Collective Communication in Rich Internet Applications based on HTML5 to Form Web Based Overlay Network (WOvNet)

By
Saqib Rasool
2011-NUST-MS PhD-IT-50

Supervisor
Dr. Hamid Mukhtar
NUST-SEECS

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Information Technology (MS IT)

In
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(February 2014)

Approval

It is certified that the contents and form of the thesis entitled "**Collective Communication in Rich Internet Applications based on HTML5 to Form Web Based Overlay Network (WOvNet)**" submitted by **Saqib Rasool** have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Hamid Mukhtar**

Signature: _____
Date: _____

Committee Member 1: **Dr. Aamir Shafi**

Signature: _____
Date: _____

Committee Member 2: **Mr. Mohsan Jameel**

Signature: _____
Date: _____

Committee Member 3: **Mr. M. Usman Ali**

Signature: _____
Date: _____

Abstract

Proprietary protocol of flash, RTMFP can be used to share servers' load by using overlay of peer-to-peer (p2p) connected browsers for streaming. Hence, huge audience can be entertained using comparatively less server resources and every new coming client is also considered an addition to overall system resources. However, RTMFP is not open source and is also not natively supported in browsers and thus end users need to install Adobe flash player before using it. It was also built for supporting audio/vedio streaming and overlay formation for propagation of data is not supported. Moreover, it works on UDP and thus cannot be used if all UDP ports are blocked. Although RTMP can be used as fallback option but it follows the client/server architecture and thus cannot be used to build an overlay network based on p2p communication. Moreover Adobe has also announced to cease the development of flash player for smart phones and is facilitating the replacement of flash player with HTML5 features on these devices.

We have focused on mentioned limitations of RTMFP and came up with a solution which is based on HTML5 standards and thus plugin free in nature. We have used communication options of HTML5 to implement a plugin free web based overlay network (WOvNet) and further optimized it using Battery Status API and Geolocation API of HTML5. Communication standards of HTML5 includes WebSocket, WebRTC, cross document messaging and server sent events. WebSocket follows the client/server architecture and is doing signalling and maintaining the network topology of WOvNet, along with server sent events.

DataChannel API of WebRTC is used for data propagation over the established overlay network and cross document messaging enables the integration of WOvNet within other applications. Moreover, WebRTC operates over UDP but can also be configured to work in scenarios where all UDP ports are blocked and only 443 port of TCP is available. We have further improved WOvNet for mobile devices by using adaptive switching of traffic based on Battery API of HTML5. Link stretch of WOvNet is also reduced by connecting geographically closer peers and placing these peers in similar islands or clusters. Geolocation API of HTML5 returns the approximate geographic locations of peers in longitudes and latitudes and thus helps in reducing link stretch of WOvNet.

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at National University of Sciences & Technology (NUST) School of Electrical Engineering & Computer Science (SEECS) or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Saqib Rasool**

Signature: _____

Acknowledgment

Up and above everything all glory to **ALMIGHTY ALLAH**. The Beneficent, The most Merciful and Most Compassionate. It's a great blessing from Almighty Allah that gives me the health and strength to do this research work. After this I would pay my sincere thanks to my **parents**, who have given me their unequivocal support throughout my life and this thesis is also the result of their endless prayers, motivations and sacrifices for me.

I am really appreciative to my supervisor and advisor **Dr. Hamid Mukhtar** who had been a tremendous mentor for me. I would like to thank him for encouraging me and giving me immense knowledge and guidance in completing this research work. I am paying my special gratitude to **Dr. Aamir Shafi**, **Mr. Mohsan Jameel** and **Mr. Usman Ali** for serving me as my committee members and for giving me brilliant suggestions relevant to my thesis work.

My co-workers and colleagues also deserve my acknowledgment as they have been really supportive to me for completing my thesis. Contributors' list is too large to mention here and I am very thankful to everyone in supporting me for completing this research.

Saqib Rasool

Contents

List of Abbreviations

Abbreviations	Descriptions
AMF	Action Message Format
API	Application Programming interface
b2b	Browser to Browser
CORS	Cross-Origin Resource Sharing
DOM	Document Object Model
DVMRP	Distance Vector Multicast Routing Protocol
GPS	Global Positioning System
HTML5	Hyper Text Markup Language 5
ICE	Interactive Connectivity Establishment
IGMP	Internet Group Management Protocol
IPTV	Internet Protocol Television
JS	JavaScript
LAN	define
MEAN	MongoDB Express Angular Node
NAT	Network Address Translation
p2p	Peer-to-Peer
RIA	Rich Internet Application
RTMFP	Real Time Media Flow Protocol
RTMP	Real Time Media Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
STUN	Session Traversal Utilities for NAT
TCP	Transmission Control Protocol
TURN	Traversal Using Relays around NAT
UDP	User Datagram Protocol
UE	Unit of Execution
W3C	World Wide Web Consortium
WebGL	Web Graphics Library
WAN	Wide Area Network
WebRTC	Web Real Time Communication
WOvNET	Web based Overlay Network
XHR	XMLHttpRequest

List of Figures

Chapter 1

INTRODUCTION

Rich Internet Application (RIA) are web based applications having ample functionality that was previously constricted only to desktop applications (1). Since the release of version 4 of HTML in 1997, no serious efforts were being observed to release the next version of HTML, till 2005. Three main reasons for this delay are discussed in section 1.2.2 of this chapter (1). As there is time span of more than one decade between the release of HTML4 and HTML5, therefore third party plugins gained huge importance in RIAs by offering rich features that were missing in HTML4. Collective communication is also one of the missing functionality in HTML4, implemented by third party Plugin.

Collective communication is the one based on cooperative information exchange between units of execution (UEs) abstracted by some high-level operations(2). These high level operations maybe a pure communication event like broadcast or multicast or it can also include processing like in the case of min or max functions. In this research we have used iterative unicast based multicast to implement broadcast, which is a high level operation of collective communication.

As it has been mentioned that collective communication can be implemented using third party plugin. Hence end users need to install corresponding plugin before running any functionality implemented over that particular plugin. Real time media flow protocol (RTMFP) is a proprietary protocol of Adobe that introduced a plugin based solution for offering collective communication in RIAs. Although this protocol is not open source however, it is clear from available documentation that it utilizes p2p (peer-to-peer) connections between browsers for both downward and upward streams. (3)

Strength of p2p based overlay network is inherently scalable as every new coming level L will add the capacity of facilitating to XL destination peers, assuming that every peer can multicast to X number of destination peers. Power of this simple concept can be understood by considering that if X is number of destination peers that are directly connected to the source peer

(such that X >1) and every destination peer can also multicast to further X peers then X2 + X destination peers can get the streaming with 2 overlay hop count of the source peer. Similarly, if all destination peers at level 2 can also send streaming to further X peers, it will result in the delivery of streaming to X3 + X2 + X destination peers within 3 overlay hop count of the source peer.(4)

HTML5 is the latest version of HTML and it has introduced many new standards and APIs that has tremendously increased the native power and capacity of web browsers. Hence, many features that were previously confined only to third party plugins are now natively available in web browsers. This native browser support encourages the web developers to implement plugin free applications that were previously developed only after the support of third party plugins. This trend has initiated a transitional era where focus is on replacing plugin based features with native standards and APIs of HTML5. Browser vendors have also participated in this transition by improving their native strengths to replace the need of third party plugins.(5)(6)

Although, HTML5 doesn't offer any built-in support for overlay formation but we have contributed in this transition of plugin based features to plugin free solutions by implementing an algorithm for creating web based overlay network (WOvNet). WOvNet creates p2p connections between browsers and then broadcast data between these interconnected peers using iterative unicast based overlay multicast. This overlay network is further optimized on the basis of available battery of portable devices using Battery Status API of HTML5, by implementing adoptive switching of network traffic in WOvNet. Moreover, Geolocation API of HTML5 helps in placement of geographically closer peers in same island or cluster to reduce the link stretch of connected peers in WOvNet, which eventually results in reducing packet delay.

The rest of the chapter is organized such that in Section 1.1 Collective Communication is defined. Section 1.2 explains the Rich Internet Applications. Section 1.3 discusses the problem statement. Section 1.4 enlists the contributions which contain motivations and goals of this research . Section 1.5 shares the challenges that were faced during this research. It is followed by the last section 1.6 containing details of further thesis organization.

1.1 Collective Communication

Communication involving different execution units which cooperate with each other in exchange of information controlled by a high level operation is known as collective communication. Collective communication construct is a high level operation which involves a group of UEs(units of execution) and exchange of information between them. This high level operation can

be a communication event purely (e.g., broadcast) or some computation (e.g., reduction). (2) Three types of operations are usually performed using different constructs of collective communication.(7)

1. Synchronization, in which processes have to wait until all group members have reached a point of synchronization.

2. Data movement, that is performed using broadcast, scatter/gather and all to all.

3. Reduction, in which data is collected by one member from all other members and any operation is performed on the data collected. The operation can be min, max, add, multiply, etc.

Broadcast, that falls in type 2 of collective communication has been introduced in RIAs through the implementation of WOvNet. Its implementation is executed by using p2p iterative unicast based overlay multicast.

1.2 Rich Internet Applications

RIA is basically a web application which is designed to perform those features and functions that are normally performed by a desktop application. In RIA the client side does not require any kind of software installation to work, this is because RIA runs inside a web browser. However, there are many features of RIA that require the installation of third party plugin, by the end users. Yet, there are number of RIAs that can only work properly with some specific browsers.

1.2.1 Flash Based RIA

HTML 4 did not have the functionality for running most of the RIAs so some methodology was needed to enable internet browsers to perform like desktop application software. Hence, most of the features of RIAs were only confined to third party plugins. Applications build on these plugins required the installation of corresponding plugins by the end user to run this functionality. These applications included video on demand, online streaming, online gaming etc. Some of the third party plugins are Adobe Flash, Java FX and Microsoft silver light etc. The most common way of delivering RIA features to desktop users is Adobe Flash. Statistics of 2012 revealed that Flash has an almost complete market penetration on desktop computers. (8)

RTMFP is also an examples of third party plugin that is used for building RIA applications. It creates a p2p based overlay network between clients' browsers. This overlay network is then used for realtime video streaming.

1.2.2 HTML5 Based RIA

HTML5 is a core mark-up internet language used for content presentation and structuring of World Wide Web and it introduced many new features including support for RIAs. Previously this cannot be done in HTML4 due to following 3 main factors (1)

1. The limited capacity of computers at that time. Hence, not much need was felt for advance features in HTML4. But, computers are blindly fast now, with many of us running smart phones faster than the desktop PCs used just six years ago

2. The limited speed of internet at that time, as it was dependent on the user's dialup modem. Now a day, mostly clients are using broadband based internet services and internet connections are also being measured in megabits per second.

3. At that time web was passing through its revolutionary phase and being molded into new medium. But now it has been more standardized now and hence it is more convenient to introduce specifications for native implementation of rich features of HTML5.

1.3 Problem Statement

RTMFP (Real Time Media Flow Protocol) which is Adobe's proprietary protocol provides an option of web based overlay network but it has few limitations

1. It is based on third party plugin of Adobe flash player.(9)

2. It is not open source.(10)

3. It cannot serve Android and iOS based clients as Adobe flash player is blocked by both of these.(11)

4. It only serves UDP and cannot operate if all UDP ports are blocked.(12)

5. The overlay network only supports audio/video so arbitrary data cannot be sent or received.(10)

HTML5 doesn't offer any option of overlay network creation but there are sufficient ingredients available that can be integrated to offer an option of plugin free overlay network creation. The target of this research is to implement a web based overlay network which is both plugin free and open source. As existing option of web based overlay network creation cannot be executed without third party plugin and is also not supporting smart

phones (Android and iOS). Hence, there is need to implement a plugin free web based overlay network, which is also optimized for smart phones.

Main idea behind WOvNet is to implement web based overlay network is to use the native features of HTML5. Its algorithms share the burden of server with connected clients by shifting traffic from client/server to WebRTC based p2p, in case of permanent power source. It also brings burden back from peer, if power source of that peer is removed. It not only helps in decreasing the server burden but also consider the preservation of battery for portable devices.

Flash playback has never been supported by Apple both in the iPhone and iPod Touch, released in 2007.(13) This caused some of the issues like following:

1. At that time a large part of video present on the Internet was Flash-based video.

2. Most of the animated content, on the internet was flash powered including games, banner ads and e-learning content. Flash plugin was required for this content which could be installed on desktop computers but not on iOS devices.

