International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 4, Issue 3, Jun 2014, 129-138

TRANS STELLAR
• Journal Publications • Research Consultancy

HIGH SPEED REAL TIME VIDEO ACCESS OVER WEB SOCKETS

SHWETA THAKUR & V. A. GULHANE

G.H. Raisoni College of Engineering, Nagpur, Maharashtra, India

ABSTRACT

The web has been largely built around the so-called request/response model of HTTP. A client loads up a web page and then not anything happens until the user clicks onto the next page. AJAX started to make the web more dynamic. Still, all HTTP message was steered by the client, which required user interaction or interrupted polling to load new data from the server. Web Socket is a protocol providing full-duplex communications channels over a single TCP connection. The Web Socket specification—developed as part of the HTML5 defining a full-duplex single socket connection over which messages can be sent between client and server from both the direction simultaneously. In this paper the websocket protocol is designed to have video streaming fast compare to http. It shows that the throughput is increased by 54% than http. Also the delay factor is improved by 2% than http. The Web Socket simplify the complexity around bi-directional web communication and connection management. Also the websocket protocol removes much of the overhead of Http protocol.

KEYWORDS: Web Sockets, HTML 5, Full Duplex Network, Web Browser, SIP

I. INTRODUCTION

Video Streaming is a technology for distributing and watching digital video over the network usually, digital video can be played after downloading the entire file to one's PC. Digital video data is enormous compared to text and still image data and it is time-consuming process to download the video. In order to solve the problem, a streaming skill has been developed which receives the data from a server and replays the video at the same time enabling the users to watch the video.

To have web applications require two way communications between client and server on some protocol, using http protocol for this communication have some problems like: A disadvantage of statelessness is that it may be necessary to include additional information in each request, and this added information will need to be interpreted by the server. A clarification would be to use a single TCP connection for traffic in both directions. That is what the Web Socket Protocol provides combined with the Web Socket API; it provides a substitute to HTTP polling for two-way communication from a web page to a remote server. The above problems can be solved using the new websocket protocol. The websocket protocol is a full duplex bidirectional protocol which provides simultaneous connection between the client and the server.

The HTTP protocol is unidirectional, in which we need to establish the connection every time between the client & server. The new websocket API resolves the issue of sending data directly to the client by allowing the browser to maintain an asynchronous socket connection to a server. By maintaining this connection the client is able to instantly send data to the server without needing to re-establish a connection. Additionally, the sever is able to send data to client at any time while the connection remains open. The websocket API defines a simple protocol to transfer information & provides a method for creating secure connection which is beneficial for authentication propose.

In this section, we propose the architecture of continuous player for HLS on various networks. The player is based on the standard HTML5 tags, JavaScript, and built-in video Player of web browser. Many implementations use a web browser to support direct and interactive communications, with voice, video, and gaming. In these implementations, the web server acts as the signaling lane between these applications. Up till now, such applications have typically required the installation of plug-ins or non-standard browser extensions. To attain real-time experience, developers use techniques polling, long polling and streaming. All of these methods use HTTP protocol to communicate with the server. Each request send to server in excess of HTTP contains bunch of redundant header information describing where this request came from, where it's bearing, the user information. This information adds a bundle of overhead on bandwidth in real-time scenarios. Secondly, this is not a full-duplex. This paper presents the design and implementation of websocket for video streaming. Section II discusses the related works of existing technologies. Section III presents the design and system architecture of websocket for video streaming. Section IV presents implementation details. Finally Section VI concludes the paper.

II. RELATED WORK

Virtualization [1] Virtualization techniques are used to migrate the server state and connection established with clients. Virtualization avoids the pre- reservation of resources in the failure host. Virtualization using Web Socket uses new technologies to make remote virtualization accessible to anyone without the need to neither download any of the data nor need to specialized hardware to visualize it. Client side rendering can be become limited both in terms of storage space and processing power.

Http polling and http long polling [6] http polling consists of a sequence of request reply messages. The client asks for a server. Upon getting this request, the server responds with an empty response if no message is available for that client. After a short time, called the polling interval the client polls the server again to see if any new messages are available. http long polling one problem associated with polling is the number of unnecessary requests made to server when it has no new messages for a client. Long polling emerged as a deviation on the polling technique that efficiently handles the information push from server to clients.

