en la década de 1970. Gracias a su uso se pudo dar vida a lo que hoy conocemos como teléfonos móviles, aunque por aquel entonces se les dio el nombre de 'celulares', denominación que actualmente se mantiene en algunos países, ya que la señal transmitida por estos dispositivos se dividía en celdas hexagonales que podían ser reutilizadas según el caso. La señal 1G solamente permitía la realización de llamadas de voz, el intercambio de texto o datos más pesados aún era imposible.

Más adelante, a finales de los noventa, los teléfonos móviles habían penetrado muy fuerte en la sociedad. Esto, unido al fuerte crecimiento que habían experimentado las tecnologías que hacen parte de dichos dispositivos, provocó que las redes móviles del momento se quedasen un poco cortas, pues no aprovechaban todo el potencial que presentaban los dispositivos creados a su alrededor. Entonces se empezó a usar una tecnología que, pese a que se había desarrollado durante los 80 aún no se había puesto en práctica en estos dispositivos. Esta fue el estándar GSM, que permite la transferencia de datos a una velocidad mayor.



Con el uso de este estándar entraron en juego capacidades como el correo de voz y los mensajes de texto, también conocidos como SMS. Además, algunos de los teléfonos más avanzados de la época pudieron acceder a portales web creados específicamente para su uso en tales dispositivos (portales WAP).

Antes de la llegada de la tercera generación solamente se podían ver vídeos o realizar videoconferencias en teléfonos móviles en las películas de ciencia ficción. Pero a principios del siglo XXI, cuando empezaron a aparecer los primeros smartphones, la idea de que en algún momento iba a aparecer una nueva tecnología que iba a permitir el intercambio de datos a alta velocidad mediante las redes móviles era algo inevitable. Entonces llegó el año 2003 y todo cambió. Con la aparición de la nueva generación las redes alcanzaron velocidades de hasta 2Mbps, esto suponía un aumento de más de cien veces lo permitido con el estándar GSM. A partir de ese momento los teléfonos inteligentes empezaron a llegar de forma masiva a los consumidores. Estos dispositivos se convirtieron en accesorios increíblemente valiosos para nuestras vidas. Gracias a la unión de esta nueva tecnología y unos dispositivos potentes y accesibles, los usuarios podían acceder a versiones completas de los sitios web que solo podían usar desde ordenadores. La tercera generación siguió evolucionando y gracias a los nuevos estándares HSPA y HSPA+ consiguieron alcanzar velocidades de hasta 12Mbps, y en aquellos países donde la infraestructura de la red móvil lo permitía se lograron alcanzar los 72Mbps. Esta generación fue la que dio un cambio radical al uso de los teléfonos móviles, además de que fue la que por primera vez permitió que desde cualquier lugar y en cualquier momento pudiésemos consumir contenidos en streaming.

Con la tercera generación se produjo un cambio de paradigma en el uso de Internet y los teléfonos, ahora lo que predomina es el contenido multimedia, por lo que la cuarta generación, en la que actualmente nos encontramos, no es más que una evolución natural de las bases que la conectividad 3G había establecido. El principal y más relevante cambio de lo que se conoce como conectividad Long-Term Evolution o 4G LTE es la velocidad. Actualmente las redes móviles superan con facilidad los 100 Mbps. Esto, unido al también significativo cambio en las capacidades de nuestros dispositivos, permite procesar mucha más información en un considerable menor tiempo, por lo que el streaming de contenidos multimedia es algo que forma parte de nuestro día a día. Ya sean contenidos bajo demanda o en vivo, las características de las conexiones actuales permiten el consumo de tales contenidos, incluso en resoluciones 4K, sin ningún problema.

A día de hoy esta es la situación en la que nos encontramos, el mundo se está transformando en torno a las redes móviles. Quizá en unos años, cuando la quinta generación se una realizad en todo el mundo, desaparecerán las antenas de radio y televisión y todos los contenidos multimedia se consuman lo harán mediante streaming bajo demanda o en vivo.

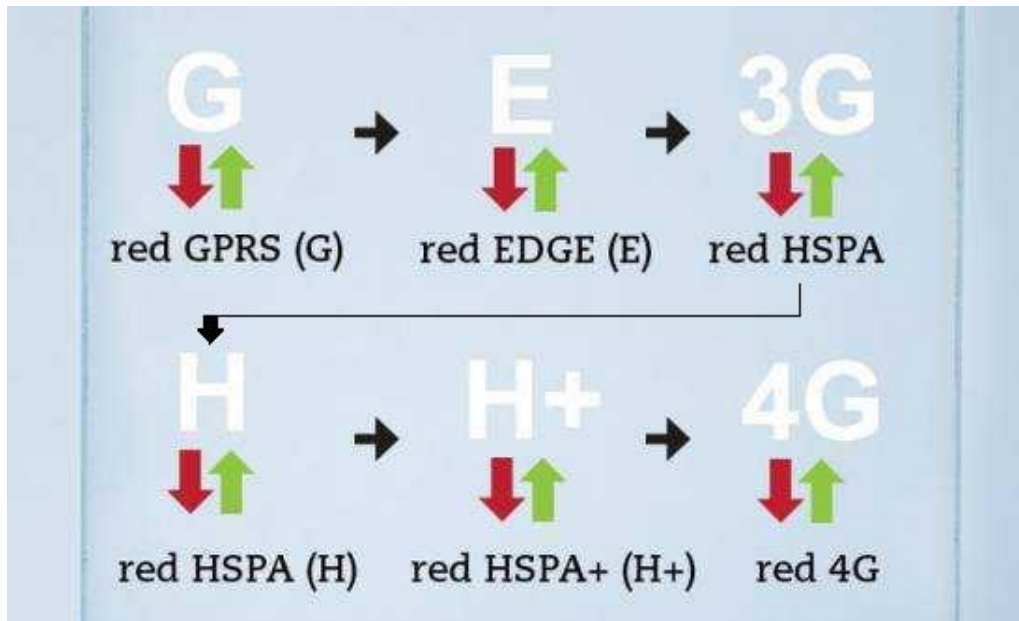


Figura 1: Esquema de la evolución de las redes móviles

2.3 Estándares y tecnologías

2.3.1 GPRS

GPRS (General Packet Radio Service) fue la primera red que permitía el intercambio de datos más allá de lo que suponía una llamada telefónica. Gracias a esta tecnología surgieron los servicios de mensajes cortos (SMS) y la mensajería multimedia (MMS).

Proporciona una cobertura inalámbrica completa y la velocidad a la que opera está entre los 56 y los 144 Kbps. Se estima que se pueden mandar unos 30 SMS cada minuto mediante esta red.

2.3.2 EDGE

La tecnología EDGE, que es el acrónimo de 'Enhanced Data Rates for GSM of Evolution', no dista mucho de lo que es GPRS. De hecho, podemos considerar a EDGE como un GPRS mejorado, ya que funcionaban bajo las mismas condiciones, pero con un aumento considerable de la velocidad. Con esta tecnología se logran alcanzar velocidades de hasta 384 Kbps, lo que supone que se puedan recibir archivos pesados y que por primera vez la navegación por paginas web complejas sea accesible a dispositivos móviles.

2.3.3 3G

Con la llegada de la tecnología de tercera generación el principal objetivo era facilitar la transferencia de datos multimedia. Con velocidades hasta siete veces mayor que su predecesora, llegando a alcanzar los 2 Mbps, esta red puede cargar una web compleja en 10 segundos, enviar y recibir archivos multimedia sin problemas, permite la mensajería instantánea y, por primera vez, el streaming de recursos multimedia, aunque con una calidad no muy alta.

2.3.4 HSPA y HSPA+

HSPA (High-Speed Packet Access) es un estándar que funciona bajo la tecnología 3G. Aportó una cobertura más estable además de un notable incremento de la velocidad. Bajo este estándar por primera vez se habla no solamente de la velocidad de descarga, que alcanza los 14,4 Mbps, sino también de la velocidad de subida, que llegaba a un máximo de 5,76 Mbps. Gracias a este estándar el streaming dio un salto de calidad más que considerable, pues las calidades de imagen alcanzaban sin problemas la calidad HD o 1080p.

HSPA+ es una evolución de HSPA que aporta mayor estabilidad y velocidad en la conexión.

2.3.5 4G LTE

El estándar LTE (Long Term Evolution) es el predominante en la actualidad. Las características del mismo han hecho que los dispositivos móviles puedan alcanzar velocidades de bajada de hasta 325 Mbps mientras que la subida a ascendido a unos considerables 86,5 Mbps respecto al estándar predecesor. ¿Qué implican estas velocidades? Pues bien, unido a la potencia de los dispositivos móviles del momento, estas características hacen que ya seamos consumidores o creadores de contenido multimedia, el streaming de tales contenidos en las máximas calidades del momento es una realidad. Sin importar el lugar en el que nos encontremos y siempre que tengamos conexión con este tipo de redes, podremos acceder a cualquier contenido que se encuentre disponible al instante. No es de extrañar que plataformas que se dedican a proveer al consumidor de una amplia gama de recursos como Netflix, HBO o Amazon Prime hayan experimentado un gran aumento en el número de suscriptores a sus servicios conforme las redes móviles han ido evolucionando y mejorando.



Implementación y evaluación de MPEG-DASH bajo las características de redes móviles.



3. Streaming

Durante este capítulo hablaremos del concepto de streaming y sus variantes 'en vivo' y 'bajo demanda'. Además, hablaremos de la evolución histórica del mismo, haciendo un breve análisis de los principales protocolos implicados en este servicio.

3.1 Streaming

El concepto de streaming se puede definir como la distribución de contenido multimedia, ya sea audio o vídeo, de forma continua. Esta distribución se realiza a través de la red de manera que el recurso que se está retransmitiendo va siendo descargado por el usuario al mismo tiempo que se consume. Este funcionamiento permite que no se requiera del espacio total del recurso para poder acceder a él ni del tiempo necesario que tomaría una descarga tradicional. El streaming puede ser en vivo o bajo demanda.

3.2 Evolución histórica

El streaming tiene origen en la década de 1990 junto a los primeros ordenadores que, pese a presentar unas prestaciones muy inferiores a lo que hoy tenemos en un ordenador medio, eran capaces de reproducir contenido multimedia. El streaming no era nada popular por aquella época, había ordenadores capaces de llevarlo a cabo, pero su precio era tan elevado que los hacían inaccesibles para el usuario de a pie. A este escollo había que añadir la poca potencia de las redes del momento, pues su poco ancho de banda no permitía demasiado.

Estas limitaciones se iban a superar pocos años después, pero seguía sin existir una plataforma que contase con un catálogo de recursos a los que el usuario pudiese acceder o simplemente disfrutar del contenido que se transmitía en vivo. Entonces apareció Mbone, una red experimental para la difusión de contenidos, gracias a su red virtual se pudieron llevar a cabo los primeros conciertos de música en directo a través de Internet, entre ellos se llevó a cabo uno de los Rolling Stone que, pese a que hoy no sería considerada una retransmisión de mucha calidad, hizo historia.



El streaming había llegado para quedarse. Poco tiempo después del éxito del concierto surgió Real Player, el primer sistema de reproducción de vídeo en streaming que además fue el primero en retransmitir contenido deportivo en vivo, en este caso un partido de baseball de los New York Yankees.

Ya con la entrada en el nuevo siglo la tecnología iba evolucionando, lo que directamente supuso una mejora en todos los aspectos del streaming. Se fueron eliminando limitaciones costosas como contar con equipos técnicos especializados y que varias personas trabajasen a la vez en los diferentes aspectos relacionados con la grabación, la edición y la retransmisión. Llegamos al año 2005 y apareció YouTube, la plataforma que se iba a convertir en el portal número uno en lo referente al consumo de contenido multimedia. YouTube, junto a Spotify que surgió en 2008 y se dedicaba exclusivamente a streaming contenido musical. Llevaron a este servicio a la fama mundial. Actualmente se estima que el 60% del tráfico mundial de internet es de naturaleza audiovisual [3].

3.3 Streaming en vivo y bajo demanda

Hoy por hoy el streaming se presenta en estas dos variantes. El más popular y extendido uso del streaming es para consumir recursos bajo demanda, es decir, una plataforma cuenta con un amplio catálogo de contenido al que el usuario puede acceder y reproducir en cualquier momento sin tener que esperar a la descarga completa del mismo. Plataformas como Netflix, Amazon Prime, YouTube o la recién surgida Disney+ se dedican prácticamente en su totalidad a proveer a sus usuarios de contenido bajo demanda.

Por otra parte, también existe el streaming de recursos en vivo, que consiste en la reproducción del contenido a la vez que este se captura. Esta forma de streaming no es tan popular, pero se encuentra presente en múltiples aspectos de uso diario como puede ser el simple hecho de ver la televisión o, más exitosamente y con un crecimiento cada vez mayor, la retransmisión de eventos deportivos. En España por ejemplo la plataforma líder es Movistar, que cuenta con una amplia gama de contenidos bajo demanda, aunque obtiene su éxito gracias al streaming en vivo, principalmente de eventos deportivos.

3.4 Protocolos

Para el correcto funcionamiento del streaming es necesaria la intervención de diferentes protocolos. En este apartado mencionaremos y explicaremos brevemente en qué consisten los más relevantes.

3.4.1 UDP (User Datagram Protocol)

UDP ha sido utilizado por un gran número de servicios de transmisión como protocolo de transporte. En realidad, UDP no es mucho más que eso, un simple protocolo de transmisión. UDP no ofrece garantía de entrega ni establece un diálogo de intercambio. Aunque por su funcionamiento se pueda suponer que es un protocolo con muy baja fiabilidad, son estas características las que lo han hecho famoso, pues su simplicidad le permite llevar a cabo una transmisión de datos mucho más sencilla que con otros protocolos, además de reducir muy considerablemente el retardo de extremo a extremo de la conexión. Todo esto hace que sea idóneo para el streaming de contenido en vivo o el uso de aplicaciones en tiempo real.

3.4.2 TCP (Transmission Control Protocol)

TCP es el protocolo utilizado en la capa de transporte por HTTP. TCP es un protocolo orientado a la conexión, a diferencia de UDP que se orienta a la transmisión. TCP intercambia información de control antes del comienzo del intercambio de información, asegurando así que tanto cliente como servidor van a trabajar sobre las mismas condiciones. Además, con TCP funcionando correctamente tendremos la certeza de que todos los mensajes enviados han llegado a su destino sin problemas y que lo han hecho en el orden correcto. Dentro del contexto del streaming, sobre todo del streaming bajo demanda, TCP es fundamental para poder prestar un servicio de calidad. En el caso de estar viendo una película/serie o cualquier video de internet, el que el audio y video no llegasen sincronizados o que hubiese saltos temporales durante la reproducción no proporcionaría una buena experiencia de usuario, con TCP garantizamos que esto sea correcto.

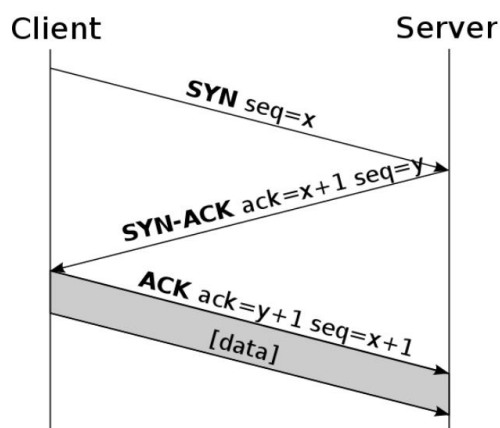


Figura 2: Establecimiento de conexión TCP

3.4.3 HTTP (HyperText Transfer Protocol)

HTTP es el protocolo de transmisión de información de la World Wide Web, es decir, el protocolo que se establece para que tanto el creador de la conexión como el destinatario puedan hablar un mismo idioma a la hora de transmitirse información. Funciona como un protocolo de la capa de aplicación basado en un modelo computacional cliente servidor.

