

Review Paper on AJAX Comet and Websocket Uses for Web HMI/SCADA

Ayyaj .I .Nadaf¹, S. V. Kulkarni². Dr. P. P. Shaha³. Dr. M. K. Bhanarkar⁴

Department of Electronics, Shivaji University, Kolhapur^{1,4},

Department of Electronics, Devchand College, Nipani^{2,3}

ayyaz.nadaf@gmail.com

Abstract— This paper presents review of Ajax Comet and Websocket technology and its uses for industrial control application. This could help for selecting the correct protocol in righteous application. AJAX is asynchronous JavaScript and Extensible Markup Language used for pushing data on a web. Comet is another dominant technology evolved for real time data access on web platform. Websocket is a new trend in which data is transferred in a bidirectional link established with lower layer model. Human Machine Interface (HMI) and Supervisory Control and Data Acquisition (SCADA) are the two major part in control application after Programmable Logic Controller (PLC).

Keywords— PROTOCOL, AJAX, COMET, WEBSOCKET, SCADA, PLC.

INTRODUCTION

IN Automated industry there are lots of process variable need to be controlled so as to get process under control. Simply saying there are lots of real time protocols available for real time monitoring of data. Prevalent manufacturer in the field of automation are Siemens, Allen-Bradley, Mitsubishi, Fanuc, Yamaha, Panasonic and much more stated their own protocols and dedicated hardware/software. New trend is of open source in this case they are building their platform for data interoperable. Same race found for the browser based technology. Basically they are platform independent. Some of supervisory control and data acquisition (SCADA) systems giving human interaction and data visualization for local and remote site using the html JavaScript functions. Websocket is evolution in server client web technology. It allows single TCP socket connection between server and client with little overhead. Advantage of using websocket is that it is full duplex bidirectional message sender receiver protocol with very low latency in transmission and reception. Websocket technology is now standardized which mean web technology is agreed standards for real time processing. Websockets are now reaching beyond for real time communication with cross platform standard. HTML5 is the key solution for modern platform independency. It is widely used in mobile and desktop applications. HTML5 with some advance add-on technic fulfill the job requirement.

BRIEF OVERVIEW.

Internet wasn't built for to be dynamic. It is gestated collection of Hyper Text Markup Language (HTML) pages. Over the time it became necessity to have rich components as a part of web designs like images, audio video etc. On the other hand server becomes more advance providing responses on the query basic leading this to dynamic HTML (DHTML) pages. Further development to this

leads developing of pages inbuilt with JavaScript. After few advances in DHTML requirement dynamic content updating took place by means of http polling within frames. Soon it is found that this technology is very hectic. As pages disappear and reappear for a little change. With reloading of complete page for minor update all information had to be resent which was already transmitted. This placed additional load on server with excess use of bandwidth. Things started to get exciting with introduction to live connect. Term Asynchronous JavaScript become familiar for very long period. The utility of page being updated in a background with background Hypertext Transmission protocol (HTTP) request remain obscure for very long time. In AJAX some Document Object Models (DOM), Cascaded Style Sheet (CSS) and HTML is used. The term 'AJAX' was publicly stated on 18th Feb 2005 by Jesse James Garrett in article titled "AJAX: A New Approach to Web Application". The first draft officially published on 5th April 2006 by World Wide Web Consortium (W3C) standard specifying XMLHttpRequest.

Comet is a web application programming technology used for making communication between client and server with pushing data from server to client. Http long-held request allows server to push data to client without browser's explicit request. Comet approach is differing to original web technology where client request complete web page at a time. This approach has other names http server push, two way web, reverse Ajax, http streaming, Ajax push. Comet is not an acronym it is known as low latency data for web. Comet is helpful to minimize the latency between server clients by sustaining long lasting http connection. But holding connection till server's event is not worth. Browser may be blocked for new request until its previous query gets satisfied. This makes a question on browser's usability. Comet technology falls in streaming and long polling category. Many comet application uses long polling as it is very straight forward than streaming algorithm. XMLHttpRequest Hidden iframe and Ajax long polling are the measures in this technology.

