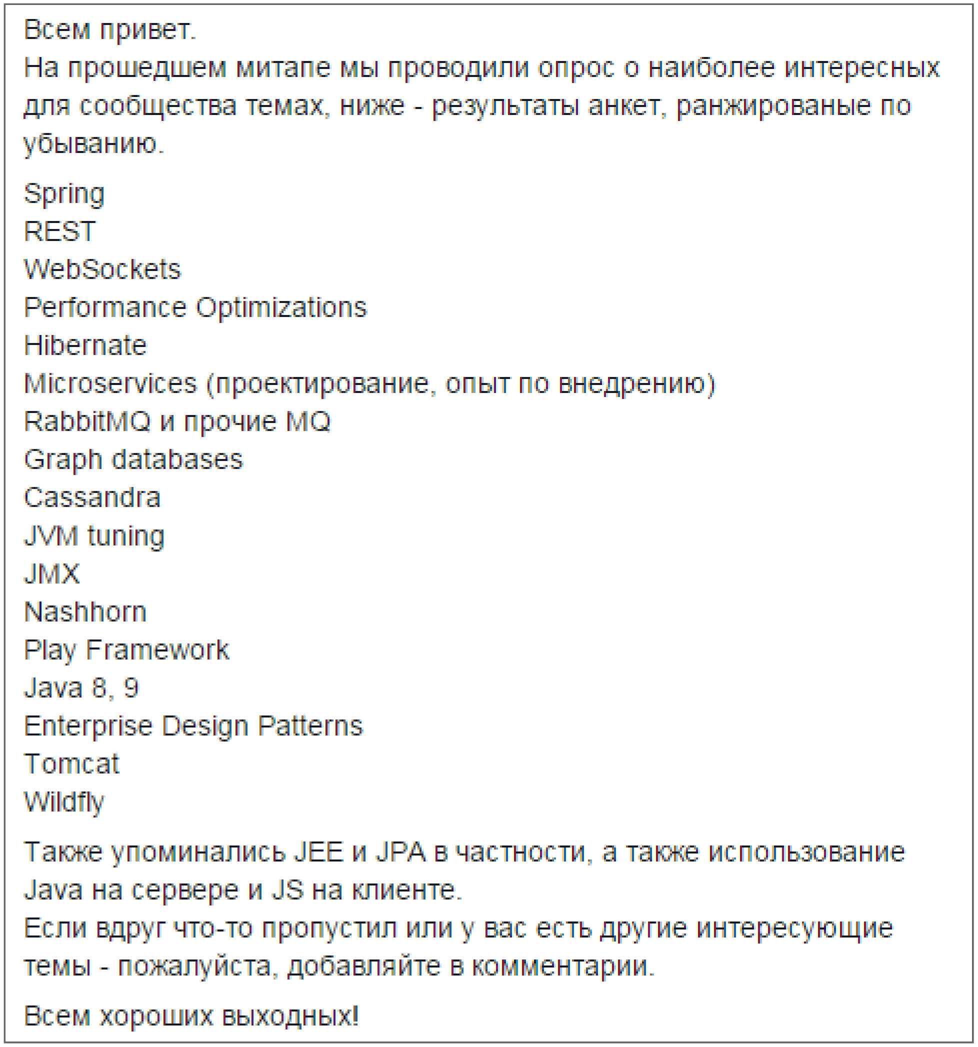
INTRODUCING WEBSOCKETS

Ibragimov Ruslan, ruslan@ibragimov.by



Всем привет.
На прошедшем митапе мы проводили опрос о наиболее интересных для сообщества темах, ниже - результаты анкет, ранжированые по убыванию.

Spring
REST
WebSockets
Performance Optimizations
Hibernate
Microservices (проектирование, опыт по внедрению)
RabbitMQ и прочие MQ
Graph databases
Cassandra
JVM tuning
JMX
Nashhorn
Play Framework
Java 8, 9
Enterprise Design Patterns
Tomcat
Wildfly

Также упоминались JEE и JPA в частности, а также использование Java на сервере и JS на клиенте.
Если вдруг что-то пропустил или у вас есть другие интересующие темы - пожалуйста, добавляйте в комментарии.

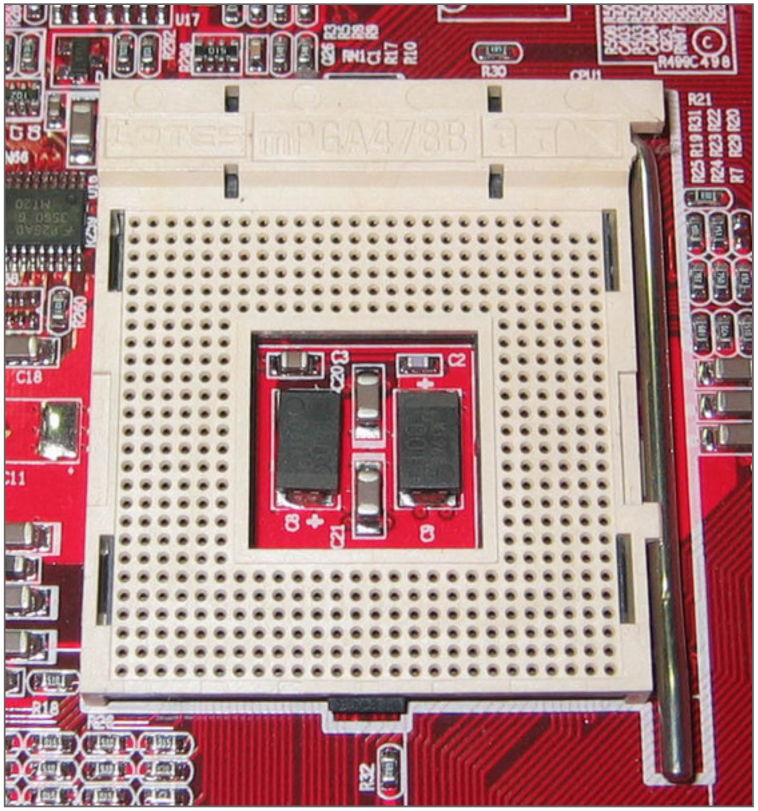
Всем хороших выходных!

AGENDA

1. Socket
2. WebSocket
3. Как мы жили без них раньше
4. Применения WebSocket
5. Стандарты: RFC 6455, JSR 356, W3C WebSocket API
6. Библиотеки
7. spring-websocket
8. STOMP
9. Security
10. Масштабирование

SOCKET?

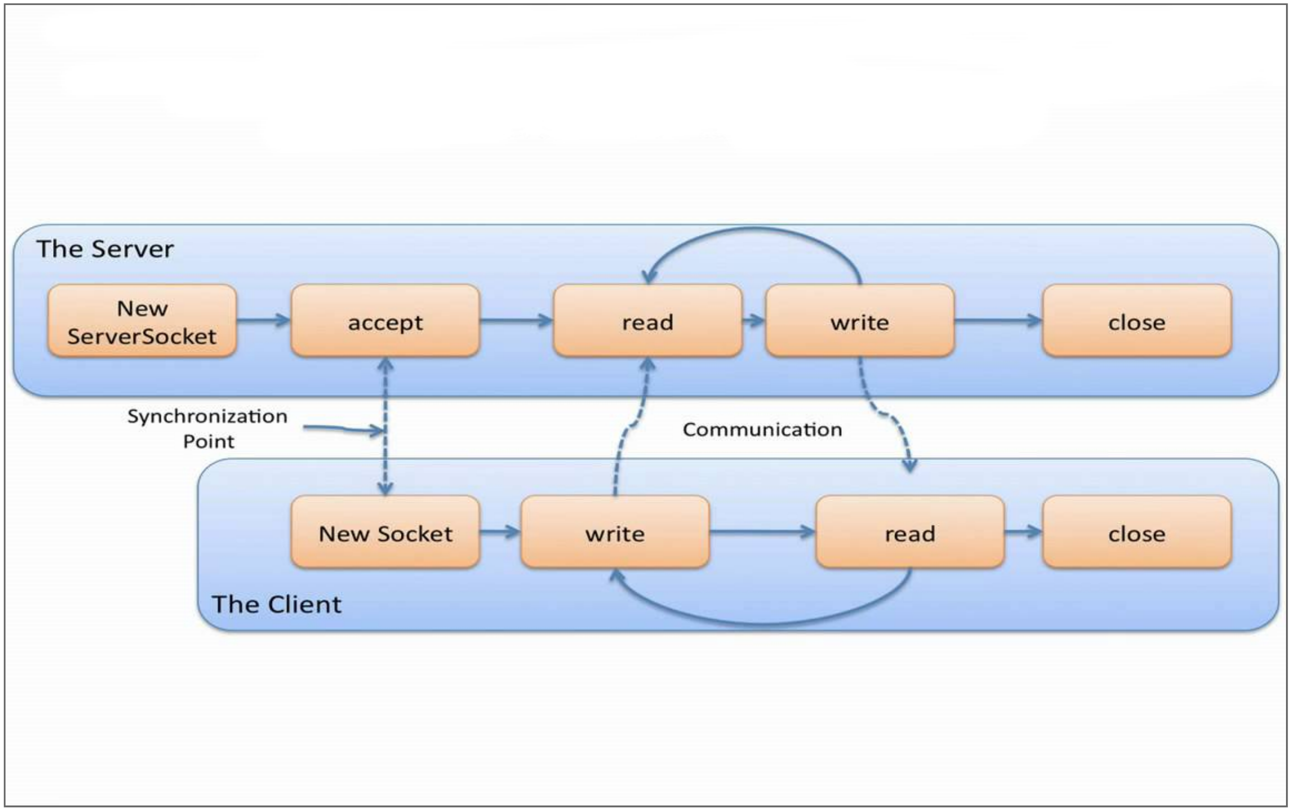
SOCKET?



