

Application Research of WebSocket Technology on Web Tree Component

JIANG Feng-yan

School of Information Science and Engineering,
Shandong Provincial Key Laboratory for Distributed
Computer Software Novel Technology.
Shandong Normal University
Jinan, China
jiangfengyan1209@126.com

DUAN Hui-chuan

Basic Education Department,
Shandong Normal University,
Jinan, China
mailsdnu@126.com

Abstract—In the development of information systems, Web tree component is an important way to show information, provide a convenient and intuitive information queries, which has become one of the essential features in the information system. At present, Web tree component still has a series of problems in communication efficiency, displaying and updating rate, etc. This paper uses MyEclipse and MySql as development tools to design and implement web tree component based on WebSocket, the component can display and update tree structure dynamically, Theory analysis and experimental results show that the improved scheme provides a better user experience, comparing with the previous traditional long polling based on AJAX technology, it effectively reduces the amount of data transmission, communication overhead, etc.

Keywords—Websocket; Web tree component; AJAX; Long-polling

I. INTRODUCTION

The tree structure [1] is a very popular way to show information, it is widely used in web applications, has the advantages of intuitive and convenient, especially in expressing the directory structure, hierarchy, etc. Currently, most web tree component is based on AJAX technology, AJAX has proven successful for providing dynamic content without reloading pages, but, using XmlHttpRequest to periodically query the server, requires that a new socket connection and a new HTTP message be created for each update [2], which is resource-intensive. Comet technology [3] achieving bidirectional communication is a way to solve the problems mentioned. However, when data is received or a timeout terminates the connection, the browser reconnects and waits for the next update, this could result in more communication overhead, and comet technology consumes server's bandwidth and resources greatly because of maintaining two connections, WebSocket [4] is a web technology, which defines a full-duplex communication channel that operates through a single socket over the web [5]. It only establishes a TCP Socket connection after the first request for connection, which can be able to save server's resources, network bandwidth, and achieve real-time communication well, it is a better way to solve the above problems. Web tree component based on WebSocket technology has achieved great improvement in the aspects of real-time, communication overhead and the delay in time.

II. WEBSOCKET

WebSocket uses single socket to push and pull information between the browser and the server in full-duplex communication[6], this will not only avoid the problem of comet's connection and portability, but also be able to achieve higher efficiency than AJAX long polling. At present, the HTML5 WebSocket is the principal mechanism for promoting the web full-duplex real-time communication [7].

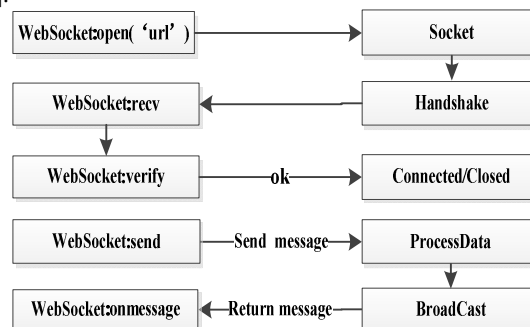


Figure 1. WebSocket connection establishment model

Fig 1 is a WebSocket connection establishment model, WebSocket protocol is essentially a TCP. To establish a WebSocket connection, the client and server upgrade from the HTTP protocol to the WebSocket protocol during their initial handshake [8], once established, WebSocket data frames can be sent 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 WebSocket API defines mainly four callback methods: onopen, onmessage, onclose and onerror to deal with event trigger during the WebSoccket connection. It is very suitable for real-time, event-driven web applications.

III. COMPARISON OF TECHNOLOGIES IMPLEMENTING WEB TREE COMPONENT

The basic operations of Web tree include expanding nodes, collapsing nodes, adding nodes, modifying nodes, deleting nodes, Web tree component based on AJAX must send requests to the server at the basic operation of the tree, frequent HTTP requests can result in unnecessary traffic, reduce the efficiency of network communications and the

speed of displaying and updating the web tree structure. Meanwhile, AJAX technology updates the tree structure through sending request to the server at set intervals, which will increase server load when a large scale of clients access server for updates based on a predefined interval.