The iPad has the most popularity as a tablet on the market. In 2011 two third of tablets sold were iPads(14). Also, the trend for buying and paying per application is more in iPad users .(15) It is clear from these stats that variety of platforms may be created without ignoring the iPad users.

There is another factor which has increased concern in delivering flash content to mobile and tablet users. In the past, Android has supported flash Plug in but in November 2011 Adobe announced that it was stopping future development of flash Plug in for mobiles.(16) then it was further announced by Adobe in June 2012 that support for flash mobile has been terminated for use on new versions of Android.(18).

A formal problem statement is given as:

"Existing collective communication option in RIA is based on Adobe's flash player, which is not supported in smart phones. Hence, we are targeting to introduce a collective communication construct for RIA which will be plugin free and optimized for portable devices. It will also incorporates algorithms for building adaptive topology of WOvNet to reduce link stretch, which eventually results in reducing packet delay."

1.4 Contributions

There is no existing plugin free web based overlay network in our knowledge thus we are going to explore different options that are natively available in browsers to implement a WOvNet. Motivations for this research are

mentioned in coming portion and it will be concluded by mentioning goals of our research.

1.4.1 Motivations

Adobe itself believed to replace its Plugin based features with HTML5(17) standards(18) and existing overlay network in web is based on Adobe Plugin. Hence this research will not only focus an implementing Plug in free WOvNet, but also it improves the following limitations of RTMFP based overlay network:

- RTMFP only supports UDP solution was needed which can operate on both TCP and UDP.
- RTMFP cannot serve Android and iOS based clients and WOvNet not only support these smart phone client with latest browsers but also provide optimization for battery consumption.
- RTMFP cannot be operated cross-domain(19) so there is a need for a solution which can also be configured for cross domain implementation and WebRTC can be configured for cross domain so WOvNet can also operate cross domain as it is based an WebRTC.
- RTMFP creates overlay network for only audio/video streaming. An overlay network which can be used to transfer arbitrary data is required. This option can be utilized for creating multiplayer games, and co-browsing to support collaboration at a larger scale.

1.4.2 Goals

This research work includes the implementation of Plug in free web based overlay network (WOvNet). The main goals of this research are

- Implementation of Plug in free collective communication construct for RIA based on web based overlay network (WOvNet).
- Adaptive switching of traffic between HTML5 standards on the basis of charging status of portable devices.
- Improvements in network topology to reduce link stretch by grouping geographically closer peer in similar cluster.

Moreover, WOvNet is optimized for data transmission as it is based on WebSocket and dataChannel API of WebRTC. For the same reason, it can be used for implementation of many interesting real time applications like multiplayer games, co-browsing and collaborative e-learning solutions.

1.5 Challenges

The target of this research is to build an overlay network based on HTML5 standards but it is not natively supported in HTML5. Hence, no existing support over the Internet was available. Moreover, it is really difficult to build the proper evaluation environment due to non existence of global clock in WOvNet.

1.5.1 Global Clock

This research involves implementation of an overlay network which is based on distributed browsers all over the internet. And there is no global clock available on internet browsers to have comparative readings on multiple browsers. This challenge has been tackled by using round trip delay. In this way, packets are transmitted to the targeted browser and then sent back to the source. Since, the source has same clock to maintain the record of sending and receiving time, therefore, we can have accurate clock readings. More detail of this approach is coming in chapter 4 of "results and discussion".

1.5.2 Less Support for Latest Technologies

It has been already mentioned that HTML5 has no built-in support for overlay formation and we have implemented it by our own, hence there is no relevant content available online. First we need to study different related approaches that are in practice for the similar system, then we have molded these approaches to adopt in browser environment. Details of existing approaches are coming in chapter 2 of background and related work and architectural details of WOvNet are coming in chapter 3 of design and implementation.

1.6 Thesis Organization

The rest of the thesis is organized as follows:

Chapter 2 discusses background and history related to the current research, and reviews the relevant literature aimed at finding the solution of the proposed problem.

Chapter 3 focuses at the design and implementation of WOvNet and also elaborate the proposed methodology.

Chapter 4 enlists the results along with detailed discussions.

Chapter 5 explains the possible improvements in WOvNet in future and also concludes this thesis.

Chapter 2

BACKGROUND AND RELATED WORK

Peer to peer (p2p) collective communication reduces the load on server, as all the users in the network are not receiving their data form the server and peers are sharing the burden of server. Server only creates and monitors network connections. The data transmission between the peers, is maintained by a virtual network called overlay network.As discussed in the previous chapter that there is no web based solution for creating a plugin free overlay network and existing option of overlay creation is dependent on third party plugins.

With the development of HTML5 many RIAs are now do not depend on third party plugins, as in the era of HTML4. However, HTML5 still not offer any functionality for WOvNet creation. Besides this, existing functionalities of HTML5 can be integrated to create WOvNet. This chapter focuses on these functionalities in details. Moreover, this chapter also discusses the ideas that inspired design and implementation of WOvNet.

2.1 History

RTMFP based collective communication introduced in RIA to minimize the server load. However, it is already mentioned that HTML4 was not capable enough to implement the said functionality therefore, third party plugins are used for the same purpose. This plugin of RTMFP was introduced in flash player 10. (20)

2.1.1 HTML4

Version 4.0 of HTML was got published on 18 December 1997 and was reviewed and revised on 24 April 1998 (21). HTML4 supports a lot of extra multimedia options, languages for scripting, style sheets, enhanced printing

facilities, and documents that have more access to disabled users. HTML4 has also made progress towards document internationalization, and targets the goal to make World Wide Web truly worldwide. HTML4 has also ruled for more than a decade due to the reasons mentioned in section 1.2.2.

2.1.2 HTML5

In 2004 the Web Hypertext Application Working Group (WHATWG) was started by a small group of people. This group created the HTML5 specification.(22) Work was started on new features specifically related to web applications on the area which was felt to be most lacking. Web 2.0 was also coined in this era. And it really was a second new web, as more dynamic web sites like social media which required a lot of new features began to develop. The first working draft of HTML5 was developed in 2008 by W3C(23). Because HTML5 gives a solution to very practical problems, browser developers are implementing its new features, even though the specification is not yet complete and new features are emerging day by day. The browser vendors' feedback helps in improvement of the specification. HTML5 is developing rapidly to implement practical improvements to the web platform. (26)

2.1.3 Transition from HTML4 to HTML5

HTML5 continues on the success started by HTML4. It is not necessary to throw away the existing markup and it is also not necessary to relearn things that are already known. Web applications that worked earlier in HTML4, will still work in HTML5. HTML5 has included new input controls but it also has almost all controls of HTML4. Some of these controls are additions that are long awaited. For example, sliders and date pickers but many are more fine additions. As an example we can say, the email input type has an appearance like a text box, but mobile browsers will be able to customize their on screen keyboard so that they can easily type email addresses.

2.1.4 Plug in Based RIA

Earlier, most of the features of RIAs were only confined to third party Plugins because of the limited functionality of HTML4. Hence, many famous RIA were build using third party plugins. Following are two famous plugin based RIA applications:

BigBlueButton build specifically for elearning and is dependent on Adobe flash player and Java. (24)

Screenr is a famous web based screencasting tool and it requires the installation of Java runtime. (25)

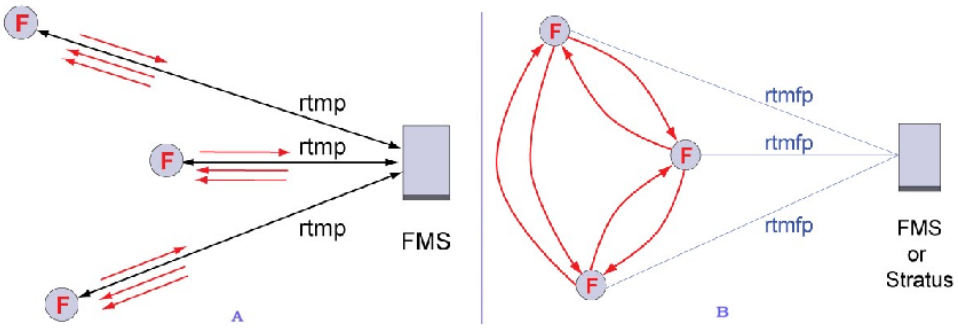

Figure 2.1: Working of RTMP and RTMFP

2.1.5 Plug in free RIA

The internet applications which do not need any third party Plug in to run content on the internet are plug in free RIAs. HTML5 along with JavaScript,is the core mark up language for development of such applications. Using these applications one can natively enjoy that features in browsers that were earlier confined only to desktop applications.

2.1.6 Transition from Plug in Based to Plug in free RIA

The transition from Plugin based RIA to Plugin free RIA has been possible by the tremendous increase in the capabilities of HTML5. These features enable browsers to run such content on web based applications which previously can only be executed using third party Plugins. Some examples of transition from Plug in based RIA to Plug in free RIA are given below.

- WebGL and Canvas to replace flash games (27)
- WebRTC to replace realtime communication.(28)
- Audio and video tags to replace flash based players.(29)

2.2 Plug in based communication options

Third party plugins are used to enhance the capabilities of browsers. However, there are security issues while installing these Plug ins and also the services that are build on these Plug ins cannot be executed without the installation of these Plug ins by the end users. In coming section, we are going to discuss RTMFP in detail and also RTMFP which was originated after RTMP.

2.2.1 RTMP

Real time media protocol (RTMP) is a plugin developed by Adobe over TCP and by following client/server architecture. RTMP is a proprietary protocol developed by Macromedia for audio/ video streaming and Action Message Format (AMF) data over the Internet, between a Flash player and a server. RTMP is designed for high-performance in audio, video, and other data transmission. RTMP is an efficient open source networking protocol works at real time and supports the efficient exchange of messages, synchronized data, audio and video. This protocol employs remote Shared Objects technique so that connected users can share data and user interfaces in real time. (32) RTMP follows client/server architecture and its shown is shown in part A of fig 2.1 (33)

2.2.2 RTMFP

RTMFP is a proprietary protocol of Adobe that enables the p2p communication between browsers and develops an overlay network for streaming (38). It works on UDP and thus helps in reducing latency in realtime applications. However, no one can join this overlay network of RTMFP without installing the third party plug-in of Adobe flash player. Hence, we can say that peer software for RTMFP is a browser equipped with Adobe flash player. Working of RTMFP explained in part B of fig 2.1 (33)

2.3 Plug in free communication options

HTML5 has introduced many new communication options that are natively supported in latest browsers. Some of these options are mentioned below:

XHR2 is the improved version of XHR1 which implies cross-origin resource sharing. (34)

Cross document messaging enables different DOMs to interact with each other. (35)

WebSocket enables bidirectional communication between browser and server. (36)

WebRTC enable p2p realtime communication between browsers.(37)

We have utilized 2 of these HTML5 communication options to build a browser-to-browser (b2b) overlay network. As this overlay network is built on HTML5 APIs therefore it doesn't require the installation of any third party plug-in and only a latest browser at client side , supporting WebSocket and WebRTC, is enough for a peer to participate in this overlay network.

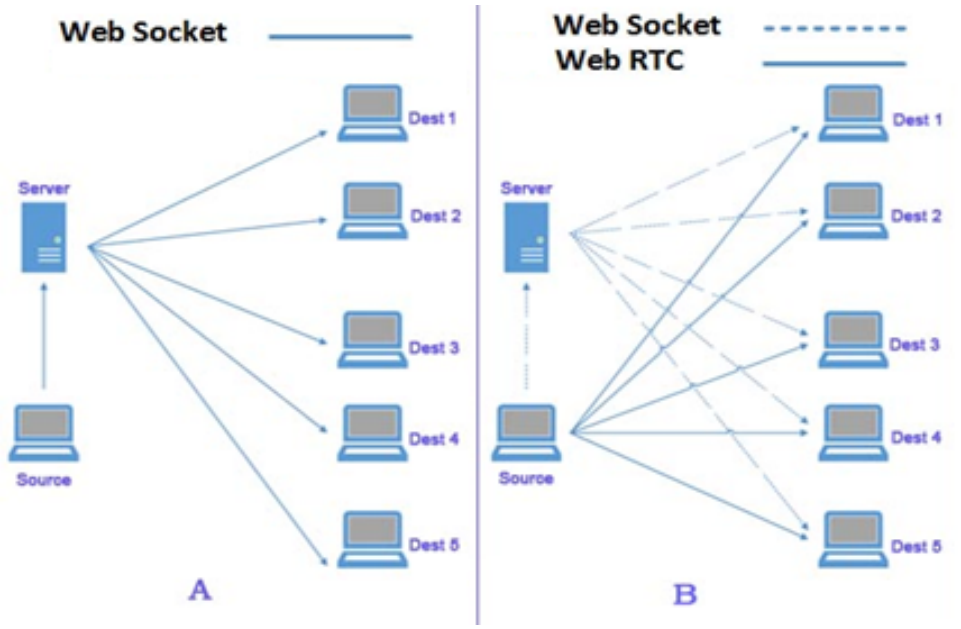

Figure 2.2: Working of WebRTC and WebSocket

2.3.1 WebSocket

In case of WebSocket, source browser will send the data to the server and then it's solely server's responsibility to broadcast it to all of the connected destination browsers.(40) Communication flow of WebSocket is also available in part A of figure 2.2. In this scenario, source browser will bear the minimum burden and thus gains the advantage of preserving its resources (including battery). However it shifts all burden of broadcasting from client's device to the server and thus server will experience all load of broadcasting. Moreover, as data is initially sent from source browser to server and then it will be forwarded to all other destination browsers, therefore increasing member of destination browsers will increase the server's burden.