HTML5 Web Socket [8] this technique is designed to overcome the limitation of XHR polling. The Web Socket runs over a long-lived TCP socket to deliver the bidirectional stream and this makes it significantly more efficient than XHR polling. The websocket maintains a session for each client & the session remains open until the client or server terminates it. The video tag of HTML5 is suitable for sending static file but not for dynamic file.

III. DESIGN AND ARCHITECTURE

The HTML5 Web Socket specification defines an API that enables web pages to use the Web Socket protocol for two-way communication with a isolated host. It introduces the Web Socket interface and defines a full-duplex communication channel that operates through a single socket over the Web. HTML5 Web Socket provide an enormous reduction in unnecessary network traffic and latency compared to the unsalable polling and long-polling solutions that were used to simulate a full-duplex connection by maintaining two connections. The Web Socket Protocol enables two-way communication between a client and remote host. The protocol has two parts handshake and data transfer.

The handshake from the client looks as follows:

GET /echo HTTP/1.1

Host: server.example.com

Upgrade: web socket

Connection: Upgrade

Origin: http://example.com

The handshake from the server looks as follows:

HTTP/1.1 101 Switching Protocols

Upgrade: web socket

Connection: Upgrade

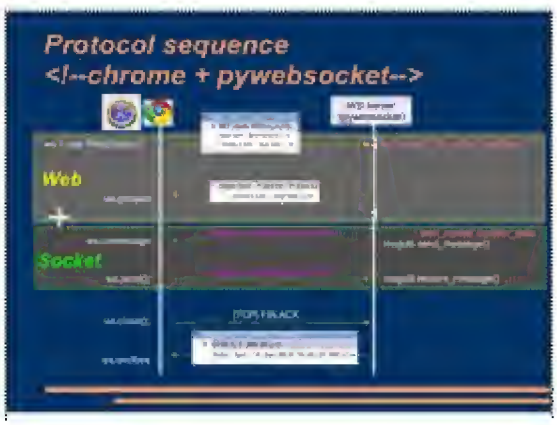

Figure 1: Handshaking in Websocket Protocol

Once the client and server have both sent their handshakes, and if the handshake be successful, then the data transfer starts. This is a two-way communication channel where each side can, independently send and receive the data. After a successful handshake, clients and servers move information back and forth in conceptual units referred to the pattern as "messages". A frame has a related type. Each frame belonging to the same message contains the same type of data. The data may be of type textual data, binary data and control frames.

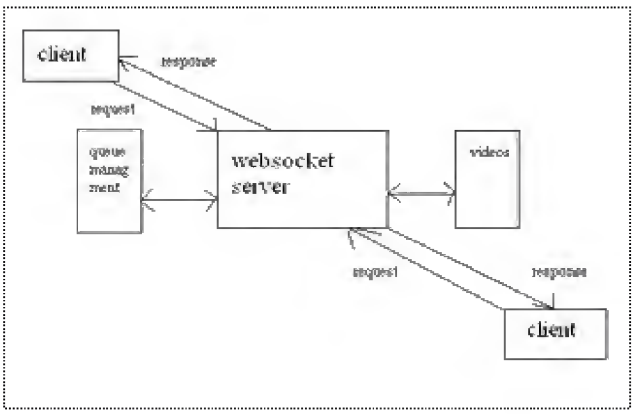

Figure 2: Architecture of Web Socket

Web Socket uses single socket to push and pull information between the browser and the server in full duplex communication. The HTML5 Web Socket is the principal mechanism for promoting the web full-duplex real-time communication. It uses single socket to push & pull information between the browser and server in full duplex communication. Web Socket protocol is essentially a TCP. To establish a Web Socket connection, the client and server upgrade from the HTTP protocol to the Web Socket protocol during their initial handshake once established, Web Socket data frames can be send back and forth between the client and the server in full-duplex mode, and this connection will continue to exist until the client side or server side close the connection initiatively. The Web Socket API defines mainly four callback methods: on open, on message, on close and on error to deal with event trigger during the Web Socket connection. It is very suitable for real-time web applications. We can have our own test, by using init(), test Web Socket(), on Open(), on Close(), on Message(), etc functions.