Compared with solutions based on AJAX technology, Web tree components based on Comet technology can reduce network traffic and server load to some extent, this way also exists much limitations. First, if an event occurs, the server sends the data to the client and the client reconnects, it can increase the numbers of socket opening and closing operation. Second, data updating may be a little delay, it needs time to re-establish connection to the server after client completed one update, which may undermine the consistency of the data in the case of multiple-person cooperation; and comet also wastes server resources for management of connections, thereby directly impacting the responsiveness to the user. WebSocket technology provides a better solution to display the Web tree structure, greatly reduces the server load and the quantity of data transmission, achieves effective communication of web tree component.

IV. THE IMPLEMENTATION OF THE WEB TREE COMPONENT BASED ON THE WEBSOCKET TECHNOLOGY

Adopt Jetty as WebSocket server in this scheme, correlative WebSocket class is quoted from the package named jetty-websocket-7.4.5.v20110725.jar, Jetty is a pure Java Web container, component of this paper can also be deployed in Jetty directly. Implement web tree component based on WebSocket technology, using MyEclipse and Mysql as development tools.

A. Improvement of Component's overall Sstructure of Class Diagram

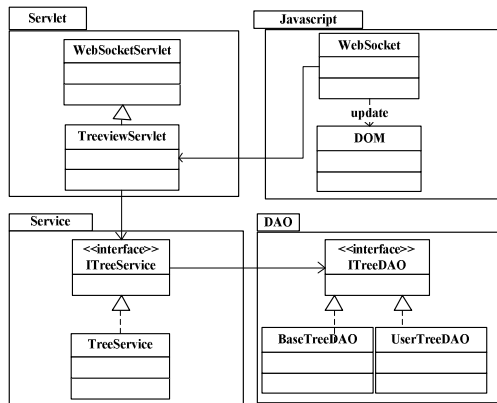


Figure 2. Component's overall structure of class diagram

The design and implementation of component on the server side is based on J2EE [9-10] architecture. Server-side use tree node object-TreeViewElement to store tree node information, which contains node number, parent node, node name, node description, node attribute set, etc, and passed as a parameter between DAO and Service classes in the system.

Fig 2 is a component's UML overall structure of the class diagram, in which all methods and properties of class

are omitted due to space reasons. The component consists of four packages, Javascript package is API functions supported in the internal of JavaScript language, using the API component client can realize AJAX to read, write, alter the DOM of the currently active document, and make response to the event; WebSocketServlet subclass in the servlet package receive all the client requests, they call different Service according to different requests, pass request object and response object as parameters, and end in returning processing results to the client. Service package defines the API achieving the service logic of web tree component, each service class accesses database using dependent class in DAO and returns processing results; DAO is an object that provides an abstract interface to some type of database or persistence mechanism, providing some specific operations without exposing details of the database. the interface exposed by the DAO to clients does not change when the underlying data source implementation changes, this pattern allows the DAO to adapt to different storage schemes without affecting its clients or business components. Essentially, the DAO acts as an adapter between the component and the data source.

In addition to the classes in Fig 2, the server-side uses the JSP Tree Tag technology to print HTML and JavaScript code to client side. JSP Tree Tag is a customizable tags and can generate tree structure dynamically, tag's parameters can be used to set the type of web tree, whether to support right-click menu, and whether the node has checkbox, which can simply the configure of client-side javascript.Using Dojo component library to display the tree structure and operation menus in the detailed client-side implementation. Dojo is an excellent open-source JavaScript tool collections, it is intended to provide a better, easier and faster tool collection to facilitate Web development. It provides rich page display and supports the development of dynamic pages; Dojo also supports event handling mechanisms and WebSocket technology, etc.