2.3.2 WebRTC

WebRTC provides an alternate option of broadcasting, without involving the server and it is achieved by directly sending the data from source browser to all of the destination browsers. Although, all clients will remain connected to the server using web socket and also use it for signaling, however data sending will only take place in peer-to-peer fashion using WebRTC. These details can also be checked in "Part of Figure 2.2". WebRTC connects the browsers in peer to peer fashion; therefore it shifts the burden from server to the sender's device. Hence it consumes the resources of source browser's device and consequently device of source browser will experience more bat-

tery consumption, resulting in earlier drainage of battery. Moreover, limited number of WebRTC based p2p connection can be created using WebRTC. We have shifted traffic from sender to server and then created WOvNet to reduce the burden of both sender and server. Part B of Fig 2.2 shows the working of WebRTC and more details of WOvNet are coming in next chapters. As, data communication in WOvNet is based on WebRTC, therefore, we need to know the following 3 APIs of WebRTC:

2.3.2.1 GetUserMedia API

WebRTC has an API to access local devices like webcam and microphone. This API only accesses local devices and does not provide real time communication. It is natively supported in browsers and thus removes the need of plugins to be installed for capturing system devices. By using this API, media can be accessed by JavaScript code and it generates a stream object. getUserMedia API method returns an object known as media stream which includes media stream tracks containing actual video and audio. Media Stream also handles the synchronization of all media tracks included in it.So,a proper playback is available at application level. This assures that audio and video will always be synchronized.The video tag of html5 can play both local and remote streams and thus provides a complete Plug in free solution for implementing video in browser.(41)

2.3.2.2 Peer Connection

Peer connection API is used to transfer data between peers. This API includes all the mechanism needed to transfer real time data as well as it also handles some signaling messages with certain JavaScript methods. In WebRTC interactive connectivity establishment(ICE) is a modified version of (SDP) session description protocol that allows multiple media descriptions. By using this, it multiplexes all traffic on a single port. Peer connection API also provides signaling and NAT traversal techniques through ICE, using STUN and TURN. ICE try to connect through STUN server and in case of failure, it uses TURN server as proxy to channel the traffic through this server. STUN works in almost 85% cases and remaining 15% are served through TURN. Signaling is not standardized in WebRTC and is chosen by developer at application level.(42)

2.3.2.3 DataChannel API

(43) Data channel API can be used for sending arbitrary data. It also shares the peer channel API for establishing connections with peers. We have also used this API in our research to implement WOvNet.

Figure 2.3: Browser Support of Geolocation API

2.4 More HTML5 APIs

2.4.1 Geolocation API

(44) Geolocation is a technique for finding someone's location in this world. There are more than one way to figure out the target. E.g. IP address, wireless network connection, which cellular service tower your phone has a connection with, or a hardware designed for finding latitude and longitude of the person you want to find by using GPS.

Your location is shared with websites you trust via Geolocation API. JavaScript on the page has the latitude and longitude available and it sends it back to remote web server so that one's location can be found. It is clear from table in fig 2.4, that most browsers support the Geolocation API. (46)

2.4.2 Battery API

Battery API Provides 4 different values to get states of client's device and also four different events to bind functionality against any change in battery status. it checks the current charging status of the device. Battery level of the device and tells how much time will the battery take to drain completely. Battery API has following four different properties and four different Events. (47):

2.4.2.1 Properties

Four properties of Battery API can be called at anytime to get their particular values. Details of these corresponding values are coming next:

Charging It returns true if device is attached to some power source and returns false if no power source is available and device is running on battery.

ChargingTime It represents the time left in seconds for complete charging of battery.

DischargingTime enables bidirectional communication between browser and server. (47)

Level It shows the current level of battery from 0 to 1.0

2.4.2.2 Events

Battery API also offers four different events that can be used for detecting changes in battery. Following are these four events of battery API:

OnChargingChange It returns true if device is attached to some power source and returns false if no power source is available and device is running on battery.

OnChargingTimeChange It represents the time left in seconds for complete charging of battery.

OnDischargingTimeChange enables bidirectional communication between browser and server.

OnLevelChange This event automatically triggers once there is any change in the charging level of battery.

2.5 Related Topologies

A topology is an arrangement of various nodes of a network linked together through different conjurations. Topology can be considered as a virtual network structure or shape. This structure focuses on the logical map of network and it does not match the physical layout.

Topologies are used to concentrate interfaces, extend cable connections and convert data formats. Normally routers are used for routing the data packets within LAN or WAN from source to destination.

For example the establishment of the call is decided by NAT traversal problems. In WebRTC this problem can be solved with the usage of STUN and TURN. Yet the establishment of the call is impossible in some restricted environments. The usage of NAT in WebRTC increases the complexity of the browser internals.

The servers of STUN and TURN must be reachable to provide the best candidates that are evaluated by ICE mechanism. But all these mechanism may fail in some restricted environments. To enable connectivity in very restrictive environments, WebRTC allows UDP and TCP(using TURN) packet transport.(48):

2.5.1 Point-to-Point

The simplest topology to communicate between two given points or node is point to point topology. The users are provided with simple real time communication between them. The common use of this topology is two way communication between users. It has the following two types:

2.5.1.1 Dedicated

As the name suggests the dedicated topology is seen by the user as a permanent channel for communication between two endpoints. Many switched telecommunication systems have the possibility of establishing a permanent circuit.

2.5.1.2 Switched Topology

A point-to-point circuit that can set up a dynamic path and dropped when no longer needed is defined as switched network. this can be done by either circuit switching or packet switching

2.5.2 One-to-Many

It is also known as star topology which is used for the synchronous and asynchronous media streaming.The one-to-many topology consists of a central node which transmits streams to the rest of the nodes connected to it. In WebRTC,the central node is providing (server) the connection to all the clients.So by connecting all of the clients to a central device the chance of network failure is reduced .

But it provides a high dependency on the central node,if the central node is disconnected the network becomes inoperable and all the clients will loose their connectivity.

One-to-many topology gives a better performance to improve the recovery process.

Resilience against the endpoints fails this isolation also prevents non-centralized failure from affecting the rest of nodes.Star topologies also make it easier to detect and localize faulty nodes and adding or removing from the end points is easier.

In WebRTC star topology we have video,audio and data transfer from one source to multiple endpoints.It will cause a heavy burden on the central node and hold-up behavior of the central node can be a limitation in this case.Due to one-way communication the media delay occurs which is not a good indicator for this topology. Star topology is commonly used for the audio and video streaming from one source to multiple peers.Tv media, conferencing are the best examples of this scenarios.

A common example is live streaming,i.e the media providers can use this type of topology to send different media files from one source to multiple endpoints.This in turn helps the subscribers to watch any desired channel.

2.5.3 Many-to-Many

In this topology each node has an interconnection with every other node in the network. It is also known as full mesh. The data received by a node can be be from every other node in the network. In a full mesh topology all the peers are interconnected by increasing the number of connections and resources.The value of full meshed networks depends upon the number of subscribers and the total number of the peer connections available in a mesh network.The number of peer connections depend upon the number of people in the conference which can grow exponentially to the square amount of the nodes.

The major advantage of Many-to-many topology is that the data can be transmitted from different devices at the same time. But there are some disadvantages as well like this topology is more costly than the other topologies and the maintenance of this topology is difficult and also has a high traffic load. Wireless networks use this type of topology. It can also be used as a file delivery network for data transfer between peers.

2.5.4 Multipoint Conferencing Unit (MCU)

MCU is a device which is used to bridge streams in conferences it mixes multiplexes and encodes different types of media source to be sent over one gate way.

MCU is a good alternative for designing video applications in webRTC.MCU is a very common component in real time media topologies which multiplexes all streams of the call into same channel it helps the end devices to handle less load for the source in which multiple peers are connected to same MCU which can multiplex media into one unique stream and later forward it to all participants of the call it also gives good scalability from the client's perspective. Multipoint conferencing Units might have to encode or decode in a very little time which can be difficult in real time applications but it provides different encoding options to stream output to the link conditions.(49)

There are some drawbacks of MCUs, in MCU the call quality is affected if the nodes do not show good performance and the receivers do not get expected response. The total number of participants and the number of conferences decide the load in MCU.A scalable implementation is needed when handling multiple conferences. An alternative is the usage of large amount of MCUs to provide more scalability. Figure 2.4 shows an example of MCU.(50)

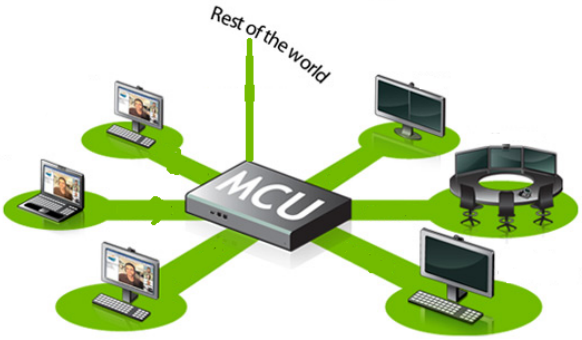

Figure 2.4: Multicast Conferencing Unit

2.5.5 Overlay

The ability of a peer to transfer data to a third party is called overlaying media streams.The implementing systems of overlay are those which require the media to be forwarded from one peer to another. Multiple topologies can be constructed using overlaying. In WebRTC,Overlay is used to forward media streams from one node to another.This feature is very helpful in building scalable applications.Multiple layers of nodes are designed to transmit the media to each other. The usage of overlaying in mesh or one-to-many topologies can be another alternative in such environment nodes can forward the streams to provide reliability on the network. But WebRTC does not provide native support for the media overlay,and it is planned to implement those versions in future versions of API.Considering WebRTC,this scenario can be implemented in group communication. Overlay has some drawbacks , those topologies that rely on the overlay are slow in distributing the data.And also overlay may provide some duplicate packets at certain points. Some examples of overlay are Hub-spoke and Tree.(51)

2.5.5.1 Hub-spoke

Hub-spoke is an example of overlay which is made by multiple nodes and arranged like a chariot wheel.Spokes are connected to the central hub and the traffic flows along these spokes. Hub-spoke performs very well in some cases because it requires less connection to perform a full communication in network. Hubs are designed to forward traffic from one node to another. Some advantages of hub-spoke are: the number of paths which are used to connect all the nodes is significantly less than the other topologies.Due to less amount of paths this topology is more efficient in terms of resources.And the new spokes can easily be created without effecting the existing nodes in network. The reliability of the network is based on the central node.The dynamic hubs can reduce this dependency.

2.5.5.2 Tree

This type of topology is based on node hierarchy,the highest level of this tree consists of a single node which is further connected to one or more nodes which forward the traffic to other layers of topology.The tree topology can adapt to the required number of end users. The tree topologies are scalable and manageable. If any branch fails it is easy to find the broken branch of the tree and it is easily repairable and the expansion of this network is easy. There is a problem of connectivity if a node fails to connect all the other layers under that node will be affected and traffic will not be forwarded further.This is why overlay is a bit crucial for tree topology which is widely used in media streaming. Tree topologies are widely used not only for media streaming but to provide wireless coverage in difficult areas.It acts as a hot spot and extends the coverage of the wireless in remote areas.

2.5.6 Summary

Different topologies can be used for real-time communication. This section has described the ones that can adapt to WebRTC in different use cases. However, some of them are still not possible to implement due to technology constraints. During this research the focus of study was on point-to-point, many-to-many and MCU topologies. These topologies can adapt to WebRTC application requirements.