Websocket became game changer for browser war. The websocket protocol was standardized by Internet Engineering task force (IETF) in 2011. Originally it was designed to be used by web technology but now it can be deployed in any client server application. Websocket protocol provides full duplex communication over single TCP connection. It takes http handshake connection and upgrades the connection to websocket. Instead of using new page for new information websocket allows to remain on the same page with updated information. Users need not to refresh the page or different pages to send receive information. Websocket provide raw socket capabilities to web that's why it is possible to use it any real time monitoring purposes. Custom protocol can be implemented on the top of basic architecture of websocket. Web tier application as well as embedded application can be built on the top of websocket. It is best for the real time monitoring.

RELATED WORK

HTML5 was designed to be platform independent, and can be used on an increasing number of mobile devices for creating both mobile websites and mobile applications [1]. This thesis presents a flexible and configurable database management system designated COMET, suitable for mbedded systems and in particular, vehicle control-systems. To be able to handle the varying requirements imposed by different systems, COMET emphasizes configurability and tailorability, by adopting a component-based architecture. The result of this research is the implementation of COMET BaseLine, which is an instance of COMET suited to a particular vehicle control-system [2]. One of the most familiar SCADA (supervisory control and data acquisition) application protocols now is OPC protocol. This interface is supported by almost all SCADA, visualization, and process control systems. There are many research efforts tried to design and implement an approach to access an OPC DA server through the Internet. To achieve this goal they used diverse of modern IT [3]. It is feasible to create a realtime Human Machine Interface, HMI, solution to a Command & Control System in HTML5. The conclusion is based on some HTML5 features that is of interest for a system of this kind, a comparison between some of the most relevant web server solutions and a prototype. It shows that it is possible to create a HMI in HTML5 for a Command & Control System but with some delimitations. A proposal would be to create a HMI in HTML5 as a supplement to the current instead of replacing it. The JavaScript API used to present the map in the prototype is releasing an upcoming version with support for WebGL,

allowing smoother and graphical richer maps in the future [4]. Regarding the limitations of traditional web real-time communication solutions such as polling, long-polling, flash plug-in, propose that using new coming Web Socket technology in the web real-time communication field, introduce the features of Web Socket technology, analysis the difference between Web Socket protocol and HTTP protocol, offer an approach to implement the Web Socket both in client and server side, prove Web Socket can decrease network traffic and latency greatly by an experiment, made the prospect of future application of Web Socket in web real-time communication[5].

HOW IT WORKS AJAX COMET AND WEBSOCKET

All these technology were add-on to standard HTML. HTML5 version released with websocket Application Programming Interface (API) specifications. Polling, long-polling and streaming is done when HTTP request sent to server. Server then responds with web pages after acknowledgement. Many times when there is requirement of real time data for example medical instrument data readings, stock exchange, ticket sale, bookings update the client generally need to refresh the pages to updated information. But of course this is not the great solution. Real time communication attempts largely by polling and long polling by delaying the http connection completion. This is nothing but the comet based push technology. In comet push technology polling is done by method in which browser sends request at a regular interval and receive the response immediately. This is helpful if client and server have a scope of synchronization. Due to this approach unnecessarily application withdraws much bandwidth by opening new connection request and closes if data is not available. Streaming is done with connection kept open for definite time but streaming encapsulated in HTTP. Firewall and other filter mechanism chooses to buffer result; hence providing increase in latency to message delivery. Secure Socket Layer SSL provide additional shield for data security. All this push technology involve http request and response header which is unnecessary data forming latency. This adds overhead to bandwidth utilization.

With AJAX client can send and receive data in asynchronous manner without interfering existing display. Data is usually retrieved from XMLHttpRequest object need not to be a JavaScript Object Notation (JSON consumes less Bandwidth). DOM is accessed using JavaScript function and represents data to user on current page. JavaScript and XMLHttpRequest object both interchange data asynchronously between browser and server.

