基于 Websocket 的消息实时推送设计与实现*

齐 华[1],李 佳[1],刘 军[2]

(1. 西安工业大学电子信息工程学院,西安 710000;
2. 武警工程学院通信工程系,西安 710086)

摘 要:随着互联网技术的快速发展,基于 B/S 架构的实时通讯和消息推送的应用范围越来越广泛,服务器消息推送是很多应用中的一项重要功能,服务器推送技术的优劣直接影响着消息推送的效率。传统的解决方案有 html refresh,定时轮询和 comet,但这些实现方案存在着资源消耗大、执行效率低、服务器负担过大等缺陷。Html5 中引入了 websocket 这一全双工通信协议,利用此技术实现的方案能够在减轻服务器负担的同时高效地对服务器消息进行实时推送。使用 node. js 平台和 socket. io 库技术设计并实现了一种基于 websocket 协议的服务器实时消息推送平台。运行测试结果显示能快速稳定地实现消息的实时服务器推送。

关键词:服务器推送;轮询;全双工协议;websocket 协议;node. js 技术;socket io 框架
DOI:10. 3969/j. issn. 1002 - 2279. 2016. 03. 011
中图分类号:TP311. 1 **文献标识码**:B **文章编号**:1002 - 2279(2016)03 - 0036 - 04

Design and Implementation of Real - time Server Message Push Based on Websocket

Qi Hua[1], Li Jia[1], Liu Jun[2]

(1. *College of Electronic and Information Engineering, Xi'an Technological University, Xi'an* 710000, *China*;
2. *Dept. of Communications Engineering, China Engineering College of Armed Police Force, Xi'an* 710086, *China*)

Abstract: With the rapid development of Internet technology, the application scope of the real - time communication and message push based on B/S framework is becoming more and more widely, and the server message push is one of the most important functions and the server push technology has a great influence on the efficiency of message pushing. The traditional solutions, such as html refresh, timing polling and comet, have disadvantages of resource wasting, low implementation efficiency and high server burden. Html 5, employing the websocket technology, as a full - duplex communication mechanism, is implemented to decrease server burden for the message push in real time. In this paper, the real - time message push system is designed and completed by the node. js platform and socket. io technology. The running test result shows that it can implement the real - time message server push rapidly and stably.

Key words: Server push; Timing polling; Full - duplex; Websocket protocol; Node. js technology; Socket io frameworks

1 引 言

随着互联网技术的快速发展,基于 B/S 架构的应用越来越广泛,如体育赛事的图文直播,网页版 QQ 和微信等在线聊天应用及 web 监控系统等。在此类应用中,人们对于消息获取的实时性要求很高。由于 Web 应用是建立在 HTTP 协议的基础上的,而 http 协议的工作模式是“请求 - 响应”,即用户通过浏览器客户端主动发出数据请求,服务器处理请求后将响应发送给用户,在这种模式下,服务器并不能主动推送消息至客户端,使得消息的实时推送成为困难。常见的解决方案有基于轮询技术的消息推送和基于 http 长连接的 comet[1]。但基于轮询的消息推送是一种伪服务器实时推送,实际上是一种客户

* 基金项目:陕西科学技术研究发展计划项目(2014K05 - 19)
作者简介:齐华,(1963 -),女,陕西省咸阳市人,博士,教授,主研方向:无线传感器网络,信息传输技术。
收稿日期:2015 - 05 - 11

端拖拽的方式。其实现原理是浏览器每隔一段时间刷新一次来更新网页内容,这种方法存在着很多缺点,如用户体验差,刷新时间间隔的设置缺乏灵活性,易浪费服务器资源,推送消息的实时性差等[2]。comet 是一种基于 http 长连接的服务器推技术,通过 ajax 引擎来进行浏览器请求和服务器响应的传送。浏览器通过 ajax 发送请求至服务器,服务器保持请求,直到有新的消息到达时才会返回响应到浏览器。浏览器收到返回响应后重新发送请求建立连接。这种方式存在着长时间保持连接浪费资源,连接数目过多时服务器负担大等缺点[3]。Websocket 技术的出现,很好地解决了服务器推送面临的诸多困难,有着很好的应用前景[4]。