SOCKET IN JAVA

```
// On client side
// client socket (TCP or UDP, default TCP)
// @since JDK 1.0
Socket socket = new Socket(hostName, portNumber);

// On server side
// server socket
ServerSocket serverSocket = new ServerSocket(portNumber);
Socket connectedClient = serverSocket.accept();
```

SOCKET IN JAVA



SOCKET IN JAVA

```
// Server logic:
while (true) {
    accept a connection;
    create a thread to deal with the client;
}

// Or in case of NIO:
ServerSocketChannel
```

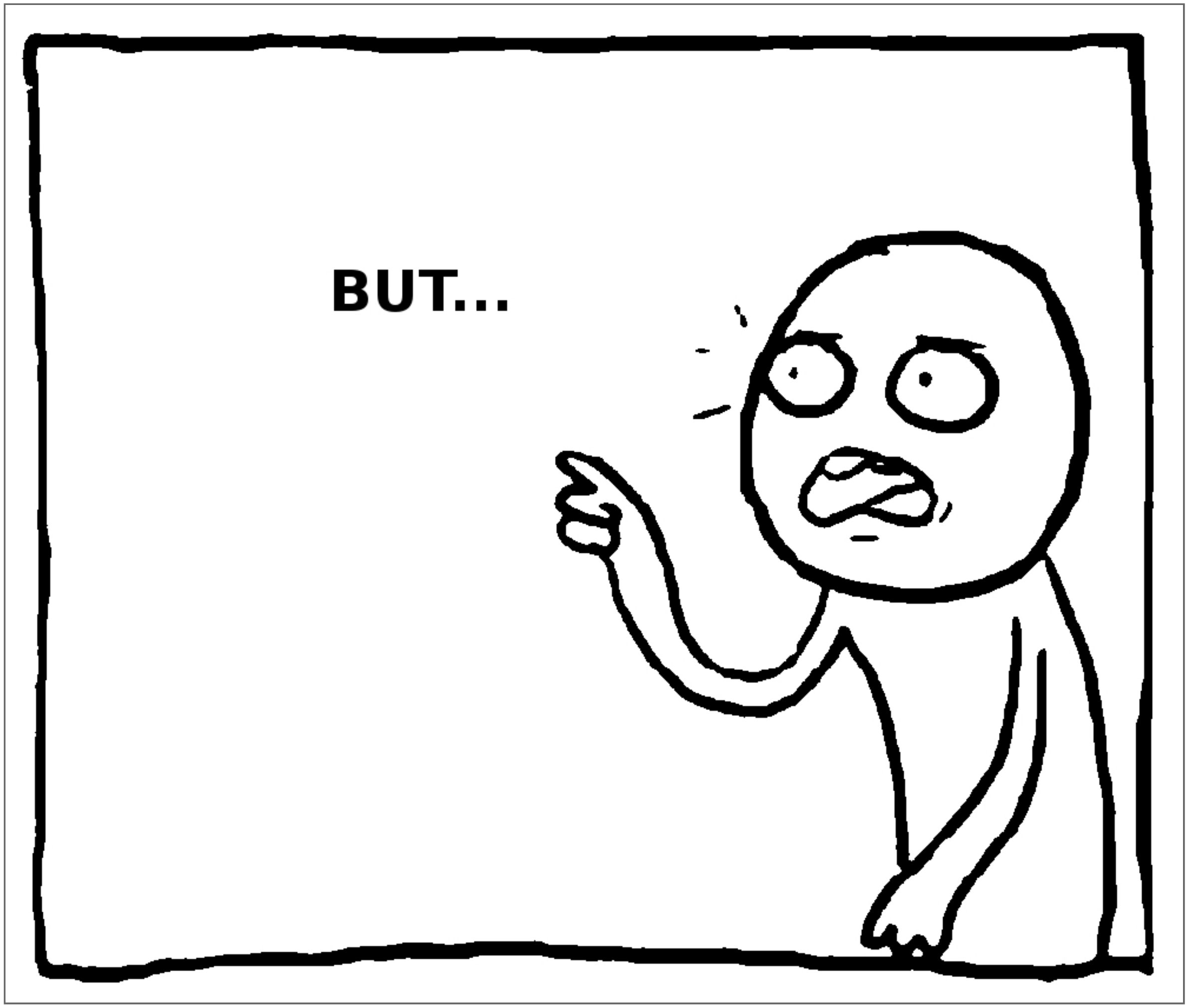
Alexey Diomin - Need for Speed: Netty & Protobuf (youtube)

> *A socket is one endpoint of a two-way communication link between two programs running on the network. A socket is bound to a port number so that the TCP layer can identify the application that data is destined to be sent to.*

What Is a Socket?

HTTP

> *An HTTP session is a sequence of network request-response transactions.*



Okay

HTTP/1.1

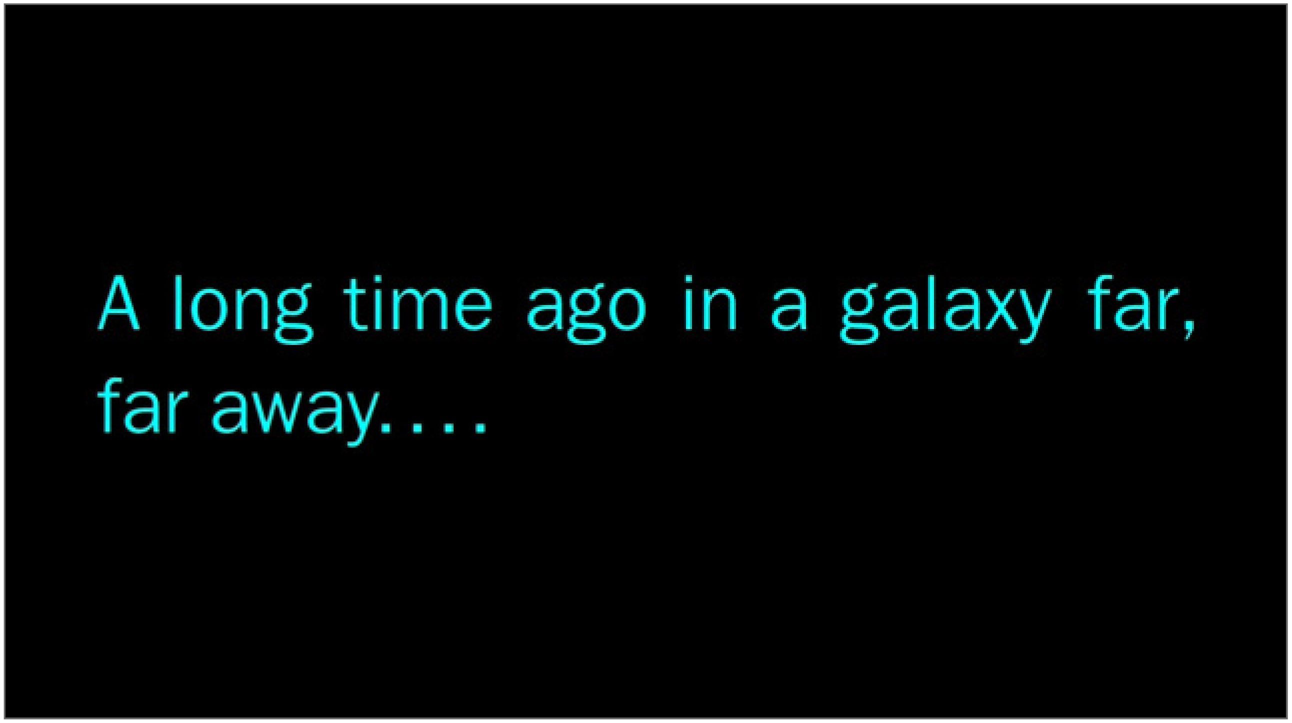
> *An HTTP/1.1 session is a sequence of network request-response transactions.*