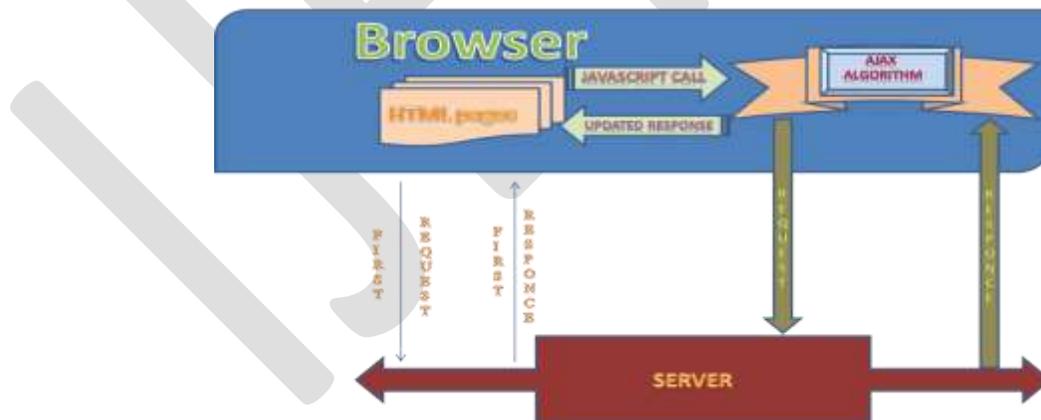


Figure 1: How Ajax works.

- Client browses pages from server and loads requested pages to client for first time.
- JavaScript function makes call to AJAX
- AJAX makes request to server asynchronously
- Server send information to AJAX

- Finally, Ajax respond back with updated information to the JavaScript call.

Simple jquery AJAX example:

```
var url='http://domainname/ajaxService';  
jQuery.ajax(  
  url:url,  
  success:function(data)  
  {  
    alert(data);  
  }  
);
```

In Comet Based web Application long held http request generated by browser get data from server without further request. Comet application eliminates the traditional polling and page by page modeling by real time persistent HTTP connection. According to HTTP 1.1 specification browser should not have more than two simultaneous http connections to web server. Therefore keeping connection ON for real time server events can lead other problems. Browser might be kept idle and cant able to generate new request until previous request is satisfied. This is a negative impact on browsers usability. In streaming application comet opens a single persistent connection to all comet events. For every new comet event from server side client handle this incrementally without closing connection. Streaming has negative impact on all modern browsers.

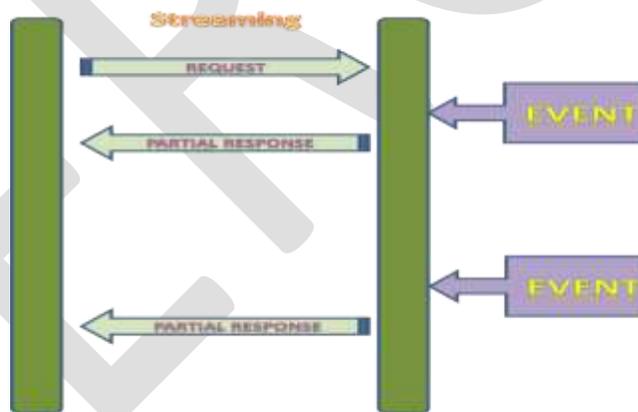


Figure 2: Streaming

Many implementation of comet application on different web browser uses long polling method. This is easier to deploy on any browser. It is very clear that from its name long polling involves client makes a polling request for comet events occur. This is similar to AJAX request style where browser keeps connection open until comet event occurs.

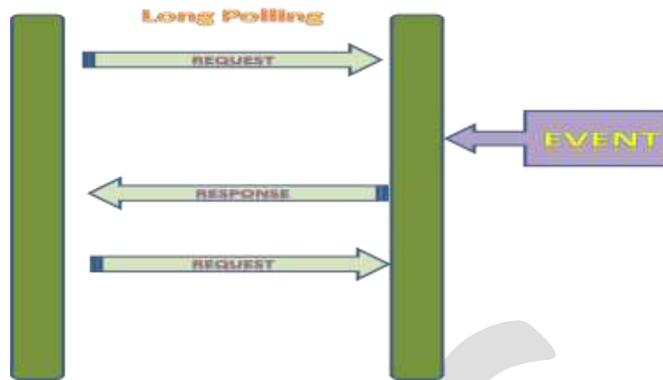


Figure 3: Long Polling

Advantage of this comet technology is that there is always a link between client and server; Server fetches the data to client as soon as it is available at client.