2　Websocket 技术

Websocket 是 html5 中的一种新协议,它可以实现真正意义上的浏览器与服务器之间的全双工通信,服务器可以主动实时地将消息推送至浏览器端。浏览器与服务器之间只需经过一次握手连接建立一条通道,此后服务器就可以通过这条通道推送消息[5]。Websocket 协议的工作过程如图 1 所示。

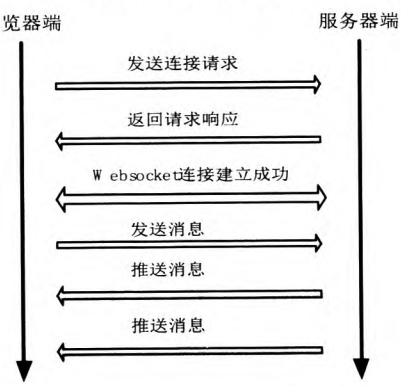

图 1　Websocket 协议工作流程

在 websocket 连接建立过程中,首先执行浏览器与服务器之间的第一次握手,浏览器通过 js 发送连接请求,第一次请求包含以下格式的内容:

GET /chat HTTP/1.1

Host: server. example. com

Upgrade: websocket

Connection: Upgrade

Sec - WebSocket - Key: dGhlIHNhbXBsZSBub25jZQ = =

Origin: http://example. com

Sec - WebSocket - Protocol: chat, superchat

Sec - WebSocket - Version: 13

服务器处理请求后返回以下格式的响应:

HTTP/1.1 101 Switching Protocols

Upgrade: websocket

Connection: Upgrade

Sec - WebSocket - Accept: s3pPLMBiTxaQ9kYG - zzhZRbK + xOo =

Sec - WebSocket - Protocol: chat

此时连接建立成功[6]。以下是对其中各项内容的解释:

Connection:Upgrade 向服务器说明这是一个协议升级请求;

Upgrade:websocket 说明升级为 websocket 协议;

Sec - websocket - Key:客户端生成的长度为 24 字节的随机字符序列,用于握手过程的验证;

Sec - websocket - Version:13 表明客户端支持的 websocket 协议版本;

Sec - websocket - Extensions:客户端希望收到的数据格式。

服务器在收到请求后,根据头信息判别是否为 websocket 请求,随后提取出其中的 Sec_websocket - key 字段信息,按照一定的算法生成一个新的字符序列,包含在响应头信息中,即 Sec - websocket - Accept。客户端按照同样的算法处理发送到服务器端的 Sec - websocket - Key,生成一个字符序列,再与收到的响应中的 Sec - websocket - Accept 进行对比,若一致则验证成功,连接建立成功,否则连接建立失败。Websocket 连接建立成功后,会一直保持,双方通过此连接通道收发消息,直到有一方主动关闭连接为止[7]。

3　node. js 和 socket. io

Node. js 是一个 javascript 运行环境,封装了 google V8 引擎,使得其执行 javascript 代码的速度和性能都有了很好保障。具有事件驱动,非阻塞 I/O 模型等特性,非常适合数据密集型的实时应用[8]。

Socket. io 是一个 websocket 库,提供了实现 websocket 协议的客户端 javascript API 以及服务器端的 node. js API,具有跨平台特性,通过它可以方便地进行实时通讯方面的开发[9]。

4　服务器推送系统的设计和实现

4.1　系统整体设计

在 Windows 7 平台下实现基于 websocket 协议的服务器实时消息推送。首先安装 node. js 环境和 socket. io 库,利用 socket. io 提供的 javascript api 来搭建客户端和服务器端的功能。服务器端中,请求

http 模块和 socket. io 模块的功能，监听特定端口，根据自定义的特定事件推送消息至客户端中，并设定好收到来自客户端的消息时的响应函数以及连接断开时的响应函数。同理，在客户端搭建中，通过 js 发送连接建立请求，设定收到服务器端的返回消息的响应函数并自定义发送消息的事件。