Con el uso de este protocolo se establecen criterios de sintaxis y semántica que permitan la comunicación entre los diferentes elementos que constituyen la arquitectura web (servidores, clientes, proxies). HTTP fue creado en 1989 por Tim Berners-Lee, el también creador del lenguaje HTML(HyperText Markup Language) y de la WWW(World Wide Web).

El funcionamiento del mismo comienza con el cliente estableciendo una conexión mediante TCP con el servidor. Tras el intercambio de mensajes de control el cliente envía una petición con la que, si cuenta con los permisos necesarios, logra acceder a los recursos que busca en el servidor. El servidor no cierra la conexión hasta que no lo haga el cliente, logrando así lo que se conoce como una conexión persistente, este tipo de conexiones mejoran el rendimiento tanto del cliente como del servidor, puesto que reducen el uso de memoria, cpu, latencia, etc. Sin embargo, también se pueden utilizar conexiones no persistentes, aunque esto supone que para cada intercambio de información se tuviese que crear una nueva conexión entre cliente y servidor, esto es lo que más sobrecarga genera.

HTTP es un protocolo sin estado, es decir, que no almacena información de las diferentes conexiones que puede realizar el mismo cliente al mismo servidor. Para salvar esta información los servidores hacen uso de las conocidas cookies.

Algunos de los principales métodos de petición HTTP son:

GET: Recupera cualquier información identificada en la petición. Si en la petición se identifica a un proceso se recuperarán los datos producidos por ese proceso. Todas las peticiones que utilizan el método GET recuperarán únicamente datos.

HEAD: Funciona de forma similar al método GET, con la diferencia de que en la respuesta solo se recuperarán líneas y cabeceras sin incluir el cuerpo de la respuesta.

PUT: Solicita al servidor que almacene el cuerpo de la petición en una ubicación específica indicada por url.

POST: Envía al servidor un recurso determinado, causando a menudo un cambio en el estado del servidor.

PATCH: Es utilizado para aplicar modificaciones parciales a un recurso.

TRACE: Realiza pruebas de eco de mensajes en el camino existente hacia un recurso determinado.

OPTIONS: Representa una solicitud de información acerca de las opciones de comunicación disponibles, es decir, este método es el que se utiliza para describir las opciones de comunicación existentes de un recurso destino.

DELETE: Solicita al servidor la eliminación de un fichero en una ubicación específica dada por url.

CONNECT: Utilizado por el cliente establece una conexión de red con un servidor indicado en forma de túnel.



Figura 3: Esquema de métodos de petición HTTP [4]

3.4.4 Evolución de HTTP

En un principio HTTP era un protocolo con pocos métodos disponibles, por lo que no era muy utilizado. Esto fue cambiando conforme llegaban nuevas versiones del mismo, proporcionando una mayor versatilidad al protocolo. Las mejoras de HTTP unidas al aumento del ancho de banda y al tremendo crecimiento que experimentó Internet hicieron que se convirtiese en el protocolo por varias razones.

Primero, el contenido multimedia ahora se puede entregar de manera eficiente en segmentos más grandes usando HTTP. En segundo lugar, en HTTP es el cliente quien controla la transmisión. Esto se basa en el hecho de que el cliente, en la petición que envía, indica la cantidad de información que

espera recibir en la respuesta, de modo que el firewall puede comprobar que el servidor está enviando esa cantidad, por lo que el bloqueo de la conexión debido a problemas de seguridad rara vez ocurre. Finalmente, otro beneficio de HTTP streaming es el uso de servidores web, además, el cliente es quien administra la transmisión, lo que significa que el servidor no necesita mantener la información de estado de la sesión del cliente. Esto se traduce en una muy buena escalabilidad.

HTTP ha contado con las siguientes versiones:

HTTP/0.9

La versión inicial de HTTP no tenía número de versión; aunque posteriormente se la denominó como 0.9 para distinguirla de las versiones siguientes. HTTP/0.9 es un protocolo extremadamente sencillo: una petición consiste simplemente en una única línea, que comienza por el único método posible, GET, seguido por la dirección del recurso a pedir (no la URL, ya que tanto el protocolo, el servidor y el puerto, no son necesarios una vez ya se ha conectado al servidor). La respuesta también es muy sencilla: solamente consiste el archivo pedido.

Al contrario que sus posteriores evoluciones, el protocolo HTTP/0.9 no usa cabeceras HTTP, con lo cual únicamente es posible transmitir archivos HTML, y ningún otro tipo de archivos. Tampoco había información del estado ni códigos de error: en el caso un problema, el archivo HTML pedido, era devuelto con una descripción del problema dentro de él, para que una persona pudiera analizarlo.

HTTP/1.0

La versión 0.9 del protocolo era muy simple y limitada, por lo que tras su revisión se amplió añadiendo al método GET los métodos HEAD y POST. Además de los métodos se añadieron algunas mejoras que dotaron al protocolo de una mayor flexibilidad:

- En cada petición se envía la versión del protocolo.
- Al comienzo de la respuesta se añade un código de estado para que se pueda responder al éxito o al fracaso de la petición y actuar en consecuencia.
- Se introdujeron las cabeceras, permitiendo la transmisión de metadatos.
- Gracias a las cabeceras se permite la transmisión de otros ficheros más allá de HTML.

Cada una de estas mejoras no se desarrollaron a la vez ni de forma planeada. Un servidor y un navegador iban añadiendo una nueva funcionalidad y, si su adaptación era positiva se establecía. A consecuencia de esta forma de funcionar surgieron problemas de interoperabilidad, por lo que tuvo que publicarse el documento RFC 1945 para definir las prácticas adecuadas.

HTTP/1.1

Esta versión es la primera versión estándar del protocolo HTTP. Se añadieron la mayoría de las características principales, desde la versión 1.1 una conexión podía ser reutilizada, ahorrando el tiempo de establecimiento de la conexión para cada petición, es lo que conocemos como conexión persistente. Se permite realizar más de una petición sin haber finalizado la anterior, disminuyendo así la latencia, esto es el pipelining. Se introdujo la negociación entre cliente y servidor para que ambos pudiesen acordar las características y condiciones de las peticiones y el contenido a intercambiar.

Esta es la versión más utilizada hoy en día, se ha ido actualizando y añadiendo todos los métodos de conocemos y está funcionando de forma estable durante más de 15 años.

HTTP/2.0

Esta versión surgió con el objetivo de suplir o mejorar las carencias existentes en sus versiones predecesoras. Entre ellas tenemos el uso de una única conexión. Con cualquiera de las versiones anteriores de HTTP es necesario utilizar varias conexiones para poder descargar todo el contenido de una web. HTTP 2.0 utiliza una única conexión que ofrece múltiples solicitudes y respuestas en paralelo. Se elimina el envío de información duplicada durante una conexión, así se reduce el consumo de recursos. Introduce multiplexación para poder enviar y recibir varios mensajes a la vez, esto aumenta considerablemente la velocidad de descarga. Se convierte en un protocolo binario, esto facilita la localización del principio y el fin de cada frame, además es menos propenso a errores y más sencillo de optimizar. Con HTTP 2.0 se comprimen las cabeceras, además se incluyen la priorización de servicios y el server push.

Conclusión streaming sobre HTTP

Todo esto ha llevado al desarrollo de nuevas investigaciones por parte de las grandes empresas tecnológicas sobre el streaming con HTTP. Algunas de las plataformas que han desarrollado han sido MSS(Microsoft Smooth Streaming) de Microsoft, HLS(HTTP Live Streaming) de Apple o HDS(HTTP Dynamic Streaming) de Adobe.



3.5 Diferentes streaming en HTTP

En este apartado se van a explicar los tres principales tipos de streaming basados en HTTP que existen.

3.5.1 Descarga progresiva

Todos los servidores web son capaces de llevar a cabo este tipo de streaming, Su funcionamiento consiste básicamente en la descarga de un archivo de datos indiferentemente de que sea audio, video o ambos. Tan pronto como el navegador haya descargado una cantidad suficiente de vídeo, suele estar entre los dos y diez segundos, comenzará la reproducción. Aquellas partes que se hayan descargado y aún no se tengan que reproducir quedarán almacenadas en el búfer de memoria.

Almacenar el vídeo en memoria permite que este se pueda pausar, adelantar o atrasar a placer, siempre y cuando la parte que se quiera visualizar ya haya sido descargada.

Para este streaming no es necesario el uso de protocolos especiales, sin embargo, si hace falta definir un formato para poder procesarlo en función del contenido parcial que ya se haya descargado. Sobre la descarga, esta se realiza utilizando el máximo ancho de banda posible tanto en el cliente como en el servidor hasta el momento en el que se llena el búfer, tras ello se descargará a la velocidad que se vacíe el búfer en la reproducción. Al no controlar que la reproducción se adapte a la descarga, si nuestro ancho de banda no alcanza la velocidad con la que se vacía el búfer este llegará a vaciarse, provocando así que la reproducción sufra parones mientras esperamos a que el búfer vuelva a llenarse.

3.5.2 HTTP Pseudo Streaming

Este tipo de streaming funciona de forma similar a la descarga progresiva con la ventaja de que permite al usuario realizar saltos temporales en la reproducción sin la necesidad de descargar el contenido intermedio. Esto supone un ahorro tanto de ancho de banda como de memoria. La plataforma a la que se más se asemeja este funcionamiento es YouTube, donde se muestra una barra roja para indicar el punto de la reproducción en la que nos encontramos y una barra gris que señala hasta donde se ha descargado el vídeo.

3.5.3 HAS (HTTP Adaptive Streaming)

Este método funciona eligiendo un fragmento de vídeo para cada instante definido en función del ancho de banda y la capacidad de la cpu del cliente en cada momento. Para su correcto funcionamiento requiere de que un mismo vídeo se tenga codificado, dividido y almacenado en múltiples tasas de bits diferentes, es decir, es necesario tener el mismo vídeo repetido tantas veces como opciones de visualización se quieran dar.

Una transmisión de HAS comienza como cualquier conexión a un servidor web, con un mensaje de petición HTTP. En esta primera petición se solicita información sobre las diferentes codificaciones disponibles de audio y vídeo que el servidor contiene para un recurso concreto. Esta información se guarda en un fichero llamado manifiesto, en él se informa al cliente de las opciones disponibles y de cómo solicitarlas. El cliente procesa la información recibida y envía un mensaje HTTP GET para cada segmento del vídeo que desea descargar, indicando en dicho mensaje las características del segmento en función de su ancho de banda disponible y la ocupación del búfer.

Con este tipo de streaming se busca que el cliente pueda disfrutar del recurso solicitado en las mejores condiciones posibles sin importar su tipo de conexión, el dispositivo en el que se reproduzca o el software que tenga disponible. Este funcionamiento dinámico es muy ventajoso cuando se utilizan dispositivos como smartphones, tabletas o cualquier otro que esté conectado a una red móvil. Las redes móviles cuentan con mayores variaciones de señal, tanto en intensidad como en velocidad, que las redes domésticas. Esto unido a la gran variedad de dispositivos hace que sea fundamental un buen control del ancho de banda disponible en cada momento. Para solventar este problema en HAS cada cliente realiza comprobaciones en la red de forma periódica, permitiendo así optar por una calidad inferior si se sufre una bajada en el ancho de banda y aumento inmediatamente tal calidad en el momento que se recuperan unas buenas condiciones en la conexión. Este funcionamiento ha permitido a HAS solucionar el problema del vaciado completo del búfer durante la reproducción, permitiendo que, mientras haya conexión, la reproducción no se va a detener. Se logra manteniendo el búfer lo más estable posible. Si por algún problema empieza a vaciarse a una velocidad igual que se consume el cliente solicitará segmentos de menor calidad para poder recuperar la estabilidad del búfer lo antes posible.

Al inicio de la conexión en la que se va a consumir un recurso con este tipo de streaming el reproductor solicitará segmentos con la menor calidad posible para, una vez se haya llenado el búfer, iniciar la reproducción. No se solicitan más segmentos hasta una parte del búfer haya sido consumida. Gracias a esta técnica HAS puede soportar también la transmisión en vivo, pseudo streaming y descarga progresiva solo permiten transmisión bajo demanda. HAS también trata de equiparar la velocidad a la que el búfer se llena con la que lo hace al vaciarse, dotando así de mayor estabilidad a la reproducción. Al hacer esto la velocidad de bits de codificación se ajusta la de visualización, permitiendo que el contenido se disfrute con la mayor calidad posible a la vez que la conexión cuenta con una gran estabilidad.

En la actualidad existen principalmente tres soluciones creadas por grandes empresas tecnológicas. MSS de Microsoft, HLS de Apple y HDS de Adobe. Estas soluciones son muy efectivas, pero al haber sido creadas para los productos propios de cada empresa presentan grandes incompatibilidades entre los dispositivos propios y los del resto del mercado. Esto impulsó a crear un estándar que fuera capaz de aunar estas tecnologías y eliminar todo tipo de restricción. Este estándar es MPEG-DASH.



3.6 MSS, HLS y HDS

En este apartado presentaremos las tres soluciones privadas que existen para el streaming adaptativo.

3.6.1 MSS (Microsoft Smooth Streaming)

Para su solución Microsoft almacena un único archivo en formato MP4 para cada resolución que se quiera ofrecer. Los archivos internamente se dividen en pequeños fragmentos de dos segundos de duración. Cabe destacar que estas divisiones del archivo completo no se almacenan, sino que cuando llega la solicitud del cliente se realiza una división virtual para proporcionar las partes del archivo solicitadas. De esta forma se facilita considerablemente la gestión de los archivos en el servidor.

En cuanto a su funcionamiento, en primer lugar el cliente solicita al servidor el manifiesto, una vez lo obtiene solicita los fragmentos que necesita mediante una url.

Microsoft solamente ha dotado de esta solución a dispositivos propios como XBOX o smartphones y tabletas que utilicen Windows Phone como sistema operativo.

3.6.2 HLS (HTTP Live Streaming)

Apple basa su solución en tres elementos principales. En primer lugar, cuentan con un servidor que recibe los archivos de audio y vídeo originales, cuando tiene lo necesario se encarga de codificar y segmentar estos archivos. De ahí se envían a otros servidores que forman parte de una red de distribución de contenidos a la que se conecta el cliente para solicitar los segmentos que necesite.

El primer servidor codifica y encapsula el archivo en formato MPEG 2TS, posteriormente éste es dividido en segmentos de diez segundos. A cada uno de los segmentos generados se le asigna un número que aumenta progresivamente con cada uno de ellos, de esta forma es mucho más sencilla su organización. Tras finalizar este proceso en cada archivo, los segmentos son enviados a la red de distribución de contenidos en formato .ts. Una vez el recurso ya se encuentra disponible el cliente es el que elige el formato en el que quiere descargar cada segmento en cada momento.

Como era de esperar, Apple lanzó esta solución de forma que únicamente fuese compatible con sus dispositivos iPhone, iPod, iPad y Mac. Sin embargo, con el tiempo incluyeron el formato MPEG4 en su solución para así reunir a un rango de dispositivos compatibles mucho más amplio.

3.6.3 HDS (HTTP Dynamic Streaming)

La solución de Adobe es similar a la de Apple en el sentido de que cuando se fragmentan los archivos originales de audio y vídeo se les asigna un número que se incrementa con cada nuevo fragmento, pero en este caso son fragmentos de cuatro segundos y no de diez. Este número sirve para identificar a los fragmentos en cada momento.

Toda la información que permite relacionar a los segmentos generados con los números de fragmentos que se requieren se encuentra en el archivo manifiesto del servidor. En este caso el manifiesto contiene submanifiestos en el que se pueden ver las diferentes calidades disponibles para cada recurso. Estos submanifiestos indican como solicitar un determinado fragmento con una calidad concreta al servidor.