2.6 Multicast

Multicast refers to the one-to-many or many-to-many communication. In a multi cast system with n clients, the server does not need to send n identical packets. With one-to-many communication, only one identical packet is sent from the server. Each packet is replicated and forwarded on networks node such as router or client. For many-to-many communication, a client obtains some packets from other clients instead of from the server. Multicast was first proposed by Dr. Stephen Deering in his doctoral work 1988 (52).

His work proposed two protocols: IGMP (Internet Group Management Protocol) and DVMRP (Distance Vector Multicast Routing Protocol). These two protocols are multicast solution suite at the network layer. Motivated by Dr. Deerings work an experimental Multicast Backbone (MBone) was proposed. Its first practical application was implemented by IETF (Internet Engineering Task Force) in March (53)

The application sent the audio data of a meeting in San Diego to twenty sites worldwide. In the beginning the MBone was only a virtual multicast network in which the multicast routing function was provided by workstations.

Furthermore, many protocols for IP multicast have been proposed and deployed, for both Intra-domain and inter-domain routings. At present IP multicast has been included with most commercial routers' standard set of protocols . The commercial deployment of IP multicast has become possible. However, the deployment in the Internet is very slow due to many commercial and political problems.

To solve the deployment problem of IP multicast, it has been proposed to implement multicast at the application layer. In the beginning the end hosts take over the functionalities of multicast enabled routers. The end hosts replicate and forward the packets. They also manage the group membership and establish the multicast tree. This multicast is named Application Layer Multicast (ALM). Some examples using ALM are Narada, NICE and TAG, etc.(58): ALM also has its drawbacks such as unfair resource distribution and difficulties in handling frequent churn.

The implementations of ALM is not very successful in real world. Recently, peer-to-peer (P2P) technology has been adopted for live video streaming. P2P has been successfully implemented in file sharing and Voice over IP (VoIP). Both commercial and academic implementations of P2P streaming systems have achieved business success, for example, pplive, ppstream, Cool Streaming, etc. Multicast implemented at the application layer is known as P2P multicast. P2P multicast is divided into two categories one is tree-based and the other is mesh-based. The tree-based P2P multicast is to establish multicast tree at the application layer, such as the traditional ALM. The mesh-based P2P multicast implements P2P technology similar to Bit Torrent. This section presents multicast both at the network layer - Internet Protocol (IP) Multicast and at the application layer - Application Layer Multicast and P2P Multicast.

2.6.1 Internet Protocol (IP) Multicast

Internet Protocol (IP) multicast has a wide use in enterprisers, commercial stock exchanges and multimedia content delivery networks. IPTV such as distance learning and video conferences is a common application of IP multicast. Multicast service is implemented at IP routing level in IP multicast i.e. at network layer. With this service the source only transmits one copy of individual packet. Each packet is replicated at routers band forwarded to multiple receivers simultaneously. The key concepts of IP multicast include multicast group address, multicast distribution tree and group membership management.

2.6.2 Overlay Multicast

Overlay multicast is the Multicast service which is implemented at overlay network layer . The overlay is a logical layer for communication between

peers. Users which participate in a multicast service form an overlay network in which unicast is utilized among user pairs for communication of data. The users in overlay multicast separately can handle routing of peers, manage groups and tree creation with no help from internet routers.

An overlay network of different Nodes connected at application layer can be used to implement Application Layer multicast. The use of ALM has increased so that the problem of non-scattered deployment of IP multicast across heterogeneous networks can be overcome resulting in improvement of wide-area service availability and performance. Quick deployment simplicity in configuration and better access control is offered in the network through an overlay-based approach. The general method of building an application level multicast architecture employs network characteristics tracking and suitable topology building by compelling the end users to self-organize themselves into logical overlay networks for enhanced delivery of data. Data is delivered via a data delivery tree which can either be the overlay network itself or it can be embedded in the overlay. (58)

2.6.2.1 Comparison with IP multicast

As Compared to overlay multi cast IP multi cast shows higher efficiency and performance. But due to factors like cost of deployment issues in inter-domain deployment IP multicast has not found a wide use by ISPs especially in WAN (wide area networks). Overlay multicast is easier to implement than IP multicast because it has no dependency on deployment of routers. In overlay multicast the overlay network is created over available internet services. Multicast group members are connected to the overlay network.

2.6.2.2 Classification of Overlay Multicast

There are three general aspects of overlay multi cast architecture (55)

- Multicast agency
- Topology usage
- Tree construction approach

Multicast agency can be further classified into

- End system multicast
- Proxy based multicast
- Hybrid multicast

Tree construction approach is the method of designing an overlay structure it has further two approaches which are described below

2.6.2.3 Tree

In tree or direct tree method the tree is built directly i.e. the parents of the members are selected from existing members which are known to them. The tree approach is the most efficient in terms of delay and bandwidth optimization. The disadvantage of this approach is its vulnerability to dynamics. If a node leaves or fails it can result in unavailability of data in its descendants. This can further result in extra cost as the tree has to repair itself so it is not optimal.(56):

2.6.2.4 Mesh

Another approach for overlay multicast is the mesh approach. Multi cast tree is created in two steps. Initially, connected overlay network which comprises of virtual paths that are basis of building a spanning tree is constructed finally, multicast routing protocol is run over the mesh and a multi cast tree is formed. it is data driven randomized overlay and has better resilience

2.6.2.5 Hybrid

Hybrid proxy/end-system model (HSM) is another type of overlay multicast which has the advantages of both proxy-based and end system multicast to compensate performance and cost of the network. In hybrid overlay the tree approach can be merged with mesh overlay to obtain better utilization of bandwidth (57)

2.6.3 Resilient Overlay Network (RON)

A Resilient Overlay Network is a network in which application traffic is routed over the network by finding those available and low latency paths which usual routing protocols cannot identify. A RON requires peers which are placed in different autonomous systems (ASs) and the latency and connectivity state with neighboring peers is determined by these placed peers. When it is required by an application to send packets to any other application the sender connects itself to the nearest RON peer known as the "ingress peer" and the packet is issued. The preference of routing network over overlay network is determined by this peer. If yes then the packet will be sent directly to the end point otherwise it is routed at overlay level using low latency path. RONs are small networks comprising of at most 50 nodes and are dedicated for specific applications (4)

2.6.4 Summary

This section discusses about multicast phenomenon and types. Multicast means transfer of data from one user to many users or it can be many to

many communication. There are two main types of Multicast, IP multicast and overlay multicast. IP multicast has better efficiency but it is not cost effective so it is not used by network service providers. On the other hand overlay multicast is easy to implement because it does not depend on router deployment. Overlay multicast can be deployed by using two approaches i.e. tree approach and mesh approach. Overlay multicast implemented at application level is termed as application layer multicast. ALM can be used in non-homogeneous networks to improve service availability and performance.

2.7 Inspirations for architecture

Following are some of the implemented ideas which gave the inspiration for the designing architecture of WOvNet.

2.7.1 Spotify

Spotify is a good case study that is also using a communication backbone based on both peer-to-peer and client server architectures for delivering the Music-on-demand streaming to millions to of clients. One of the major reasons for their success over competitors is to offer free service to millions of users. According to a study only 8.8% data comes from their servers 35.8%(59) comes from peer-to-peer communication and 55.4% comes from local cache. In this way, Spotify reduces the operational cost by shifting 35.8% servers load to the clients and can offer free or cost efficient services to millions of clients.(60) Spotify connects smart phone clients as free riders and doesn't create outgoing connections for them. Because of this fact that these devices have limited resources.

2.7.2 RELOAD

RELOAD (Resource Location and Discovery)(4) has been adopted by the working group of P2P SIP as initial point1 for the primary P2P-SIP protocols. Overlay functions of RELOAD are divided into forwarding layer generic routing layer and a topology module which is plug able for supporting many different network topologies. Routing layer takes the decision of next hop on the basis of routing table which is implemented in network topology module. A central registration server in RELOAD provides a digital signature to the peers before joining the overlay. We have also followed the same approach by assigning a unique identifier to each peer of b2b overlay before joining the overlay network. List of all peers along with their positions in the topology and their unique identifiers is also maintained at server for sustaining the overlay network of b2b.

2.7.2.1 Forwarding layer

Forwarding layer in RELOAD manages the algorithm independent message transportation and also managing the NAT traversal and secure transportation of messages. This is implemented in b2b overlay using WebRTC as it ensures the secure delivery of messages between the peers independent of the network topology of b2b overlay. WebRTC also provide NAT traversal.

2.7.2.2 Generic routing layer

The second layer in RELOAD is the generic routing Layer.The generic routing layer determines next hop decisions by using the routing table in the topology module. It means that it does not make decisions by itself it is dependent on the routing table to do so.We have also maintained a routing table at server which is responsible for next hoop decision.

2.7.2.3 Plug able routing module

Topology module of RELOAD is plug able and thus any supporting topology can be plugged into RELOAD for building the overlay network. As this layer of module doesn't provide much information about the topology therefore, we explored different options of topologies for implementing browser to browser(b2b) overlay in section 2.6.

2.7.3 Responsibilities of Service Provider

Three main responsibilities are focused for the WOvNet as as server provider of overlay network. (4) It has to connect peers in form of a network and provides service advertisement to add third party services. It has to provide network resources and service activation to the users. It further checks the quality of service and takes measures to improve quality in future. Next is the explanation of these three responsibilities.

2.7.3.1 Service Creation

In service creation, service provider must ensure the initiation of service for the newly arrived client.

2.7.3.2 Service Provisioning

In service provisioning, service provider must ensure the quality of service according to the agreement with the client.

2.7.3.3 Service Assurance

Service assurance means that service provider must ensure the continuity of seamless services.

2.7.4 Summary

In this section two examples which use overlay architecture are discussed. Spotify and RELOAD are excellent applications of peer to peer networking. By using peer to peer architecture Spotify reduces the load on its servers by a large amount. RELOAD uses three overlay functions which are forwarding layer a generic routing layer and a plug able topology module. Responsibilities of an overlay ISP have also been discussed in this section. An overlay ISP provides service to different peers connected to the overlay network it also ensures the quality of service being provided and performs auditing to improve the services in future.

Chapter 3

DESIGN & IMPLEMENTATION

The previous chapter described about the history and background required for implementing the WOvNet. Concepts discussed in previous chapter have contributed in designing and implementing WOvNet. This chapter explains the methodology used for modeling WOvNet. Inspired from the RELOAD, WOvNet is also divided into three layers. Details of these three layers are mentioned in next section, which is followed by the details of implementation and responsibilities of WOvNet.

3.1 Three Layered Architectural Design

RELOAD has been a major inspiration in the design of WOvNet as it is also divided into three layers. Node.js provides the base for generic routing layer similar to RELOAD. WebRTC helps in implementing the forwarding layer by doing the data propagation. While the pluggable topology is maintained by the WebSocket. Next sections explain all of these three layers in detail.

3.1.1 Routing Layer

Generic routing layer described earlier in RELOAD is replicated in WOvNet by implementing a routing table at Node.js server. This routing table further contains 7 different entries for peers of different levels. A peer that is not a free rider can hold up to maximum of 5 connections. Although this number is static but it is not a fixed value and the number of outgoing connections on WOvNet can be changed explicitly. However, constant number of connections help in maintaining the overlay network. (4)

There are also some peers that are not considered for creating any outgoing connection and these peers are known as free riders. Overall routing table will have following seven lists, all maintaining peers at different levels.

list -1: This list holds all the free riders. Free riders are the peers that are not connected to any power source and are working on a battery e.g. smart phones, laptops

list 0: This list holds all the peers that are running on 5 outgoing connections and no further connection can be made with these peers.

list 1: This list holds all the peers that are running on 4 outgoing connections and can provide 1 more outgoing connection.

list 2: This list holds all the peers that are running on 3 outgoing connections and can contribute 2 more outgoing connections.

list 3: This list holds all the peers that are running on 2 outgoing connections and can facilitate 3 more outgoing connections.

list 4: This list holds all the peers that are running on 1 outgoing connection and can offer 4 more outgoing connections.

list 5: This list holds all the peers that are not used for creating any outgoing connection and can create 5 more outgoing connections.

Research shows that maintaining a constant degree for outgoing connection helps in better management of an overlay network. (4) Hence, we have also adopted the same approach by maintaining "5" as fixed number of maximum outgoing connections.

Dividing peers in different categories also helps in efficient searching at run time as searching algorithm starts from list 5 and if its empty only then it needs to see other list otherwise top most peer is returned for giving outgoing connection to the recently arrived peer. As a result, peer from list 5 needs to be shifted to list 4, as it has now created an outgoing connection while the new peer is pushed to list 5 incase of permanent power source and is being shifted to list -1 in case it is running on battery.