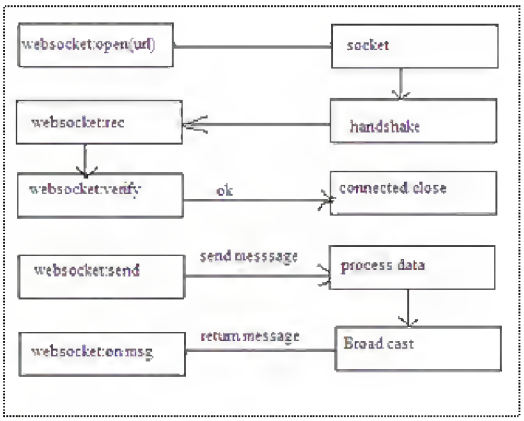

Figure 3: WebSocket Connection Establishment Model

IV. IMPLEMENTATION DETAIL

The main aim is to provide the user the facility of connected architecture. The http protocol is half duplex and disconnected architecture, in which if the user want to access some videos he has to send a request to the server. The server will accept the request and will reply to the client. If the same user wants to access another file he will have to follow the same procedure again which is a time consuming one. In short the same users have to follow the process of handshaking. Also the http add extra overhead.

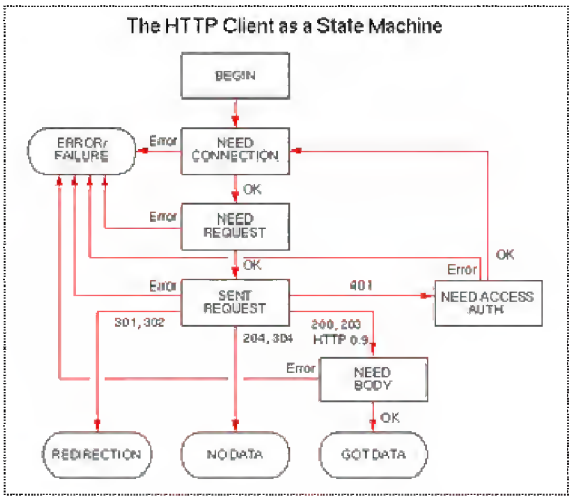

Figure 4: Http Client as a State Machine

The http polling method also uses the mechanism used by http, where as in case of websocket protocol which is a full duplex and connected architecture. In which if the user want to access the video he will first follow the procedure of handshaking once the handshaking is done between client and server, the client can access the video, if the same user wants to access another file he doesn't have to follow the procedure of handshaking which saves the time also overcomes the limitation of http. The websocket protocol initially starts on http it will use the upgrade header of http and will upgrade it to websocket. The whole communication now will be move on to websocket. Currently many browsers support websocket.

Table 1: Compatibility with Browser

Show All Versions	IE	Firefox	Chrome	Safari	Opera	IOS Safari
Current	10.0	22.0	28.0	6.0	15.0	6.0-6.1
Near future	11.0	23.0	29.0	7.0		7.0

A. Test Environment

The experiment on websocket has been done using a computer i.e. user's own desktop will act as server with a Intel(R)core(TM)Duo cpu T6600 @ 2.20Ghz 2.20 Ghz & 3 GB of RAM. The test mobile device is HTC desire 700, and the operating system is Android (version 4.1.2) Mobile. But it's not necessary to have the client (mobile) with the same configuration.

Table 2: System Details

Processor	Intel (R) Core (TM) Duo CPU T6600 @ 2.20Ghz 2.20 GHz
RAM	3 GB
System Type	32 bit OS

The SIP is also used to provide some facilities of websocket. The Session Initiation Protocol (SIP) is a signaling communications protocol, generally used for managing multimedia message sessions like voice and video calls in excess of Internet Protocol (IP) networks. Also the websocket server is used here, in which we are uploading a file on the server once we have done with the uploading we can broadcast it and the clients which are connected to it can have the experience of same video. The client can be any machine or any Smartphone. We can also stop the process from the server, if the process is stop from the server it won't be access by any client.

Table 3: Simulation Parameters

No. of Request	100
Data Transfer Rate	30 kbps
Buffer Size	32
Resolution	357/442
No. of Frames	95

V. EXPERIMENTAL RESULT

The experimental results are performed using .net to evaluate the performance of proposed technique. The user can access the video very fast. The time required for extracting the video is less. To play the video, every time it is not necessary to download the flash player. The values of the experimental parameters are listed in Table 2. The streaming of video is done over websocket. The parameters are executed several times to analyze the performance by varying the number of request.