Respecto a la compatibilidad de esta solución, era posible con la mayoría de dispositivos que contaban con Adobe Flash Player, pero debido a los problemas de seguridad que tuvo se ha ido quedando en desuso.

3.7 MPEG-DASH

DASH o Dynamic Adaptive Streaming over HTTP es la principal solución de streaming basada en HTTP con tasa de bits adaptable. La desarrolló MPEG. Su desarrollo comenzó en 2010, pero no fue publicada como un estándar internacional hasta abril de 2012. Es una solución que aporta compatibilidad con cualquier dispositivo existente independientemente de la red a la que se conecte, además, elimina las limitaciones que presentaban las principales soluciones existentes hasta el momento.

A diferencia de las anteriores soluciones que marcaban los formatos en los que el audio y vídeo se codificaban, en esta solución se permite al cliente definir estos parámetros.

A la hora de iniciar una sesión funciona de forma similar que las soluciones anteriores, el cliente vía HTTP solicita la información necesaria al servidor en el que se encuentran los recursos que queremos. Esta información se envía al cliente en el manifiesto de MPEG-DASH, para esta solución toma el nombre de MPD (Media Presentation Description).



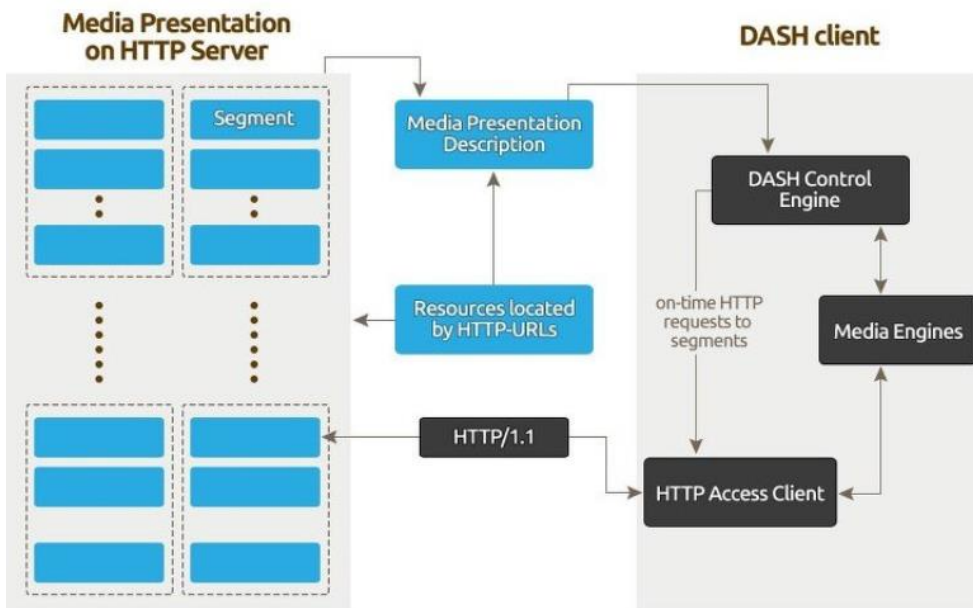


Figura 4: Esquema de la arquitectura de MPEG-DASH [8]

El cliente, habiendo analizado la información contenida en el MPD, envía una nueva petición HTTP con las características que ha considerado idóneas en función del ancho de banda actual. En este momento comienza la descarga de los segmentos y la reproducción. Si se sufren cambios en las condiciones bajo las que está funcionando el cliente, este cambiará las características en las solicitudes de los segmentos para adecuar la calidad del contenido y que el streaming no se interrumpa.

3.7.1 MPD (Media Presentation Description)

El MPD es el manifiesto del estándar MPEG-DASH, es un documento que contiene toda la información relacionada con los diferentes segmentos disponibles para que el cliente pueda decidir en cada momento cual es el más adecuado y así solicitarlo. Este archivo tiene un formato xml [10].

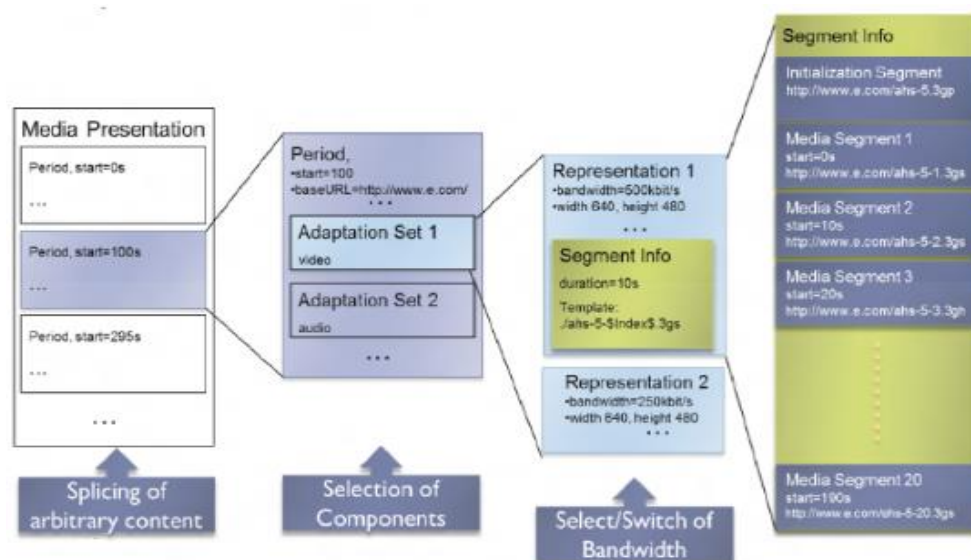


Figura 5: Contenido archivo MPD [12]

En la figura 5 se puede observar cómo está formado el modelo de datos del fichero MPD de forma jerárquica:

- En primer lugar, contiene una secuencia de uno o más Periods
- Un Period está formado por uno o más Adaptation Set
- Cada Adaptation Set a su vez se divide en Representations
-

Los principales elementos y atributos que puede contener el fichero MPD son:

Nombre del elemento o atributo	Uso	Descripción
MPD		Elemento raíz
@id	OP	Especifica un identificador para el MPD. Si no se especifica se puede utilizar, por ejemplo, la URL del MPD como identificador.
@profiles	OB	Especifica una lista de Media Presentation profiles (3.3).
@type	OPD default: static	Especifica si el MPD debe ser actualizado (@type=“dynamic”) o no (@type=“static”).
@availabilityStartTime	COB Obligatorio para @type=“dynamic”	En el caso de que @type=“dynamic” este atributo debe estar presente. En este caso especifica la base para el cálculo del tiempo (en UTC) en el que está disponible cualquier segmento del MPD. En el caso de que @type=“static” este atributo especifica el tiempo en el que están disponibles todos los segmentos del MPD. Si el atributo no aparece entonces todos los segmentos del MPD están disponibles en el momento en el que está disponible el MPD.
@availabilityEndTime	OP	Especifica la última disponibilidad del tiempo final de un segmento para cualquier segmento del MPD. Si este atributo no está presente, su valor es desconocido.
@mediaPresentationDuration	COB Obligatorio para type=“static”	Especifica la duración de todo el MPD. Debe estar presente en el caso de que el atributo MPD@minimumUpdatePeriod no aparece.
@minimumUpdatePeriod	OP	Especifica el periodo más pequeño entre posibles cambios en el MPD. Si no aparece significa que el MPD no cambia.
@minBufferTime	OB	Especifica una duración que se utiliza para saber cuando se puede iniciar la reproducción.

ProgramInformation	0..N	Contiene información descriptiva sobre el contenido multimedia.
BaseURL	0..N	Contiene una Base URL que puede ser utilizada para resolver todas las URLs del MPD.
Location	0..N	Especifica una ubicación donde el MPD está disponible.
Period	1..N	Especifica la información de un Period (3.1.1).
Leyenda Para los atributos: OB = obligatorio, OP: Opcional, OPD= opcional con valor por defecto, COB=condicionalmente obligatorio Para los elementos: <mínimo>...<máximo> (N = ilimitado) Los elementos aparecen en negrita y los atributos van precedidos de @		

3.7.1.1 Period

El archivo MPD contiene varios Periods, podrái contener solamente uno si el recurso multimedia no tuviese una gran extensión, pero dado que los elementos que suelen consumir mediante MPEG-DASH suelen tener una extensión considerable por norma general hay más de un Period. Un Period contiene un intervalo del contenido multimedia con instante de inicio y un instante final.

Los principales elementos y atributos que puede contener un Period son:

Nombre del elemento o atributo	Uso	Descripción
Period		Especifica la información del Period
@xlink:href	OP	Especifica una referencia a un elemento Period externo.
@id	OP	Especifica un identificador para el Period, que debe de ser único dentro del MPD.
@start	OP	Especifica el PeriodStart del Period.
@duration	OP	Especifica la duración del Period.
@bitstreamSwitching	OPD default: false	Si su valor es 'true' es equivalente a que el atributo @bitstreamSwitching de cada AdaptationSet dentro del Period sea 'true'.
BaseURL	0..N	Contiene una Base URL que puede ser utilizada para resolver las URLs del Period.
SegmentBase	0..1	Especifica la información de Segment Base. Se sobrescribe por la información de AdaptationSet.SegmentBase y Representation.SegmentBase, en el caso de que existan estos elementos.
SegmentList	0..1	Especifica la información de Segment List. Se sobrescribe por la información de AdaptationSet.SegmentList y Representation.SegmentList, en el caso de que existan estos elementos.

SegmentTemplate	0..1	Especifica la información de Segment Template. Se sobreescribe por la información de AdaptationSet.SegmentTemplate y Representation.SegmentTemplate, en el caso de que existan estos elementos.
AdaptationSet	0..N	Especifica un AdaptationSet.
Subset	0..N	Especifica un Subset.
Leyenda Para los atributos: OB = obligatorio, OP: Opcional, OPD= opcional con valor por defecto, COB=condicionalmente obligatorio Para los elementos: <mínimo>...<máximo> (N = ilimitado) Los elementos aparecen en negrita y los atributos van precedidos de @		

3.7.1.2 Adaptation Set

Un Period se divide en Adaptation Sets, hay tantos Adaptation Sets por Period como componentes multimedia y alternativas de codificación haya para el mismo. Para un fichero multimedia corriente, por ejemplo, un capítulo de una serie, tendríamos un Adaptation Set para el audio y otro para el vídeo. Si contásemos con diferentes codificaciones para el vídeo tendríamos un Adaptation Set por cada una de ellas. En el caso del audio, si tenemos disponibles varias opciones de idioma, del mismo modo que con el vídeo habría un Adaptation Set para cada idioma disponible.

Los principales elementos y atributos que puede contener un Adaptation Set son:

Nombre del elemento o atributo	Uso	Descripción
AdaptationSet		Descripción del Adaptation Set.
@xlink:href	OP	Especifica una referencia a un elemento AdaptationSet externo.
@id	OP	Especifica un identificador para el AdaptationSet, que debe de ser único dentro del Period.
<i>CommonAttributesElements</i>		Especifica los atributos y los elementos comunes.
@contentType	OP	Especifica el tipo del componente del contenido multimedia para el Adaptation Set actual.
@bitstreamSwitching	OP	Si su valor es true indica que se puede cambiar de Representation dentro del mismo Adaptation Set durante la presentación
ContentComponent	0..N	Especifica las propiedades de un componente del contenido multimedia contenido en el Adaptation Set actual.
BaseURL	0..N	Contiene una Base URL que puede ser utilizada para resolver las URLs del Adaptation Set.

SegmentBase	0..1	Especifica la información de Segment Base. Se sobrescribe por la información de Representation.SegmentBase, en el caso de que exista este elemento.
SegmentList	0..1	Especifica la información de Segment List. Se sobrescribe por la información de Representation.SegmentList, en el caso de que exista este elemento.
SegmentTemplate	0..1	Especifica la información de Segment Template. Se sobrescribe por la información de Representation.SegmentTemplate, en el caso de que exista este elemento.
Representation	0..N	Especifica una Representation.
<p>Leyenda</p> <p>Para los atributos: OB = obligatorio, OP: Opcional, OPD= opcional con valor por defecto, COB=condicionalmente obligatorio</p> <p>Para los elementos: <mínimo>...<máximo> (N = ilimitado)</p> <p>Los elementos aparecen en negrita y los atributos van precedidos de @.</p> <p>Las listas de elementos y atributos aparecen en negrita y cursiva.</p>		

3.7.1.3 Representation

Por cada Adaptation Set existente tendremos tantas Representations como diferentes opciones de calidad de reproducción queramos ofrecer. Una Representation es una alternativa de codificación con diferentes características a las demás para un mismo componente del recurso multimedia. Entre ellas, las Representations de un mismo Adaptation Set, se diferencian por su resolución, número de canales, tasa de bits y otras características como el número de fotogramas por segundo.

Dado el carácter adaptativo de MPEG-DASH la Representation que se va a utilizar es solicitada de forma automática en función del ancho de banda disponible en cada momento, sin embargo, algunos reproductores como el de Youtube permiten elegir una opción concreta de calidad de reproducción, aunque por defecto funcionen con la opción automática.

Los principales elementos y atributos que puede contener una Representation son:

Nombre del elemento o atributo	Uso	Descripción
Representation		Descripción de la Representation.
@id	OB	Especifica un identificador para la Representation, que debe de ser único dentro del Period, a no ser que la Representation sea funcionalmente idéntica a otra dentro del mismo Period.

@bandwidth	OB	Bandwidth mínimo necesario.
<i>CommonAttributesElements</i>		Especifica los atributos y los elementos comunes.
BaseURL	0..N	Contiene una Base URL que puede ser utilizada para resolver las URLs de la Representation.
SubRepresentation	0..N	Especifica la información sobre una Sub-Representation que se encuentra dentro de una Representation.
SegmentBase	0..1	Especifica la información de Segment Base.
SegmentList	0..1	Especifica la información de Segment List.
SegmentTemplate	0..1	Especifica la información de Segment Template.
<p>Leyenda</p> <p>Para los atributos: OB = obligatorio, OP: Opcional, OPD= opcional con valor por defecto, COB=condicionalmente obligatorio</p> <p>Para los elementos: <mínimo>...<máximo> (N = ilimitado)</p> <p>Los elementos aparecen en negrita y los atributos van precedidos de @.</p> <p>Las listas de elementos y atributos aparecen en negrita y cursiva.</p>		

3.7.1.4 Segment

Una Representation está formada por varios Segments. Los Segments son la última subdivisión del archivo MPD, representan cada una de las partes en las que se ha dividido un recurso multimedia concreto. Contienen una dirección URL con la que se permite al usuario acceder al fragmento que dicho Segment representa.

La Representation en la que nos encontremos tiene asignado lo que se conoce como Segment Information, que nos proporciona información en lo referente a la ubicación, disponibilidad y las propiedades de cada uno de los Segments que forman la Representation. Esta información se puede representar utilizando alguna de las alternativas Segment Base, Segment List o Segment Template.

Segment Base

En el caso de que la Representation solamente albergue un único elemento es suficiente con utilizar este tipo de Segment. En el caso opuesto este no sería un tipo válido y habría que utilizar Segment List o bien Segment Template. De haber más de uno la Representation tendrá un atributo para indicar la duración(@duration).



Segment List

Este es un elemento que a su vez está compuesto por uno o más elementos Segment List. Por cada elemento Segment List tendremos una lista de elementos SegmentURL, estos últimos contienen las direcciones de los segmentos en los que está dividido el recurso multimedia, además pueden contener un rango de bytes y/o un Index Segment.

Segment Template

Este elemento utiliza identificadores específicos que a los que posteriormente se les asignan valores de forma dinámica con el objetivo de crear una lista de segmentos.