Better

HTTP/2

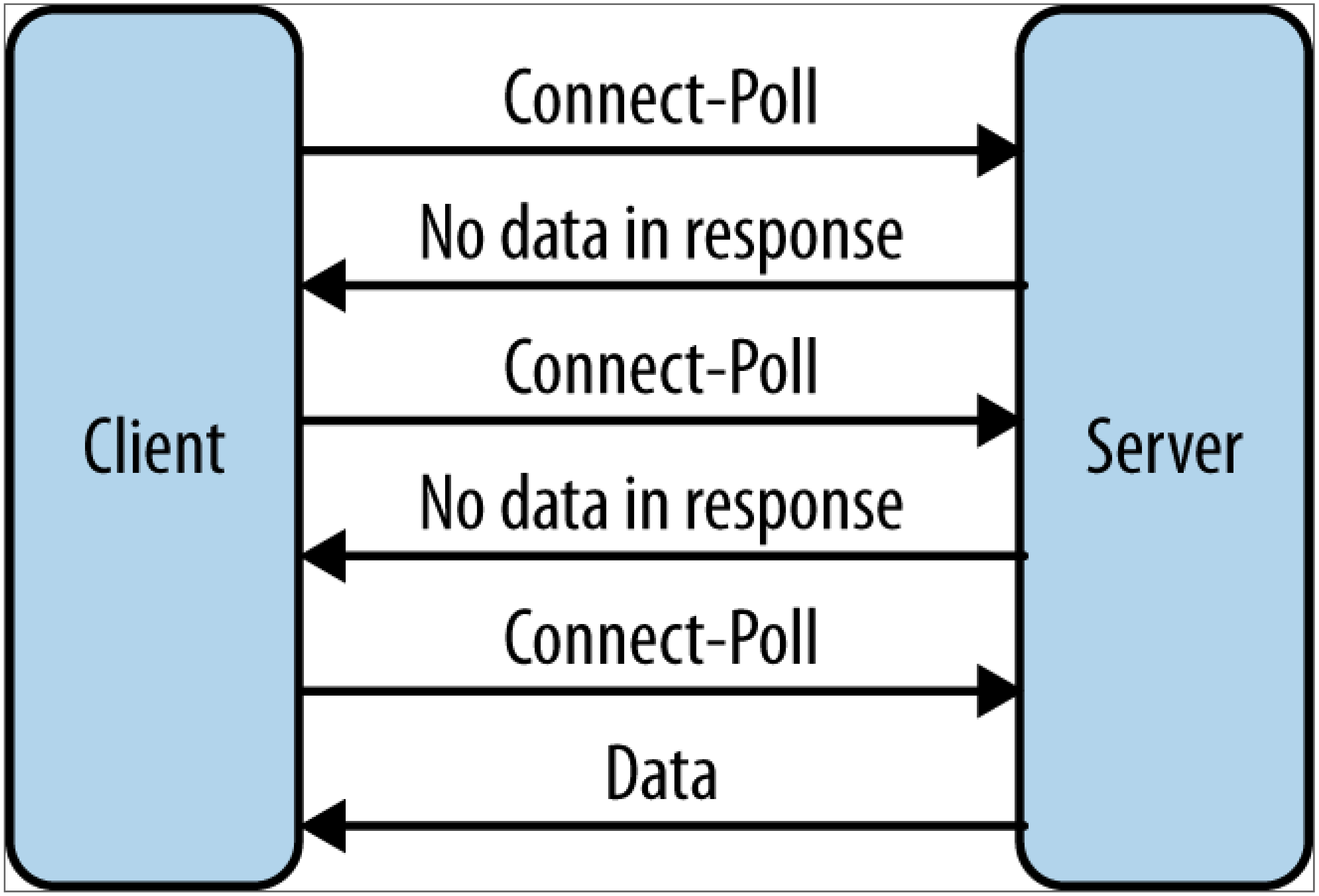
- ...
- Decrease latency to improve page load speed in web browsers by considering:
  1. Data compression of HTTP headers
  2. Server push technologies
  3. Fixing the head-of-line blocking problem in HTTP 1
  4. Loading page elements in parallel over a single TCP connection
- ...

HTTP/2

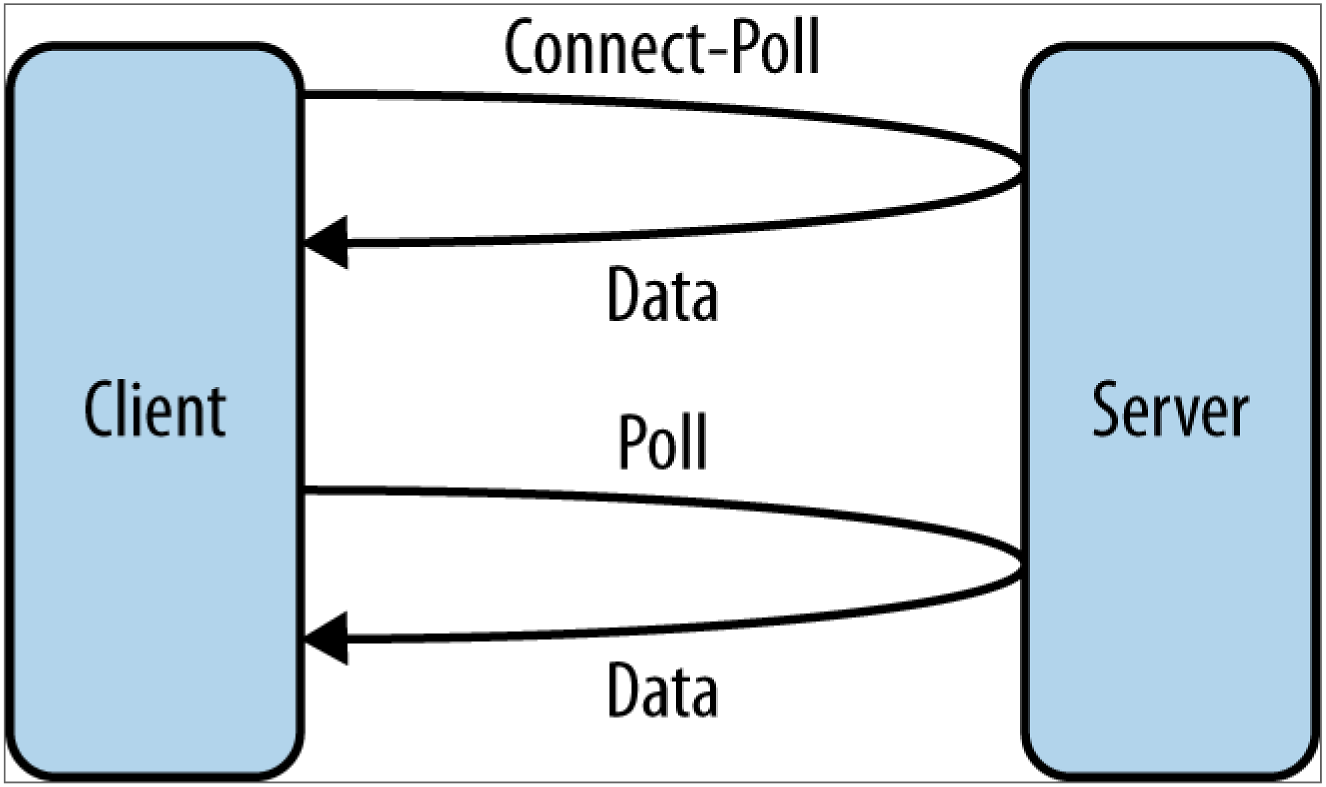


A long time ago in a galaxy far, far away....