3.1.2 Overlay Network Layer

WebSocket provides the bidirectional client server communication and it is used for passing the control information from clients to server and vice versa. This control information helps in maintaining a stable overlay network. Whenever any change needs to be transferred from routing table to overlay network, it is done through WebSocket. This concept is further elaborated in part A of figure 3.1.

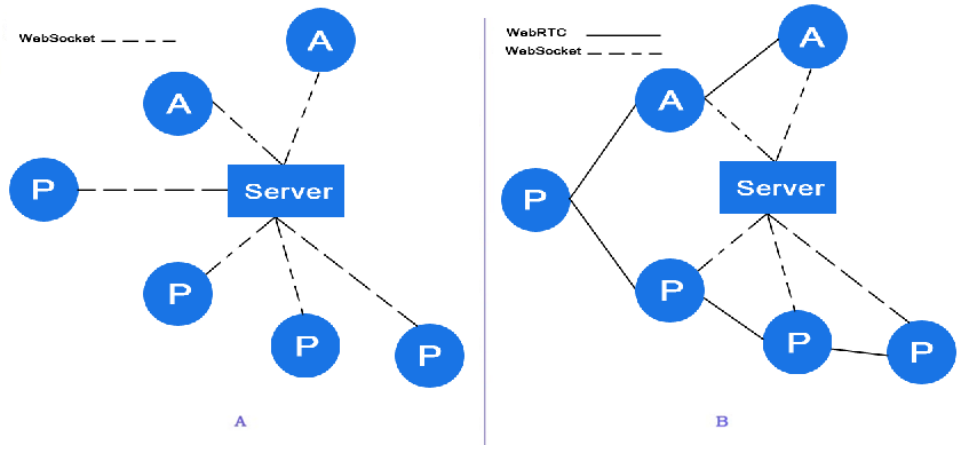

Figure 3.1: WebSocket

3.1.3 Forwarding layer

Unlike WebSocket, WebRTC doesn't use the client/server architecture and is based on peer-to-peer connections between browsers. Hence WebRTC is used for implementing overlay multicast by using iterative unicast. The server is only responsible for maintaining the hierarchy of overlay network as peers are connected to each other according to the routing algorithm applied by the server. Hence forwarding layer is based on p2p connections between browsers, using WebRTC. This is further elaborated in part B of figure 3.2.

3.2 Implementation

Implementation of WOvNet can be divided into five further phases. First phase consists of development of a simple tree based topology, where every new peer added in the existing hierarchy without considering any improvements.

Second phase introduced a peer ranking algorithm for filtering the free riders to reduce the burden on devices that are not connected to the permanent power source. Battery Status API of HTML5 is used for finding the charging status of peer in peer ranking algorithm. This algorithm also dynamically shifts a peer to free rider if permanent power source is removed and vice versa.

Third phase uses an algorithm of neighbor selection which results in creation of clusters or islands using the Geolocation API of HTML5. This algorithm ensures the placement of geographically closer peers in same cluster. This concept of cluster creation not only reduces the link stretch but also helps in reducing the packet delay.

Fourth phase introduced the algorithm of super peer finding. This phase

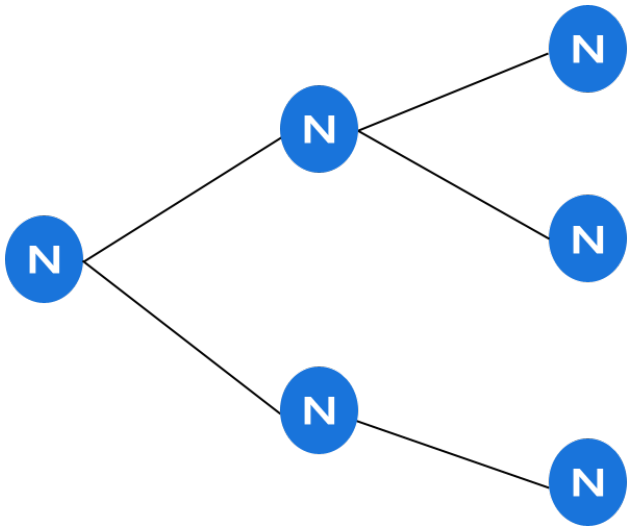

Figure 3.2: Simple Tree Based Topology

incorporates the efficient searching for super peer at the arrival for any new peer. It also maintains the changes in routing table at the arrival of new peer.

Fifth and last phase equipped the WOvNet with three responsibilities hence it can be further divided into three different sub-phases. These three responsibilities are service creation, service provisioning and service assurance. Algorithms implemented in second, third and fourth phase are also utilized for the implementation of three mentioned responsibilities.

First four implementation phases are described in this section while next section of responsibilities describes the last phase of implementation along with its 3 sub divisions.

3.2.1 Simple tree based overlay network topology

WOvNet was started by the implementation of a simple tree based overlay network topology. In this approach the incoming peers are joined in a network in the form of a tree. It does not consider the two algorithms of peer ranking and neighbor selection while building the network topology and overlay network topology is built by filling the earliest level first, with statically defined fixed number of outgoing connections.

Figure 3.2 shows a simple tree based network topology for static number of maximum 2 outgoing connections and is having three levels. First level is initially filled with two outgoing connections and this is the maximum number of possible outgoing connections in our scenario. After filling the first level, outgoing connections are being started from second level as displayed in figure 3.2. It is also clear from the figure that initially first peer from

second level has offered two outgoing connections and then second peer of second level started offering outgoing connection for peers of level 3.

WebSocket is used for maintaining the topology of the overlay network and peer connection is established through WebSocket. WebSocket does not take part in transfer of data between peers connected to the overlay network and it only provides a wiring mechanism for the peers. Since WebSocket provides bidirectional client/server communication, therefore server can update any peer of overlay network to maintain the simple tree based topology.

Transfer of data between peers connected to the overlay network takes place through dataChannel API of WebRTC. WebSocket is used for signalling to create peer connections and then dataChannel API provides the data sending interface in unicast. We have implemented iterative unicast based multicast to enable the collaborative communication between connected peers of WOvNet. Hence, first phase of implementation ensures following two main features:

- Creation of simple tree based overlay network topology.
- Implementation of iterative unicast based multicast.

3.2.2 Peer Ranking Algorithm

Peer ranking Algorithm enables the filtering of free riders from normal peers. A free rider is a peer running on battery and is placed in "list -1" of routing table. Thus a free rider never offers even a single outgoing connection to help the peers with portable devices to maintain battery for longer time intervals. Battery Status API of HTML5 is used for finding the charging status of peers.

Routing table with many different lists is also introduced in this phase and this helps in ranking peers according to their capacity for outgoing connections. Any peer which is facilitating one or more outgoing connections is considered as super peer. The peers that are connected to permanent power source are eligible for being a super peer. Hence there are following three different types of peers at this phase:

Free riders: That are running on battery and thus cannot considered for outgoing connections.

Super Peers: Peers that are running on permanent power source and are offering one or more outgoing connections. These peers are placed between "list 0" to the list number of one less than the constant number of maximum possible outgoing connections.

Peers eligible for being a super peer: These peers are also connected to permanent power source and are eligible for outgoing connections. However, currently they are not supporting even a single outgoing connection. These peers will always remain in the list number equal to the constant number of maximum possible outgoing connections.

Second phase of implementation results in introduction of following two main features:

- Establishment of routing table by considering a constant number of maximum possible outgoing connections.
- Filtration of free riders from other peers using Battery Status API of HTML5.

3.2.3 Neighbor Selection Algorithm

The battery API helps in peer ranking but it does not provide any solution for reducing the link stretch. It can be defined as the ratio of path of packet in IP network with path of packet in overlay network. (61) However, we have focused on decreasing the overall length of overlay network path as less length will surely result in reduced link stretch. Greater link stretch increase delay which eventually may cause the more latency(62). Hence, we introduced a neighbor selection algorithm, in third phase, which helps in reducing the link stretch caused by large geographical distances of the peers. After the implementation of this algorithm, the closely located peers are put in a cluster which is a square shaped area on the Globe. This square is formed by the intersection of longitudinal and latitudinal lines. The decision for placing a peer in some certain cluster is made by finding the longitude and latitude of that peer using Geolocation API of HTML5. Floor and ceil functions of maths are then applied on longitude and latitude to find the boundaries of a cluster and then the peer is added to that particular cluster.

Clusters are designed in a way so that every cluster has a minimum of one super peer which is getting streaming from server and forwarding it to further connected peers. This peer is super peer in a sense that it is getting streaming from server and forwarding to other peers in that cluster. Super peer doesn't experience extra burden and outgoing connections are equally distributed among all the peers of a cluster. However, if a cluster has too many free riders, it may become saturated by filling the maximum number of outgoing connections of every peer. In this scenario next coming peer will also become a super peer and starts serving the same cluster with its outgoing links. If second super peer also get 5 free riders, this cluster will again become saturated and there is need for adding more super peer to this cluster, in case of the arrival of new peer. In this way, a cluster may have one or more gateway peers that are getting data from server and forwarding this to all connected peers of that cluster.

Third phase of implementation results in introduction of following two main features:

- Establishment of clusters or islands to place geographically closer peers in same cluster by implementing a neighbor selection algorithm.
- Introduction of neighbor selection algorithm results in reduction of link stretch and packet delay.

3.2.4 Super Peer Finding Algorithm

Fourth phase of implementation introduced an algorithm for efficient searching of super peer for a newly arrived peer. This is done by placing a peer which is eligible for outgoing connections but haven't created any outgoing connection into the list equal to maximum number of possible outgoing connections. It means that the list number is actually representing the capacity of a peer for outgoing connections. Similarly, if the same peer is going to serve one outgoing connection, it will be shifted to the list one less than the maximum number of outgoing connections. Hence, updated list is now again representing the number of available connections for that peer. In this way, all peers are actually placed in the lists that are corresponding to their capacity for making outgoing connections.

This placement of super peers helps in efficient searching at the arrival of new peer. Hence, list equal to maximum number of outgoing connections is checked initially at the arrival of any new peer. In case of availability of peer in that list, no more searching is done and the connection to the peer from that list is established. If there is more than one peer in that list then the oldest peer is selected for creating outgoing connection. This is done to reduce the churn in WOvNet by allowing some quarantine time for the newly coming peers. If there is no peer found in the list with maximum number of outgoing connections, algorithm is going to search in the list one less than the previous and again repeat the whole details mentioned for the first list.

Fourth phase of implementation results in introduction of following two main features:

- Efficient searching for super peer at the arrival of new peer.
- Making appropriate changes in the routing table at the arrival of new peer.

3.3 Responsibilities of WOvNet

3 responsibilities of overlay service provider discussed in section 2.8.3 have been employed on WOvNet. These responsibilities are Service Creation,

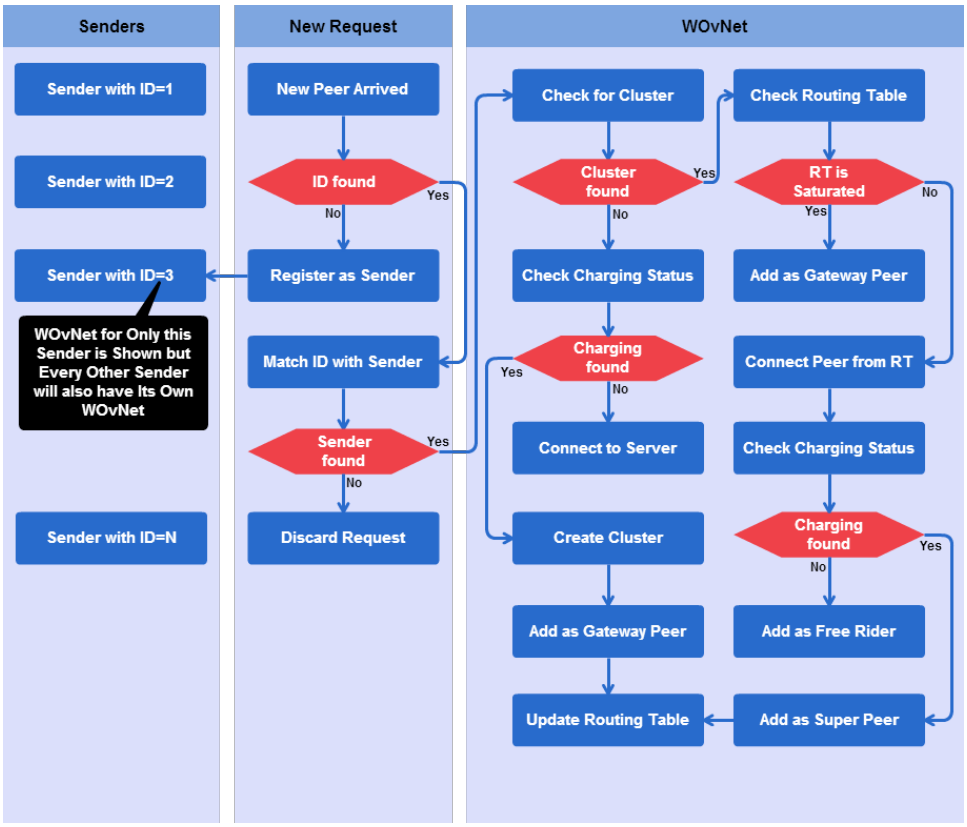

Figure 3.3: Flow Chart of Service Creation

Service Provisioning and Service assurance. Before discussing the algorithms for these three responsibilities one should be aware of following four types of peers in WOvNet.

