



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Escola Tècnica Superior d'Enginyeria
de Telecomunicació de Barcelona



QUIC over IP Multicast

Treball Final de Grau realitzada a
l'Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona
Universitat Politècnica de Catalunya
per

Enric Perpinyà Pitarch

En compliment parcial
dels requisits per al Grau en
Serveis i Sistemes de Telecomunicacions **ENGINYERIA**

Director: Jorge Mata Díaz
Barcelona, Data 2022



Contents

List of Figures	3
List of Tables	3
1 Introducció	7
1.1 Transfons del project	8
1.2 Objectius	9
1.3 Requeriments i especificacions	10
1.4 Pla de feina	10
1.4.1 Estructura de la feina	11
1.4.2 Paquets de feina, tasques i fites	12
1.5 Diagrama temporal (Diagrama de Gantt)	16
2 Live-streaming state of art	17
2.1 Technologies used at the moment	17
2.2 The new standard: QUIC	18
2.3 Benefits of using TCP for live-streaming	18
2.4 UDP	19
2.5 Topic	19
2.6 Topic	19
3 Section 3	20
3.1 Subsection	20
4 Section 4	22
4.1 Subsection 4.1	22
4.2 Subsection 4.2	23
5 Section 5	24
5.1 Overview	24
6 Experiments and results	25
7 Budget	26
8 Environment Impact (Optional)	27
9 Conclusions	28
10 Future Work	28
References	29
Appendices	30

List of Figures

1	Trafic d'Internet	7
2	Trafic d'Internet	9
3	Diagrama de Gantt del projecte	16
4	Prototype setup	22

Listings

List of Tables

1	This is the caption	23
---	-------------------------------	----

Abbreviations

CBR Constant BitRate

CDN Content Delivery Network

ETSETB Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona

EU European Union

HTTP Hypertext Transfer Protocol

IETF Internet Engineering Task Force

IP Internet Protocol

ISP Internet Service Provider

RFC Request For Comments

RTCP Real Time Control Protocol

RTP Real Time Protocol

TCGI Transport Control i Gestió d'Internet

TCP Transmission Control Protocol

WebRTC Web Real Time Communications

Abstracte

El present document descriu i compara la implementació d'una estació de radio IP fent ús de la tecnologia QUIC sobre multicast i sobre unicast per retransmetre contingut en temps real. L'objectiu és demostrar que l'ús de multicast és millor per distribuir aquest tipus de contingut en termes d'escalabilitat, energètics i econòmics. També es discuteix les dificultats i avantatges tècniques de l'ús de QUIC sobre multicast (Draft QUIC over Multicast, Pardue i Hurst) com la gestió de Connection IDs o l'ús d'RTP i RTCP per capes superiors i la seguretat.

Revision history and approval record

Revision	Date	Purpose
0	05/03/2022	Document creation
1	dd/mm/yyyy	Document revision

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Enric Perpinyà Pitarch	enric.perpinya@estudiantat.upc.edu
Jorge Mata Díaz	jorge.mata@entel.upc.edu

Written by:		Reviewed and approved by:	
Date	05/03/2022	Date	dd/mm/yyyy
Name	Enric Perpinyà	Name	Jorge Mata
Position	Project Author	Position	Project Supervisor

1 Introducció

En els darrers anys, els serveis de contingut en temps real han experimentat un creixement quasi exponencial, sobretot arran de la popularitat de pàgines web com [youtube.com](https://www.youtube.com), [twitch.tv](https://www.twitch.tv) o sobretot amb l'ús de les videoconferències durant la pandèmia. No obstant això, les tècniques actuals per distribuir aquest tipus de contingut, malgrat haver millorat bastant amb el desenvolupament d'eines com WebRTC, encara és un problema molt greu tant a la part dels servidors, principalment per les CDN, com també pel volum de tràfic que implica a les xarxes, principalment pels ISP. De fet, s'està tornant una qüestió tecnològica tant greu que molts serveis d'aquests han decidit apujar els seus preus per poder fer front a la demanda.

Al mateix temps que hi ha aquest augment de la demanda per serveis nous d'Internet, també hi ha altres que estan intentant canviar de plataforma de distribució tecnològica com la televisió i la ràdio; és a dir, les cadenes de televisió estan intentant transmetre el seu contingut utilitzant Internet en lloc d'antenes o per cable utilitzant un canal freqüencial destinat per elles. De fet, fer-ho a través d'Internet permet estalviar costos significatius de distribució al mateix temps que permet tenir una retroactivitat i perspectiva més acurada de la quantitat de gent que està veient l'emissió.