4.2 客户端实现

```
<html xmlns = "http://www.w3.org/1999/xhtml" >
<head >
<meta http - equiv = " Content - Type" content = "text/html;charset = utf - 8" / >
<title > 服务器消息查看 </title >
<body >
<h1 align = "center" > 服务器信息查看 </h1 >
  
<p > <strong > 与服务器连接状态: </strong >
<div id = "state" > </div > </p >
  
<p > <strong > 实时信息: </strong >
<div > 时间: <span id = "time" > </span > </div >
 
<div > 内容: < span id = " content" > </span > </div >
</p >
< script src = " http://localhost: 888/socket. io/socket. io. js" > </script >
<script type = "text/javascript" >
var rexian = io. connect ( ´http://localhost: 888´ ); //创建 socket. io 实例，建立连接;
rexian. on( ´connect ´,function( ) {
console. log( ´成功建立 websocket 连接´ );
document. getElementById ( " state" ). innerText = "websocket 连接建立成功";
} );//连接建立成功后的响应;
rexian. on( ´stc´,function( data) {
console. log( data) ;
document. getElementById( ´time´ ). innerText = data. warningtext[0]. time;//获取服务器消息发送时间;
document. getElementById( ´content´ ). innerText = data. warningtext[1]. content;
} );//获取服务器消息具体内容;
rexian. on( ´disconnect´,function( ) {
document. getElementById( ´state´ ). innerText = " 与 websocket 服务器连接断开" ;} );//与服务器断开连接后的响应;
</script >
</body >
</html >
```

4.3 服务器端实现

```
var http = require( ´http´ );//请求 http 模块;
var io = require( ´socket. io´ ) ;
var server = http. createServer( function( req,res) {
res. end( ´ </body > </html > ´ ) ;
} ). listen( 888) ;//创建 http 服务器的主函数，监听“888”端口号;
console. log( ´Http Server start at 888´ ) ;
var websocket = io. listen( server) ;
websocket. on( ´connection´,function( client) {
console. log( ´websocket 连接建立成功´ ) ;
client. emit( " stc" , { " warningtext" : [ { " time" : new Date( ) } , { "content" : "温度超标" } ] } ) ;
} ) ;
websocket. on( ´disconnect´,function( ) {
console. log( ´与客户端连接已断开´ ) ;
} ) ;
```

4.4 运行测试

目前很多主流的浏览器端都已经支持websocket 协议，常见的有 chrome，Firefox，IE10，Oprea，Safari。本文选用 chrome 浏览器平台，运行服务器端程序，再通过 chrome 浏览器访问客户端页面，运行测试。服务器端与浏览器端运行结果分别如图 2 和图 3 所示。

从图 2 和图 3 中可以看出，在运行服务器端和浏览器端程序后，成功建立 websocket 连接，与服务器连接状态一栏更新为“websocket 连接建立成功”。Websocket 提供的 emit 发送方式允许用户自定义事件来触发发送函数，可以设置不同的事件来触发不同的消息推送函数，这极大地提高了服务器消息实时推送的灵活性。能够在各类 web 监控系统中得到广泛应用，实时将预警信息推送至客户端。此实时消息推送系统中，服务器端在发生“stc”事件时，将实时消息以 json 格式发送至浏览器端，浏览器端顺利地接收到实时消息的发送时间以及内容。此外，根据 websocket 协议的工作流程，连接通道建立之后会一直保持，直到某一方主动断开连接。本文设计的服务器推送系统在服务器端主动断开连接后，浏览器端会向控制台发送连接断开的消息，更新与服务器的连接状态，如图 4 所示。