- Sender: This is the peer which is sending the information on the first hand. It can be compared to a server in client server architecture.
- Gateway Peer: It is the peer which is transferring data to a cluster and is directly connected to the sender
- Super Peer: It is a peer in cluster which may have outgoing connections.
- Free Rider: It is the peer which cannot develop outgoing connections.

3.3.1 Service Creation Algorithm

Service creation algorithm is explained through a flow chart in figure 3.3 and a sequence diagram for the same algorithm is presented in figure 3.4.

This algorithm starts by checking the ID of newly arrived peer. If ID is not present the peer is registered as a sender. If peer arrived with an ID, it is matched with the senders that are transmitting the data. If no match of ID is found the peer's request to join the network is discarded. If ID found, the next step is to find a cluster for the new peer and it is done by neighbor selection algorithm explained in section 3.2.3. If cluster is not found then the charging status is checked for peer ranking algorithm in section 3.2.2. If permanent power source is found, new cluster is formed and the peer is added as a gateway peer and routing table is updated. However, if the newly arrived peer is running on a battery, it is connected to the sender as a free rider.

Considering that cluster is found for the newly arrived peer, we need to check the lists of that cluster from the routing table. If routing table is saturated then the new peer is connected as a gateway peer and routing table is updated. If routing table is not saturated then power status of the peer is checked. if permanent power source is found then the peer is connected as a super peer and routing table is updated. Alternatively, if the peer is running on battery, it is added in the cluster as a free rider. For adding a peer in cluster, we are using the algorithm of super peer finding which is already explained in section 3.2.4.

3.3.2 Service Provisioning Algorithm

Service provisioning algorithm utilize the Battery Status API of HTML5. The charging status of all the peers is constantly being monitored by mentioned API. There are two scenarios that needed to be looked upon.

3.3.2.1 Power Source Added

The first scenario is when a power source is added to a free rider. As the charging status has changed the free rider is now connected as a super/gateway peer and it can develop outgoing connections. The decision of connecting as a gateway or super peer is made on the basis of cluster. Following two possibilities exist in this scenario:

Peer in a cluster If the peer was in a cluster then after power is connected it will be updated as a super peer.

Peer not in a cluster If the peer was directly connected to the source, then a new cluster is formed and the peer is connected as a gateway peer.

3.3.2.2 Power Source Removed

The power source removal has different effect on gateway peer compared to super peer. Scenarios for both gateway peer and super peer are discussed

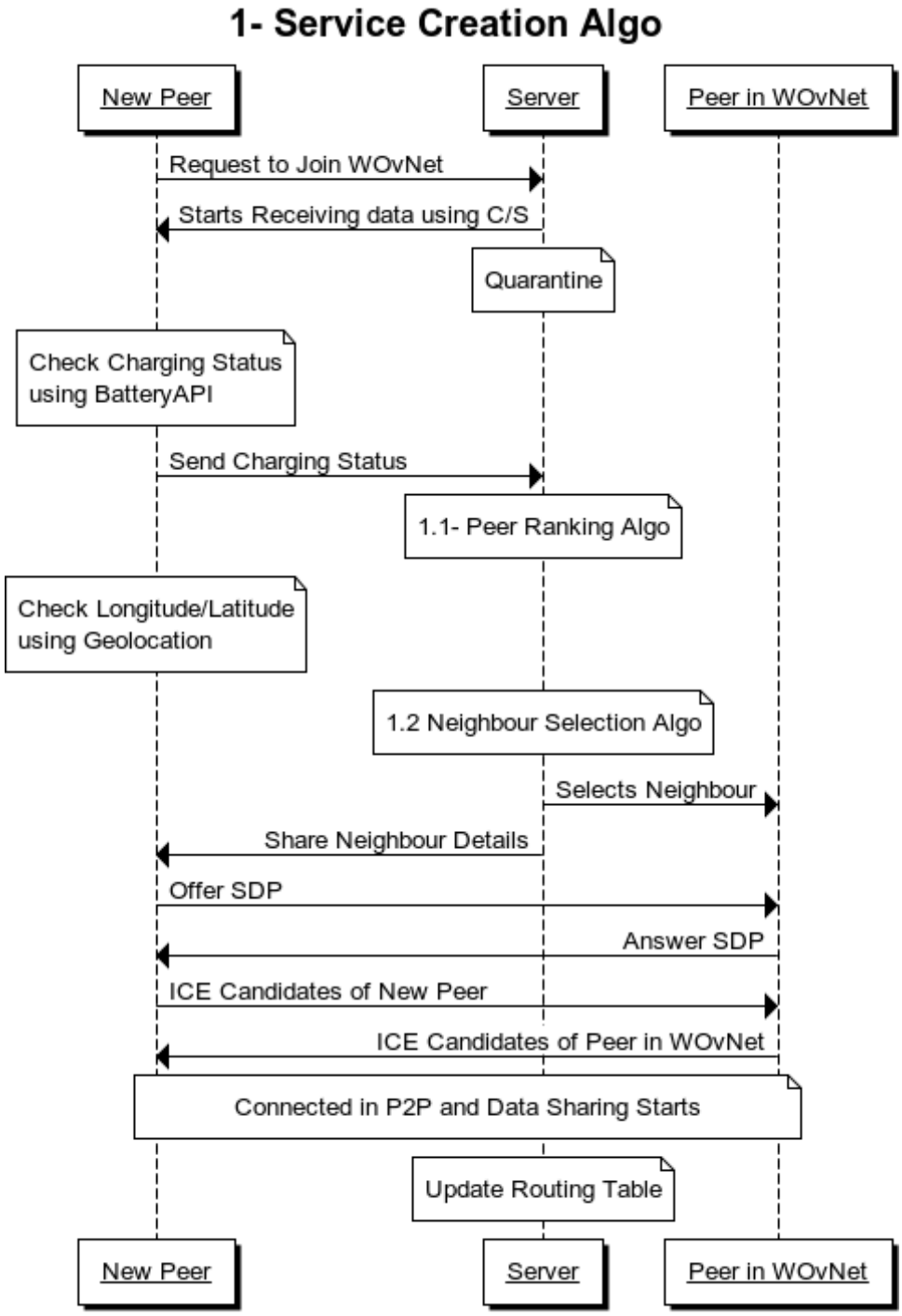

Figure 3.4: Sequence Diagram of Service Creation

in detail below:

Gateway peer can go through one of the following four scenarios:

1. If the power source is removed from a gateway peer with no outgoing connections the the gateway is connected as a free rider to the sender.
2. If the gateway peer whose power is removed has outgoing connections which are all free riders then gateway as well as its outgoing connections are connected to the sender as free riders.
3. If there is a super peer connected to the corresponding gateway peer the super peer becomes the gateway and earlier gateway peer is connected as a free rider to the newly formed gateway.
4. If more than one super peer is present the super peer belonging to the highest list is connected as a gateway peer and the previous gateway and all other super peers in the cluster connect to that peer. If it gets saturated without facilitating all other peers in the cluster a new gateway peer is formed by another super peer in the cluster staying at the top of the list.

Super Peer can go through one of the following three scenarios:

1. If the power source is removed from a super peer with no outgoing connections it remains at the same place in the routing table and it only converts into a free rider.
2. If the power source is removed from a super peer which has outgoing connections to the free riders only then it converts into a free rider and along side its outgoing connections are also connected to another super/gateway peer as free rider, using super peer selection algorithm.
3. If the power source is removed from a super peer which has outgoing connections to other super peer, that other super peer takes the place of its parent and all the outgoing connections and the earlier parent itself are now update in WOvNet using super peer finding algorithm explained in 3.2.4.

3.3.3 Service Assurance Algorithm

Service assurance allows the continuity of service if a peer drops during operations . Now as there are 4 types of peers then the drop scenario is different for each type. All these scenarios are explained below.

3.3.3.1 Sender Drop

The first scenario has the most devastating effect as it is for the sender. So if the sender which has created the overlay network drops, there is no way to maintain the overlay for that service. Thus, the whole overlay is removed and communication to all the peers is terminated.

3.3.3.2 Gateway drop

If a gateway peer is dropped, it causes one of the following three effects:

1. If a gateway peer is dropped with no outgoing connection then it will have no effect on the overlay network.
2. If it has outgoing connections which are all free riders then all of them are connected to the sender as free riders.
3. If super peers are present then the one with the highest list number becomes the gateway peer.

3.3.3.3 Super Peer Drop

If a super peer is dropped, it causes one of the following three effects:

1. If a super peer with no outgoing connection is dropped, it has no effect on the overlay.
2. If it has all free riders as outgoing connections those peers connect to the parent peer of the peer dropped.
3. If the parent becomes saturated then they are connected to other super peers in the cluster based on list number.
4. If there are super peers in the outgoing connections of the dropped peer then one with the highest list number takes its parent's place.

3.3.3.4 Free Rider Drop

The last scenario is if a free rider connected to the overlay drops. This scenario has no effect on the overlay network because a free rider has no outgoing connection so there is no need for rerouting the traffic.

Chapter 4

RESULTS AND DISCUSSION

This chapter presents experiments and results along with relevant discussion. All experiments discussed in this section are performed at machines having CPU of Intel (R) Pentium (R) B950 @ 2.10 GHZ 2.10 GHZ, with memory of 2.00 GB. MEAN stack is used for the implementation.

Following is the organization detail of this chapter:

Section 4.1 explains the impact of overlay creation on server.

Section 4.2 explains the impact of cluster creation on link stretch of overlay with detailed discussions.

Section 4.3 explains the impact of cluster creation on packet delay in detail.

4.1 Impact of WOvNet on server

This section presents impact of WOvNet on server. Experiments are performed to find the memory and CPU consumption of server by shifting a file from 1 to 15 clients, by adding one client just after the other. Similarly 15 peers are also connected through WOvNet and sent same file. Comparison of both approached shows that WOvNet is helping in reducing the server's burden.

4.1.1 Evaluation Environment

Experiments are performed by sending a file from server to multiple clients using WebSocket by following client/server architecture. Load on the server including CPU usage and memory consumption is examined for that file sending process. The effect of increase in number of peers is calculated for

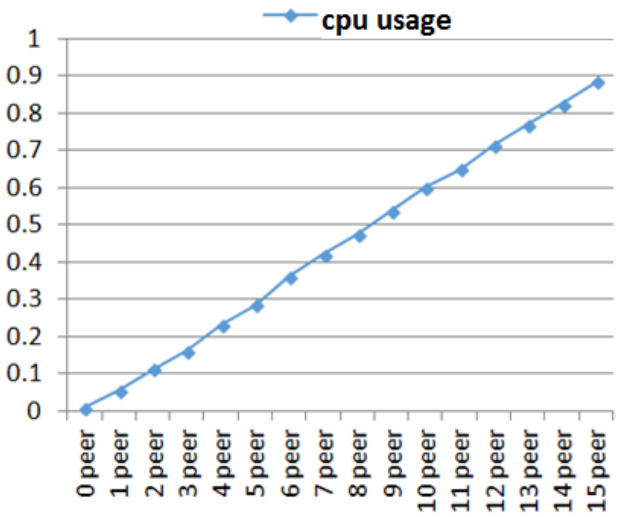

Figure 4.1: CPU Usage

both scenarios and is graphically presented in figure 4.1 and 4.2 respectively. This helps in finding the approximate load increased by adding an extra peer to the server.