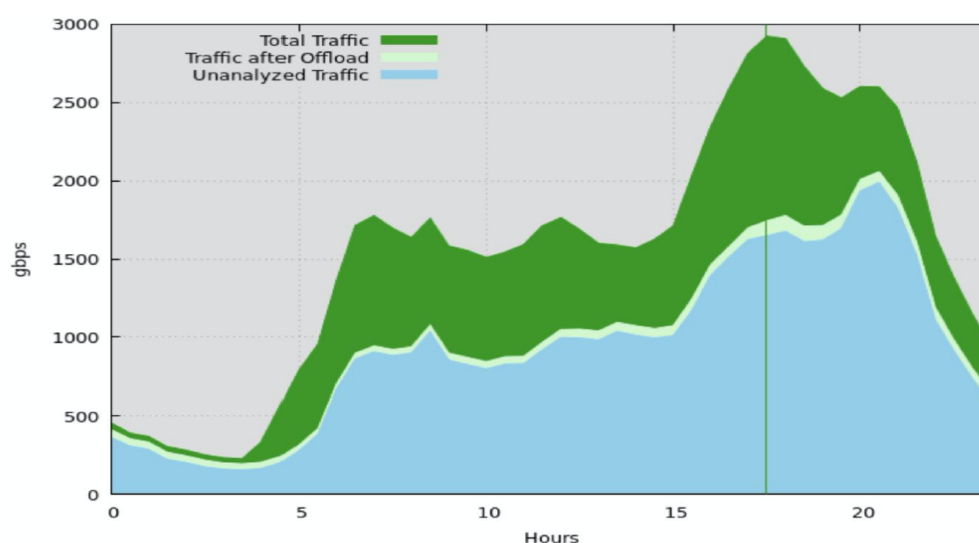


Figure 1: Volum de tràfic en un dia amb la sortida d'un nou videojoc. En blau el tràfic no analitzat, en verd clar el tràfic normal, en verd fosc el tràfic si la descarrega es fes amb multicast. Imatge d'Akamai.

L'any 2015, el tràfic de serveis de vídeo en temps real, com Netflix, YouTube o Twitch, va representar pràcticament el 70% del tràfic. Normalment, aquest tipus de tràfic va per pics, ja que quan una nova temporada d'una sèrie famosa, quan un youtuber important puja un nou vídeo o s'està fent un esdeveniment en directe la quantitat de gent veient el contingut sol augmentar de manera sobtada sobretot als primers moments. Això sol implicar en general problemes per donar accés a tothom al servei depenent de quanta gent sigui, mentre que al mateix temps altres serveis també es veien afectats (les CDN solen tenir molts serveis simultàniament distribuïts en microserveis).

Per ficar dades i posar-nos en context del volum de tràfic, el màxim que s'ha arribat a donar simultàniament per part de la CDN Akamai ha estat un pico de 167 Tbps en abril de 2020. Suposant un tràfic lineal, per simplificar, ja que normalment s'utilitzen codificacions CBR en directe en temps real, amb una resolució de 1080p el tràfic és de 5 Mbps/connexió i fins a 20 Mbps/connexió en cas de 4k. Llavors, en 1080p suposant aquell tràfic màxim suportat llavors es podrien arribar fins a 33.4 milions de clients en 1080p i fins a 8.35 milions en 4k. Si tota Espanya volgués mirar un contingut en directe, seria impossible donar el servei.

Aquests problemes de subministrament solen ser a causa que les connexions que es fan són punt a punt; dit d'una altra manera, són unicast. Això implica que el servei que pots donar en gran part està limitat al nombre de connexions simultànies que es pot donar. Malgrat que l'ús d'unicast té grans avantatges com la facilitat en la seguretat, la simplificació en l'arquitectura del programa o que pràcticament tots els dispositius del món suporten, també comporta un gran inconvenient.

Si el contingut que has de distribuir als clients és exactament el mateix com en el cas de la televisió o la ràdio, unicast pot arribar a saturar un servei si el volum d'usuaris és molt alt. Arran d'aquesta qüestió de distribuir el mateix contingut a tothom que sol·licita, ja en els anys 80 amb la creació d'IP, es va pensar en una tecnologia que soluciona aquests casos: **IP Multicast**. Malgrat no estar tan establerta com IP unicast, sí que la gran majoria de dispositius l'accepten.

Per altra banda, al llarg dels anys, a les connexions unicast normalment han fet ús d'un protocol anomenat TCP. Aquest protocol va ser especialment important en els anys 90 i a principis dels 2000 perquè Internet pogués funcionar i popularitzar-se, ja que les connexions eren lentes i poc fiables. Aquest protocol incorpora una sèrie de mecanismes per intentar pal·liar aquests qüestió. No obstant això, amb la millora de les connexions i la seva fiabilitat hi ha altres protocols que poden ser més útils i millors en l'Internet que tenim actualment, incorporant funcionalitats que antigament no s'havien plantejat o que no semblaven rellevants. Un protocol que pareix que substituirà TCP és QUIC, ja que millora bastant certs aspectes de TCP com l'establiment de la connexió molt més ràpidament (menys RTTs) o el poder restablir la sessió amb el servidor en canviar de xarxa.