Los identificadores de los atributos y el valor que deben tomar son los siguientes:

- RepresentationID: @id de la Representation actual
- Number: número del Segment correspondiente
- Bandwidth: @bandwidth de la Representation actual
- Time: @t del elemento para el Segment que se está utilizando

Destacar que Number y Time son excluyentes, por lo que no se pueden utilizar ambos a la vez.

Un ejemplo del elemento Segment Template podría ser este:

```
<SegmentTemplate
  timescale="1000" duration="10000"
  media="mp4-main-ogop-$$$RepresentationID$$$Number$.m4s"
  startNumber="1"
  initialization="mp4-main-ogop-mpd-V-BS_init.mp4"/>
<Representation
  id="h264ogop_low" mimeType="video/mp4"
  codecs="avc1.64000d" width="320" height="180"
  frameRate="25" sar="1:1" startWithSAP="3"
  bandwidth="43921">
</Representation>
<Representation
  id="h264ogop_mid" mimeType="video/mp4"
  codecs="avc1.64001e" width="640" height="360"
  frameRate="25" sar="1:1" startWithSAP="3"
  bandwidth="163912">
</Representation>
. . .
```

Figura 6: Ejemplo de elemento Segment Template

Segment Information

Segment Information nos aporta la información relacionada principalmente con los elementos Initialization Segment, Index Segment y Bitstream Switching Segment, además, nos aporta las direcciones URL de cada segmento y, si se trata de streaming de contenido en vivo, su disponibilidad.

3.7.1.5 Tipos de Segments

Una vez conocidas las diferentes alternativas en la representación de la información de los segmentos, a continuación, vamos a mencionar los diferentes tipos de segmentos y a comentar la información que nos proporciona cada uno de ellos.

Initialization Segment

Aporta la información necesaria para que se puedan inicializar los segmentos de una representación concreta.

Tanto Segment Base como Segment List o Segment Template pueden proporcionar el Initialization Segment Information de dos formas diferentes, o utilizando el elemento Initialization Segment o a través del atributo @initialization. En el caso de que esta información no se tratase todos los segmentos de la Representation se iniciarían de forma automática.

Media Segment

En este tipo de segmentos en el que se encuentra encapsulado el contenido multimedia, es decir, cada parte de un vídeo que deseamos reproducir está en un Media Segment. Por cada parte que contenga, este segmento ha de tener un Stream Acces Point(SAP), este contiene la información necesaria para reproducir sin problemas el recurso multimedia.

Se puede dividir un Media Segment en varios subsegmentos a través de un Segment Index, según el formato empleado se puede incluir este último en el Media Segment o es necesaria la existencia de un Index Segment.

Index Segment

Guarda la información necesaria para ordenar los Media Segments. En caso de que el formato así lo requiera la existencia de Index Segment es obligatoria.

Bitstream Switching Segment

Contiene la información necesaria para poder moverse a la Representation que el mismo segmento está asignado.

4. Implementación

En este capítulo se va a preparar un escenario típico de consumo de un recurso multimedia vía streaming bajo demanda haciendo uso del protocolo MPEG-DASH. Contaremos con un servidor origen, en el que se almacenará el contenido necesario para llevar a cabo el streaming y un cliente donde reproducir tal recurso. Además, se aplicarán al ancho de banda disponible las condiciones de las redes móviles mencionadas anteriormente para dotar al escenario del mayor realismo posible.

4.1 Escenario

Nuestro escenario va a contar con tres elementos a configurar: el servidor HTTP, el cliente y el router. Al estar simulando un escenario en con redes móviles, todos nuestros equipos estarán conectados entre sí vía WiFi en un entorno sin interferencias.



Figura 7: Esquema del escenario a implementar

Además de los equipos necesitaremos una serie de aplicaciones para simular el entorno deseado, así como software específico que nos permita codificar el audio y el vídeo que vayamos a utilizar, y hacerlo compatible con el formato que ha de tener un contenido DASH.

4.1.1 Servidor

Aprovechando que contamos con un ordenador que tiene como sistema operativo Windows 10 Home, explotaremos el servicio IIS(Internet Information Services) que incluyen los sistemas operativos de Microsoft desde la versión Windows XP. Gracias a dicho servicio este ordenador actuará como nuestro servidor, almacenando el contenido multimedia que vamos a transmitir en todas sus codificaciones.

Implementación y evaluación de MPEG-DASH bajo las características de redes móviles.

El servicio IIS no viene por defecto instalado, por lo que nuestro primer paso será activarlo. Para ello tendremos que acceder al Panel de control de nuestro equipo, entrar en Programas y características y hacer clic en Activar o desactivar las características de Windows. Nos aparecerá una lista en la que debemos buscar Internet Information Services, hay que marcar tal opción.

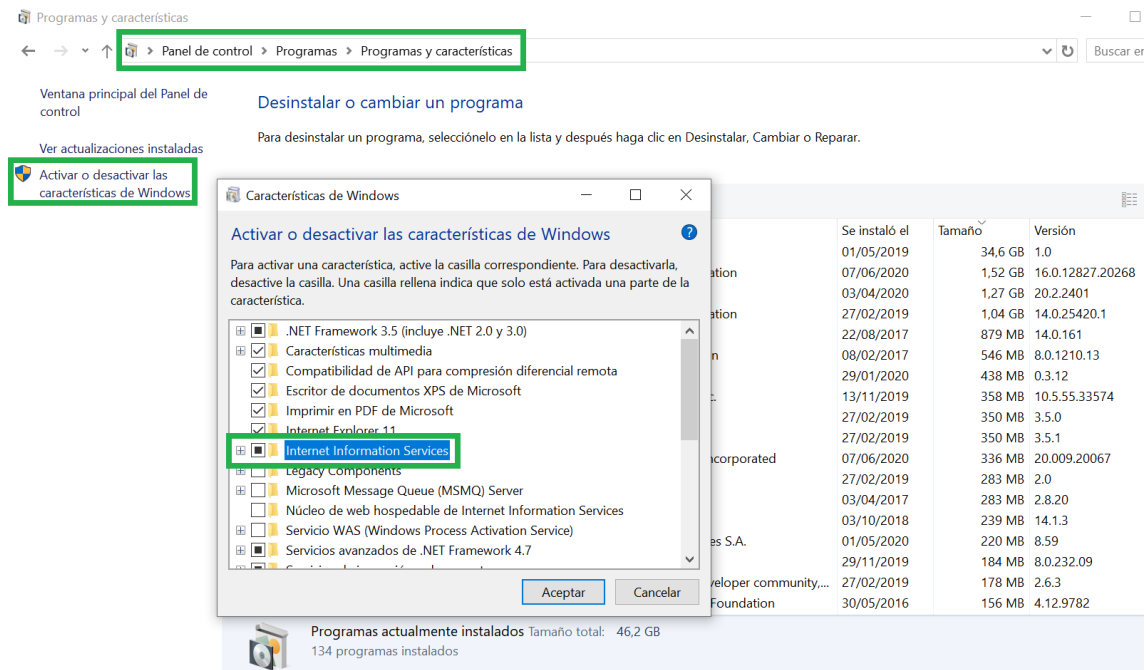


Figura 8: Activar servicio IIS

Esta activación provoca que se genere el directorio donde se almacenará toda la información del servidor en la ruta C:\inetpub\wwwroot\. En este punto de la preparación del entorno solo se puede acceder al servidor en nuestra red local introduciendo la dirección IP privada que tenga el equipo en la barra de direcciones de nuestro navegador.

Nuestro servidor ya está montado, pero si tratásemos de reproducir un contenido DASH en él no podríamos. Primero hay que configurarlo para que pueda leer el manifiesto o fichero con extensión .mpd que habremos de generar. Esta configuración se realiza a través del Administrador de IIS que tiene nuestro sistema operativo. Dentro de la configuración hay que añadir un nuevo tipo MIME indicando como extensión .mpd y el tipo MIME dash.

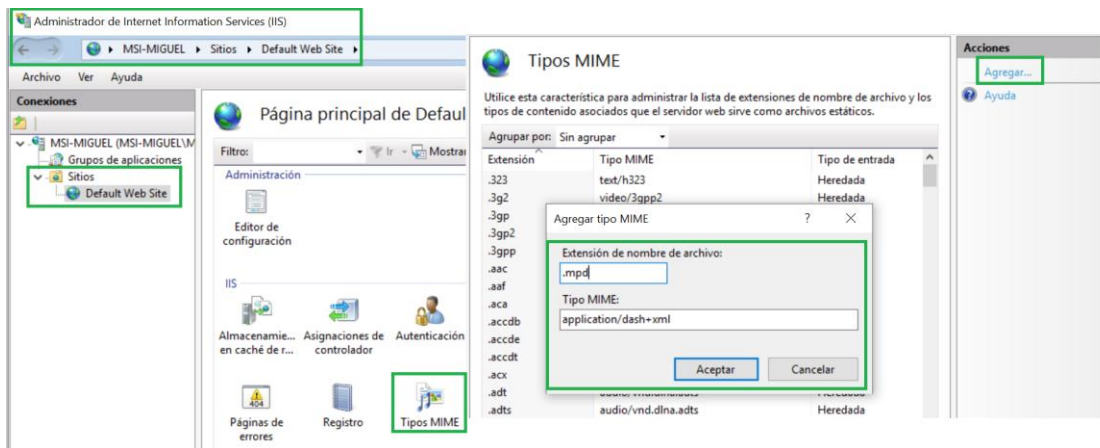


Figura 9: Configuración IIS

Con esto ya hemos terminado de configurar nuestro servidor para que sea capaz de interpretar correctamente los archivos en formato DASH. Lo siguiente es codificar y preparar el archivo que queremos utilizar para el streaming en varios formatos diferentes. Para ello utilizaremos el programa FFmpeg que podemos descargar de forma completamente gratuita desde su propia web [13].

Una vez lo tengamos descargado tendremos un fichero comprimido en formato .zip. Los archivos que contiene es recomendable extraerlos en una carpeta situada en una ruta de fácil acceso, pues esto hará más sencilla la configuración necesaria para su uso. Como este programa se ejecuta por línea de comandos es necesario añadirlos a las variables de entorno del equipo en el que vayamos a realizar la codificación, en este caso en el mismo que utilizamos como servidor. Esto se lleva a cabo desde la Configuración avanzada del sistema en la opción Variables de entorno. En ella hay que añadir la ruta donde hayamos colocado los ficheros descargados a la variable de nombre Path.

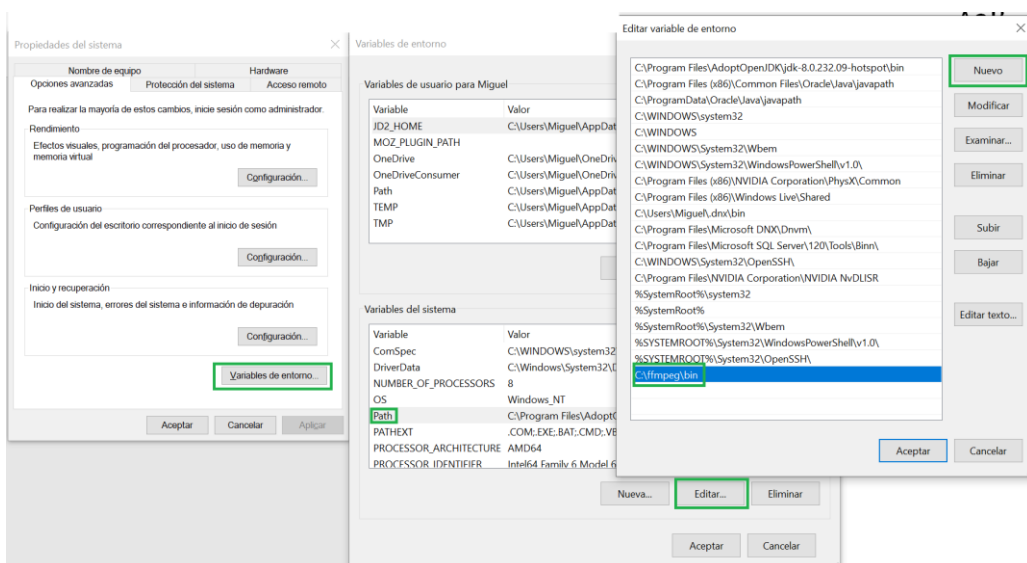


Figura 10: Configuración FFmpeg

Implementación y evaluación de MPEG-DASH bajo las características de redes móviles.

Con esto FFmpeg ya está listo para ser usado a través de nuestro terminal:

```
C:\Users\Miguel>ffmpeg -h
ffmpeg version 4.2.3 Copyright (c) 2000-2020 the FFmpeg developers
built with gcc 9.3.1 (GCC) 20200523
```

Figura 11: FFmpeg en terminal

Ya podemos codificar nuestro audio y vídeo en el formato que queramos, pero seguimos necesitando poder aplicarle al conjunto de archivos un formato DASH. La herramienta, también gratuita, que vamos a utilizar para lograr este objetivo es MP4box. Para poder utilizarla es necesario instalar el framework GPAC, que nos proporcionará este componente entre aquellos que incluye su instalación.

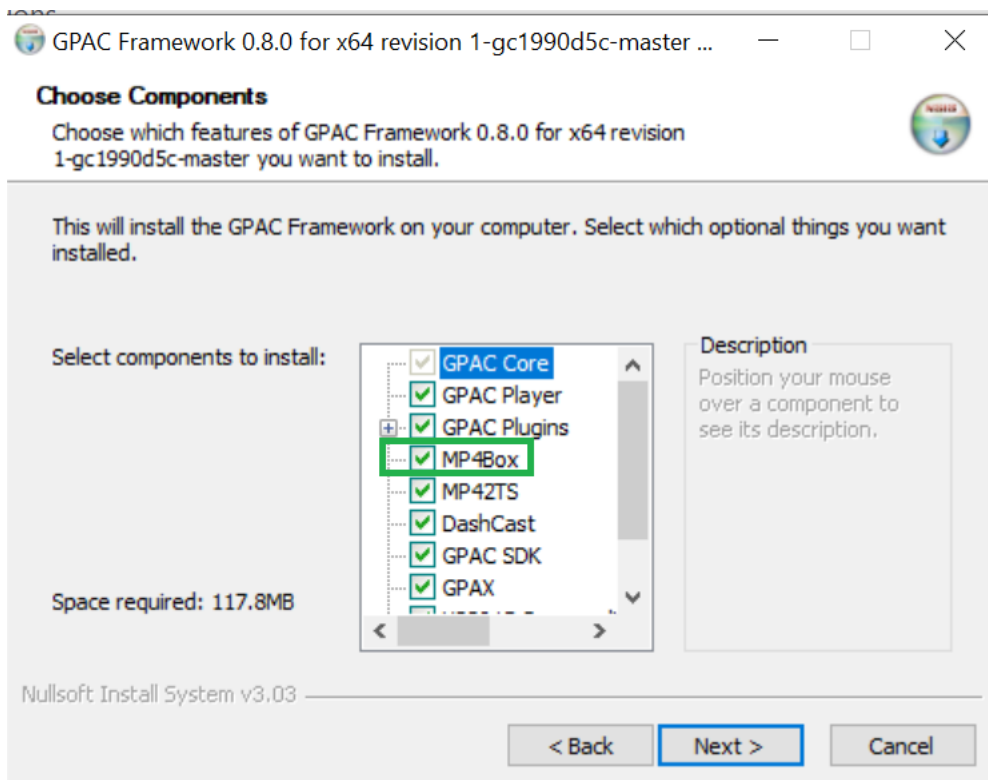


Figura 12: Instalación framework GPAC

Ya tenemos todo el software necesario en nuestro equipo, ahora tenemos que preparar los archivos que utilizará el servidor para hacer posible el streaming. Para ellos vamos a utilizar un vídeo original que tiene una resolución de 1920x1080 y cuenta con una tasa de bits de 2879 kbps.

