

Java Websockets

Stick a spork in the web socket, it's done.

Brian S. Sheldon

me

- 30 years of ~~mistakes opportunities~~ work experience
- 10+ years of Cobol
- 13 years of Java
- 10 months of Javascript and Node.js

I am no expert

Feel free to tell me how to improve code, etc...

This talk was slapped together over the weekend

WebSocket uses TCP

WebSocket is a computer [communications protocol](#), providing [full-duplex](#) communication channels over a single [TCP](#) connection.

The WebSocket protocol was standardized by the [IETF](#) as [RFC](#) 6455 in 2011, and the WebSocket [API](#) in [Web IDL](#) is being standardized by the [W3C](#).

<https://en.wikipedia.org/wiki/WebSocket>

- HTTP
- TLS
- TCP

Publish Subscribe

In [software architecture](#), **publish–subscribe** is a [messaging pattern](#) where senders of [messages](#), called publishers, do not program the messages to be sent directly to specific receivers, called subscribers, but instead characterize published messages into classes without knowledge of which subscribers, if any, there may be. Similarly, subscribers express interest in one or more classes and only receive messages that are of interest, without knowledge of which publishers, if any, there are.

https://en.wikipedia.org/wiki/Publish%E2%80%93subscribe_pattern

Uses for Websockets

- Instant messaging
- Games
- Notifications (fire/bomb threat/fire bomb threat/new message)
- Financial ticker
- Sport scores (go team!)

<http://www.infoworld.com/article/2609720/application-development/9-killer-uses-for-websockets.html>

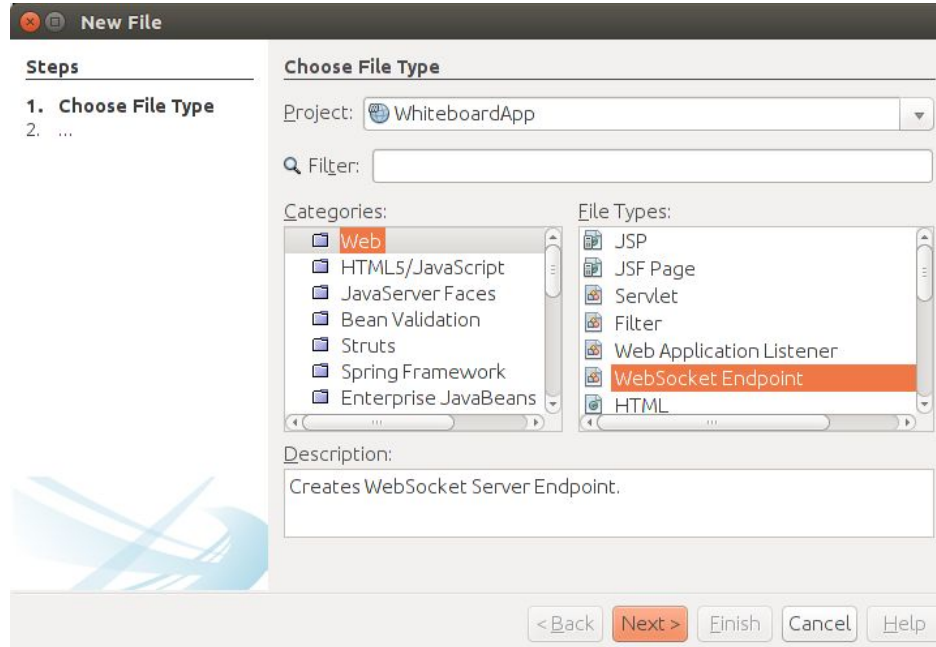
Use Caution

1. **Proxy servers can wreak havoc with WebSockets running over unsecured HTTP**
2. **Web browsers allow huge numbers of open WebSockets**
3. **WebSockets and HTTP/2 transport are not unified**
4. **Sophisticated WebSocket implementations end up re-inventing HTTP**

<https://samsaffron.com/archive/2015/12/29/websockets-caution-required>

Creating a websocket endpoint in NetBeans

File/New File



New WebSocket Endpoint

Steps

1. Choose File Type

2. **Create WebSocket Endpoint**

Name and Location

Class Name:

Project:

Location: ▼

Package: ▼

Created File:

WebSocket URI:

< Back

Next >

Finish

Cancel

Help



The image shows a code editor window with two tabs: "NewMarker.java" and "SomeEndpoint.java". The "SomeEndpoint.java" tab is active, displaying the following Java code:

```
1 package org.sample.whiteboardapp;
2
3 import javax.websocket.OnMessage;
4 import javax.websocket.server.ServerEndpoint;
5
6 @ServerEndpoint("/endpoint|")
7 public class SomeEndpoint {
8
9     @OnMessage
10     public String onMessage(String message) {
11         return null;
12     }
13 }
14
```