1.1 Transfons del projecte

Aquest projecte es fa des de zero, encara agafa principalment dos projectes de codi lliure d'Internet i un borrador del IETF:

1. ***Hypertext Transfer Protocol (HTTP) over multicast QUIC***, de Lucas Pardue, Richard Bradbury i Sam Hurst.
2. ***NGHQ***: Llibreria escrita en C que implementa part de l'esborrany de *HTTP sobre multicast QUIC* fins la versió 7.
3. ***NGTCP2***: Llibreria escrita en C que implementa el protocol QUIC segons l'estàndard escrit en el RFC 9000.

La idea original ha estat de l'autor, encara que l'enfocament i la metodologia han estat proposades pel professor, Jorge Mata.

Tot el codi utilitzat és de codi lliure desenvolupat per enginyers i que està pràcticament tot allotjat en plataformes online com Github. Tot el codi desenvolupat en aquest projecte està allotjat també al Github i també és de codi lliure.

1.2 Objectius

El principal objectiu d'aquest projecte és desenvolupar un servidor de contingut en temps real que faci ús de la tecnologia QUIC sobre IP multicast utilitzant la llibreria de software lliure. Al mateix temps, es preten fer una comparativa de les avantatges i inconvenients front al desenvolupament d'un servidor similar utilitzant QUIC sobre IP unicast; és dir, el perfil de QUIC genèric que proposa el RFC 9000.

El sistema estarà conformat per un servidor que rebrà peticions dels clients que demanaran el contingut de radio en directe a través d'IP. Aquest servidor redireccionarà en el cas de multicast al fluxe multicast o enviarà directament la informació en cas d'unicast.

[ESBORRANY - FALTA L'IMATGE FINAL]

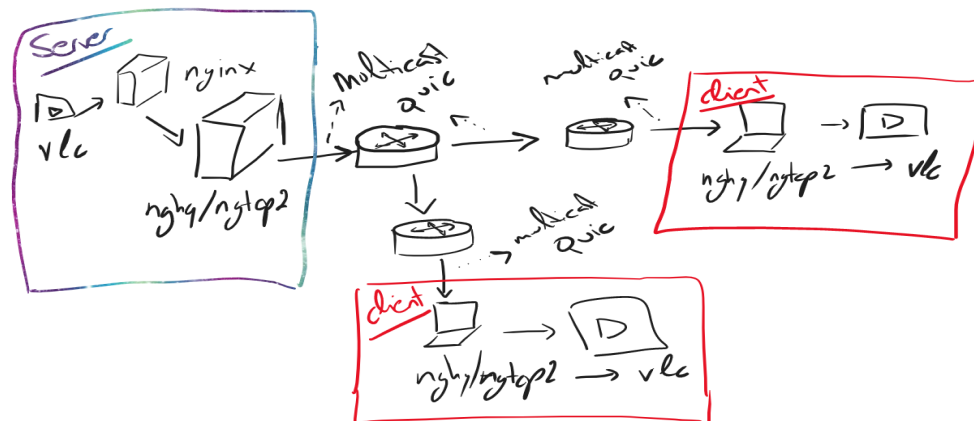


Figure 2: Topologia de la configuració plantejada.

La arquitectura del sistema està basada en la següent escenari: s'utilitzarà un reproductor d'àudio en directe, aquest serà captat per un servidor intermig i li passarà el contingut al servidor que hem desenvolupat perquè el transmeti a través de la xarxa fins als clients perquè ho reproduixin.

El que es pretén demostrar és que hi ha un límit d'usuaris en el cas d'unicast i, en canvi, no hi ha un límit en el cas d'usuaris multicast. Donat que la majoria de la gent accedeix a través d'HTTP als serveis d'aquest tipus de contingut és farà ús de QUIC, un protocol desenvolupat pensat en HTTP. Cal indicar que QUIC permet l'ús de fils simultànies que

s'utilitzaran per enviar tràfic també a través de multicast. El perfil de QUIC en el cas de multicast difereix en alguns paràmetres i configuracions d'un perfil de QUIC normal.

1.3 Requeriments i especificacions

Requeriments del projecte:

- Proveir un sistema integrat de transmissió de contingut en directe tant per la part del client com la part del servidor que permeti la transmissió via unicast i/o multicast.
- Donar una guia als usuaris finals per fer la instal·lació tant simple com sigui possible (annexe ?)
- Tot el software extern o implementat ha de ser gratuït i de codi lliure.
- Evaluar la solució final emulant-la en diferents escenaris i proveir uns resultats consistents.
- La transferència del contingut ha de ser en el màxim possible fluida i continua.
- El software desenvolupat ha de ser el més flexible possible per poder utilitzar-lo en futures implementacions d'altres softwares si es volgués.
- S'ha de poder avaluar el tràfic de manera quantitativa i qualitativa.

