Websocke

# 简介

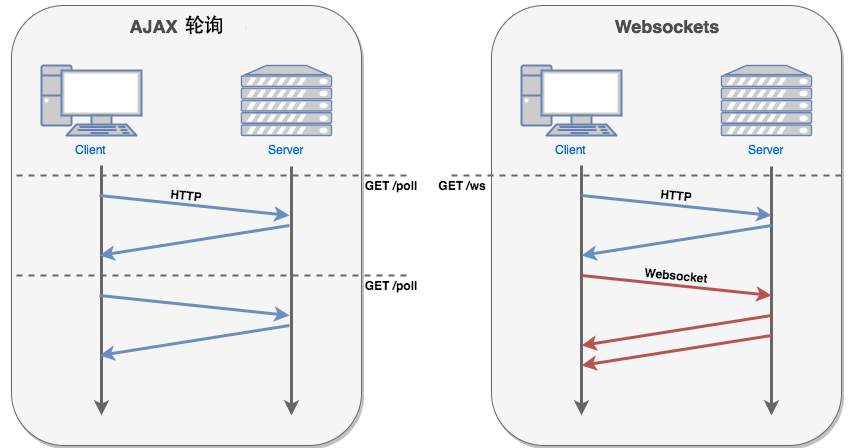
（<https://www.runoob.com/html/html5-websocket.html>）

WebSocket 是 HTML5 开始提供的一种在单个 TCP 连接上进行全双工通讯的协议。

WebSocket 使得客户端和服务器之间的数据交换变得更加简单，允许服务端主动向客户端推送数据。在 WebSocket API 中，浏览器和服务器只需要完成一次握手，两者之间就直接可以创建持久性的连接，并进行双向数据传输。

现在，很多网站为了实现推送技术，所用的技术都是 Ajax 轮询。轮询是在特定的的时间间隔（如每1秒），由浏览器对服务器发出HTTP请求，然后由服务器返回最新的数据给客户端的浏览器。这种传统的模式带来很明显的缺点，即浏览器需要不断的向服务器发出请求，然而HTTP请求可能包含较长的头部，其中真正有效的数据可能只是很小的一部分，显然这样会浪费很多的带宽等资源。

HTML5 定义的 WebSocket 协议，能更好的节省服务器资源和带宽，并且能够更实时地进行通讯。



# 前端websocket的实现

var socket=new WebSocket(url) //创建websocket对象

socket.onopen=function(){ //打开连接，创建与服务的通信

}

socket.onmessage=function(data){ //服务返回的数据

}

socket.onerror=function(){ //通信发生错误

}

socket.onclose=function(){ //连接关闭时触发

}

socket.send() //主动给服务器发送数据

socket.close() //关闭连接

# 后端websocket服务提供

1. 安装cnpm install ws
2. 创建服务

var WebSocketServer = require('ws').Server;//引入websocket

//开启服务9088端口

var wss = new WebSocketServer({

port: 9088

});

//连接通信调用

wss.on('connection', function connection(ws, req) {

//前端请求send调用事件

ws.on('message', function incoming(message) {

//message 前端请求参数

wss.clients.forEach((client) => {

client.send(message)

})

});

})