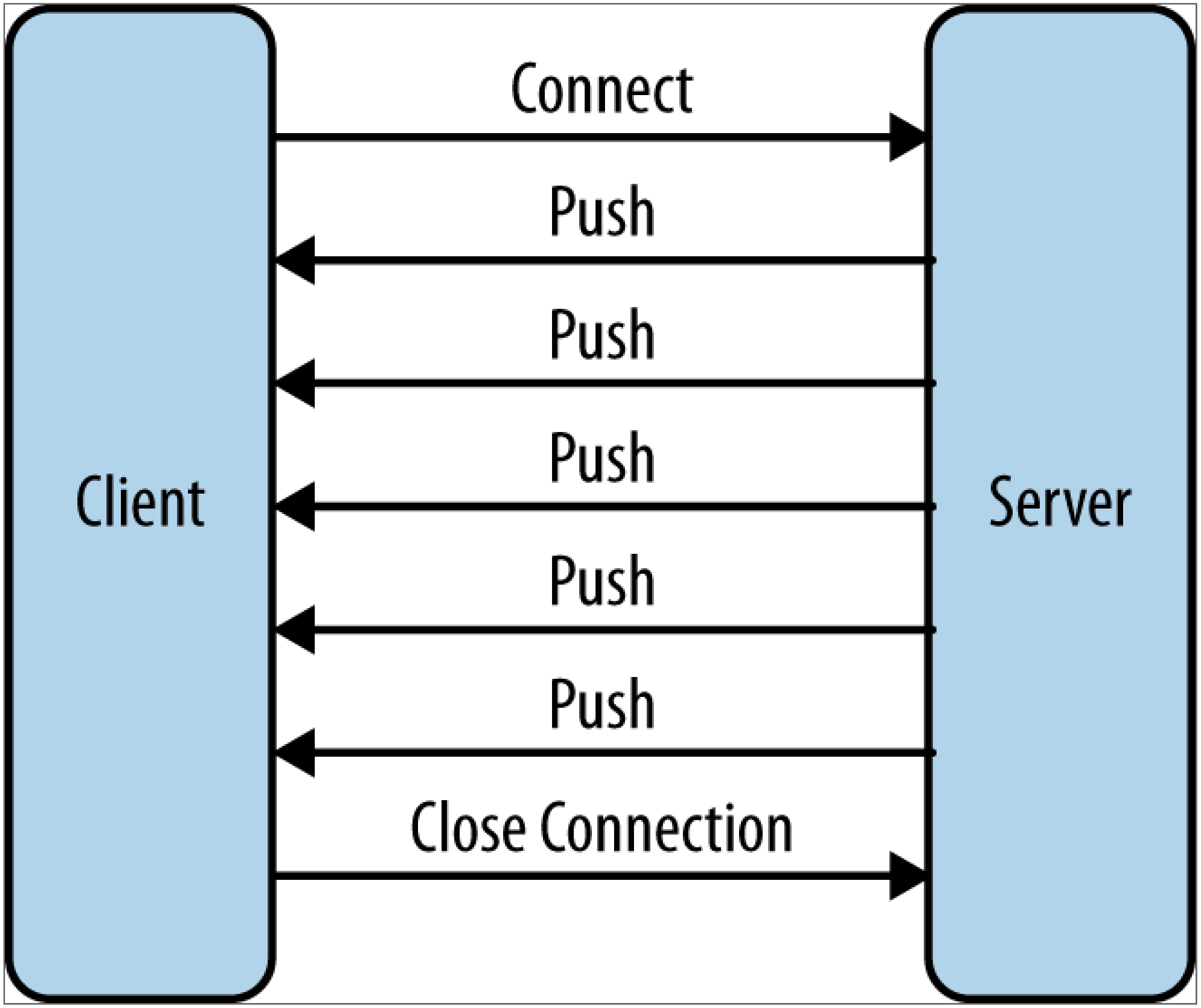
POLLING


Client
Connect-Poll
No data in response
Connect-Poll
No data in response
Connect-Poll
Data
Server

LONG POLLING



HTTP STREAMING AKA COMET



PROBLEMS

- Low Latency Client-Server and Server-Client Connections
- Waste

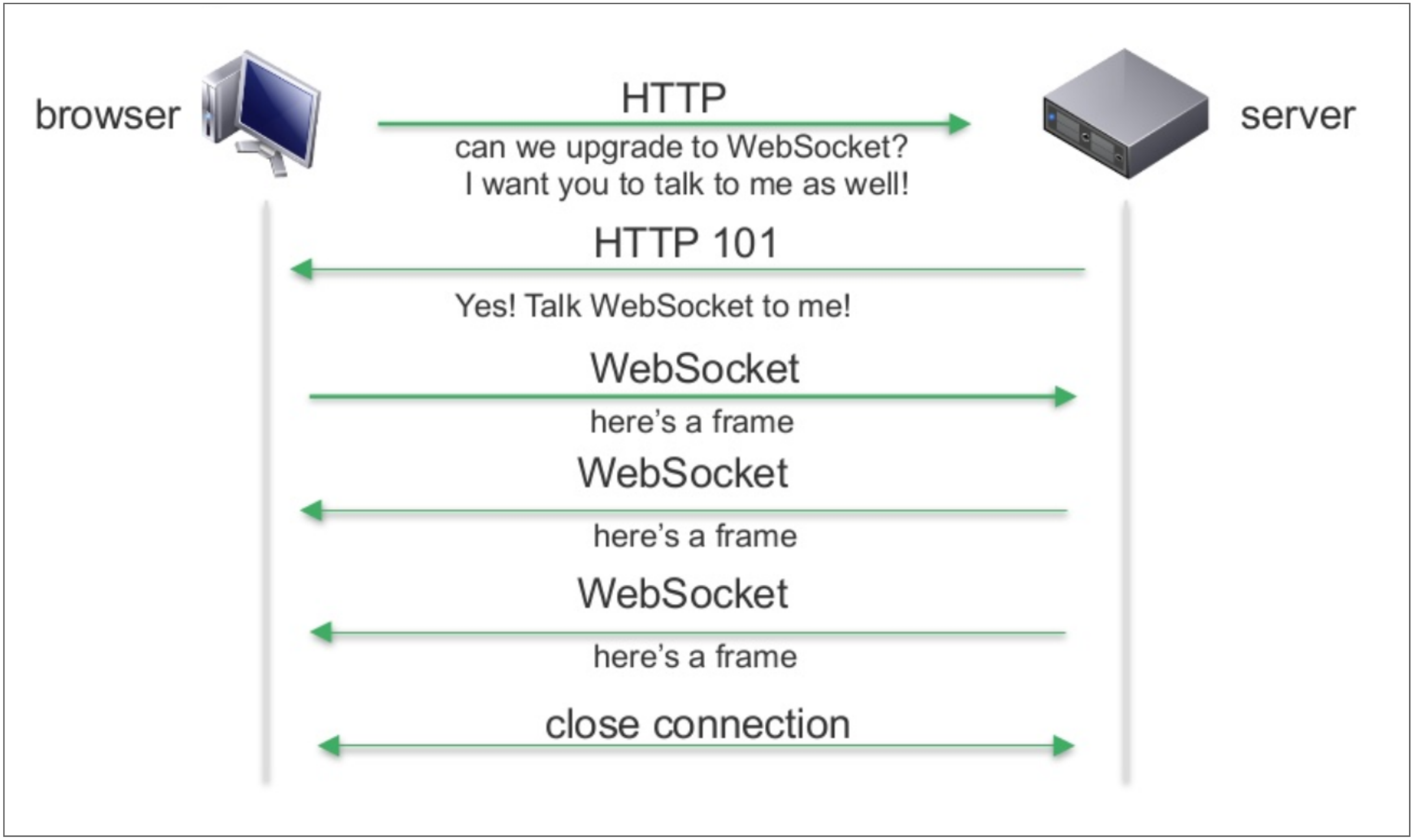
WEBSOCKET

A layer on TCP

Full-duplex, stateful connection

Stream of messages (rather than bytes)

HTTP used for the initial handshake


browser
server
HTTP
can we upgrade to WebSocket?
I want you to talk to me as well!
HTTP 101
Yes! Talk WebSocket to me!
WebSocket
here’s a frame
WebSocket
here’s a frame
WebSocket
here’s a frame
close connection

HANDSHAKE

```
HANDSHAKE REQUEST

GET /mychat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: x3JJHMbDL1EzLkh9GBhXDw==
Sec-WebSocket-Protocol: chat
Sec-WebSocket-Version: 13
Origin: http://example.com
```