Especificacions del projecte:

- Hi ha d'haver com a mínim dos clients reben el contingut, ja que si no multicast deixa de tenir sentit encara que és viable fer-ho per un.
- El mostreig de la ràdio ha de poder variar durant la transmissió.
- Per evaluar el sistema en un entorn controlat, el servidor i els clients seran màquines virtual Linux que corren damunt el mateix sistema operatiu; Linux també.
- S'evaluarà el tràfic amb eines d'ús extensiu, proveint els filtres necessaris per decodificar el missatges transmesos.

1.4 Pla de feina

En aquesta secció de document es descriu en detall tots els blocs de treball en el que s'ha dividit la feina i els temps (Diagram de Gantt) en els que s'ha fet la tasca corresponent.

Al llarg del projecte, hi hagut varis canvis en respecte al pla original. Els principals canvis han estat els següents:

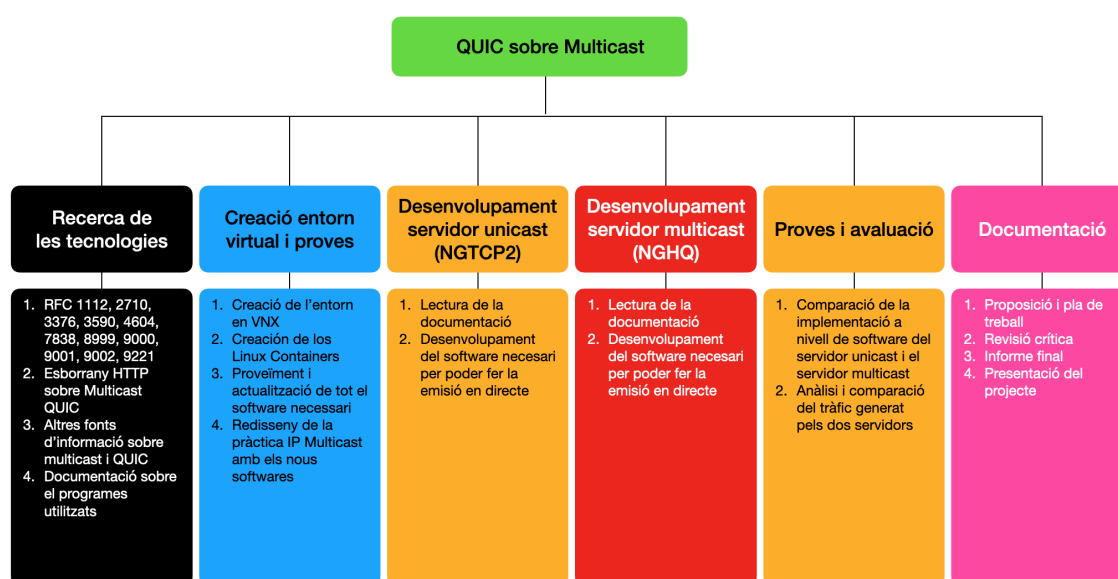
- **Lectura de RFCs.** Donat la complexitat del projecte i la profunditat d'aquest s'ha vist que la quantitat de documentació al respecte que s'hagut de llegir ha estat molt major de la que s'esperava, en gran part, ja que el nombre de tecnologies necessaries per poder entendre i desenvolupar una plataforma com la que es demana és molt major del que cabria esperar a priori. També, a causa de que s'utilitzen tecnologies noves com QUIC, hi han hagut RFCs que han aparegut durant el transcurs com el 9221.

- **Certs programes no existien o no compilaven.** Per fer l'escenari de proves s'ha volgut utilitzar un software nou per virtualitzar la xarxa intentant imitar la de la pràctica IP multicast de l'assignatura TCGI del grau d'Enginyeria de serveis i sistemes de telecomunicacions del ETSETB. Es va trobar que les màquines virtuals i el software utilitzat en aquelles màquines és antic i qualcún ja no està disponible. S'ha buscat alternatives al respecte.

El que més ha dificultat i endarrerit el desenvolupament ha estat la lectura de la documentació ja que era molt més extensa del que s'esperava a priori además de que també entrava en molt detalls que semblaven contradictoris amb la idea, encara que sembla que no. S'ha de pensar que la proposta és una tecnologia que encara no s'ha desenvolupat del tot i encara està en procés, llavors això és més un prova del concepte que una demostració per ficar-ho a producció. Encara s'està lluny d'aquest punt com es veurà al llarg del treball.