4.1.2 Reduction in CPU consumption per peer

This experiment reveals the extent to which WOvNet helps in reducing the CPU consumption of server. We have shifted peers from 1 to 15 on server using WebSocket and logged the CPU usage at server. Figure 4.1 shows the CPU usage of server for 0 to 15 connections of WebSocket for transmission of a same file from server to clients. It is obvious from this graph that increase of every single connection on server increases the CPU usage by approximately 0.062%. Thus, every connection that is sending data through WebRTC is actually reducing about 0.062% CPU processing of server.

4.1.3 Reduction in memory consumption per peer

Figure 4.2 contains the details of memory consumption at server from 0 to 15 connections. It reveals that every newly arrived connection utilizes around 573.44KB more memory at server. In this way, every connection sending data through WebRTC reduces the memory consumption of server by approximately 573.44KB per connection.

4.1.4 Summary

Graph in fig 4.1 and 4.2 depicts that every new coming connection increases the CPU consumption about 0.062%. Moreover, every new peer also cap-

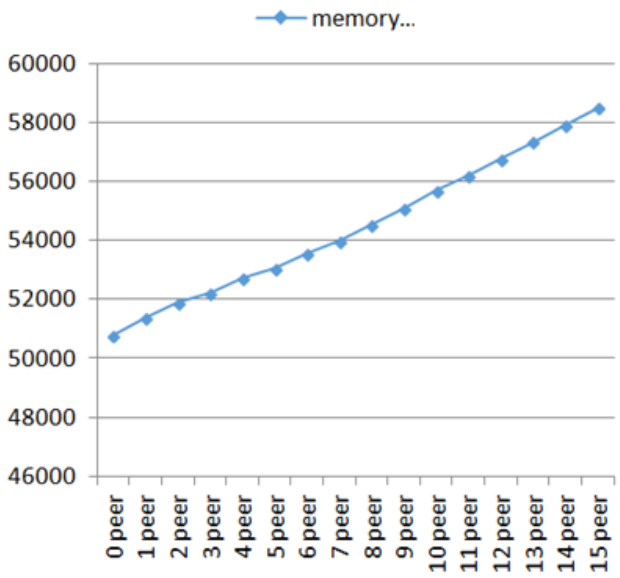

Figure 4.2: Memory Consumption

turing 573KB memory at the server. Hence, more peers we are shifting to the WOvNet the less burden will be experienced by the server. 1 to 15 connections are created using WebSocket and CPU usage and memory consumption is measured. 15 connections are also being made using WebRTC and it has been observed that only those peers are creating burden on server that are getting data directly from server, using WebSocket. All other peers that are getting data through dataChannel API of WebRTC are not creating burden on server. However, every peer is sending a ping request to server for keeping WebSocket connection alive,which creates burden of 0.001% in CPU utilization, after every 20 seconds, which is almost negligible.

4.2 Impact of Cluster creation on Link stretch in overlay

This section describes the effect of cluster creation on link stretch in WOvNet. Link stretch is a major cause of delay in packet transfer in peer to peer networking (62). Link stretch calculated using simple tree based approach, explained in section 3.2.1. This tree based approach is then compared with the WOvNet having all three algorithms of peer ranking, neighbor selection and super peer finding. The evaluation Scenario and results of this experiment are discussed ahead.

Figure 4.3: Evaluation Scenario for Link Stretch

4.2.1 Evaluation Environment

Five different connections were created between six peers from Pakistan and USA, having locations mentioned in figure 4.3. Two different overlay networks are created using simple tree based algorithm mentioned in section 3.2.1 and WOvNet containing three algorithms of peer ranking, neighbor selection and super peer finding. Results show almost 50% reduction in link stretch, in best case scenario .

4.2.2 Evaluation Scenario

Following two overlay networks are built using mentioned order of peers' arrival in part A of figure 4.4. Two different overlay networks are built that are mentioned below:

Overlay based on Simple Tree based Topology is mentioned in part B of figure 4.4 by having "2" as fixed number of maximum outgoing connections. It follows the sequence of peers' arrival as it is mentioned in part A of figure 4.4.

Overlay based on WOvNet is mentioned in part C of figure 4.4 by following the sequence of peers' arrival in part A of figure 4.4.

Coordinates of all connected peers are calculated using Geolocation API of HTML5 and then geographical distances are calculated from level 1 peers to the level 3 peers.

4.2.3 Results

Results show reduction in link stretch by use of WOvNet as compare to simple tree topology is almost 50% in case best case scenario. Comparison of both overlay networks is available in figure 4.4. The link stretch in the case

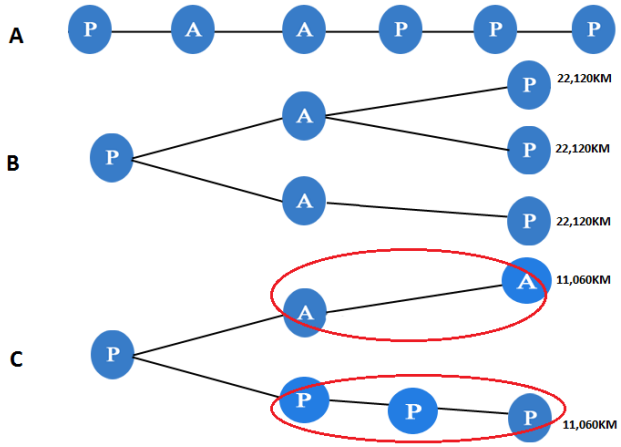

Figure 4.4: Reduction in Link Stretch

of simple tree based topology appears to be 22,120 km while it is reduced to 11,060 km, after using WOvNet for serving the same peers arriving in same sequence.

4.2.4 Summary

Introduction of neighbor selection algorithm has tremendously optimized WOvNet because now we are well aware of the geographical locations of the connected peers and we can build network topology more effectively. This also has great impact on network traffic in terms of reducing the packet delay and this is what we have discussed in next section.

4.3 Impact of Overlay on Network Traffic

We have again created the same two overlay networks that we have already discussed in previous section and sent data packets from peers of level 1 to peers of level 3 and have calculated the packet arrival time. Results show that WOvNet has also reduced the packet delivery time.

4.3.1 Evaluation Scenario

We have no clock synchronization present in the peers connected in overlay network. Hence we can say that no global clock is available in our test bed. The problem which arises from this limitation is that we cannot determine packet delivery time from the sender to the receiver. In order to overcome this limitation we used the round trip delay. (63)

The same technique which was used to find the link stretch in previous section is used here, but instead of finding the distance we have calculated

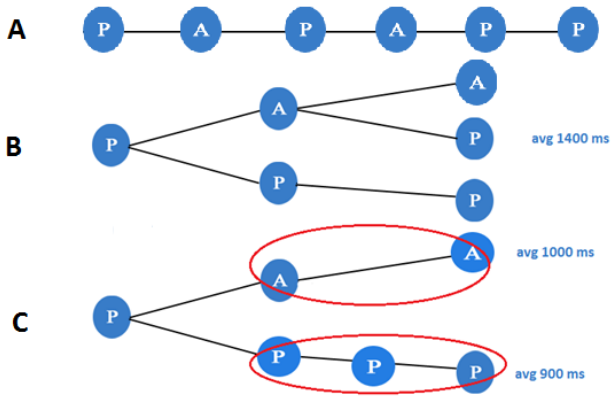

Figure 4.5: Reduction in Packet Delivery Time

the packet delivery time. By using round trip delay, we found that the average packet delay for the peers located at the coordinates mentioned in figure 4.1 was appeared to be 700ms while the peers that are placed at the same physical location were taking about 300ms.

4.3.2 Results

The results of delay in packet delivery are mentioned in figure 4.5 . These results show that if a simple tree based network is used then the maximum packet delivery time between peers of level 1 and level 3 reached to approximately 1400ms, as it is mentioned in part B of figure 4.5. However, in case of WOvNet, the packet delivery time between peers of level 1 and level 4 appeared to be 900ms, as it is mentioned in part C of figure 4.5.

4.3.3 Summary

In this section packet delivery time has been calculated for WOvNet and for simple tree based topology. Results show that cluster creation not only reduces the link stretch but also contributes in reducing the traffic delay in WOvNet. As a result, in the discussed example, although WOvNet is adding and extra level in overlay network but still it is having less packet delay than the simple tree based topology, where the number of levels are less as compare to WOvNet.

Chapter 5

CONCLUSION AND FUTURE WORK

In this chapter, the conclusion with a summary of the research findings along with future directions is presented. We have implemented HTML5 based browser to browser (b2b) overlay that is totally plug-in free and based on native functionalities of latest browsers. This b2b overlay can be used for peer casting service and it will also provide a base for building services over plug-in free overlays in future. We have added optimization algorithms in WOvNet to reduce the link stretch and packet delay and all of these functionalities are purely based on the features of HTML5 to build a plugin free web based overlay network known as WOvNet.

5.1 Summary

Since the sale of mobile devices (laptops, tablets and smart phones) crossed PCs' sale in 2011, it has become more important to offer cross platform services by targeting these platforms. One of the reasons for this transition is the enhanced capabilities of portable devices that were previously constricted to only desktop PCs. Latest portable devices are powerful in many folds as compared to earlier devices in terms of processing power, storage capacity, battery backup etc. These enhanced features resulted in introduction of many new trends like Bring Your Own Device (BYOD) that eventually reflected in improved user experience for facilitating resource sharing in daily life.

Although Client/Server architecture is still leading in communication of online applications however, improved resource sharing experience results in introduction of an alternative in terms of peer to peer (p2p) communication. Hence application developers are now targeting p2p along with client/server architecture to provide hybrid solutions that results in decrease of server load and eventually helps in building more cost effective solutions. Spotify

is a free advertisement driven music on demand service and it is a good example of adopting a hybrid approach to offer a cost effective solution. Only 8.8% of music playback at Spotify comes from its servers and hence it can engage more users with comparatively less server resources. (64)

Similar concept of load sharing hit Rich Internet Application (RIA) after the introduction of real time media flow protocol (RTMFP) by Adobe. It uses flash plugin and enables web browsers to build a p2p based network topology which results in reducing server load by serving audio/video streaming through web based overlay network. As a result RTMFP can be used to serve huge audience using comparatively less server resources, as every new coming client is also considered as an addition to overall system resources. However, RTMFP is not open source and is also not natively supported in browsers. This is so because it requires the installation of Adobe flash player by end users therefore, use of plugin based solutions is going to be depreciated after the introduction of latest HTML5 & JavaScript standards.

Any solution based on HTML5 standards also becomes platform independent due to native support of these standards in latest browsers running on different devices. There are few more strengths of plugin free overlay network over the RTMFP based overlay network. One of these strengths is plugin free overlay is based on WebSocket and dataChannel API of WebRTC and both of these can be used to propagate data and we have also optimized it for data propagation while RTMFP based overlay network is focusing on audio/video streaming. This feature of plugin free overlay network can be utilized for implementing multiplayer games and also in e-learning for collaborative tasks. It can also be used for implementing efficient co-browsing systems for huge audience.

One more limitation of RTMFP based overlay network is that it works on UDP only and cannot be used if all UDP ports are blocked. Although RTMP can be used as a fall back option but it follows the client/server architecture and thus cannot be used to build an overlay network based on P2P communications. We have implemented P2P communication between browsers based on WebRTC. Although it works on UDP by default but it can also be configured to work on TCP.

Another limitation of RTMFP is that it can only work intra-domain and there is no support for inter-domain communication. However, WebRTC can be used to implement inter-domain communication. Because an overlay network based on latest HTML5 features has been implemented in this research therefore, strengths of these features over RTMFP becomes the strength of our implemented overlay network.

Emergence of latest HTML5 standards has tremendously increased the strength and capacity of the web based applications. Moreover, many of the features that were previously confined only to third party plug-ins are now natively available in web-browsers. Two Plug in free solutions are

available in HTML5 that can be used for communicating from one browser to multiple browsers. One option is WebRTC that creates p2p connections between browsers and it transmits data using all resources of the client's devices through their browsers. Other option is WebSocket that follows client/server architecture and uses server resources for data transmission. If only WebSocket is used for data transmission it shifts burden to the server and if only WebRTC is used for the similar behavior it consumes all of the client's resources. Battery is one of the resources that are limitedly available on client's device and if a solution which depends only on client's resources is applied. It eventually results in earlier drainage of battery.