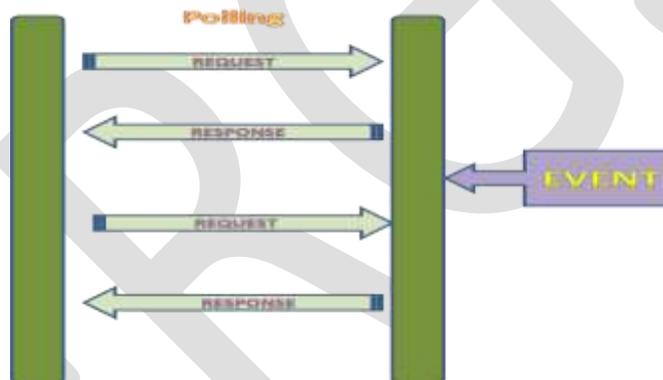


Figure 4: Polling

Websocket is a full duplex real time protocol over a single TCP connection. It provides few methods for data extraction and encapsulation. Following API creates websocket object and its associated event methods.

```
var url='http://domainname/ajaxService?';  
var Socket = new WebSocket(url);  
Socket.onopen=function()  
{  
    //This method is called when socket connection is established.  
}  
Socket.onclose=function()  
{  
    //This method is called when socket is closed  
}  
Socket.onmessage=function()
```

```
{  
    //This method is called when message is received  
}  
  
Socket.onerror=function()  
{  
    //This method is called when error occurred during communication  
}  
Socket.send(data)  
{  
    // This method is called to transfer the data from client.  
}
```

CURRENT TREND OF DEVELOPING WEB SCADA

Basic process control application starts with sensing the physical process sending this information to controller then according to the controller algorithm outputs are generated. This is the basic close loop system. SCADA system is involved for user intervention and data logging. This should be in real time because SCADA client might be at distinct places and in distinct number. Human Machine Interface (HMI) is used for same expect for data logging. Unlike SCADA HMI requires special type of hardware and software. Basic features of HMI/SCADA should

- Provide real time access to plant floor parameters.
- Provide major alarms of the control system.
- Provide colourfully recognisable machine parts.
- Provide numerical display and input for user interaction.
- Provide multiple protocol support.
- Provide fail safe design

Industrial controller requires transferring the real time data to controller. In many case there is need of storing the real time information to server for like historical trends. Another example is that to view historical alarms it is necessary to store current alarms. Many industrial scada systems use these technologies.

OPC system .net is a scada software developer company providing SCADA solution for the smartphones like android and apple. The solution is based on AJAX technology design and development of SCADA for oil gas, nuclear, facilities management, food processing industry.



Figure 5: OPC SCADA https://www.opcsystems.com/help/ImagesExt/image5_34.jpg

Inductive automation serves ignition web based SCADA.



Figure 6: Inductive SCADA

<https://inductiveautomation.com/static/images/product/webbased/wb3.png>

Ignition's web-based architecture is platform independent browser based scada runs on any platform and it is powered by JAVA.

Another open source SCADA is SCADA BR is developed by open source community of OPEN SCADA and MANGO automation open source project of serotonin.

PVBROWSER is yet another open source web based SCADA framework developed in c and c++. All above support many industrial communication protocol but they are not yet supporting websocket.

CSWORK is yet another framework for browser based scada. Support variety of data protocol format for example BACnet, Modbus, SQL database, OPC data access server. Support real time historical trending, alarming.

CONCLUSION

Websocket is new trend in real time web based data communication. It must be a part of modules in PLC based protocol communication. Currently industry is using AJAX web based scada system. Some open source scada software are using their own framework for it. Internet browser is being used by many mobile and laptop. If web socket is used in industrial environment lot of additional cost and energy can be saved. Browser based solution (HTML5 and websocket) can be easily deploy on any platform.