HANDSHAKE RESPONCE

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: HSmrc0sMlYUkAGmm5OPpG2HaGWk=
Sec-WebSocket-Protocol: chat
```

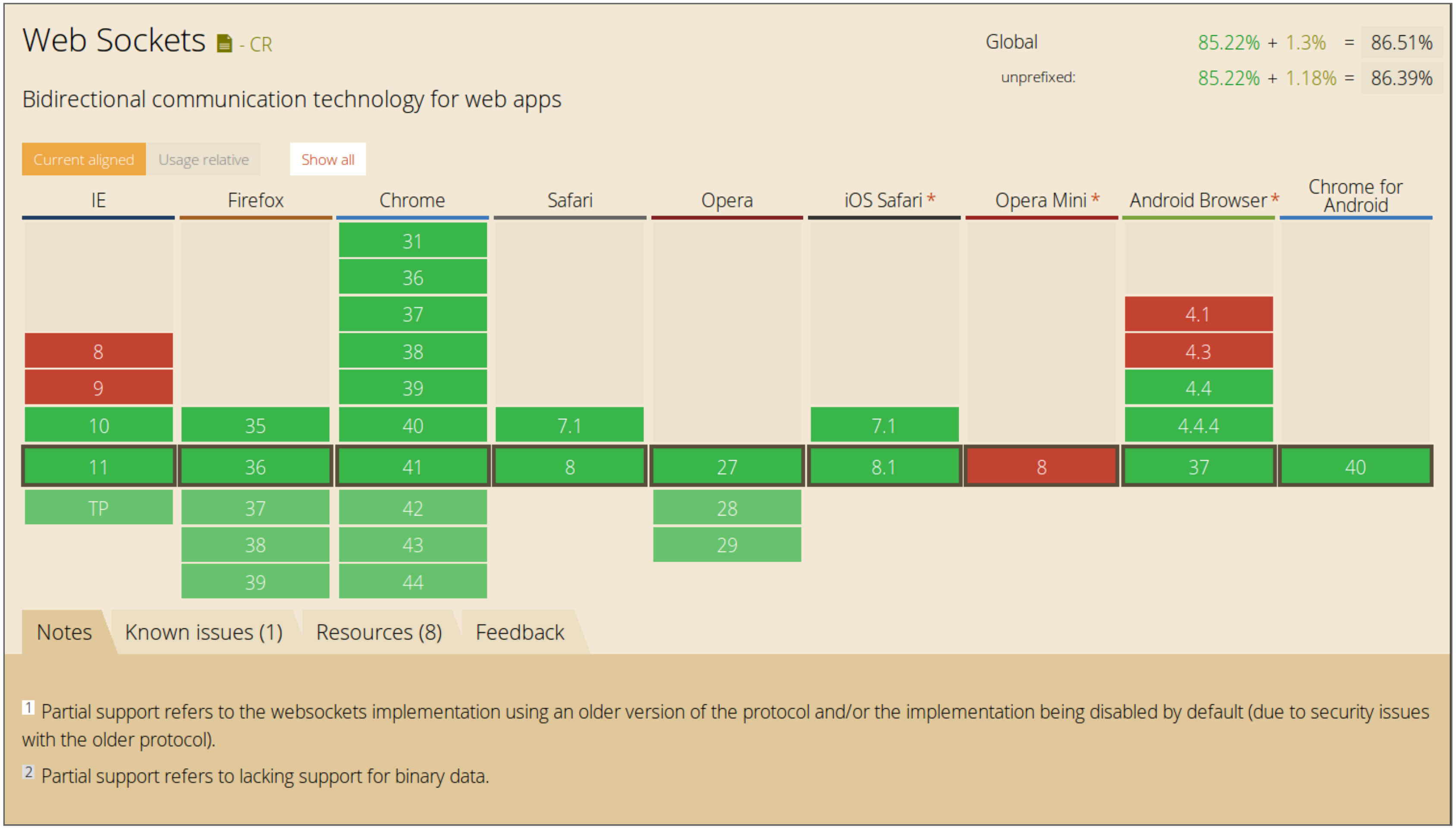
КТО УЖЕ ИСПОЛЬЗОВАЛ WEBSOCKETS?

USE CASES

- Multiplayer online games
- Chat applications
- Push notifications
- Realtime updating social streams
- ...

С КАКОЙ ВЕРСИИ ПОДДЕРЖИВАЮТСЯ WEBSOCKETS В IE?

CAN I USE?



Web Sockets - CR

Bidirectional communication technology for web apps

Global	85.22% + 1.3%	=	86.51%
unprefixed:	85.22% + 1.18%	=	86.39%

IE	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
		31						
		36						
		37					4.1	
8		38					4.3	
9		39					4.4	
10	35	40	7.1		7.1		4.4.4	
11	36	41	8	27	8.1	8	37	40
TP	37	42		28				
	38	43		29				
	39	44						

[1] Partial support refers to the websockets implementation using an older version of the protocol and/or the implementation being disabled by default (due to security issues with the older protocol).

[2] Partial support refers to lacking support for binary data.

caniuse.com/#feat=websockets

В КАКОМ ГОДУ IETF СТАНДАРТИЗИРОВАЛА WEBSOCKET ПРОТОКОЛ КАК RFC 6455?

DEC-2011

RFC 6455

В КАКОМ ГОДУ БЫЛ УТВЕРЖДЕН JSR 356?

MAY-2013

JSR 356

Part of JavaEE 7

Not using Servlet API

JSR 356

- Tomcat 8 + backport to Tomcat 7.0.47
- Jetty 9.1
- Glassfish 4 with Tyrus WebSocket engine
- WildFly 8
- Any Servlet 3.1 container
- Any JavaEE 7 compatible appserver

EXAMPLE

```
@ServerEndpoint("/ws")
public class Getter {

    @OnMessage
    public void getValue(String data, Session client) {
        String returnText = getTimeStamp();
        client.getAsyncRemote().sendText(returnText);
    }
}
```

JSR 356

No fallback

No sub-protocol support

Too low level

Single WebSocket connection per client

results in single @ServerEndpoint per application

КТО ПИШЕТ ПРИЛОЖЕНИЯ НА ГОЛЫХ СЕРВЛЕТАХ?

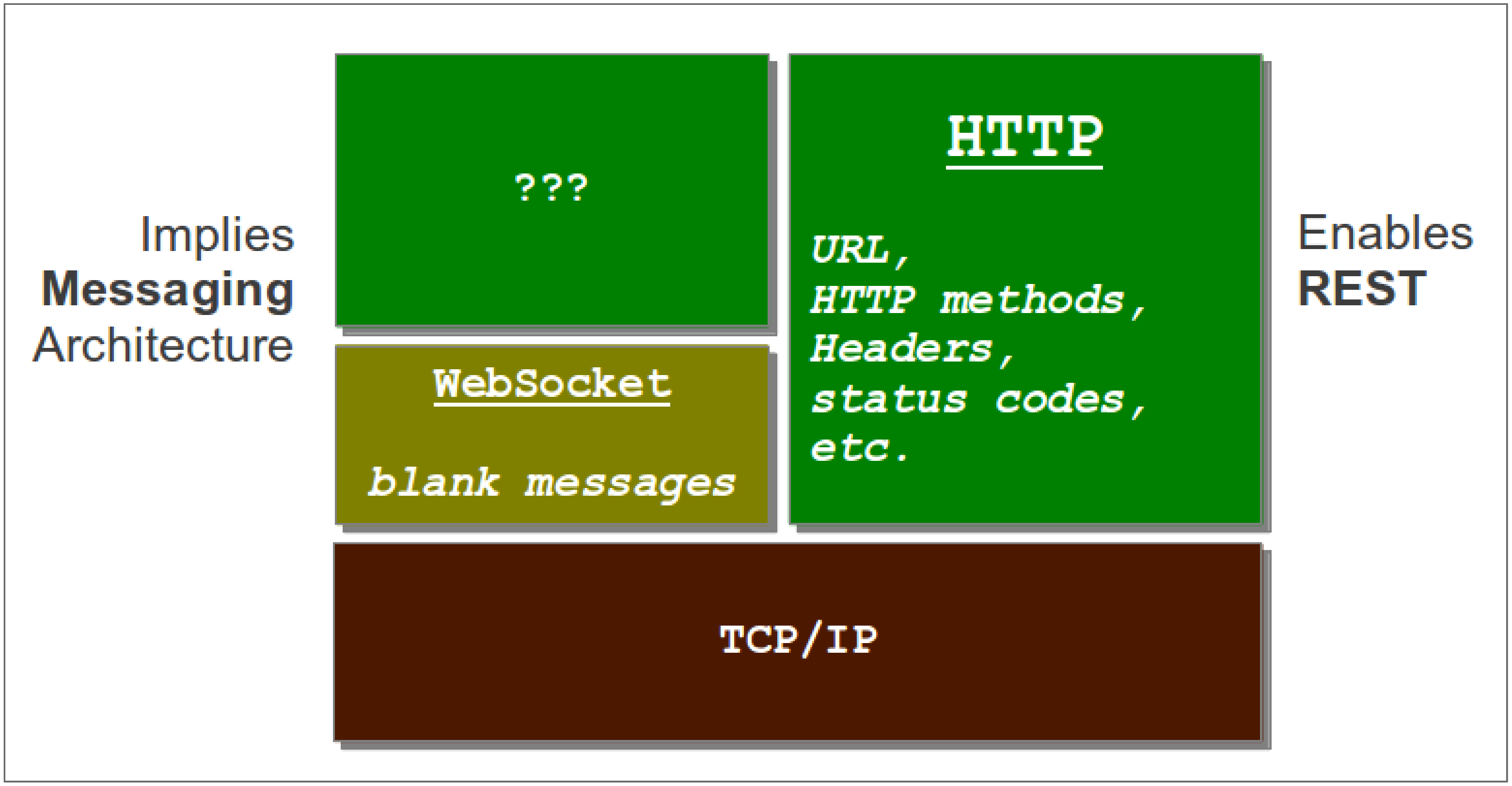
JAVASCRIPT WEBSOCKET API