Haciendo uso de la herramienta FFmpeg vamos a convertir el vídeo original en una serie de vídeos donde la resolución y la tasa de bits son menores. Esto se realizaría mediante línea de comandos y un ejemplo de uso podría ser la siguiente instrucción.

```
> ffmpeg -i original.mp4 -s 1280x720 -c:v libx264 -b:v 1500k -g 90 -an original_video_1280x720_1500k.mp4
```

Cada parámetro significa lo siguiente:

“**-i original.mp4**” es el archivo de entrada

“**-s**” es la resolución a la que se va a codificar el archivo

“**-c: v**” indica el códec de video que va a usar

“**-b: v**” es la tasa de bits a la que queremos codificar el video

“**-g 90**” le dice a ffmpeg que queremos un intervalo de fotogramas clave (longitud de GOP) de 90

“**-an**” para no codificar el audio

Por último, escribimos el nombre con el que queremos que se guarde nuestro nuevo archivo.

Es recomendable dar a los archivos generados un nombre con el que se les pueda identificar fácilmente en el análisis de la red.

Cómo resultado hemos obtenido una versión del vídeo original sin audio, en formato mp4, con una resolución de 1280x720 y una tasa de bits de 1500 kbps. Todo esto codificado en H264.

Esto lo vamos a hacer tantas veces como archivos diferentes queramos tener, para poder realizar un buen estudio del funcionamiento de MPEG-DASH es recomendable utilizar una cantidad no muy elevada que permita ver claramente los cambios en la calidad del vídeo en función del ancho de banda disponible.

En lo referente al audio, solo vamos a crear un archivo de audio utilizando el siguiente comando:

```
> ffmpeg -i original.mp4 -c aac -b 128k -vn original_audio_128k.mp4
```

Cada parámetro significa lo siguiente:

“**-i original.mp4**” es el archivo de entrada

“**-c**” es el codec de audio

“**-b**” la tasa de bits del audio

“**-vn**” para no codificar video

Al igual que en el caso del vídeo, al final del comando introducimos el nombre con el que queremos que quede guardado el archivo de audio.

Con este comando hemos obtenido un fichero de audio que contiene únicamente el audio de vídeo original, con una tasa de bits de 124kbps y codificado en AAC.

Dado su tamaño habitual los ficheros de audio no suponen una gran carga en la red. Puesto que también es complicado detectar grandes cambios en la calidad del audio solo vamos a contar con este fichero para todas las pruebas.

Tras finalizar este proceso tendremos en el directorio en el que hayamos llevado a cabo todos los comandos una serie de archivos de vídeo de diferentes calidades y tamaños:

original	10/06/2020 18:13	Archivo MP4	110.686 KB
original_video_1280x720_2500k	10/06/2020 19:24	Archivo MP4	101.690 KB
original_video_1280x720_1500k	10/06/2020 19:23	Archivo MP4	61.084 KB
original_video_640x360_1000k	10/06/2020 19:20	Archivo MP4	40.647 KB
original_video_320x180_500k	10/06/2020 19:17	Archivo MP4	20.216 KB
original_video_160x90_250k	10/06/2020 19:16	Archivo MP4	10.089 KB
original_audio_128k	10/06/2020 19:28	Archivo MP4	4.985 KB

Figura 13: Resultado codificación con FFmpeg

Estos archivos todavía no pueden ser utilizados con DASH, solamente son copias de un vídeo original, pero con diferentes calidades. Para que el protocolo DASH las pueda usar es necesario que conozca donde están y como solicitarlas en función de las necesidades del momento. Tareas que se encarga el ya conocido manifiesto o fichero con extensión .mpd. Este es el fichero que contiene toda la información necesaria para que el cliente DASH pueda reproducir el contenido solicitando la versión que más se adecua a la situación de la red, provocando la sensación al observador de que se trata de un único archivo continuo de vídeo.

Para darle a estas versiones de un mismo vídeo el formato que los haga compatibles con el cliente DASH y para generar el archivo mpd, vamos a hacer uso de la herramienta mp4box. Al igual que FFmpeg se utiliza por línea de comandos, para nuestro escenario hemos llevado a cabo la siguiente instrucción:

```
> mp4box -dash 5000 -rap -profile dashavc264:onDemand -out  
manifiesto2s.mpd -frag 2000 original_audio_128k.mp4  
original_video_160x90_250k.mp4 original_video_320x180_500k.mp4  
original_video_640x360_1000k.mp4 original_video_1280x720_1500k.mp4
```

Cada parámetro significa lo siguiente:

“**-dash 5000**” divide los archivos de entrada en segmentos de 5 segundos

“**-rap**” obliga a los segmentos a comenzar con puntos de acceso aleatorios. En otras palabras, permite la búsqueda del video.

“**-profile dashavc264: onDemand**” usa el perfil bajo demanda visto en el apartado 2.7.7.2.4.

“**-out manifiesto2s.mpd**” el nombre del archivo mpd de salida

“**-frag 2000**” establece la longitud del fragmento en 2 segundos. Siempre menor que el valor especificado en “**-dash**”

A este comando hay que añadirle a continuación el archivo de audio y los vídeos que queremos utilizar en esta solución DASH.

Como resultado obtendremos un fichero .mpd junto con tantos vídeos como hayamos indicando en el comando, en nuestro caso este sería el resultado:







 original_video_1280x720_1500k_dashinit	10/06/2020 22:56	Archivo MP4	61.082 KB	00:05:13
 original_video_640x360_1000k_dashinit	10/06/2020 22:56	Archivo MP4	40.648 KB	00:05:13
 original_video_320x180_500k_dashinit	10/06/2020 22:56	Archivo MP4	20.224 KB	00:05:13
 original_video_160x90_250k_dashinit	10/06/2020 22:56	Archivo MP4	10.103 KB	00:05:13
 original_audio_128k_dashinit	10/06/2020 22:56	Archivo MP4	5.010 KB	00:05:13
 manifiesto2s.mpd	10/06/2020 22:56	Archivo MPD	3 KB	

Figura 14: Resultado codificación con mp4box

Estos son los ficheros que debemos colocar en la carpeta de nuestro servidor para así tener preparado nuestro contenido perfectamente reproducible mediante DASH.

4.1.2 Ciente

Existen diferentes alternativas que podemos utilizar como cliente DASH. Podemos utilizar un navegador de internet tradicional (Chrome, Firefox, etc) o un reproductor compatible con DASH. Si optamos por utilizar un navegador, deberemos configurarlo para que sea capaz de reconocer un fichero .mpd e interpretarlo correctamente para que pueda llevar a cabo la reproducción de nuestro contenido DASH.

Para configurar el navegador podemos elegir alguna de las siguientes alternativas:

- Instalar una extensión que, al introducir en la barra de dirección la url de un fichero .mpd nos lleve a un reproductor DASH que reproduzca el contenido señalado en la dirección. Esta extensión se llama Native MPEG-Dash + HLS Playback y se puede obtener de forma completamente gratuita.
- Usar alguno de los reproductores DASH que se ofrecen también de forma gratuita en internet, en la dirección <http://reference.dashif.org/dash.js/> podemos encontrar varios de ellos.
- Añadir un reproductor DASH min.js a nuestro servidor local y llamarlo desde el archivo HTML del servidor.

La segunda opción es utilizar un reproductor multimedia que pueda reproducir archivos en red y además sepa leer un fichero .mpd, es decir, que sea compatible con DASH.

Una opción puede ser el reproductor VLC, que es compatible con una amplísima variedad de códecs y es capaz de reproducir archivos a partir de una dirección URL, ya sea bajo demanda o en vivo.

Usar el reproductor VLC es la opción más sencilla, pero vamos a optar por instalar la extensión Native MPEG-Dash + HLS Playback en nuestro navegador Google Chrome. El motivo principal es que dicho navegador nos proporciona en sus herramientas para capturar tráfico de red la opción de ver que se está reproduciendo y solicitando en cada momento mientras vemos como avanza el flujo de vídeo. Acceder a estas herramientas es tan sencillo como apretar la tecla F12 en la web que deseemos analizar.

Con esto podemos comenzar el streaming de nuestro contenido en formato DASH, pero para dotar de más realismo a nuestro entorno vamos a hacer uso de la herramienta NetLimiter, que nos va a permitir limitar la velocidad de la red para así poder estudiar mejor el comportamiento de DASH ante los cambios en la misma.

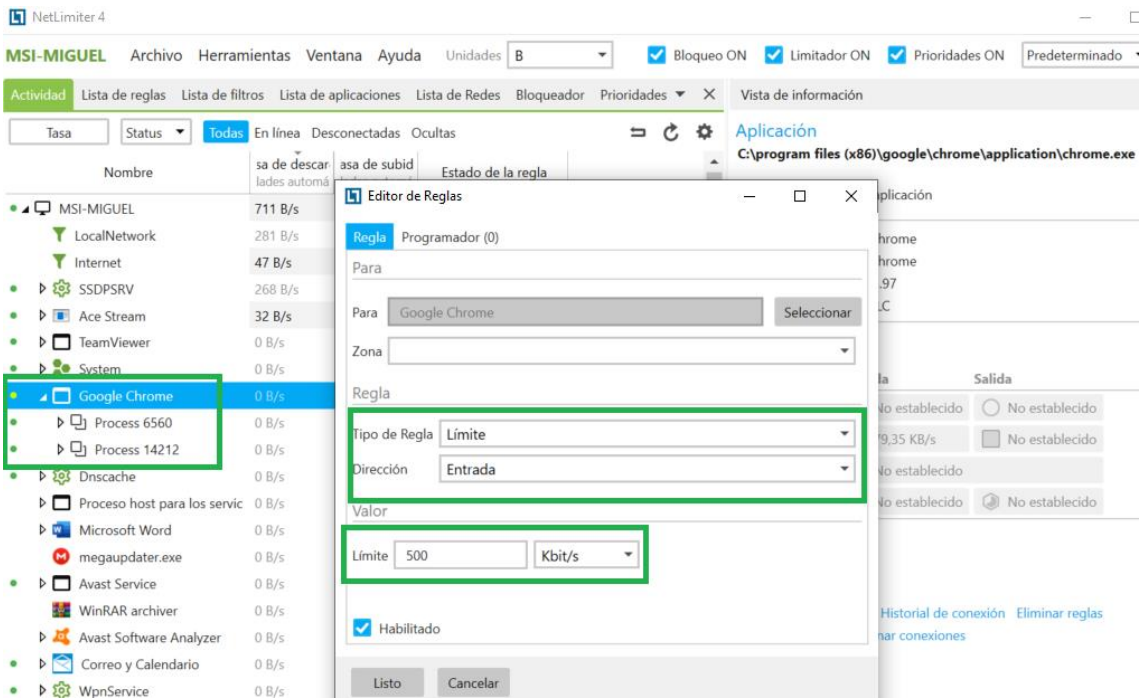


Figura 15: Uso de NetLimiter

Hay que ir con ojo a la hora de añadir la regla a nuestro navegador, pues por defecto el límite se aplica en KBps en vez de en Kbps, esta equivocación provocaría que nuestro entorno no reflejase el comportamiento que queremos simular.

Como tratamos de recrear un entorno bajo las condiciones de redes móviles, en las diferentes simulaciones aplicaremos cambios en la red entre los intervalos de máximos y mínimos de cada uno de los estándares mencionados.

4.1.3 Router

Con nuestra actual configuración solo podemos hacer uso del entorno a nivel local, por lo que vamos a crear unas reglas en la configuración NAT de nuestro router para que, utilizando una serie de puertos concretos y la IP pública de nuestro router, podamos acceder a nuestro servidor desde cualquier parte.

Por defecto el servidor creado por IIS utiliza el puerto 80, así que solamente debemos indicar que las peticiones que lleguen a nuestro router en un intervalo de puertos libres que elijamos, en este caso las que lleguen entre los puertos 3001 y 3500, ambos inclusive, se redirijan al puerto 80 del equipo en el que hayamos preparado el servidor, en nuestro caso es el que cuenta con la dirección IP privada 192.168.1.139.



Implementación y evaluación de MPEG-DASH bajo las características de redes móviles.

Configuración de NAT/PAT/CGNAT

Estas normas son necesarias para autorizar una conexión remota desde Internet que llegue a un dispositivo específico de tu red LAN. También puedes definir los puertos(s) que utilizará esta comunicación.

Para crear la regla NAT debes introducir la IPv4 asignada a tu dispositivo en la LAN. Para saber cuál es puedes consultar el listado de IPs asignadas en la pestaña "DHCP" de esta página.



Atención: Asegúrate de que no has filtrado estos puertos en el firewall.

Personalizar reglas							
estado	aplicación / servicio	puerto interno	puerto externo	protocolo	IPv4 del dispositivo	Aceptar propuesta de puerto PCP	
	FTP Server	21	21	TCP		<input checked="" type="checkbox"/>	añadir
	Web Server (HTTP)	80-579	3001-3500	TCP	192.168.1.139	<input checked="" type="checkbox"/>	delete

Figura 16: Configuración NAT router

Finalizado esto ya hemos completado todos los pasos para que sea posible llevar a cabo un streaming haciendo uso del protocolo MPEG-DASH.

5. Pruebas y evaluación

Habiendo configurado previamente nuestro escenario, vamos a llevar a cabo pruebas bajo las condiciones de los estándares 3G y 4G. Para ello, hemos codificado en formato compatible con DASH un vídeo original en diferentes resoluciones. Esto lo hemos hecho con segmentos de 4,8 y 16 segundos, de esta forma lograremos ver como la longitud de los segmentos puede ser más o menos relevante en función del estado de la red.

Iniciaremos la reproducción con una velocidad máxima limitada al máximo real que puede ofrecer cada uno de los estándares, tras unos segundos, cuando el flujo de vídeo sea estable, reduciremos la velocidad de la red acercándonos al mínimo de la red, y unos segundos después volveremos al máximo. De este modo lograremos simular una pérdida de cobertura en la red móvil y ver como actúa DASH en esas condiciones.

Como herramientas utilizaremos la herramienta de análisis de red que nos ofrece Chrome para ver que archivo está solicitando en cada momento y la versión 3.1.1 del reproductor gratuito <https://reference.dashif.org/dash.js/> que nos permite ver las variaciones del buffer del vídeo y de la calidad del mismo.

Estándar	Velocidad mínima	Velocidad máxima real
3G	0.2 Mbps	2 Mbps
4G	4Mbps	40 Mbps

5.1 3G

Para el estándar 3G vamos a simular tres sesiones de streaming, para cada uno hay una preparación diferente del formato DASH. En los tres casos el procedimiento va a ser el mismo ya explicado. Utilizando la herramienta para desarrolladores de Google Chrome, analizaremos como es el inicio de la conexión en cada caso, viendo lo que tarda en estabilizarse para, posteriormente, al disminuir el límite de la velocidad, ver cómo y con qué velocidad se vuelve a estabilizar en una calidad menor. Luego volveremos al límite original y evaluaremos la capacidad de DASH para recuperar la calidad óptima.



5.1.1 3G segmentos de cuatro segundos

En esta primera prueba, en la que aplicamos los límites marcados por el estándar 3G, hemos preparado nuestros archivos para que los segmentos tengan una duración de cuatro segundos.