1.4.1 Estructura de la feina

[ESBORRANY - FALTA L'IMATGE FINAL]



1.4.2 Paquets de feina, tasques i fites

Projecte: Recerca de les tecnologies	WP ref: WP1	
Element principal: Lectura de la documentació	Pàg. 1 de 6	
Breu descripció: Lectura del RFCs necessaris per tal d'entendre i proposar solucions adequades per crear un perfil de QUIC sobre multicast. Lectura de la documentació necesaria	Data d'inici estimada: 01-10-2021 Data de finalització estimada: 01-11-2021	
	Data d'inici real: 14-09-2021 Data de finalització real: 30-04-2022	
Tasques internes: T1. RFC 1112, 2710, 3376, 3590, 4604, 7838, 8999, 9000, 9001, 9002, 9221 T2. Esborrany HTTP sobre Multicast QUIC T3. Altres fonts d'informació sobre multicast i QUIC T4. Documentació sobre el programes utilitzats	Entregables: Tasca 1 i 2	Dates: incluídes en el document

Projecte: Creació entorn virtual i proves	WP ref: WP2	
Element principal: Entorn virtual	Pàg. 2 de 6	
Breu descripció: Creació de l'entorn virtual amb VNX, proveïment de les màquines virtuals amb el software necessari i fer la pràctica d'IP multicast de TCGI per testejar que l'entorn funciona correctament	Data d'inici estimada: 16-09-2021 Data de finalització estimada: 10-10-2021	
	Data d'inici real: 16-09-2021 Data de finalització real: 30-12-2022	
Tasques internes: T1. Creació de l'entorn en VNX T2. Creació dels Linux Containers T3. Proveïment i actualització de tot el software necessari T4. Rediseny de la pràctica IP Multicast amb els nous softwares	Entregables: Presentació al professor de l'escenari montat	Dates: 14-01-2022

Projecte: Desenvolupament servidor unicast (NGTCP2)	WP ref: WP3	
Element principal: Desenvolupament de software	Pàg. 3 de 6	
Breu descripció: Creació d'un servidor unicast que utilitzi el protocol QUIC per poder utilitzar-lo per comparar amb un que utilitzi multicast. També s'ha de fer el client.	Data d'inici estimada: 01-11-2021 Data de finalització estimada: 01-12-2021	
	Data d'inici real: 15-02-2022 Data de finalització real: 05-05-2022	
Tasques internes: T1. Lectura de la documentació T2. Desenvolupament del software necessari per poder per poder fer la emissió en directe.	Entregables: Tasca 1 Tasca 2 Tasca 3	Dates: 15-05-2022

Projecte: Desenvolupament servidor multicast (NGHQ)	WP ref: WP4	
Element principal: Desenvolupament de software	Pàg. 4 de 6	
Breu descripció: Creació d'un servidor multicast que utilitzi el protocol QUIC per poder utilitzar-lo per comparar amb un que utilitzi unicast. També s'ha de fer el client.	Data d'inici estimada: 01-11-2021 Data de finalització estimada: 01-12-2021	
	Data d'inici real: 15-02-2022 Data de finalització real: 05-05-2022	
Tasques internes: T1. Lectura de la documentació T2. Desenvolupament del software necessari per poder per poder fer la emissió en directe.	Entregables: Tasca 1 Tasca 2 Tasca 3	Dates: 15-05-2022

Projecte: Proves i avaluació	WP ref: WP5	
Element principal: Simulació i proves de qualitat	Pàg. 5 de 6	
Breu descripció: Totes les simulacions i avaluacions necessaries per aconseguir proves concloents de en quin cas és millor utilitzar unicast i quin multicast.	Data d'inici estimada: 01-11-2021 Data de finalització estimada: 15-12-2021	
	Data d'inici real: 15-02-2022 Data de finalització real: 05-05-2022	
Tasques internes: T1. Comparació de l'implementació a nivell de software del servidor unicast i el servidor multicast T2. Anàlisi i comparació del tràfic generat pels dos servidors	Entregables: Tasca 1 Tasca 2 Tasca 3	Dates: 15-05-2022

Projecte: Documentació	WP ref: WP6	
Element principal: Documentació	Pàg. 6 de 6	
Breu descripció: Documentació que s'ha d'entregar. Un manual d'usuari per poder configurar ambdós servidors serà inclòs en el document final.	Data d'inici estimada: 15-09-2021 Data de finalització estimada: 15-01-2022	
	Data d'inici real: 20-01-2022 Data de finalització real: 15-05-2022	
Tasques internes: T1. Proposició i pla de treball T2. Revisió crítica T3. Informe final T4. Presentació del projecte	Entregables: Tasca 1 Tasca 2 Tasca 3	Dates: 08-10-21 30-11-21 15-05-22 28-05-22