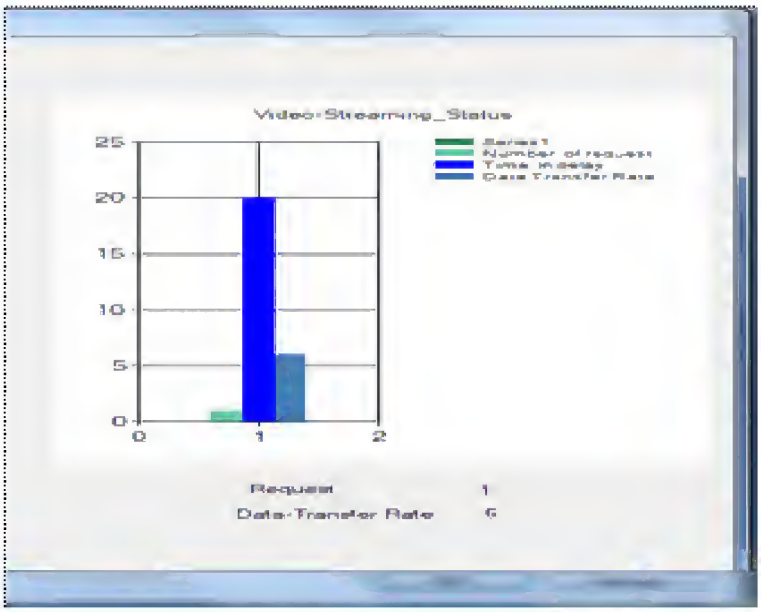

Figure 5: No of Request & Data Transfer Rate

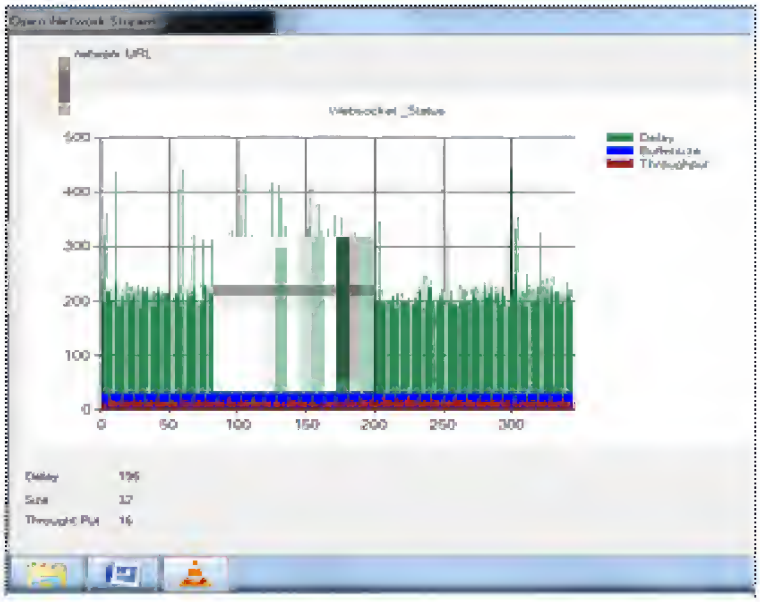

Figure 6: Web Socket Status 1

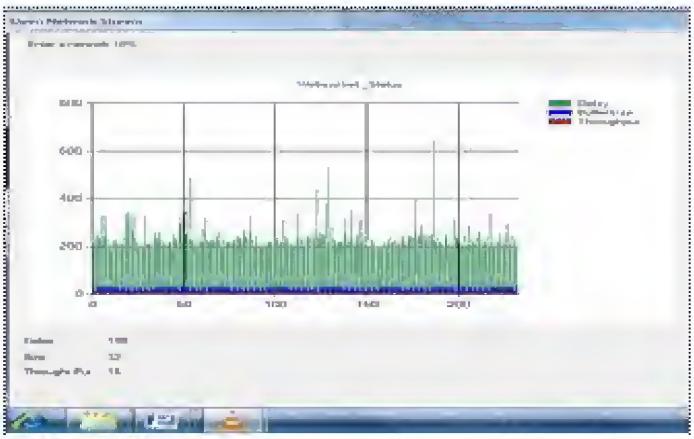

Figure 7: Web Socket Status 2

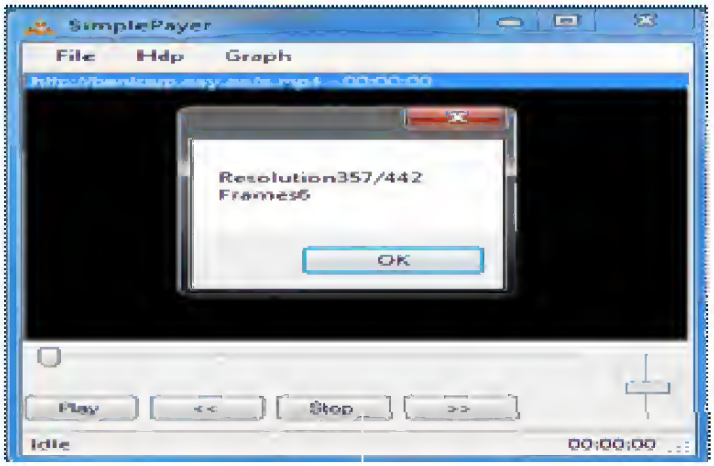

Figure 8: Resolution & Frames of a Video

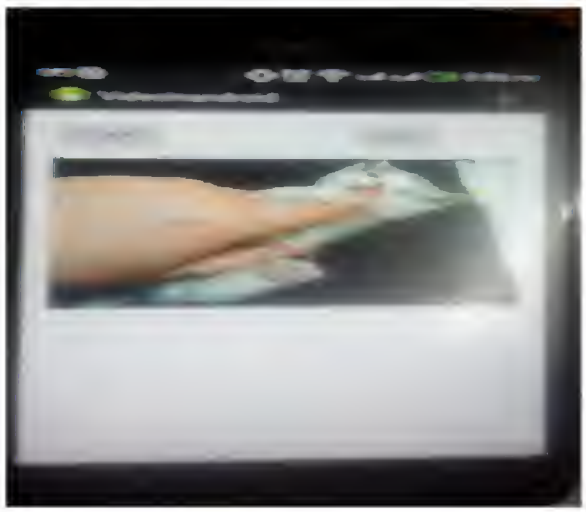

Figure 9: Client Machine (Mobile) Playing the Video Broadcasted by the Server

VI. COMPARATIVE ANALYSIS

The video streaming is performed several times by varying the number of request from 1 to 30 to analyze and compare the performance of websocket with the http. Websocket have delay performance better when number of request increases. The comparative graph of http versus websocket given in following figure