Name	Size	Time
 dash.all.min.js	137 kB	516 ms
 manifiesto4s.mpd	3.0 kB	68 ms
 original_video_160x90_250k_dashinit.mp4	1.3 kB	28 ms
 original_video_320x180_500k_dashinit.mp4	1.3 kB	26 ms
 original_video_640x360_1000k_dashinit.mp4	1.3 kB	24 ms
 original_video_1280x720_2500k_dashinit.mp4	1.3 kB	42 ms
 original_audio_128k_dashinit.mp4	1.3 kB	84 ms
 original_video_640x360_1000k_dashinit.mp4	1.4 kB	27 ms
 original_audio_128k_dashinit.mp4	1.4 kB	24 ms
 original_video_640x360_1000k_dashinit.mp4	352 kB	3.03 s
 original_audio_128k_dashinit.mp4	67.9 kB	837 ms
 original_video_160x90_250k_dashinit.mp4	1.4 kB	34 ms
 original_video_160x90_250k_dashinit.mp4	81.4 kB	852 ms
 original_audio_128k_dashinit.mp4	65.4 kB	676 ms
 original_video_160x90_250k_dashinit.mp4	125 kB	557 ms
 original_video_640x360_1000k_dashinit.mp4	523 kB	2.06 s
 original_video_320x180_500k_dashinit.mp4	1.4 kB	19 ms
 original_video_320x180_500k_dashinit.mp4	255 kB	1.04 s
 original_audio_128k_dashinit.mp4	65.5 kB	515 ms
 original_video_320x180_500k_dashinit.mp4	301 kB	1.41 s
 original_video_320x180_500k_dashinit.mp4	358 kB	1.71 s
 original_audio_128k_dashinit.mp4	65.8 kB	549 ms

Figura 17: 3G cuatro segundos. Inicio reproducción DASH

Vemos como, en primer lugar, carga el reproductor y el archivo .mpd, además de ubicar las cuatro versiones de nuestro vídeo original. Como se puede observar, el protocolo está todavía tratando de detectar cual es la mejor versión para el estado actual. Esto, unido a que son segmentos de solo cuatro segundos, hace que el streaming esté cambiando constantemente de calidad.

Name	Size	Time
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	177 kB	695 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	384 kB	1.50 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	696 kB	3.03 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	533 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	653 kB	2.66 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	525 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	241 kB	1.41 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	549 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	331 kB	1.60 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	549 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	524 kB	2.34 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	535 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	610 kB	2.71 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	546 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.0 MB	4.02 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	774 kB	3.35 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.6 kB	528 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	217 kB	1.12 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.7 kB	526 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	450 kB	2.06 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	555 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	120 ms

Figura 18: 3G cuatro segundos. Estabilización de la reproducción DASH

Tras unos segundos, el streaming se estabiliza, el protocolo ha tomado como mejor opción el streaming que tiene una tasa de bits de 1000Kbps (figura 46), es lo ideal pues contamos con una velocidad limitada a 2Kbps y la siguiente versión del archivo tiene una tasa de 2500Kbps (figura 47).

Name	Size	Time
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.3 kB	1.93 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	13.1 kB	630 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	251 kB	6.74 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	856 kB	26.33 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	66.0 kB	3.62 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	3.55 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	177 kB	4.68 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	249 kB	8.62 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	3.72 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	151 kB	4.02 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	140 kB	5.52 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	3.59 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	70.3 kB	3.29 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	3.51 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	51.8 kB	2.83 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	3.54 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	151 kB	6.61 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	3.56 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	3.58 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	162 kB	6.03 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	65.7 kB	2.71 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	32.1 kB	1.82 s

Figura 19: 3G cuatro segundos. Reducción del límite de velocidad

Al reducir el límite se estaba descargando un segmento del vídeo a 1000 Kbps, esta descarga retrasa mucho el streaming y el buffer del vídeo se vacía rápidamente. Tras finalizar la descarga de 26.33 segundos el protocolo rápidamente solicita los segmentos de menos calidad, estabilizando así de nuevo la reproducción. El tamaño de los segmentos favorece que esto se produzca de manera rápida y se establezca la reproducción con una tasa de bits de 250 Kbps (figura 44).

Name	Size	Time
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	59.1 kB	597 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	59.4 kB	575 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	1.36 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	61.6 kB	457 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	501 kB	1.65 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	622 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.1 MB	5.01 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	298 kB	1.37 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	292 kB	1.32 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	237 kB	1.06 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	236 kB	1.39 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	613 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	671 kB	3.37 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	631 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	758 kB	3.76 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	638 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.3 MB	6.36 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.3 kB	624 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	205 kB	1.24 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	618 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	696 kB	3.46 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.3 kB	606 ms

Figura 20: 3G cuatro segundos. Vuelta al límite de velocidad inicial

Volvemos a establecer el límite como lo estaba al principio del streaming y vemos como la reproducción se estabiliza rápidamente en la calidad que mejor se adapta a las características de la red. En la figura 21 se puede apreciar claramente como ha ido variando la calidad de la imagen.

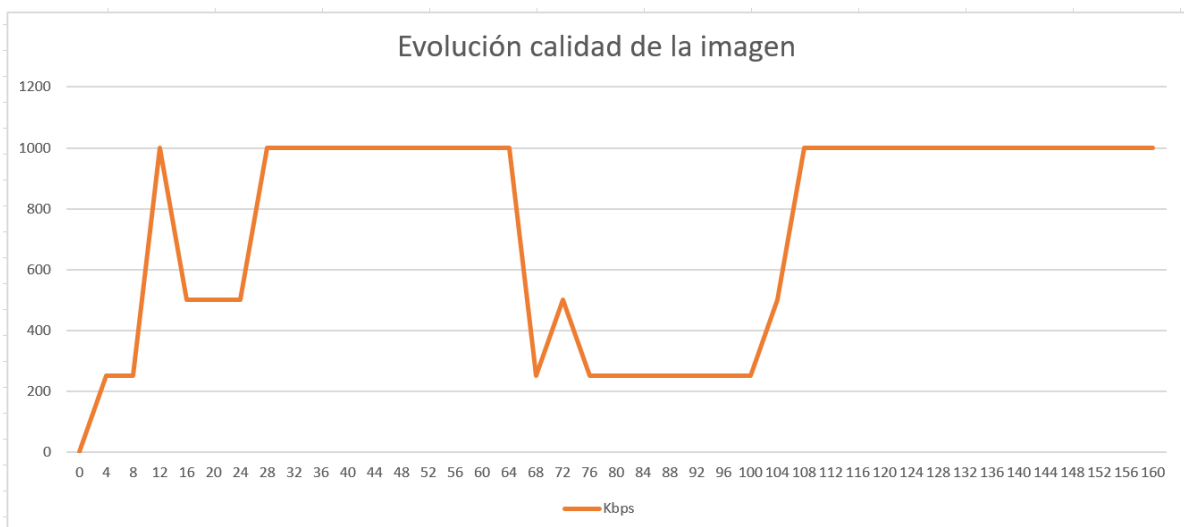


Figura 21: 3G cuatro segundos. Gráfica calidad de imagen

5.1.2 3G segmentos de ocho segundos

En segundo lugar, para las pruebas bajo las condiciones del estándar 3G, hemos preparado nuestros archivos para que los segmentos tengan una duración de ocho segundos.























Name	Size	Time
 dash.all.min.js	137 kB	581 ms
 manifiesto8s.mpd	3.0 kB	20 ms
 original_video_160x90_250k_dashinit.mp4	683 B	21 ms
 original_video_320x180_500k_dashinit.mp4	683 B	25 ms
 original_video_640x360_1000k_dashinit.mp4	683 B	36 ms
 original_video_1280x720_2500k_dashinit.mp4	684 B	34 ms
 original_audio_128k_dashinit.mp4	789 B	68 ms
 original_video_640x360_1000k_dashinit.mp4	1.4 kB	21 ms
 original_audio_128k_dashinit.mp4	1.4 kB	21 ms
 original_video_640x360_1000k_dashinit.mp4	1.1 MB	7.16 s
 original_audio_128k_dashinit.mp4	133 kB	1.64 s
 original_audio_128k_dashinit.mp4	131 kB	1.58 s
 original_video_320x180_500k_dashinit.mp4	1.4 kB	23 ms
 original_video_320x180_500k_dashinit.mp4	588 kB	2.30 s
 original_video_320x180_500k_dashinit.mp4	789 kB	3.14 s
 original_video_640x360_1000k_dashinit.mp4	1.7 MB	7.11 s
 original_audio_128k_dashinit.mp4	131 kB	1.07 s
 original_video_320x180_500k_dashinit.mp4	689 kB	2.73 s
 original_video_320x180_500k_dashinit.mp4	527 kB	2.64 s
 original_audio_128k_dashinit.mp4	131 kB	1.08 s
 original_video_320x180_500k_dashinit.mp4	709 kB	1.85 s
 original_audio_128k_dashinit.mp4	131 kB	380 ms

Figura 22: 3G ocho segundos. Inicio reproducción DASH

De nuevo vemos como nada más comenzar se descarga el archivo .mpd y se localizan todas las versiones del vídeo. Tras ello se van probando diferentes calidades para encontrar la óptima. Cabe destacar que al aumentar el tamaño de los segmentos aumentan los tiempos de descarga y cuesta más estabilizar el vídeo.

Name	Size	Time
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.05 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	289 kB	1.09 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	562 kB	2.24 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	2.6 MB	10.96 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	1.04 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.1 MB	4.76 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.07 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.4 MB	5.97 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.09 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	212 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	521 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	823 kB	2.32 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	279 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	844 kB	2.41 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	412 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.1 MB	3.59 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	104 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	513 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	2.4 MB	8.43 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	350 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	706 kB	2.83 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	313 ms

Figura 23: 3G ocho segundos. Estabilización de la reproducción DASH

Pese a haber tardado un poco más, el streaming se ha estabilizado de la forma esperada, tomando como opción habitual los segmentos del vídeo con una calidad de 1000Kbps (figura 46). Se puede observar cómo, a excepción de algún caso puntual, todos los tiempos de descarga son inferiores a la duración del segmento. Esto garantiza la continuidad del streaming, pues el flujo será constante y el buffer estará lleno a unos niveles elevados.

Name	Size	Time
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	2.4 MB	8.43 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	350 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	706 kB	2.83 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	313 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	786 kB	35.53 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	10.52 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	428 kB	22.20 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	10.45 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	189 kB	7.55 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	396 kB	13.49 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	572 kB	18.67 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	6.96 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	491 kB	20.12 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	7.04 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	7.00 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	251 kB	6.70 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	30.4 kB	346 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	193 kB	10.18 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	5.08 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	7.05 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	233 kB	10.72 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	7.03 s

Figura 24: 3G ocho segundos. Reducción del límite de velocidad

Del mismo modo que ha tardado más en estabilizar la reproducción, el tamaño de los segmentos provoca que el efecto de la reducción de la velocidad tarde más en apreciarse, pues al estar el búfer en tan buenos niveles con segmentos de ocho segundos puede seguir aguantando la calidad más tiempo. Aun así llega un punto en el que ya no es viable seguir manteniendo la calidad del vídeo y se reduce para que el streaming pueda continuar.

Name	Size	Time
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	216 kB	7.16 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	509 kB	15.69 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	6.93 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	2.99 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	296 kB	1.14 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	480 kB	2.43 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.08 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	629 kB	2.47 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	166 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	503 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.0 MB	3.20 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.7 MB	6.89 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	57 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	562 kB	1.28 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	306 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	2.6 MB	9.48 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	428 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	535 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.1 MB	3.73 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	225 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.4 MB	4.48 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	457 ms

Figura 25: 3G ocho segundos. Vuelta al límite de velocidad inicial

Volvemos a la calidad inicial y volvemos a ver los efectos de un tamaño de segmento algo mayor. El streaming no se recupera de una forma instantánea en la calidad óptima, pero aun así acaba haciéndolo de forma exitosa. En la figura 26 se pueden ver los cambios mencionados de forma resumida.

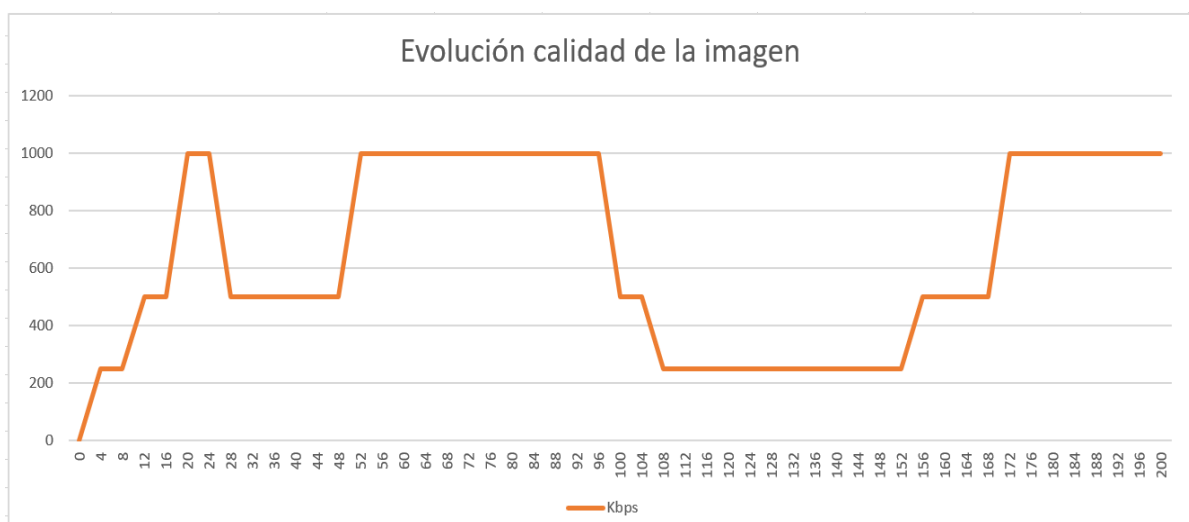


Figura 26: 3G ocho segundos. Gráfica calidad de imagen



5.1.3 3G segmentos de dieciséis segundos

Como última prueba con este estándar, vamos a realizar un análisis de cómo se comporta el protocolo DASH con segmentos de dieciséis segundos.























Name	Size	Time
 dash.all.min.js	137 kB	680 ms
 manifiesto16s.mpd	3.0 kB	23 ms
 original_video_160x90_250k_dashinit.mp4	478 B	19 ms
 original_video_320x180_500k_dashinit.mp4	479 B	21 ms
 original_video_640x360_1000k_dashinit.mp4	479 B	24 ms
 original_video_1280x720_2500k_dashinit.mp4	479 B	48 ms
 original_audio_128k_dashinit.mp4	550 B	46 ms
 original_video_640x360_1000k_dashinit.mp4	1.4 kB	65 ms
 original_audio_128k_dashinit.mp4	1.4 kB	63 ms
 original_audio_128k_dashinit.mp4	264 kB	3.52 s
 original_video_640x360_1000k_dashinit.mp4	3.3 MB	19.62 s
 original_audio_128k_dashinit.mp4	261 kB	3.37 s
 original_video_320x180_500k_dashinit.mp4	1.4 kB	18 ms
 original_video_320x180_500k_dashinit.mp4	1.6 MB	6.43 s
 original_video_160x90_250k_dashinit.mp4	1.4 kB	19 ms
 original_video_160x90_250k_dashinit.mp4	787 kB	3.54 s
 original_video_160x90_250k_dashinit.mp4	690 kB	3.14 s
 original_audio_128k_dashinit.mp4	262 kB	1.10 s
 original_video_320x180_500k_dashinit.mp4	1.2 MB	4.67 s
 original_video_160x90_250k_dashinit.mp4	665 kB	4.22 s
 original_audio_128k_dashinit.mp4	262 kB	2.44 s
 original_audio_128k_dashinit.mp4	261 kB	937 ms

Figura 27: 3G dieciséis segundos. Inicio reproducción DASH

Al igual que en todos los casos, lo primero que se ve es como se descarga el archivo .mpd para poder ubicar las versiones del vídeo, cosa que pasa a continuación. En este caso, al tener un tamaño de segmento tan grande, la reproducción tarda bastante en iniciar. Se puede ver como el primer archivo de vídeo tarda casi veinte segundos en descargarse.