B. Operation Procedure and Core Code of Component

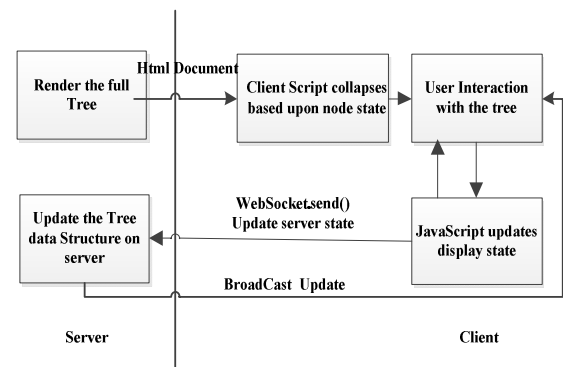


Figure 3. Operation procedure based on WebSocket

Fig 3 is operation procedure of Web tree component based on WebSocket. The Web tree component directly spans tree structure on the server side and then implements client display, if the user operates a tree node, client calls the WebSocket "send" method to transfer the message to server

which is then pushing it to all connected users. Main class and its functions implementing Web tree component is below:

TABLE I. MAIN CLASS AND ITS FUNCTIONS

class name	function
TreeviewElement	define the attributes of the tree node
DbManager	interaction with the database
Category	obtain a collection of nodes that meet certain conditions
InitServlet	define a List to store connected users
TreeWebSocketServlet	achieve WebSocketServlet and return a TreeWebSocket object
TreeviewRender	display updated tree structure
TreeWebSocket	display updated tree structure

The TreeWebSocket class realizes an OnTextMessage interface, the key method in this TreeWebSocket class is onMessage, we call it to notify all connected users by iterating on the users set when we receive a client message, main core code is given below:

```

public void onMessage(String data)
{
    List<TreeWebSocket> socketList =
InitServlet.getSocketList();
    for (TreeWebSocket socket:socketList) {
        try {
            socket.getConn().sendMessage(data);
        }catch(IOException e){
            e.printStackTrace();
        }
    }
}

```

The client side call onmessage method to obtain data from the server side and use the updates to update tree structure, main core code is below:

```

ws.onmessage=function(evt){
    var parentID=evt.data;
    if(parentID!=null&&parentID.equals("")){
        Category category = new
        Category(Integer.parseInt(parentID));
        var top =category.getChildren();
        response.setContentType("text/html");
        java.io.PrintWriter out = response.getWriter();
        for(var i= 0;i<top.length;i++){
            out.println(treeviewRender.renderTreeViewAjax(top[i],false))
        }
    }
};

```

V. ANALYSIS AND STUDY ON THE PERFORMANCE

To further prove the advantages of the WebSocket technology in the Web tree component, this section designs a practicable scheme to test data delay, CPU utilization, network throughputs etc, which generated in the operation of tree structure. The server Jetty runs on a 32-bit Windows 7 PC with quad-core Duo CPU. Compared with long polling scheme based on AJAX technology, the program based on the WebSocket technology has great advantages in minimizing server load, reducing network throughputs and

real-time communication, etc. Performance testing result is illustrated in Fig4:

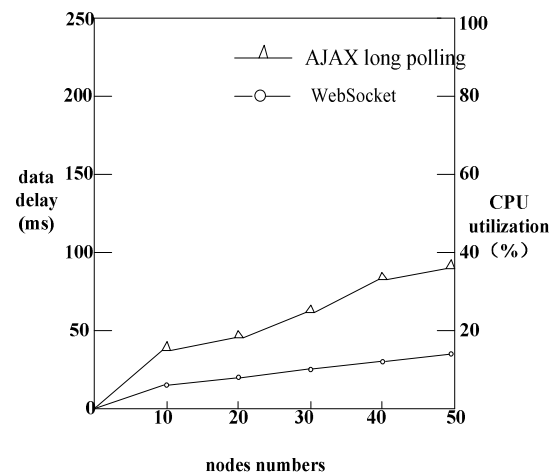


Figure 4. Performance testing result

From the Fig 4 it can be seen network delay and data delay of Web tree component based on WebSocket is far lower than that based on AJAX long polling in the case of the expansion of the same node number. And AJAX long polling needs to manage two connections that causes the increase of the CPU usage directly. Further analysis indicated that the data delay of the proposed scheme is far less than 50 ms, it is because once established WebSocket data frames can be sent back and forth between the client and the server in full-duplex mode, client needs not re-send a connection request, so requests time for getting updates again is zero. However, AJAX long polling needs to generate numerous request/response cycles to keep the connection alive, which inevitable meeting increases the additional time expenditure of real time network communication. As it is depicted in Fig 4, WebSocket program has much better performance on data delay and CPU utilization than the latter.

Assume tree structure is updated once per second, By means of the Opera Dragonfly, total HTTP requests and responses header information overhead of web tree component can be obtained, web tree component based on AJAX long polling contains 1050 bytes, However each of these messages based on WebSocket is a WebSocket frame that has as little as two bytes of overhead. If client number is 1000, network throughput based on these two approaches is shown in Table II, the units is bytes per second.

TABLE II. CMPARION OF NETWORK THROUGHPUT

	long polling	WebSocket
Connection mode	re-send request to the server after updates	keep the connection
client number	1000	1000
information overhead	1050	2
Network throughput	1050000	2000

TableII illustrates that WebSocket technology greatly reduces the network overload and improves communication

efficiency. In this paper, web tree component is designed based on WebSocket technology, client need not initiates a connection to the server when operating tree structure, which can avoid extra overhead, and, WebSocket is a light weight messaging protocol, it can reduce the quantity of node data transmission, save bandwidth and enhance the expansion speed of tree structure.

VI. CONCLUSIONS

Web tree component based on WebSocket realizes real-time data transmission between the server and client by exploiting WebSocket interface. Compared to AJAX polling and Comet solutions, the adoption of WebSokcet effectively lowers the server load, reduces node data transmission, cuts the communication cost, simplifies implementation complexity, etc. It greatly improves the web service capacity and brings users a better experience of tree operation.

ACKNOWLEDGMENT

This research was supported by the Natural Science Foundation of Shandong Province of China (No. ZR2011FM001)

REFERENCES

- [1] WANG Long,LIU Yan-heng,LI Xiao-guang,ect.Chinese high-frequency word extraction algorithm based on tree structure and weighted entropy[J]. Journal of Jilin University, 2011.41(1):188-192.
- [2] Christian Heilmann.Beginning JavaScript with DOM scriptingand AJAX: From novice to professional: Beginning: from noviceto professional [M]. Apress, 2006:61-84,299-342.
- [3] ZHOU Ting. Comet: Long Server Push Technology Based on HTTP [EB/OL].IBMDeveloperWorksChina.http://www.ibm.com/developerworks/cn/Web/wa-lo-Comet/.2007,8(31).
- [4] W3C.The WebSocket API [EB/OL].http://dev.w3.org/html5/websockets/.2010.
- [5] Peter Lubbers,Brain Albers,Frank Salim.Pro HTML5 Programming[M].Apress press,2010.
- [6] M. Bhide, P.Deolasee, A. Katkar, A. Panchbudhe, K. Ra-mamritham, and P. Shenoy. Adaptive push-pull: Disseminat-ing dynamic web data. IEEE Trans. Comput.,2002,51(6):652-668.
- [7] Mark Pilgrim. HTML5:Up and Running[M].O'Reilly press,2010.
- [8] Greg Wilkins. AJAX, Comet and Jetty [EB/OL].http://www.webtide.com/downloads/ whitepaperAJAXJetty.html,2006,06,11.
- [9] Rod Johnson.Expert one-on-one J2EE design and development (programmer to programmer)[M].New ed.Wrox,2002:113-178,251-284,441-514.
- [10] Deepak Alur,John Crupi,Dan Malks.Core J2EE patterns: Bestpractices and design strategies[M]. 2nd ed. Prentice Hall PTR,2003: 22-29,390-407,408-420.