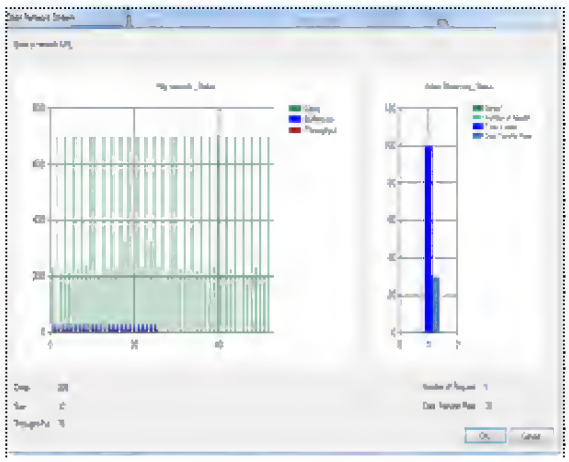

Figure 10: Http Status

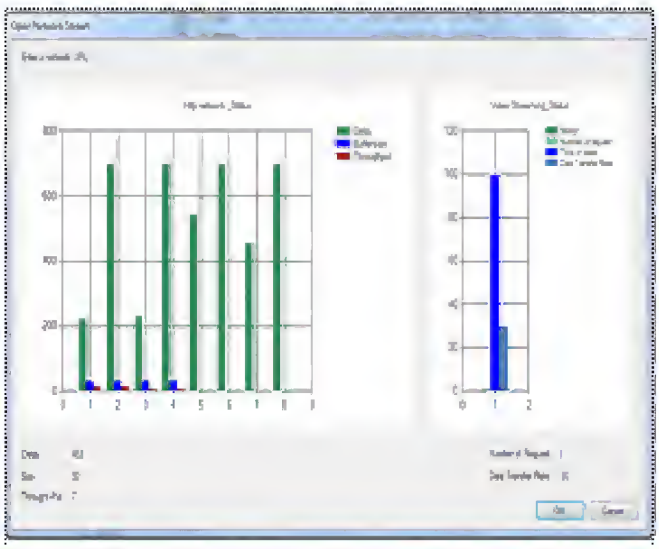

Figure 11: Delay & Throughput of http

VII. CONCLUSIONS

The previous work is reviewed and it is analyzed that there is need to adapt websocket so that it can perform well even if there are n no of request from the client. So, the main purpose of the proposed technique is to decrease delay increase the throughput, & maintain the session. Web Socket uses single socket to push and pull information between the browser and the server. In full duplex communication, this will not only avoid the problem of comet's connection and portability, but also be able to achieve higher efficiency. It also solves the problem of browser incompatibility. It shows that the throughput is increased by 48% than http. Also the delay factor is improved by 59% than http. The analysis can be extended as future work to pick up the performance by taking more number of requests into account. Further, the same can be extended by considering streaming of multiple videos simultaneously.

REFERENCES

1. SHStream :self-healing Framework for HTTP Video-Streaming 2013 13th IEEE/ACM international symposium on Cluster,Cloud, and Grid Computing .
2. Design of Rich Client Web Architecture Based on HTML5 2012 Fourth International Conference on Computational and Information Sciences
3. Application Research of Web Socket Technology on Web Tree Component. 2012 INTERNATIONAL SYMPOSIUM ON INFORMATION TECHNOLGY IN MEDICINE AND EDUCATION.
4. Application of HTML5 Multimedia. 2012 International Conference on Computer Science and Information Processing (CSIP) .
5. Research On The Integration of RTCWeb Technology with IP Multimedia Subsystem 2012 5th International Congress on Image and Signal Processing (CISP 2012)
6. Communicating and Displaying Real-Time Data with WebSocket Victoria Pimentel*Universidad Simón Bolívar* Bradford G. Nickerson *University of New Brunswick*

7. T. Stockhammer, "Dynamic adaptive streaming over http –: standards and design principles," in *Proceedings of the second annual ACM conference on Multimedia systems*, ser. MMSys '11. New York, NY, USA: ACM, 2011, pp. 133–144.
8. Remote Data Visualization through Websockets 2011 Eighth International Conference on Information Technology: New Genrations
9. A Low Latency and Intuitive Control Video Streaming System.the 1st IEEE Global Conference on Consumer Electronics 2012.
10. Application Research of Web Socket Technology on Web Tree Component. 2012 INTERNATIONAL SYMPOSIUM ON INFORMATION TECHNOLGY IN MEDICINE AND EDUCATION.
11. Design of Rich Client Web Architecture Based on HTML5 2012 Fourth International Conference on Computational and Information Sciences
12. Real-time Web Application Roadblock: Performance Penalty of HTML Sockets. IEEE ICC 2012 - Communication QoS, Reliability and Modeling Symposium.
13. Full-Duplex Relay Based on Distributed Beam forming in Multiuser MIMO Systems. IEEE TRANSACTIONS ON VEHICULAR TECHNOLGY,VOL,62,NO 4, MAY 2013
14. Research on Web Applications Using Ajax New Technologies 2008 International Conference on MulTiMedia and Information TechnoIogy.
15. Jesse James Garrett. Ajax: A New Approach to Web Applications.
16. http://adaptivepath.com/publications/essays/archives/000385.php.
17. Hybrid CDN-P2P Architectures for Live Video Streaming: Comparative Study of Connected and Unconnected Meshes. 2011 International Symposium on Computer Networks and Distributed Systems (CNDS), February 23-24, 2011
18. Efficient 3D Adaptive HTTP Streaming Scheme over Internet TV mm12-085 - Efficient 3D Adaptive HTTP Streaming Scheme over Internet TV
19. Real-time Web Application Roadblock Performance PenaIty of HTML Sockets IEEE ICC 2012 - Communication QoS, Reliability and Modeling Symposium