Objectius

WP#	Tasca#	Títol curt	Objectius/Entregables	Data (setmana)
WP1	T1	RFC 1112, 2710, 3376, 4604, 7838, 8999, 9001, 9002, 9221	Estat de l'art	Inclós en el document
WP1	T1	Esborrany HTTP sobre Multicast QUIC	Estat de l'art	Inclós en el document
WP1	T1	Altres fonts d'informació sobre multicast i QUIC	Estat de l'art	Inclós en el document
WP1	T2	Documentació sobre el programes utilitzats	Guies	Inclós en el document
WP2	T2	Creació de l'entorn en VNX	Entorn virtual montat	Setmanes 37 a 52
WP2	T2	Creació de los Linux Containers	Màquines disponibles per l'entorn virtual	Setmanes 37 a 52
WP2	T2	Proveïment i actualització de tot el software necessari	Màquines provistes del software requerit	Setmanes 37 a 52
WP2	T2	Redisseny de la pràctica IP Multicast amb els nous softwares	Comentaris per actualitzar la pràctica d'IP Multicast amb software actual (annexe)	Setmanes 37 a 52
WP3	T1	Lectura de la documentació (NGTCP2)	Explicació del software a estat de l'art	Setmanes 7 a 19
WP3	T3	Desenvolupament del software necessari per poder fer la emissió en directe	Creació del servidor per enviar la radio IP via unicast	Setmanes 7 a 19
WP4	T1	Lectura de la documentació	Explicació del software a l'estat de l'art	Setmanes 7 a 19
WP4	T3	Desenvolupament del software necessari per poder fer la emissió en directe	Creació del servidor per enviar la radio IP via multicast	Setmanes 7 a 19

WP#	Tasca#	Títol curt	Objectius/Entregables	Data (setmana)
WP5	T4	Comparació de la implementació a nivell de software del servidor unicast i el servidor multicast	Explicació de les avantatges i inconvenients del desenvolupament a l'apartat de resultats del document	Setmanes 7 a 19
WP5	T4	Anàlisi i comparació del tràfic generat pels dos servidors	Gràfica comparativa de la quantitat de tràfic generat per les dues implementacions	Setmanes 7 a 19
WP6	T1	Proposició i pla de treball	Proposició i pla de treball	Setmana 37
WP6	T2	Revisió Crítica	Revisió Crítica	Setmana 48
WP6	T3	Informe final	Informe final	Setmana 19
WP6	T4	Presentació del projecte	Presentació del projecte	Setmana 20

1.5 Diagrama temporal (Diagrama de Gantt)

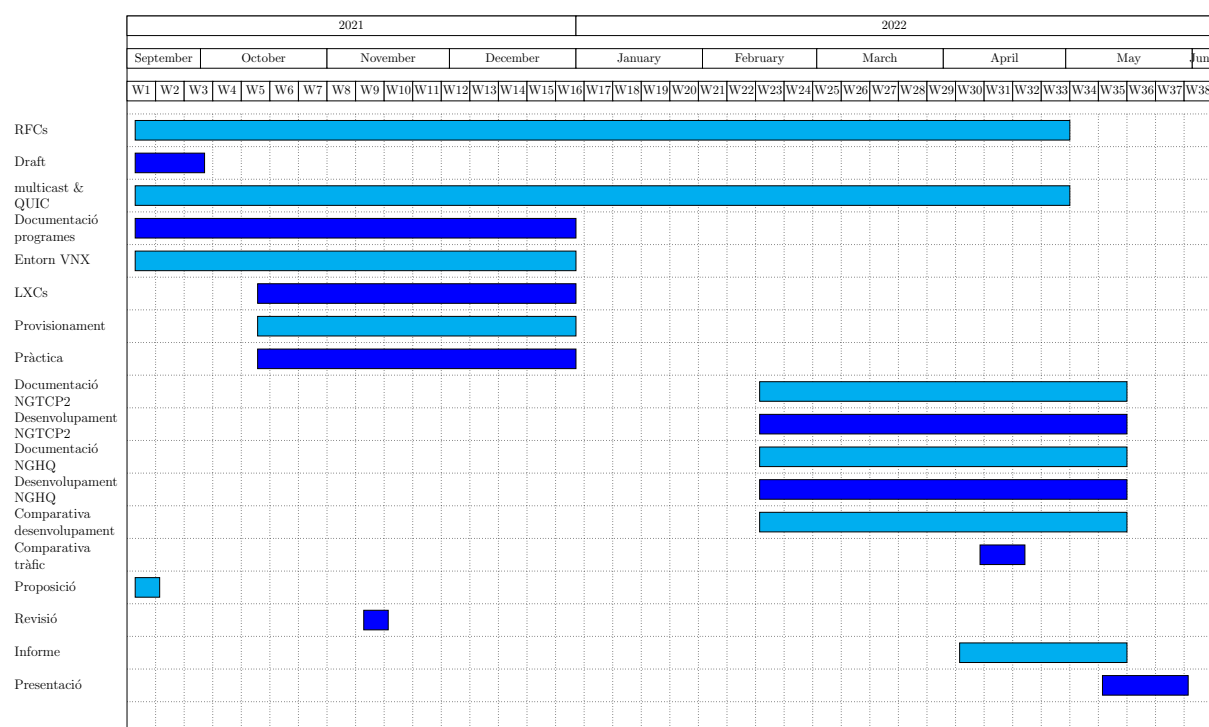


Figure 3: Diagrama de Gantt del projecte

2 Live-streaming state of art

Live-streaming content is in its dawn. Although it is possible and it is being done nowadays, real-time content in Internet is still primitive in the way we are delivering it. We are using brute-force to deliver huge amounts of real-time content. It is a matter of fact that the demand for this type of content is growing far faster than the CDN¹ (from now on called servers) are able to supply to the spectators (from now on called clients).