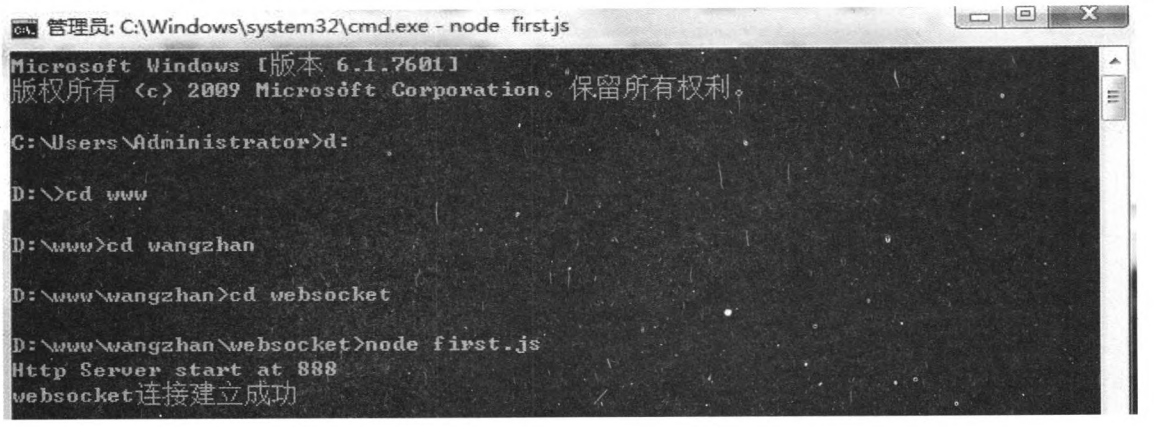

图 2　websocket 服务器运行

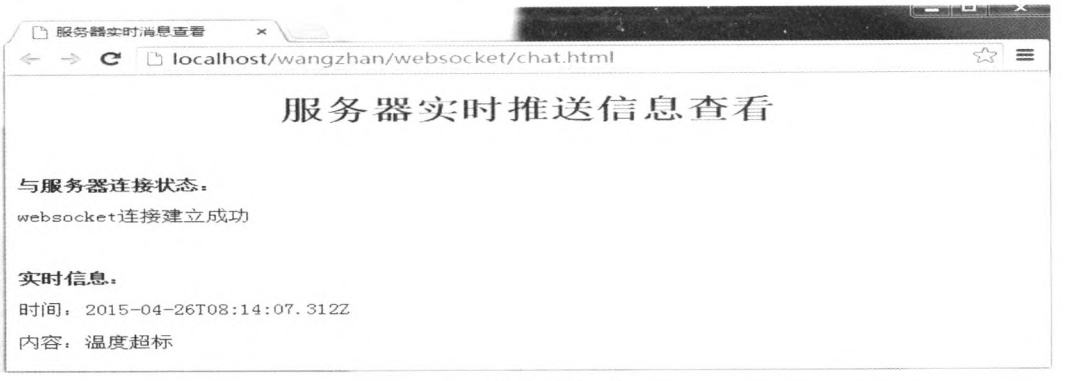

图 3　浏览器端运行

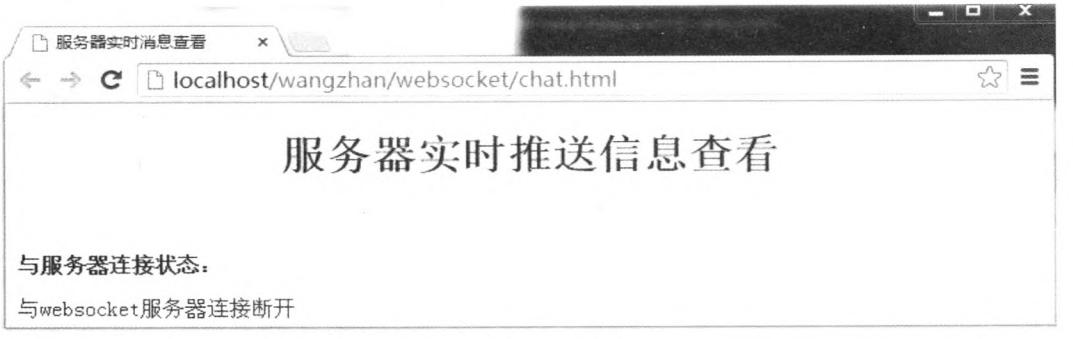

图 4　服务器主动断开连接

5　结 束 语

Websocket 为消息的服务器实时推送提供了一个高效简洁的解决方案,能实现真正意义上的浏览器与服务器之间的全双工通信。此 http 服务器通过 node. js 构建,由于 node. js 的单线程,非阻塞 I/O 调用等特性,使得其并发性能异常强大并很适合在分布式设备上运行数据密集型的实时应用。通过压力工具测试服务器的性能结果显示,其能够支持数以万计的并发连接。从 websocket 协议的运作机制可以看出,在每个实时消息的推送过程中,传送的仅仅是实时消息本身,无需重复发送 http 头信息,并且由于 http 连接会一直保持到某一方主动断开,消息的推送都是实时的,大大降低了消息传送的延时。与传统的 ajax 轮询式消息推送解决方案相比,基于 websocket 协议的实时消息推送系统具有网络吞吐量低,延时小等优势。随着 html5 技术的迅速发展,相信作为其重要特性之一的 websocket 技术必将得到更广泛的重视和应用。

参考文献:

[1] 薛真真. 基于服务器推送和事件流处理技术的实时 Web 系统研究[D]. 杭州:浙江大学,2008.
Xue Zhen Zhen. Real - Time web system research based on server push and event stream processing technology [D]. Hangzhou:ZheJiang university,2008.

[2] 丛红艺. Pushlet 网络推技术研究及应用[D]. 吉林:吉林大学,2007.
Cong Hong Yi. The research and application of Pushlet network push technology [D]. Jilin: JiLin university, 2007.

(下转第 43 页)

6 结束语

首先对 NSCT 变换后的电力设备夜间红外图像的低频系数利用直方图双向均衡技术进行增强处理,并对红外图像的高频系数利用自适应阈值的方法进行分离,对大于阈值的部分进行放大处理,对低于阈值的部分进行抑制。实验证明提出的算法可以有效提高电力设备夜间红外图像的亮度和对比度,增强图像暗区细节,取得很好的视觉效果。在以后的研究中,要着重研究电力设备红外图像的自适应增强技术,提高算法的实用性。