Name	Size	Time
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.6 MB	11.48 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	3.34 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	1.4 kB	20 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	787 kB	2.59 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.2 MB	5.61 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	1.15 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.3 MB	4.74 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	683 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.04 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.5 MB	4.99 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	847 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	697 kB	1.83 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	883 kB	3.50 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	868 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.03 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	451 kB	885 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	851 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	984 kB	2.98 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	964 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	648 kB	1.62 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	629 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.03 s

Figura 28: 3G dieciséis segundos. Estabilización de la reproducción DASH

Dado el gran tamaño de los segmentos, el streaming tarda demasiado en estabilizarse. En las pruebas, tras más de dos minutos de vídeo seguimos sin ver una calidad constante en la reproducción.

Name	Size	Time
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	847 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	697 kB	1.83 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	883 kB	3.50 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	868 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.03 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	451 kB	885 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	851 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	984 kB	2.98 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	964 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	648 kB	1.62 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	629 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.03 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	847 kB	32.78 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	17.98 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	734 kB	25.59 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	14.02 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	9.15 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	449 kB	14.42 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	7.10 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	706 kB	25.26 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	13.81 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	777 kB	10.60 s

Figura 29: 3G dieciséis segundos. Reducción del límite de velocidad

Con la reducción de la velocidad el streaming se estabiliza rápidamente en la calidad más baja disponible, pues es la única que se puede permitir descargar. Al ser el tamaño de los archivos tan grandes, llegamos a observar algunos parones durante la reproducción del vídeo.

Name	Size	Time
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	7.10 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	706 kB	25.26 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	13.81 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	777 kB	10.60 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	2.80 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	646 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	1.0 MB	3.15 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	161 kB	259 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	787 kB	5.10 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	264 kB	1.35 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	2.12 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	2.08 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	690 kB	3.01 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.2 MB	3.93 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.3 MB	6.05 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	2.11 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	960 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.5 MB	5.02 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	524 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.8 MB	6.30 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	802 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.05 s

Figura 30: 3G dieciséis segundos. Vuelta al límite de velocidad inicial

Tras recuperar la capacidad de la red, DASH cambia nuevamente la calidad del vídeo. En esta ocasión el cambio ha tardado en notarse, cosa que se produce por la larga duración de los segmentos. En la figura 31 vemos como todos los cambios durante la reproducción tardan en aplicarse.

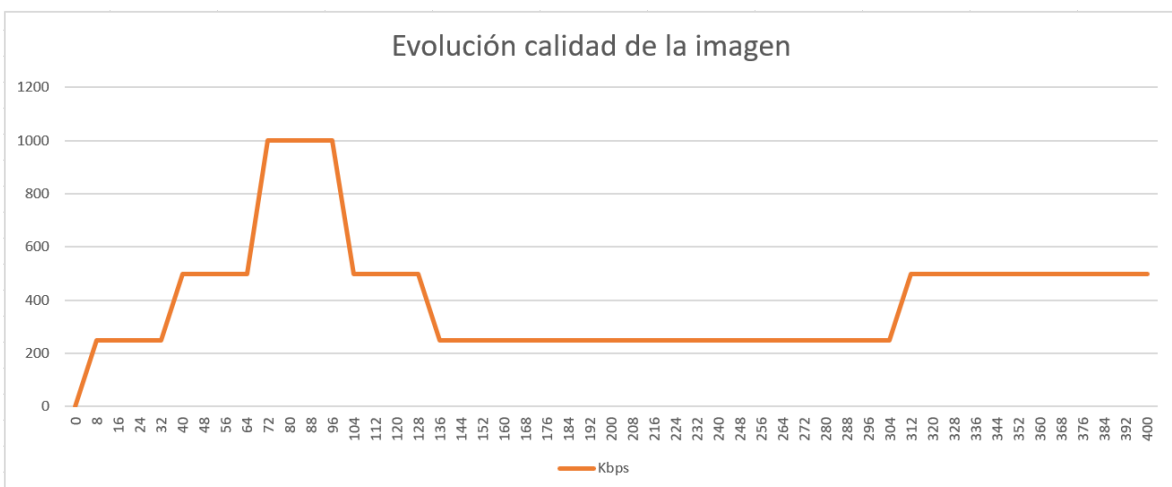


Figura 31: 3G dieciséis segundos. Gráfica calidad de imagen

5.2 4G

Para el estándar 4G vamos a simular tres streamings, del mismo modo que en las pruebas realizadas con el anterior protocolo. Cada una de las tres reproducciones será igualmente analizada con la herramienta para desarrolladores que nos ofrece Google Chrome.

5.2.1 4G segmentos de cuatro segundos

En esta primera prueba, en la que aplicamos los límites marcados por el estándar 4G, hemos preparado nuestros archivos para que los segmentos tengan una duración de cuatro segundos.













Name	Size	Time
 dash.all.min.js	137 kB	56 ms
 manifiesto4s.mpd	3.0 kB	22 ms
 original_video_160x90_250k_dashinit.mp4	1.3 kB	30 ms
 original_video_320x180_500k_dashinit.mp4	1.3 kB	39 ms
 original_video_640x360_1000k_dashinit.mp4	1.3 kB	66 ms
 original_video_1280x720_2500k_dashinit.mp4	1.3 kB	40 ms
 original_audio_128k_dashinit.mp4	1.3 kB	83 ms
 original_video_1280x720_2500k_dashinit.mp4	1.4 kB	38 ms
 original_audio_128k_dashinit.mp4	1.4 kB	34 ms
 original_video_1280x720_2500k_dashinit.mp4	902 kB	134 ms
 original_audio_128k_dashinit.mp4	67.9 kB	142 ms
 original_video_1280x720_2500k_dashinit.mp4	1.3 MB	183 ms
 original_video_1280x720_2500k_dashinit.mp4	956 kB	106 ms
 original_video_1280x720_2500k_dashinit.mp4	2.1 MB	220 ms
 original_audio_128k_dashinit.mp4	65.4 kB	51 ms
 original_audio_128k_dashinit.mp4	65.5 kB	21 ms
 original_video_1280x720_2500k_dashinit.mp4	1.6 MB	195 ms
 original_audio_128k_dashinit.mp4	65.8 kB	148 ms
 original_video_1280x720_2500k_dashinit.mp4	1.4 MB	181 ms
 original_video_1280x720_2500k_dashinit.mp4	1.9 MB	216 ms
 original_video_1280x720_2500k_dashinit.mp4	357 kB	72 ms
 original_audio_128k_dashinit.mp4	65.7 kB	94 ms
 original_video_1280x720_2500k_dashinit.mp4	618 kB	115 ms
 original_audio_128k_dashinit.mp4	65.3 kB	74 ms
 original_audio_128k_dashinit.mp4	65.5 kB	22 ms

Figura 32: 4G cuatro segundos. Inicio y estabilización de la reproducción DASH

La primera diferencia apreciable con respecto a la reproducción bajo el estándar 3G es que la calidad se estabiliza de forma casi inmediata y en la mejor versión que tenemos del vídeo (figura 47). El resto funciona del mismo modo ya observado, en primer lugar se carga el reproductor y luego se descarga el archivo .mp4 para poder ubicar las versiones del vídeo.

Name	Size	Time
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	610 kB	2.42 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.0 MB	3.07 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	774 kB	3.35 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.6 kB	542 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.7 kB	60 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	217 kB	612 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	115 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	450 kB	815 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	27 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.7 MB	5.67 s
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	2.0 MB	7.84 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.1 MB	4.38 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	302 kB	1.72 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	521 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	526 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	298 kB	220 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	72 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	292 kB	176 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	190 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	237 kB	39 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.4 kB	21 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	236 kB	77 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	60 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	671 kB	1.69 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.8 kB	242 ms

Figura 33: 4G cuatro segundos. Reducción del límite de velocidad

Al reducir la velocidad de la red, con un tamaño de segmento reducido, el cambio es prácticamente inmediato. Como con la reducción el cliente no puede aguantar la calidad a 2500 Kbps, DASH cambia rápidamente a la calidad óptima (figura 46).

Name	Size	Time
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	758 kB	111 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.3 kB	32 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	663 kB	125 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	2.0 MB	198 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	629 kB	88 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.6 MB	165 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	3.5 MB	328 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	513 kB	63 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.5 MB	221 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	217 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.3 kB	22 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.9 kB	29 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	556 kB	82 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.7 kB	34 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	877 kB	135 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.3 kB	24 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	947 kB	120 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.6 kB	24 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	3.3 MB	424 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.9 kB	24 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	317 kB	64 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	23 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	485 kB	64 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	65.5 kB	23 ms
<input type="checkbox"/> original video 1280x720 2500k dashinit.mp4	871 kB	153 ms

Figura 34: 4G cuatro segundos. Vuelta al límite de velocidad inicial

Una vez la velocidad vuelve a ser la máxima disponible bajo el estándar 4G DASH se encarga de seleccionar el vídeo con una tasa de 2500 Kbps, pues es el mejor del que disponemos y puede descargarlo perfectamente. En la figura 35 vemos como todos los cambios son prácticamente instantáneos.

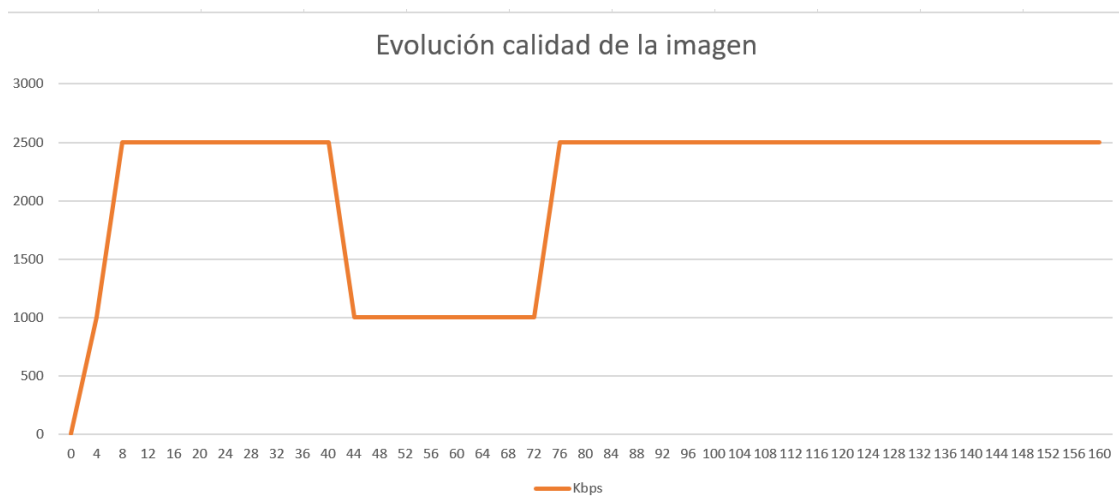


Figura 35: 4G cuatro segundos. Gráfica calidad de imagen

5.2.2 4G segmentos de ocho segundos

Vamos a poner a prueba el funcionamiento de DASH con segmentos de ocho segundos y las condiciones dadas por el estándar 4G.


























Name	Size	Time
 hls.min.js	58.3 kB	147 ms
 dash.all.min.js	137 kB	161 ms
 manifiesto8s.mpd	3.0 kB	47 ms
 original_video_160x90_250k_dashinit.mp4	683 B	54 ms
 original_video_320x180_500k_dashinit.mp4	683 B	62 ms
 original_video_640x360_1000k_dashinit.mp4	683 B	48 ms
 original_video_1280x720_2500k_dashinit.mp4	684 B	90 ms
 original_audio_128k_dashinit.mp4	789 B	101 ms
 original_video_1280x720_2500k_dashinit.mp4	1.4 kB	33 ms
 original_audio_128k_dashinit.mp4	1.4 kB	30 ms
 original_video_1280x720_2500k_dashinit.mp4	2.7 MB	547 ms
 original_audio_128k_dashinit.mp4	133 kB	82 ms
 original_audio_128k_dashinit.mp4	131 kB	51 ms
 original_video_1280x720_2500k_dashinit.mp4	4.3 MB	544 ms
 favicon_144-vfiiLAfaB.png	16.3 kB	63 ms
 original_video_1280x720_2500k_dashinit.mp4	3.7 MB	413 ms
 original_video_1280x720_2500k_dashinit.mp4	2.3 MB	273 ms
 original_audio_128k_dashinit.mp4	131 kB	34 ms
 original_audio_128k_dashinit.mp4	131 kB	118 ms
 original_audio_128k_dashinit.mp4	131 kB	34 ms
 original_audio_128k_dashinit.mp4	131 kB	31 ms
 original_video_1280x720_2500k_dashinit.mp4	2.6 MB	317 ms
 original_audio_128k_dashinit.mp4	131 kB	44 ms
 original_video_1280x720_2500k_dashinit.mp4	3.6 MB	463 ms
 original_audio_128k_dashinit.mp4	131 kB	32 ms

Figura 36: 4G ocho segundos. Inicio y estabilización de la reproducción DASH

Como en todos los casos anteriores, los dos primeros pasos en el inicio del streaming son la carga del reproductor y la descarga del manifiesto. Pese a contar con una alta velocidad de descarga, se aprecia un pequeño retraso de poco más de medio segundo al iniciar la reproducción.

Name	Size	Time	v
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	2.1 MB	7.37 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	367 ms	
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	3.7 MB	13.91 s	
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.4 MB	6.18 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.11 s	
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.4 kB	22 ms	
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	2.0 MB	7.98 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	476 ms	
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	4.3 MB	16.88 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.04 s	
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	1.4 kB	19 ms	
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	500 kB	1.95 s	
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	354 kB	1.39 s	
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.6 MB	6.72 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.09 s	
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	180 kB	695 ms	
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.4 kB	18 ms	
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	380 kB	2.02 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.11 s	
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	415 kB	1.64 s	
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.2 MB	5.30 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.06 s	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	389 ms	
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	480 kB	947 ms	
<input type="checkbox"/> original_audio_128k_dashinit.mp4	112 kB	386 ms	

Figura 37: 4G ocho segundos. Reducción del límite de velocidad

En esta ocasión la reducción del límite de velocidad provoca que, como el cambio se produce a mitad de la descarga de uno de los segmentos de mayor calidad, el búfer se vacíe casi al completo, estando cerca de llegar a parar el vídeo. DASH soluciona el problema optando por descargar el vídeo con la menor calidad posible, de este modo recupera niveles del búfer que garanticen la continuidad del vídeo. Tras esto aumenta la calidad hasta encontrar la opción óptima, en este caso con una tasa de 500 Kbps (figura 45).

Name	Size	Time
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.2 MB	5.30 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	1.06 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	389 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	480 kB	947 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	435 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	629 kB	1.54 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	33 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	531 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	854 kB	2.58 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	32 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	4.1 MB	599 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.5 MB	220 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	7.1 MB	778 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	114 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	130 kB	28 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	49 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	2.5 MB	364 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	51 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	3.2 MB	347 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	32 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	26 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.9 MB	220 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	59 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.8 MB	199 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	131 kB	26 ms

Figura 38: 4G ocho segundos. Vuelta al límite de velocidad inicial

Con los segmentos de ocho segundos la imagen tarda un poco en recuperar su calidad máxima, pues el búfer se encuentra lleno de segmentos de ocho segundos a una calidad inferior. Una vez pasado este periodo la reproducción, como se ve en la figura 39, recupera su máxima calidad.