Even though someone might say it is almost every content via internet, there are still a lot of events that can not be transmitted using internet because it cannot be supported² due to the heavy load of users that need to be managed. As Jake Holland from Akamai wrote on Github: "The capacity of CDNs and others to deliver popular content at scale is not keeping up with demand."

In april of 2020, Akamai achieved the new world record of peak traffic delivery which was 176 tbps. It seems a lot, however it is not. Making some linear and quick easy maths it is easy to see that it is not. Keeping in mind that usually for real-time video we use CBR³ encoder in order to squeeze the internet resources, for a 1080p video we need a bit-rate of 5 mbps, and for a 4k video, 20 mbps. This means that with the current world-record we can arrive to 35.2 millions of simultaneous users for live-streaming content using a 1080p encoding and 8.8 millions users using 4k. Considering that the world population with internet access is 5200 million people we are really faraway from delivering TV or Radio via Internet using the current methods. What is more, the Fifa World Cup Finals 2018 were seen by 500 million people, 200 million to 300 million people were seeing in India the Cricket World Cup in 2015 (whenever India is playing) and a large etcetera with a lot of content can be done. The brute force does not seem like a good strategy for scaling anymore, maybe only for quick deployments.

The solution that is being discussed in this Final Degree Project proposes to use multicast instead of unicast in order to avoid this scalability limitation, due to the fact that using this technology for distribute the same content to everyone is more efficient in terms of computation in the servers and of resources in general.

2.1 Technologies used at the moment

TCP is at the moment the main protocol over Internet. Almost every service at the moment uses it and HTTP is not an exception. When doing a HTTP request or response (versions 1.0, 1.1 and 2), TCP is being used. Even though, it is almost 50 years old and the internet has evolved and has grown significantly, it is still the main protocol. However, it is reaching its full capacity in two particular scenarios: websites with small pageload and real-time content such as TV and Radio. Although, the main goal of this project is to focus on the real-time content delivery, the small pageload problem will also be explained briefly because is what QUIC tries to solve as originally planned.

¹CDN: Content Delivery Network

²<https://github.com/GrumpyOldTroll/wicg-multicast-receiver-api/blob/master/explainer.md>

³Constant Bit-Rate

Most of the is requested via a navigator like Firefox or Google Chrome, which use the HTTP protocol as the main communication protocol which the use of TCP. It is clear and obvious that for large content which is not in real-time that can be stored like films, TCP is not a big bottleneck. However, in the last several years, with the raising of important services (like e-commerce) and cyberattacks, specially spoofing, the use of TLS⁴ has also became a standard. Websites that deliver real-time video like Youtube or Twitch are not an exception either. This also adds some complexity and workload delivering the website to the client.

When someone connects to a internet using a web-browser at least it need 3 packets for for stablishing the TCP connection, 3 for stablishing the TLS connection over TCP and 2 HTTP packets (request and response). It is being received the first byte of content after the 7th packet! This handshake, although being the most being used, it is not optimal. When a really small website is being delivered (imagine an html "Hello World!"), a load of resources and packets are being used for a deliver a small packet. More control stuff than content. It gets even worse, if it is the case that have already stablished a connection earlier and after a while there is a new request; all this handshake has to be done again because TCP and TLS are a connection oriented protocols, but HTTP is not. QUIC aims to solve this type of problems.

2.2 The new standard: QUIC

It is clear and obvious that for large content that can be stored in a small buffer and does not need to be real-time we can use a really known protocol that have been around for years which is TCP. It is the main protocol for most of the services in the internet and has been even implemented in the kernel of most of the devices. Many of the devices that rule internet like firewalls or load-balancers were designed bearing in mind that the protocol that had to be used was TCP. During many years it was not a priority to use other transport layer protocols except in rare case.

Nowadays, might be surprising that we use that proccol that is end-to-end to deliver the same real-time content to a large amount of spectactors (for now on they will be called clients)⁵. It is easy to see that this approach although having many withdrawals it has some benefits.

The solution that is being discussed in this Final Degree Project proposes to use multicast instead of unicast in order to avoid this scalability limitation, due to the fact that using this technology for distribute the same content to everyone is more efficient in terms of computation in the servers and of resources in general.

2.3 Benefits of using TCP for live-streaming

The main benefit with this approach for real-time content is the easiness on the programing and mounting at first time. All devices (or almost all) can bypass internet elements

⁴TLS : Transport Security Layer

⁵In the Annexe I there is a larger explanation and details on how to test this with famous websites at the moment

when they talk TCP.