So it can be concluded that the tradeoff between using WebSocket or WebRTC is that of server load versus power consumption. This research focuses on design of a communication backbone that alternatively utilizes both of these communication options to build a self-adaptive network topology for maintaining a plugin free overlay network. This communication backbone tries to shift the burden of server to the connected clients after building a self-adaptive overlay network to facilitate collective communication in RIA. This overlay uses the overlay multicast to broadcast data from one sender browser to multiple receiving browsers and it is self-adaptive as it dynamically shifts the burden to the peers and also takes back their burden, on the basis of their capacity for outgoing connections. This capacity of outgoing connections is determined by peer ranking algorithm, on the basis of their charging status. The peer ranking algorithm uses Battery Status API of HTML5 to find the charging status of the clients.

Moreover, this overlay is also well aware of the geographical locations of the clients and thus placement of clients in the network topology takes place by adding closer peers in similar groups to form islands or clusters. Geolocation API of HTML5 helps in returning approximate position of connected devices in longitude and latitudes and our algorithm of neighbor selection also keeps track of positions of previously connected peers and if a new peer arrives near to any previously added peer; it is placed near to the previous one. In this way, geographically closer peers combined in a group to form a cluster. This approach results in reducing the link stretch by minimizing the length of connections in overlay network.

5.2 Contributions

We can summarize contributions of this thesis as follows:

1. We have explored HTML5 standards that can be used for building replacement of RTMFP and integrate these options to form a plugin free WOvNet.

2. We have implemented an algorithm of peer ranking to help the portable

devices in maintaining batteries for longer time intervals, while using WOvNet.

3. We have introduced algorithm of neighbor selection and super peer finding to place geographically closer peers in same cluster and do the efficient placement within a cluster respectively.

4. We have added 3 responsibilities of overlay service provider in WOvNet and these are service creation, service provisioning and service assurance.

5.3 Future Work

WOvNet can be extended in following different directions:

5.3.1 Adding More Collective Communication Constructs

We have only implemented broadcast using iterative unicast based multicast. However, many collective communication constructs that are mentioned in section 1.1 can be implemented in future.

5.3.2 Improvements in Peer Ranking Algorithm

HTML5 has introduced many more APIs (65) that can be used for further optimization of WOvNet. Some of these APIs are System Information API and Network API. System Information APIs provide more details about the device used by peer to join WOvNet. Network API provides the network details thus both of these APIs can play vital role in peer ranking algorithm.

5.3.3 Improvements in Neighbor Selection Algorithm

Currently we are creating clusters from the squares available due to the intersection of longitudinal and latitudinal lines. An algorithm can be build for selecting the boundaries of a cluster or an island dynamically to make the cluster formation more flexible.

5.3.4 Quarantine time

Quarantine time (66) is giving some time to the peer for building trust on it and then add it to the overlay network. Some algorithms can be implemented to find the trust level of a peer during quarantine time and then take decision for adding it to the overlay network.

Bibliography

[1] David, M. (2013). HTML5: designing rich Internet applications. Taylor & Francis.

[2] http://www.cise.ufl.edu/research/ParallelPatterns/glossary.htm

[3] Lee, C. Y., & Jang, B. O. (2013). An Efficient Sport Target Scenes Extracting Methodology for P2P-Based Personal Mobile Beaming Broadcast Service. Wireless personal communications, 73(2), 299-307.

[4] Buford, J., Yu, H., & Lua, E. K. (2009). P2P networking and applications. Morgan Kaufmann.

[5] https://mozillalabs.com/en-US/

[6] http://www.chromeexperiments.com/

[7] http://wiki.ccs.tulane.edu/index.php5/Collective_Communication

[8] Van Acker, S., Nikiforakis, N., Desmet, L., Joosen, W., & Piessens, F. (2012, May). FlashOver: Automated discovery of cross-site scripting vulnerabilities in rich internet applications. In Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security (pp. 12-13). ACM.

[9] Xue, L., Wen, F., Fan, C., Wang, J., & Wang, X. (2012, August). Group Audio Application with Flash Multicast Streaming Based on RTMFP. In Proceedings of the 2012 International Conference on Computer Application and System Modeling. Atlantis Press.

[10] Abell Lozano, A. (2013). Performance analysis of topologies for Web-based Real-Time Communication (WebRTC).

[11] Rodriguez-Gil, L., Ordua, P., Garca-Zubia, J., Angulo, I., & Lpez-de-Ipia, D. (2014, February). Graphic technologies for virtual, remote and hybrid laboratories: WebLab-FPGA hybrid lab. In Remote Engineering and Virtual Instrumentation (REV), 2014 11th International Conference on (pp. 163-166). IEEE.

[12] Singh, K., & Davids, C. (2011). Flash-based audio and video communication in the cloud. arXiv preprint arXiv:1107.0011..

[13] Shank, P. (2012). Do I Need to Know about HTML5?(Nov 12). Chicago

[14] http://www.gartner.com/it/page.jsp?id=1980115

[15] http://tech.fortune.cnn.com/2011/07/11/apple-users-buying-61-more-apps-paying-14-more-per-app/

[16] http://www.engadget.com/2012/06/28/adobe-confirms-it-wont-support-flash-on-android-4-1/

[17] Jackson, D. W. (2012). Steve Jobs Was Right About the Future of HTML 5 but Was He Right About Digital Publishing?. Law Library Journal, Spring.

[18] http://blogs.adobe.com/conversations/2011/11/flash-focus.html

[19] Alexandru, C. (2014). Impact of WebRTC (P2P in the Browser). Internet Economics VIII, 39.

[20] Na, D. Y., & DeRocher, B. C. (2011). HTML5: Whats Different for User Experience Design and the Web?. Connectivity and the User Experience, 45.

[21] World Wide Web Consortium. (1999). HTML 4.01 specification.

[22] "FAQ What is the WHATWG?". WHATWG. 12 February 2010. Retrieved 24 February 2010.

[23] "HTML5: A vocabulary and associated APIs for HTML and XHTML.". World Wide Web Consortium.

[24] Coulter, S., Walters, G., & Ricketts, D. (2014). Live Streaming with Big Blue Button at UWS.

[25] Buchansky, H. (2013). Jing and Screenr (Free Screencasting Software). Journal of the Canadian Health Libraries Association, 34(02), 117-118. Chicago

[26] Hickson, I., & Hyatt, D. (2011). HTML5: A vocabulary and associated APIs for HTML and XHTML. W3C .%Working Draft edition.

[27] Curran, K., & George, C. (2012). The future of web and mobile game development. International Journal of Cloud Computing and Services Science (IJ-CLOSER), 1(1), 25-34.

[28] Davies, M., Zeiss, J., & Gabner, R. (2012, December). Evaluating two approaches for browser-based real-time multimedia communication. In Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia (pp. 109-117). ACM.

[29] Daoust, F., Hoschka, P., Patrikakis, C. Z., Cruz, R. S., Nunes, M. S., & Osborne, D. S. (2010). Towards Video on the Web with HTML5. NEM Summit, 6.

[30] http://www.zdnet.com/blog/microsoft/will-there-be-a-silverlight-6-and-does-it-matter/11180

[31] (ZHANG, K., & HUANG, J. (2011). Research and Implementation of Real-time Distance Education System Based on RTMP.)

[32] (Premchaiswadi, W., & Tungkasthan, A. (2007, June). An implementation of an interactive virtual classroom on Internet. In World Conference on Educational Multimedia, Hypermedia and Telecommunications (Vol. 2007, No. 1, pp. 1299-1304).

[33] http://broadcast.oreilly.com/2009/04/adobes-real-time-media-flow-pr.html

[34] Gracia, J., & Bayo, E. (2012). Integrated 3D web application for structural analysis software as a service. Journal of computing in civil engineering, 27(2), 159-166.

[35] Sire, S., Paquier, M., Vagner, A., & Bogaerts, J. (2009, April). A messaging API for inter-widgets communication. In Proceedings of the 18th international conference on World wide web (pp. 1115-1116). ACM.

[36] Eraiah, S., Fallows, J. R., Gaunce, D. B., & Salim, F. J. (2010). U.S. Patent Application 12/772,046.

[37] Elleuch, W. (2013, October). Models for multimedia conference between browsers based on WebRTC. In Wireless and Mobile Computing, Networking and Communications (WiMob), 2013 IEEE 9th International Conference on (pp. 279-284). IEEE.

[38] http://www.adobe.com/products/adobe-media-server-extended/rtmfp-faq.html

[39] Sarris, S. (2014). HTML5 unleashed. Pearson Education.

[40] Wang, V., Salim, F., & Moskovits, P. (2013). The definitive guide to HTML5 WebSocket (Vol. 1). Apress.

[41] Daoust, F., Hoschka, P., Patrikakis, C. Z., Cruz, R. S., Nunes, M. S.,& Osborne, D. S. (2010). Towards Video on the Web with HTML5. NEM Summit, 6.

[42] Johnston, A., Yoakum, J., & Singh, K. (2013). Taking on WebRTC in an Enterprise. Communications Magazine, IEEE, 51(4), 48-54.

[43] Nurminen, J. K., Meyn, A. J., Jalonen, E., Raivio, Y.,& Marrero, R. G. (2013). P2P media streaming with HTML5 and WebRTC. In IEEE International Conference on Computer Communications. IEEE.

[44] en.wikipedia.org/wiki/Geolocation

[45] http://searchsoa.techtarget.com/definition/Rich-Internet-Application-RIA

[46] http://caniuse.com/geolocation

[47] http://www.w3.org/TR/2014/CR-battery-status-20141209/

[48] Jones, P. (2013). BEHAVE M. Petit-Huguenin Internet-Draft Impedance Mismatch Intended status: Standards Track S. Nandakumar Expires: January 14, 2014 G. Salgueiro.

[49] Rodrguez Prez, P., Cervio Arriba, J., Trajkovska, I., & Salvacha Rodrguez, J. (2012). Advanced Videoconferencing based on WebRTC.

[50] http://www.nt-electronics.com/mcu.html#

[51] Abell Lozano, A. (2013). Performance analysis of topologies for Web-based Real-Time Communication (WebRTC).

[52] HUAQUN, G. (2007). Enhancing multicast communications with data-in-work schemes (Doctoral dissertation)

[53] Almeroth, K. C. (2000). The evolution of multicast: From the MBone to interdomain multicast to Internet2 deployment. Network, IEEE, 14(1), 10-20.

[58] Lao, L., Cui, J. H., Gerla, M., & Maggiorini, D. (2005, March). A comparative study of multicast protocols: top, bottom, or in the middle?. In INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE (Vol. 4, pp. 2809-2814). IEEE.

[55] Zhu, Y., Shu, W., & Wu, M. Y. (2005, July). Approaches to establishing multicast overlays. In Services Computing, 2005 IEEE International Conference on (Vol. 2, pp. 268-269). IEEE.

[56] Wang, F., Xiong, Y., & Liu, J. (2007, June). mtreebone: A hybrid tree/mesh overlay for application-layer live video multicast. In Distributed Computing Systems, 2007. ICDCS'07. 27th International Conference on (pp. 49-49). IEEE.

[57] Roverso, R. (2013). A System, Tools and Algorithms for Adaptive HTTP-live Streaming on Peer-to-peer Overlays.

[58] Yeo, C. K., Lee, B. S., & Er, M. H. (2004). A survey of application level multicast techniques. Computer Communications, 27(15), 1547-1568.

[59] Kreitz, G., & Niemela, F. (2010, August). Spotify–large scale, low latency, p2p music-on-demand streaming. In Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on (pp. 1-10). IEEE.

[60] https://web.archive.org/web/20140207233805/http://pansentient.com/2011/04/spotify-technology-some-stats-and-how-spotify-works

[61] Buford, J., & Kolberg, M. (2009, January). Hybrid overlay multicast simulation and evaluation. In Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE (pp. 1-2). IEEE.

[62] Ren, S., Guo, L., Jiang, S., & Zhang, X. (2004, April). SAT-Match: a self-adaptive topology matching method to achieve low lookup latency in structured P2P overlay networks. In Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International (p. 83). IEEE.

[63] Grovenburg, W. G. (2007). U.S. Patent No. 7,257,087. Washington, DC: U.S. Patent and Trademark Office.

[64] Kreitz, G., & Niemela, F. (2010, August). Spotify–large scale, low latency, p2p music-on-demand streaming. In Peer-to-Peer Computing (P2P), 2010 IEEE Tenth International Conference on (pp. 1-10). IEEE.

[65] Yan, X., Yang, L., Lan, S., & Tong, X. (2012, August). Application of HTML5 multimedia. In Computer Science and Information Processing (CSIP), 2012 International Conference on (pp. 871-874). IEEE.

[66] Brown, A., Buford, J., & Kolberg, M. (2007, March). Tork: a variable-hop overlay for heterogeneous networks. In Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07. Fifth Annual IEEE International Conference on (pp. 104-108). IEEE.