```
var socket = new WebSocket('ws://itx.by:12010/updates');

socket.onopen = function () {
  setInterval(function() {
    if (socket.bufferedAmount == 0)
      socket.send(getUpdateData());
  }, 50);
};

socket.onerror = function (error) {
  console.log('WebSocket Error ' + error);
};

socket.onmessage = function (message) {
  console.log('Server: ' + message.data);
};

socket.onclose = function() {
  console.log('Closed.');
};
```

W3C: The WebSocket API and **W3C: Server-Sent Events** | **MDN: WebSocket**

- Atmosphere **[1] [2]**
- **netty-socketio**
- **jWebSocket**
- **Kaazing**
- **Meteor**
- **CometD**
- **other servers**
- ...

КТО ИСПОЛЬЗУЕТ SPRING-CONTEXT НА ТЕКУЩЕМ ПРОЕКТЕ?



Implies **Messaging** Architecture

???

WebSocket

blank messages

HTTP

URL, HTTP methods, Headers, status codes, etc.

Enables **REST**

TCP/IP

STOMP

Simple protocol for asynchronous message passing

Originally for scripting languages (Ruby, Python)

Supported by message brokers

Suited for use on the web

STOMP FRAME CONTENT

```
COMMAND
header1:value1
header2:value2

body^@
```

COMMANDS

```
SEND ===>>>
SUBSCRIBE, UNSUBCRIBE ===>>>
MESSAGE <<<===
ERROR <<<===
RECEIPT <<<===
ACK, NACK ===>>>
```

THE "DESTINATION" HEADER

A key concept in STOMP

Opaque string, syntax left to server

Typically URI path-like ("/queue/a", "/topic/a")

USAGE

Produce messages: via SEND frame with "destination" header

Consume messages: SUBSCRIBE frame w/ "destination" + MESSAGE frames from server

Server cannot send unsolicited messages!

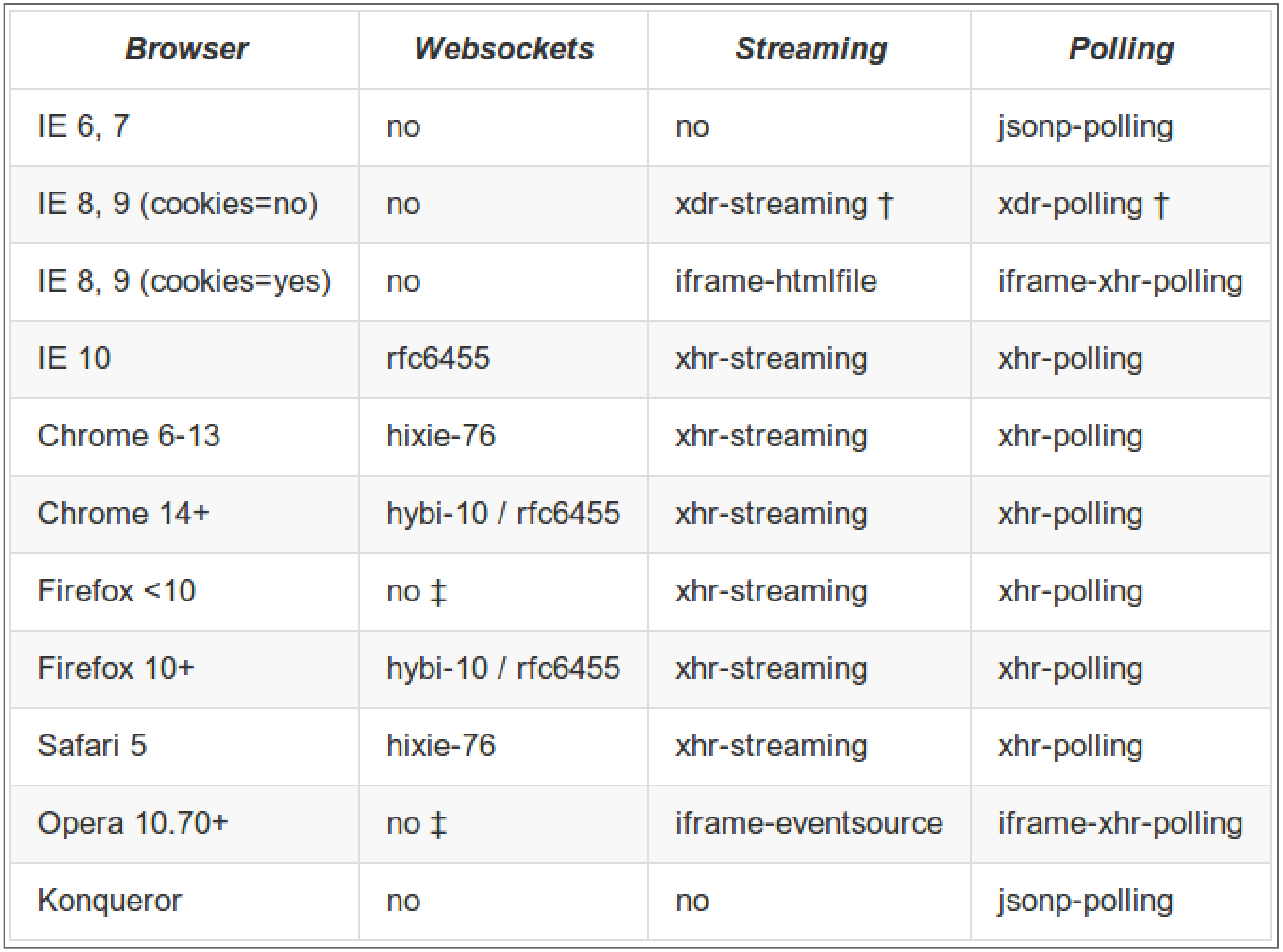
EXAMPLE

```
SEND
destination:/topic/trade
content-type:application/json
content-lenght:46

{"action":"Buy","ticker":"EMC","shares":"44"}^@
```

CONFIGURATION

```
@Configuration
@EnableWebSocketMessageBroker
public class Config implements WebSocketMessageBrokerConfigurer {

  @Override
  public void registerStompEndpoints(StompEndpointRegistry r) {
    r.addEndpoint("/ws").withSockJS(); // WebSocket URL prefix
  }