参考文献:

[1] 石丹,李庆武,倪雪,等. 基于 Contourlet 变换的红外图像非线性增强算法[J]. 光学学报,2009,29(2):342 - 346.
Shi Dan,Li Qingwu,Ni Xue,et al. Infrared image nonlinear enhaneement algorithm based on contourlet transofrm [J]. Acta Optica Sinica,2009,29(2):342 - 346.

[2] 龚昌来,罗聪,杨冬涛,等. 一种基于平稳小波域的红外图像增强方法[J]. 激光与红外,2013,43(6):703 - 707.
Gong Changlai. Luo Cong,Yang Dongtao,et al. Infrared image enhaneement method basde on stationary wavelet domain[J]. Laser & Infrared,2013,43(6):703 - 707.

[3] 吴一全,史俊鹏. 基于多尺度 Retinex 的非下采样 Contourlet 域图像增强[J]. 光学学报,2015,35(3):1 - 10.
Wu Yiquan,Shi Junpeng. Image enhancement in non - subsampled contourlet transform domain based on multi - scale retinex[J]. Acta Optica Sinica, 2015,35(3):1 - 10.

[4] 杜超本,贾振红,覃锡忠,等. 基于 NSCT 的遥感图像模糊增强算法[J]. 计算机工程,2012,38(4):188 - 189.
Du Chaoben,Jia Zhenhong,Tan Xizhong,et al. Remote Sensing Image Fuzzy Enhancement Algorithm Based on NSCT [J]. Computer Engineering, 2012, 38 (4): 188 - 189.

[5] H Demirel,G Anbarjafari. Image resolution enhancement by using discrete and stationary wavelet decomposition [J]. IEEE Transactions on Image Processing,2011,20(5):1458 - 1460.