The code is formatted with line numbers on the left. The package is `org.sample.whiteboardapp`. It imports `javax.websocket.OnMessage` and `javax.websocket.server.ServerEndpoint`. The `@ServerEndpoint("/endpoint|")` annotation is highlighted. The `SomeEndpoint` class implements the `@OnMessage` annotation with a method `onMessage` that returns `null`.

javax.websocket.server.ServerEndpoint

<https://javaee-spec.java.net/nonav/javadocs/javax/websocket/server/ServerEndpoint.html>

```
import javax.websocket.OnMessage;
import javax.websocket.server.ServerEndpoint;

@ServerEndpoint("/hello");
public class HelloServer {

    @OnMessage
    public void processGreeting(String message, Session session) {
        System.out.println("Greeting received:" + message);
    }
}
```

Mapbox-gl.js

<https://www.mapbox.com/mapbox-gl-js/api/>

<http://localhost:8080/javawebsockets/>

localhost:8080/javawebsockets/

Apps Deltek M G Mail Yahoo RMM saic NetFlx Streamin bok SAIC Self-Service E glasses stuff Tools java android NETGEAR Router readme maint request Other bookmarks

Elements Console Sources Network Performance Memory Application Security Audits

View: Preserve log Disable cache Offline No throttling

Filter 100000 ms 200000 ms 300000 ms 400000 ms 500000 ms 600000 ms 700000 ms 800000 ms

Name X Headers Frames Cookies Timing

| | Data | Le.. | Time |
|-------------------|--|------|---------|
| newmarkerendpoint | [{"id":1,"lng":-97.45443018557538,"lat":35.63751266866494}] | 58 | 10:2... |
| | [{"id":1,"lng":-97.4887874289888,"lat":35.523531614018694}] | 58 | 10:2... |
| | [{"id":2,"lng":-97.45124390211095,"lat":35.574020636093735,"randomimg":"images/a1.gif"}] | 86 | 10:2... |
| | [{"id":3,"lng":-97.06876348202532,"lat":35.86043608114372,"randomimg":"images/a2.gif"}] | 85 | 10:2... |
| | [{"id":1,"lng":-97.45124390211095,"lat":35.574020636093735}] | 59 | 10:2... |
| | [{"id":1,"lng":-97.06876348202532,"lat":35.86043608114372}] | 58 | 10:2... |

1 / 73 requests | 0 B / 645 KB...

Select frame to browse its content.

Console

top Filter Info 7 items hidden by filters

```
Object {readyState: 4, getResponseHeader: function, getAllResponseHeaders: function, setRequestHeader: function, overrideMimeType: function...}
    eennndd
    bailing my.js:30
    clickkkk 1 my.js:34
    calling makePopupPicker -97.06876348202532 35.86043608114372 my.js:38
    778 150 ▶ Object {lngLat: LngLat, point: Point, originalEvent: MouseEvent, type: "mouseup", target: e} my.js:147
    createMarker 35.86043608114372 -97.06876348202532 my.js:65
    lng -97.06876348202532 lat 35.86043608114372 my.js:127
    newMarker my.js:225
    ▶ Object {id: 3, lng: -97.06876348202532, lat: 35.86043608114372, randomImg: "images/a2.gif"}
    sendNewMarkerToServerUsingRest my.js:231
    setView ▶ (2) [Marker, Marker] my.js:177
    setView after each -97.45124390211095 35.574020636093735 -97.06876348202532 35.86043608114372 my.js:199
    successsss undefined nocontent my.js:240
    ▶ Object {readyState: 4, getResponseHeader: function, getAllResponseHeaders: function, setRequestHeader: function, overrideMimeType: function...}
    eennndd
    setView ▶ [Marker] my.js:177
    setView after each -97.06876348202532 35.86043608114372 -97.06876348202532 35.86043608114372 my.js:199
    bailing my.js:30
    setView ▶ [] my.js:177
    bailing my.js:30
```

JavaEE Async chat JAX-RS

Useful links

- My github repo <https://github.com/brianssheldon/javawebsockets>
- Where I got the java websocket code <https://netbeans.org/kb/docs/javaee/maven-websocketapi.html>
- Mapbox using a local map <http://fuzzytolerance.info/blog/2016/02/09/Vector-Tiles-Part-1/>
- Mapbox examples <https://www.mapbox.com/mapbox-gl-js/examples/>
- Map file <https://openmaptiles.org/downloads/>

I'm off to see Guardians of the Galaxy 2