  @Override
  public void configureMessageBroker(MessageBrokerConfigurer c) {
    c.enableSimpleBroker("/topic/"); // destination prefix
  }
}
```

SOCKJS TRANSPORTS BY BROWSER



Browser	*Websockets*	*Streaming*	*Polling*
IE 6, 7	no	no	jsonp-polling
IE 8, 9 (cookies=no)	no	xdr-streaming †	xdr-polling †
IE 8, 9 (cookies=yes)	no	iframe-htmlfile	iframe-xhr-polling
IE 10	rfc6455	xhr-streaming	xhr-polling
Chrome 6-13	hixie-76	xhr-streaming	xhr-polling
Chrome 14+	hybi-10 / rfc6455	xhr-streaming	xhr-polling
Firefox <10	no ‡	xhr-streaming	xhr-polling
Firefox 10+	hybi-10 / rfc6455	xhr-streaming	xhr-polling
Safari 5	hixie-76	xhr-streaming	xhr-polling
Opera 10.70+	no ‡	iframe-eventsource	iframe-xhr-polling
Konqueror	no	no	jsonp-polling

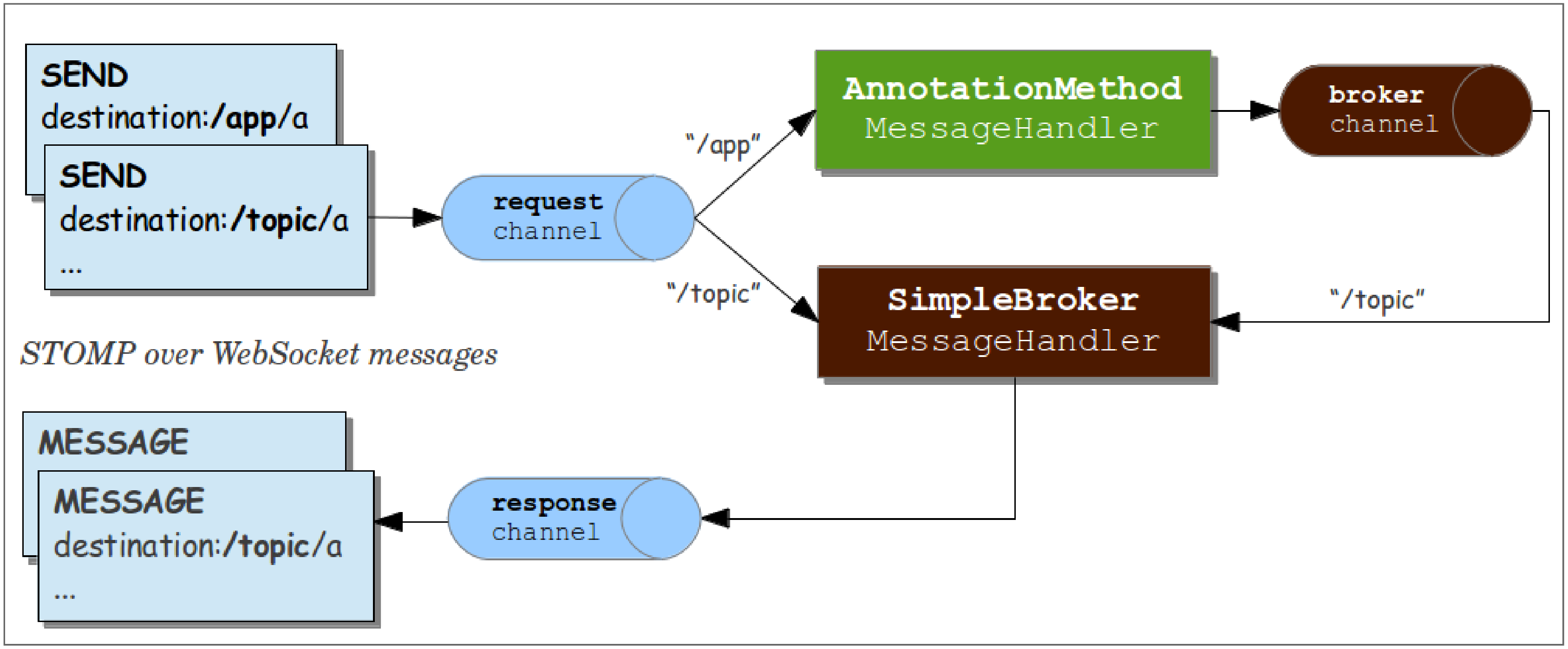
RECIEVE MESSAGE

```
@Controller
public class TradeController {


  @MessageMapping("/trade")
  @SendTo("/topic/trade")
  public String trade(Trade trade) {

      // Return value broadcast to "/topic/trade"

      return "[" + getTimestamp() + "]: ";
  }


}
```

SEND MESSAGE TO USERS

```
@RestController
public class TradeController {

  @Autowired
  private SimpMessagingTemplate template;

  @RequestMapping(value="/trade", method=POST)
  public void greet(Trade trade) {
    this.template.convertAndSend("/topic/trade", trade);
  }

}
```

MESSAGE FLOW



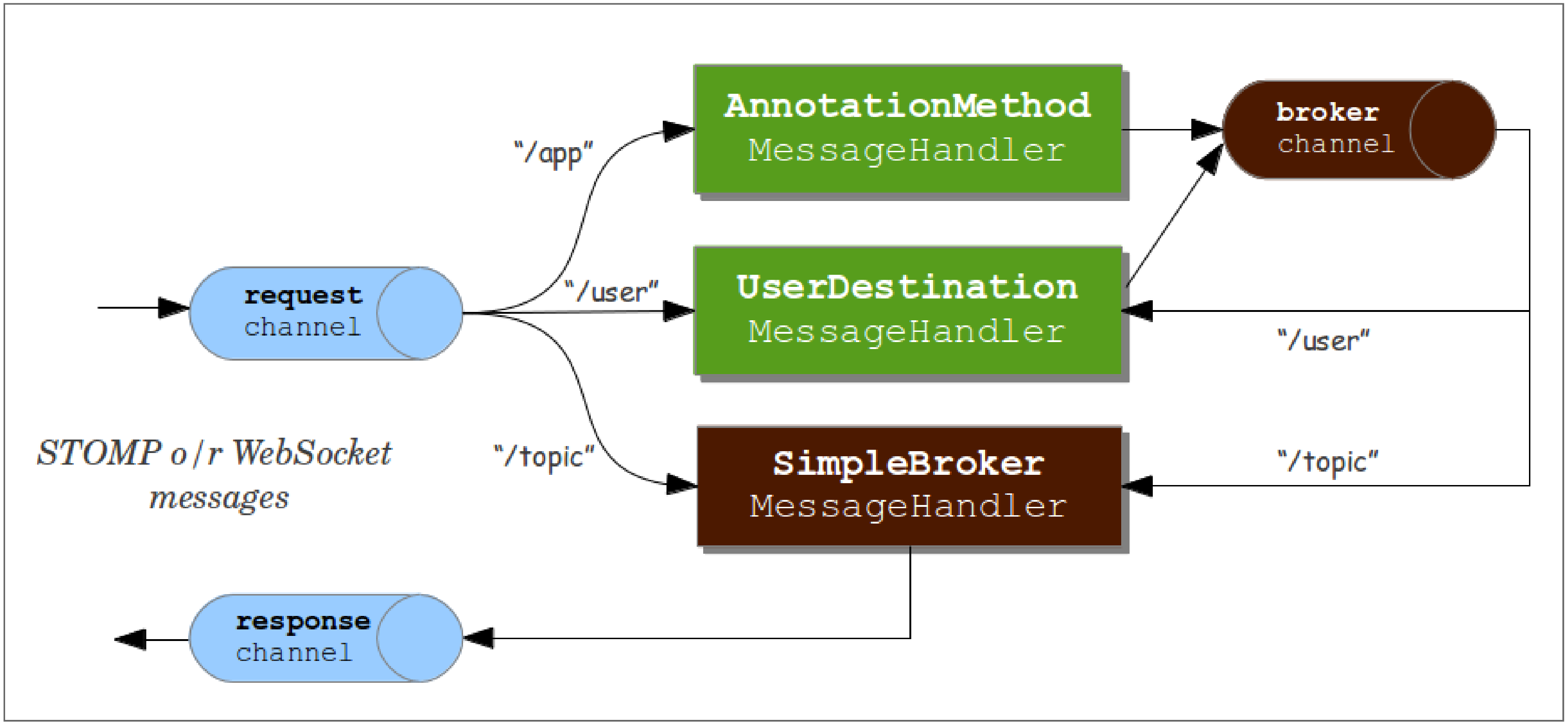
SEND MESSAGE TO USER

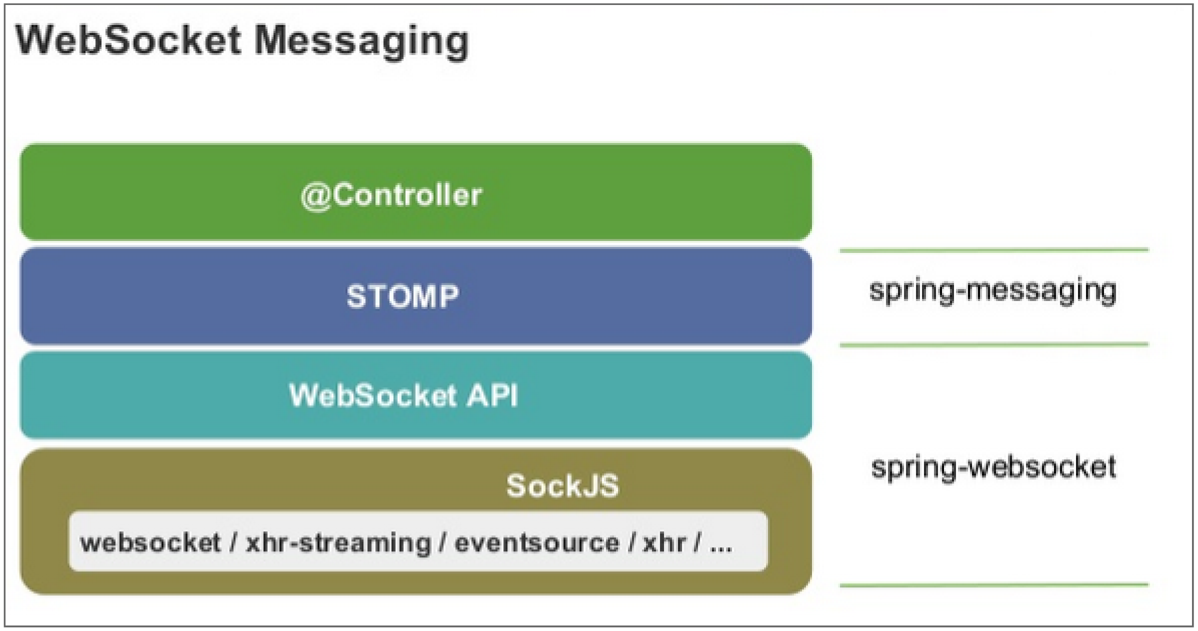
```
@RestController
public class TradeController {

  // ...