[6] Cunha A L,Zhou J,Do M N. The nonsubsampled contourlet transform: theory, design and applications[J]. IEEE Transactions on Image Processing, 2006, 15 (10): 3089 - 3101.

[7] 向静波,苏秀琴,陆陶. 基于 Contourlet 变换和形态学的图像增强方法[J]. 光子学报,2009,38(1):224 - 227.
Xiang Jingbo,Su Xiu,Lu Tao. Image enhancement based on the contourlet transform and mathematical morphology [J]. Acta Photonica Sinica,2009,38(1):224 - 227.

[8] 陈钱,柏连发,张保民. 红外图像直方图双向均衡技术研究[J]. 红外与毫米波学报,2003,22(6):428 - 430.
Chen Qian,Bo Lian Fa,Zhang Baomin. Histogram double equalization in infrared image[J]. J. Infrared Millim. Waves,2003,22(6):428 - 430.

[9] 梁栋,殷兵,于梅,等. 基于非抽样 Contourlet 变换的自适应阈值图像增强算法[J]. 电子学报,2008,36(3):527 - 530.
Liang Dong,Yin Bing,Yu Mei,et al. Image enhancement based on the nonsubsampled contourlet transform and adaptive threshold[J]. Acta Electronica Sinca,2008,36(3):527 - 530.

[10] 刘海波,汤群芳,杨杰. 改进直方图均衡和 Retinex 算法在灰度图像增强中的应用[J]. 量子电子学报,2014,31(5):525 - 532.
Liu Haibo,Tang Qunfang,Yang Jie. Application of improved histogram equalization and Retinex algorithm in gray image enhancement [J]. Chinese Journal of Quantum Electronicd,2014,31(5):525 - 532.

[11] 朱国庆,李庆武,林少飞,等. 基于人眼视觉特性的非下采样轮廓波变换域红外图像增强算法[J]. 激光与光电子学进展,2015,52,1 - 6.
Zhu Guoqing, Li Qingwu, Lin Shaofei, et al. Infrared image enhancement algorithm based human visual system characteristic via non - subsampled contourlet transform domain[J]. Laser & Optoelectronics Progress, 2015, 52:1 - 6.

(上接第 39 页)

[3] 周婷. 基于 HTTP 长连接的"服务器推"技术[DB/OL]. (2007.08.20)/[2015.05.01]. http://www.ibm.com/developworks/cn/web/wa - lo - comet/.
Zhou Ting. Server push technology based on HTTP long - time connection[DB/OL]. (2007.08.20)/[2015.05.01]. http://www.ibm.com/developworks/cn/web/wa - lo - comet/.

[4] 周乐钦,燕彩蓉,苏厚勤. 基于 Web - Socket 协议的推送数据技术在监控系统中的应用研究[J]. 计算机应用与软件,2013(5):229 - 32.
Zhou Le Qin,Yan Cai Rong,Su Hou Qin. The research of the application of the message push technology based on Web - Socket protocol in the monitoring system [J]. Computer application and software. 2013(5):229 - 32.

[5] 刘华星,杨庚. HTML5—下一代 Web 开发标准研究[J]. 计算机技术与发展,2011(8):54 - 58,62.
Liu Hua Xing,Yang Gen. HTML5—research on the next generation Web developing standard [J]. Computer technology and development,2011(8):54 - 58,62.

[6] W3C. The Websocket API[DB/OL]. (2010.03.01)/[2015.05.01]. http://dev.w3.org/html5/websockets/.

[7] Website of project Pushlets [DB/OL]. (2008.11)/[2015.05.01]. http://www.pushlets.com/.

[8] CometD Bayeux Ajax Push [DB/OL]. (2010.01)/[2015.05.01]. http://www.cometd.com/.

[9] zhangxin09. Ajax、comet、HTML5 Websocket 技术比较分析[DB/OL]. (2013.01.04)/[2015.05.01]. http://blog.csdn.net/zhangxin09/article/details/8464150/.