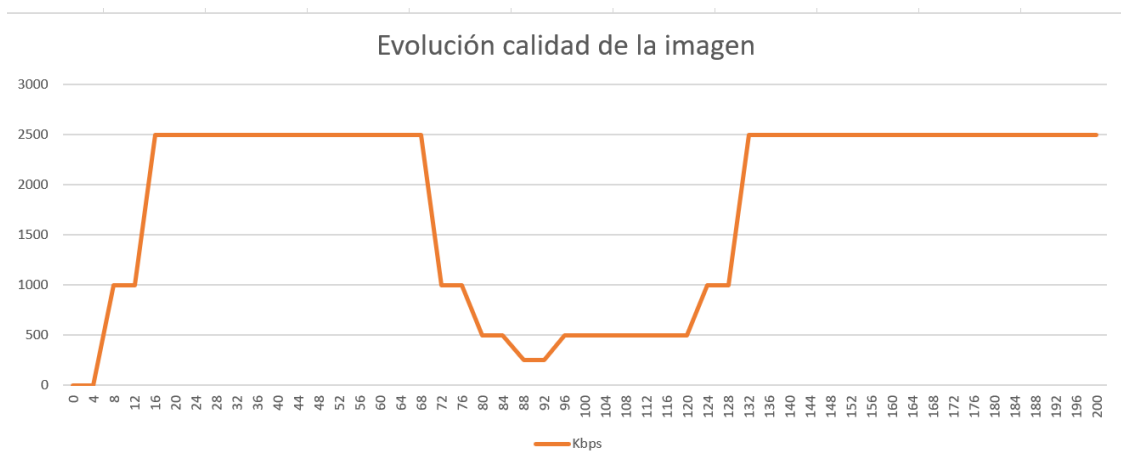


Figura 39: 4G ocho segundos. Gráfica calidad de imagen



5.2.3 4G segmentos de dieciséis segundos

La última prueba va a consistir en estudiar el comportamiento del protocolo DASH con unos segmentos de dieciséis segundos.

Name	Size	Time
<input type="checkbox"/> manifiesto16s.mpd	3.0 kB	242 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	478 B	25 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	479 B	37 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	479 B	36 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	479 B	106 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	550 B	104 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.4 kB	20 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	1.4 kB	19 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.6 MB	350 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	264 kB	123 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	115 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	1.4 kB	23 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	8.3 MB	899 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	5.3 MB	532 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	55 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	6.3 MB	677 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	56 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	54 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	7.1 MB	699 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	51 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	8.9 MB	978 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	85 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	51 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	5.6 MB	615 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	87 ms

Figura 40: 4G dieciséis segundos. Inicio y estabilización de la reproducción DASH

Tras la descarga del manifiesto y la ubicación de las diferentes alternativas del vídeo se selecciona la mejor opción posible. El tamaño del segmento provoca que la primera descarga se demore prácticamente un segundo. Además, al principio si notamos la calidad más baja pese a contar con una gran velocidad. Tras ese tiempo la calidad se establece en la máxima con la que contamos.

Name	Size	Time
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	39 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	7.1 MB	774 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	42 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	8.9 MB	924 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	37 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	41 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	5.6 MB	495 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	56 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	2.2 MB	13.27 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.4 kB	24 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	3.5 MB	15.03 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	2.16 s
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	8.7 MB	34.70 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	2.12 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	1.4 kB	81 ms
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	984 kB	3.77 s
<input type="checkbox"/> original_video_160x90_250k_dashinit.mp4	648 kB	2.57 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	887 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.4 kB	22 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.3 MB	4.22 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.7 MB	6.88 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	640 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.3 MB	4.32 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	835 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.02 s

Figura 41: 4G dieciséis segundos. Reducción del límite de velocidad

El cambio en la calidad tras la reducción tarda mucho en hacer efecto, pero el búfer llega a vaciarse y el vídeo se detiene durante unos segundos hasta que un segmento de calidad inferior es descargado. Se puede ver como hay una descarga que se demora 34 segundos, esta es la que provoca que se detenga momentáneamente la reproducción.

Name	Size	Time
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.2 MB	3.99 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	566 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.3 MB	4.02 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	546 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.04 s
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.5 MB	5.01 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	950 ms
<input type="checkbox"/> original_video_320x180_500k_dashinit.mp4	1.8 MB	6.32 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	852 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	1.04 s
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	1.4 kB	24 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	3.7 MB	547 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	2.0 MB	266 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	45 ms
<input type="checkbox"/> original_video_640x360_1000k_dashinit.mp4	3.5 MB	561 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	47 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	8.7 MB	989 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	6.3 MB	965 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	65 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	53 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	9.8 MB	1.04 s
<input type="checkbox"/> original_audio_128k_dashinit.mp4	261 kB	58 ms
<input type="checkbox"/> original_video_1280x720_2500k_dashinit.mp4	5.3 MB	741 ms
<input type="checkbox"/> original_audio_128k_dashinit.mp4	262 kB	71 ms
<input type="checkbox"/> original audio 128k dashinit.mp4	261 kB	38 ms

Figura 42: 4G dieciséis segundos. Vuelta al límite de velocidad inicial

Para finalizar, al acabar la limitación de la velocidad el gran tamaño de los segmentos provoca que, al igual que cuando reducíamos la velocidad, también tardamos en notar el aumento esperado en la calidad, aunque al final acaba llegando. En la figura 43 vemos que los cambios ante las variaciones de la red son muy lentos.

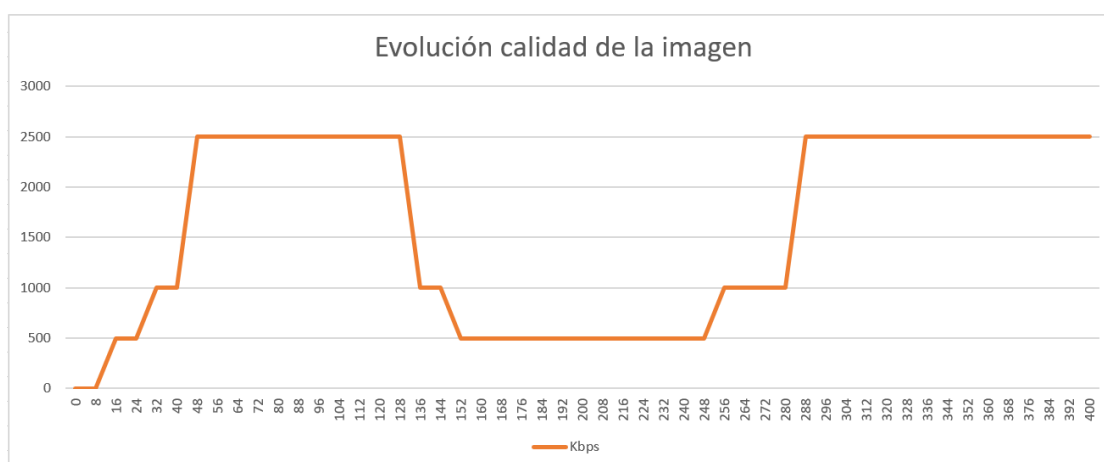


Figura 43: 4G dieciséis segundos. Gráfica calidad de imagen



Figura 44: Ejemplo imagen tasa de bits 250 Kbps

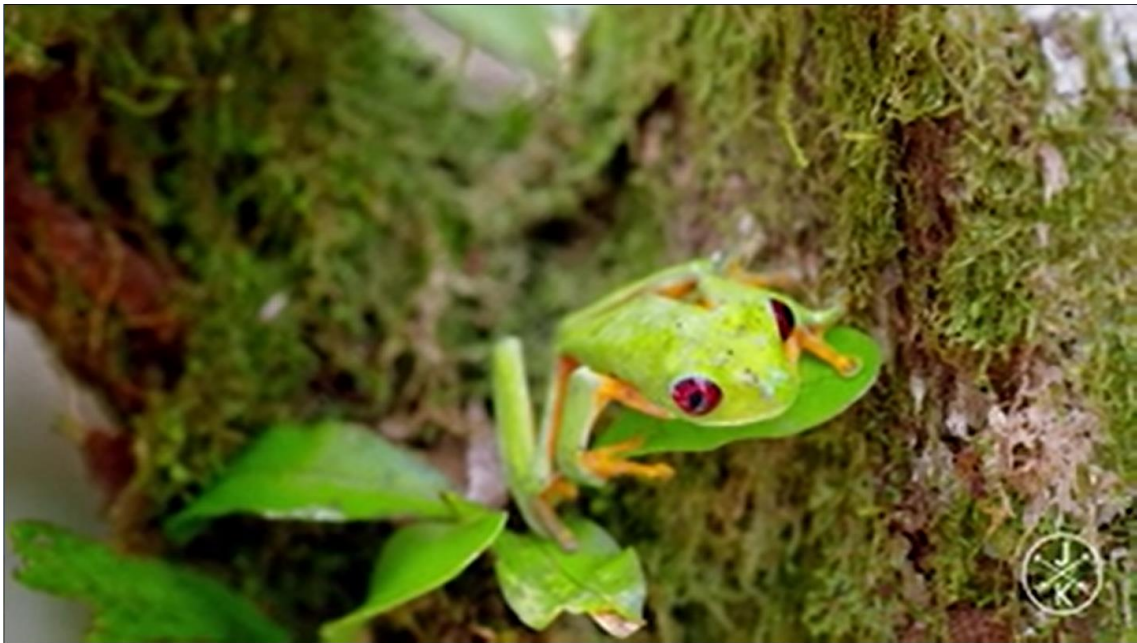


Figura 45: Ejemplo imagen tasa de bits 500 Kbps



Figura 46: Ejemplo imagen tasa de bits 1000 Kbps



Figura 47: Ejemplo imagen tasa de bits 2500 Kbps

6. Conclusiones

A lo largo de este trabajo se han estudiado las redes móviles y el streaming por Internet, haciendo un repaso desde sus inicios y destacando los principales estándares y protocolos que han hecho posible su funcionamiento.

Después de echar la vista atrás nos hemos dado cuenta de como el streaming se remonta a un pasado muy anterior al esperado, aunque funcionase de una forma muy diferente a lo que conocemos hoy en día. Hemos pasado de grandes y costosos equipos que costaban mucho de transportar a que cualquiera de nosotros pueda transmitir en vídeo desde cualquier parte del mundo solo con un teléfono móvil.

Los grandes y rápidos avances han provocado que haya grandes diferencias entre dispositivos que apenas cuentan con unos meses de diferencia. Esto, unido al gran aumento del tráfico de contenido multimedia en Internet, ha provocado la necesidad de que cualquier contenido pueda ser consumido por cualquier dispositivo moderno ya tenga una conexión muy potente o de capacidades más modestas. Con el objetivo de cubrir esa necesidad, compañías como Apple, Microsoft y Adobe crearon estándares para solucionar este problema en los equipos fabricados por la compañía, con la idea de que su solución fuese la predominante. Sin embargo, contamos con una gran variedad de dispositivos alrededor del mundo, por lo que tales compañías, pese a proporcionar una solución a sus usuarios, no lograron el objetivo de dominar el mercado del streaming. Es por ello que de ese intento de cubrir una necesidad surgió una nueva, la de que el streaming por Internet funcionase lo mejor posible en todos los dispositivos independientemente de su hardware o software. Con este objetivo global se desarrolló MPEG-DASH. Al tratarse de un estándar no se preocupa de compatibilidades entre dispositivos, se centra en ofrecer el mejor servicio posible teniendo en cuenta la capacidad del cliente en cada momento.

En lo referente a las pruebas realizadas, hemos probado bajo las condiciones de los estándares 3G y 4G. Probar con 3G nos ha arrojado resultados muy interesantes, pues pese a que DASH surgió mucho después del 3G, de haber aparecido antes el streaming podría haber crecido mucho bajo ese estándar. Con un tamaño de segmento de cuatro segundos hemos obtenido resultados prometedores, pues el protocolo actúa de forma casi inmediata ante los cambios de la red, siendo eso sí muy sensible a los posibles cortes de la misma. Con los segmentos de ocho segundos esta causa tenía menos efecto en la calidad del streaming, pues el búfer contiene una mayor cantidad de vídeo. Sin embargo, al ser segmentos de mayor tamaño se tarda más tiempo en alcanzar las condiciones óptimas. Las pruebas con los segmentos de dieciséis segundos nos dejan clara una cosa, un tamaño tan elevado no es nada recomendable para este tipo de soluciones. Las razones son claras, tarda en arrancar, tarda en alcanzar las condiciones óptimas y sufre bastantes parones durante la reproducción.



Bajo las condiciones del estándar 4G hemos obtenido resultados similares. Lo que, teniendo en cuenta la gran diferencia entre 3G y 4G, habla muy bien del protocolo MPEG-DASH. Gracias a la velocidad de este estándar, con segmentos de cuatro segundos alcanzamos de forma prácticamente inmediata la calidad óptima para las condiciones de la conexión, además de reaccionar rápidamente a los cambios de la red. Respecto a las pruebas con segmentos de ocho segundos, el inicio se retrasa un poco y, pese a que aguanta mejor las caídas de red que los segmentos de cuatro segundos, esto provoca que tarde más en adaptarse a las condiciones de la red. Las pruebas con segmentos de dieciséis segundos nos muestran mejores resultados que las realizadas con dicho tamaño bajo el estándar 3G, pero, pese a la velocidad del 4G, el streaming se sigue parando en ciertos puntos, lo que no es para nada el objetivo de DASH. En conclusión, MPEG-DASH es un protocolo con el que cualquier dispositivo puede emitir contenido que cualquier dispositivo puede reproducir, y cuando su uso esté enfocado en conexiones móviles, teniendo en cuenta la volatilidad de este tipo de redes un menor tamaño de segmento nos dará los mejores resultados.

Bibliografía

1. <https://es.slideshare.net/kahmal/internet-movil>
2. <https://www.enter.co/especiales/innovacion/de-1g-a-5g-un-recorrido-por-la-historia-del-internet-movil/>
3. <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>
4. <https://aprendiendoarduino.wordpress.com/2017/06/26/protocolo-http-2/>
5. <https://www.bbc.com/mundo/noticias-37247130>
6. <https://concepto.de/http/#ixzz6KcSk7EWd>
7. https://en.wikipedia.org/wiki/Dynamic_Adaptive_Streaming_over_HTTP
8. https://www.researchgate.net/figure/MPEG-DASH-system-overview-21_fig5_319143979
9. <https://blog.michelletores.mx/peticiones-http-get-post-put-delete-etc/>
10. <https://www.brendanlong.com/the-structure-of-an-mpeg-dash-mpd.html>
11. <https://www.microsoft.com/silverlight/smoothstreaming/>
12. <https://www.scc.uned.es/jornadasmaster/pdf/Charla1.pdf>
13. <https://www.ffmpeg.org/>
14. <https://github.com/gpac/gpac/wiki/DASH-Introduction>
15. https://www.webtv-solutions.com/support.php?s=other_docs&d=dash&lang=es
16. <https://reference.dashif.org/dash.js/>
17. Pozueco Álvarez, L, García Pañeda, X, García Fernández, R, Melendi Palacio, D, Cabrero Barros, S, Díaz Orueta, G (2015) Adaptation engine for a streaming service based on MPEG-DASH. *Springer*. Recuperado de <https://link.springer.com/article/10.1007/s11042-014-2034-y>



18. MPEG's Dynamic Adaptive Streaming over HTTP (DASH) –Enabling Formats for Video Streaming over the Open Internet. http://tech.ebu.ch/docs/events/webinar043-mpeg-dash/presentations/ebu_mpeg-dash_webinar043.pdf. pág. 26, 31
19. Vilas Paz, M. (2007) *Análisis del tráfico generado por servicios de audio y video en Internet al atravesar diferentes tecnologías de Red* (Tesis doctoral) Recuperado de <https://www.educacion.gob.es/teseo/mostrarRef.do?ref=409794>
20. Álvarez, A, Pozueco Álvarez, L, García Pañeda, X, García Fernández, R, Melendi Palacio, D, Cabrero Barros, S, Díaz Orueta, G (2014) Adaptive Streaming: A subjective catalog to assess the performance of objective QoE metrics. *Macrothink Institute*. Recuperado de <http://www.macrothink.org/journal/index.php/npa/article/view/5461>