REFERENCES:

- [1] Jin-tae Park¹, Hyun-seo Hwang¹, Jun-soo Yun¹ and Il-young Moon¹ School of Computer Science and Engineering, Korea University of Technology and Education {wlsxo05, smilebear1, yuntn55, iymoon@koreatech.ac.kr “Study of HTML5 WebSocket for a Multimedia Communication”. International Journal of Multimedia and Ubiquitous Engineering Vol.9, No.7 (2014),pp.61-72 <http://dx.doi.org/10.14257/ijmue.2014.9.7.06>
- [2] Dag Nystrom “COMET: A Component-Based Real-Time Database for Vehicle Control-Systems” Copyright Dag Nyström, 2003 ISBN 91-88834-46-8 Printed by Arkitektkopia, Västerås, Sweden Distribution: Mälardalen University Press.
- [3] Hosny A. Abbas & Ahmed M. Mohamed “Review on the Design of Web Based SCADA Systems Based on OPC DA Protocol ” International Journal Of Computer Networks (IJCN), Volume (2) : Issue (6)
- [4] Henrik Rydstedt “HTML5 as HMI in a Command &Control System” Tryckt av: Reprocentralen ITC ISSN: 1401-5749, UPTEC IT 14 004 Examiner: Lars-Åke Nordén Ämnesgranskare: Lars Oestreicher Handledare: Jakob Sagatowski
- [5] [Qigang Liu](#), [Xiangyang Sun](#), [Sydney Institute of Language & Commerce, ShangHai University, ShangHai, China](#). “Research of Web Real-Time Communication Based on Web Socket” [10.4236/ijcns.2012.512083](#) PP. 797-801.
- [6] Dilip Kumar Sharma (G.L.A. Institute of Technology and Management, India), Gopalji Varshneya (G.L.A. Institute of Technology and Management, India) and Ashwani Kumar Upadhyay “AJAX in Development of Web-Based Architecture for Implementation of E-Governance” Volume 3, Issue 3. Copyright © 2007. 14 pages.
- [7] Li-Jie Cui¹, Hui He² and Hong-Wei Xuan¹, Harbin, China ²Harbin Institute of Technology, Harbin, China andyclj1977@163.com, hehui@hit.edu.cn, henryxuan@hotmail.com “Analysis and Implementation of an Ajax-enabled Web Crawler” International Journal of Future Generation Communication and Networking Vol. 6, No. 2, April, 2013.
- [8] D. G. Synodinos, “HTML 5 Web Sockets vs. Comet and Ajax,”2008. <http://www.infoq.com/news/2008/12/websockets-vs-comet-ajax>.
- [9] Chwan-Hwa Wu, Senior Member, IEEE, J. David Irwin, Fellow, IEEE, and Fa Foster Dai, Senior Member, IEEE “Enabling Multimedia Applications for Factory Automation” IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 48, NO. 5, OCTOBER 2001.
- [10] Songchar Jiang “Wireless Communications and a Priority Access Protocol for Multiple Mobile Terminals in Factory Automation” IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 14, NO. 1, FEBRUARY 1998.
- [11] Adnan Salihbegovic, Vlatko Marinković, Zoran Cico, Elvedin Karavdić, Nina Delic ETF Sarajevo, Bosnia and Herzegovina b Bosna-S Oil Services Company, Bosnia and Herzegovina “Web based multilayered distributed SCADA/HMI system in refinery application” Computer Standards & Interfaces 31 (2009) 599–612.
- [12] Bahram Ravani, Magomed Gabibulayev, Ty A. Lasky, “Improvement of a Human-Machine Interface (HMI) for Driver Assistance Using an Event-Driven Prompting Display”, IEEE transactions on control systems technology, vol. 19, no. 3, may 2011 page no 622-627.
- [13] Angelos Amditis, Luisa Andreone, Katia Pagle, Gustav Markkula, “Towards the Automotive HMI of the Future: Overview of the AIDE-Integrated Project” IEEE transactions on intelligent transportation systems, vol. 11, no. 3, september 2010 page no 567-578.
- [14] Edwin Lughofer, James E. Smith, Muhammad Atif Tahir, “Human-Machine Interaction Issues in Quality Control Based on Online Image Classification”, IEEE transactions on systems, man, and cybernetics—part a: systems and humans, vol. 39, no. 5, september 2009 page no 960-971.
- [15] Doukas and Kleantlis Thramboulidis, “A Real Time Linux-Based Framework for Model-Driven Engineering in Control and Automation” IEEE transactions on industrial electronics, vol. 58, no. 3, march 2011 page no 914-924