At the current moment, it is the standard which means less complexity, more tools and easier to implement. For young services that are not expected to growth a lot TCP is more than enough for most cases.

At the same time, there

2.4 UDP

Nevertheless, in the last decade it has been a growing interest in other protocols that are end-to-end and allow to expand functionalities in upper layers: UDP⁶. It is a really simple protocol with almost 0 functionalities aside from passing the content to the upper layer. It has been around since the creation of the internet. That means, like with TCP, is implemented in the kernel. Changing the kernel of all the devices in the internet it is impossible

2.5 Topic

Here you have a couple of references about LaTeX [1] and electrodynamics [2].

2.6 Topic

⁶UDP: User Datagram Protocol

3 Section 3

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. EU is the European Union. Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur. Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio. ETSETB is Telecom. Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

3.1 Subsection

The book [1] Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Algorithm 1 Temperature-Distributed algorithm

```

1: procedure TEMP-SPREAD( $GN_i, HN_j, temperatures$ )  $\triangleright$  Lowest temperature priority
2:    $temperature\_list \leftarrow short(temperatures)$ 
3:    $max\_temperature \leftarrow max(temperature\_list)$ 
4:    $ThresHold \leftarrow 0.5$ 
5:    $temperature\_impact \leftarrow 0.2$ 
6:   for  $GN_i$  in  $i = 1, 8$  do  $\triangleright$  Iterate every hardware node on the given GN
7:      $it\_temperature \leftarrow temperature\_list(GN_i)$ 
8:      $temp\_weight \leftarrow \frac{max\_temperature - it\_temperature}{max\_temperature} * temperature\_impact$ 
9:      $\omega(Master - GN_i) \leftarrow ThresHold * temp\_weight$ 
10:    for  $HN_j$  in  $j = 1, n$  do
11:      if  $available\_accel_{i,j} > busy\_accel_{i,j}$  then
12:         $policy_\omega = \frac{AvailableHW}{TotalHW} * ThresHold$ 
13:         $\omega(GN_i - HN_{i,j}) \leftarrow ThresHold + policy_\omega$ 
14:      else
15:         $\omega(GN_i - HN_{i,j}) \leftarrow 1$ 
16:     $node \leftarrow find\_djistra\_shortest\_path(Master\_Node, aux\_node)$ 
17:    return  $node$   $b$   $\triangleright$  The gcd is b

```

4 Section 4

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetur at, consectetur sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

4.1 Subsection 4.1

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetur at, consectetur sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui. Read de book [2] of Einstein.

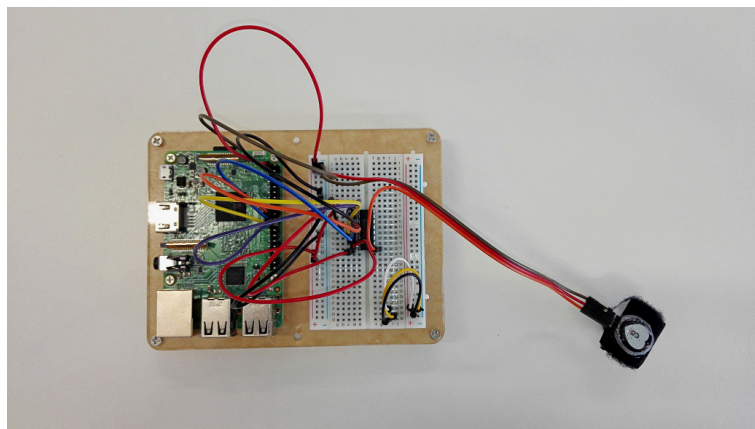


Figure 4: Prototype setup.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus

quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

4.2 Subsection 4.2

Table 1: This is the other caption. Since the trial size of the experiments showed is one second, the number of *Target* and *Impostor* data corresponds to number of trials or seconds

Dataset	Label	Train	Validation	Develop	Test
First	Target	135	45	30	30
	Impostor	5,220	1,740	1,890	2,880
	#Subjects	31			12
Second	Target	144	80	48	48
	Impostor	2,014	1,119	1,343	1,545
	#Subjects	15			5

5 Section 5

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

5.1 Overview

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetur odio sem sed wisi. Visite the Knuth repository [3].

6 Experiments and results

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetur a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetur. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

7 Budget

Depending on the thesis scope this document should include:

8 Environment Impact (Optional)

Whether the tasks that have led to the realization of this thesis, as if its results have identifiable environmental impact, describe it in this section.

9 Conclusions

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

10 Future Work

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetur odio sem sed wisi.

References

- [1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The L^AT_EX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.
- [2] Albert Einstein. Zur Elektrodynamik bewegter Körper. (German) [On the electrodynamics of moving bodies]. *Annalen der Physik*, 322(10):891–921, 1905.
- [3] Donald Knuth. Knuth: Computers and typesetting.

Appendices

Appendices may be included in your thesis but it is not a requirement.