  @MessageExceptionHandler
  @SendToUser("/queue/errors")
  public String handleException(IllegalStateException ex) {
    return ex.getMessage();
  }

}
```

CLIENT SIDE

```
var socket = new SockJS('/ws');
var client = Stomp.over(socket);

client.connect('', '', function(frame) {

  var user = frame.headers['user-name'];
  var suffix = frame.headers['queue-suffix'];

  client.subscribe("/queue/trade" + suffix, function(msg) {
    // ...
  });
  client.subscribe("/queue/errors" + suffix, function(msg) {
    // ...
  });

}
```

MESSAGE FLOW


AnnotationMethod MessageHandler
broker channel
"/app"
UserDestination MessageHandler
request channel
"/user"
"/user"
STOMP o/r WebSocket messages
"/topic"
SimpleBroker MessageHandler
"/topic"
response channel



WebSocket Messaging

@Controller

STOMP

WebSocket API

SockJS

websocket / xhr-streaming / eventsource / xhr / ...

spring-messaging

spring-websocket

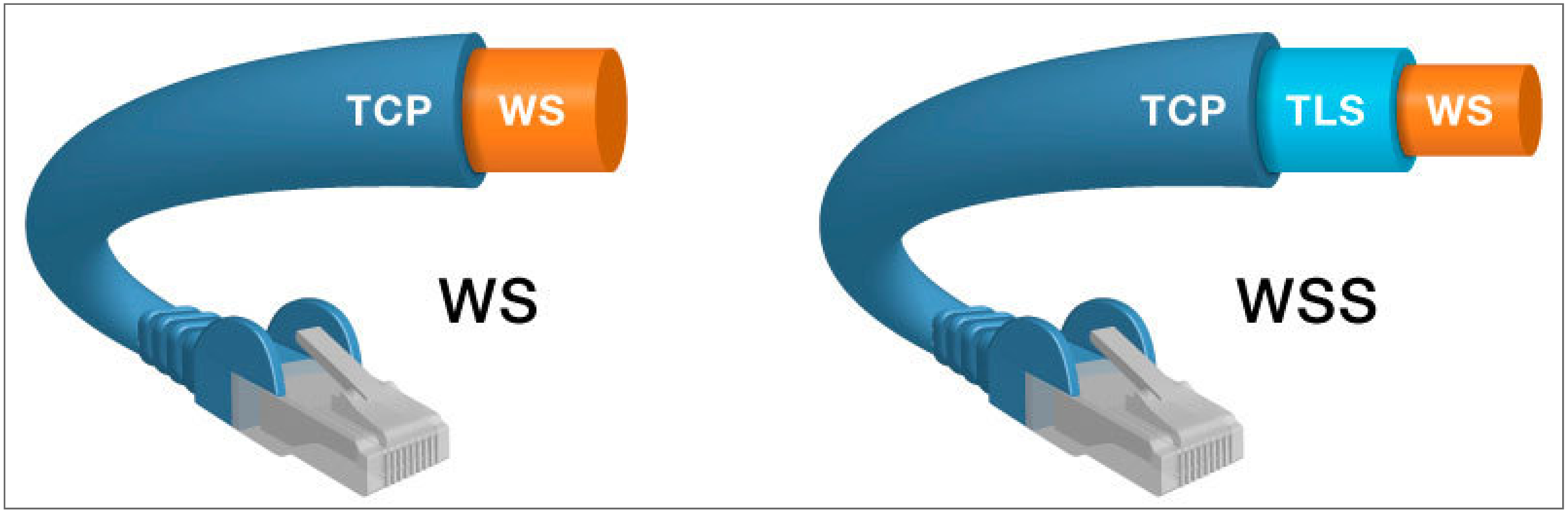
SPRING SECURITY 4.0 (REALESE SOON) AND WEBSOCKET

```
@Configuation
public class WebSocketSecurityConfig extends
          AbstractSecurityWebSocketMessageBrokerConfigurer {
    protected void configureInbound(MessageSecurityMetadataSourceRegistry messages) {
      messages
        . simpDestMatchers("/user/queue/errors").permitAll()
        . simpDestMatchers("/**").hasRole("ADMIN");
      }
  }
```

SPRING SECURITY 4.0 (REALESE SOON) AND WEBSOCKET

```
// or "hasRole('ROLE_ADMIN')"
@PreAuthorize("hasRole('ADMIN')")
@MessageMapping("/update")
public void update(Update update, Principal principal) {
        // ...
}
```

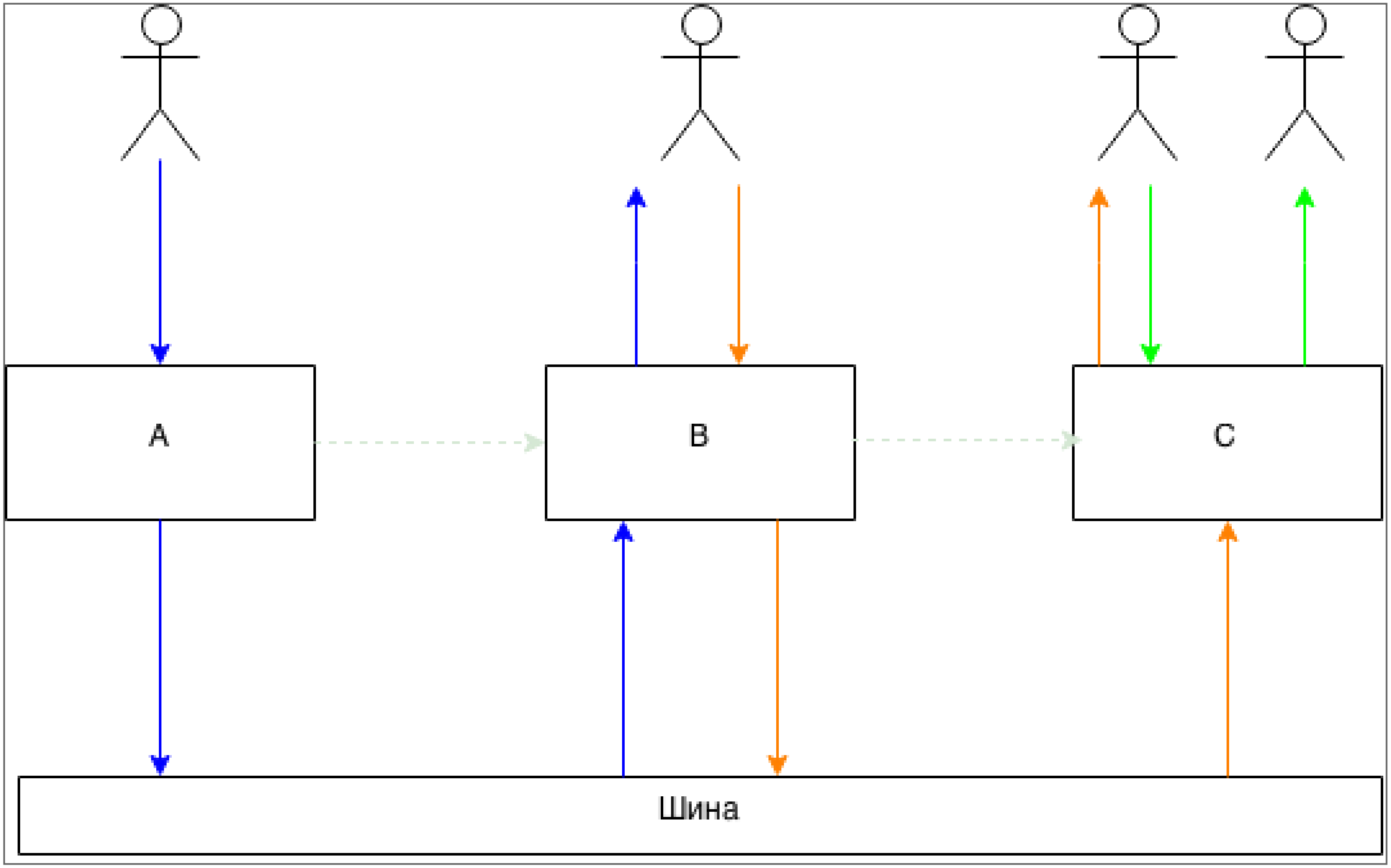
WS AND WSS



PERFORMANCE TESTING

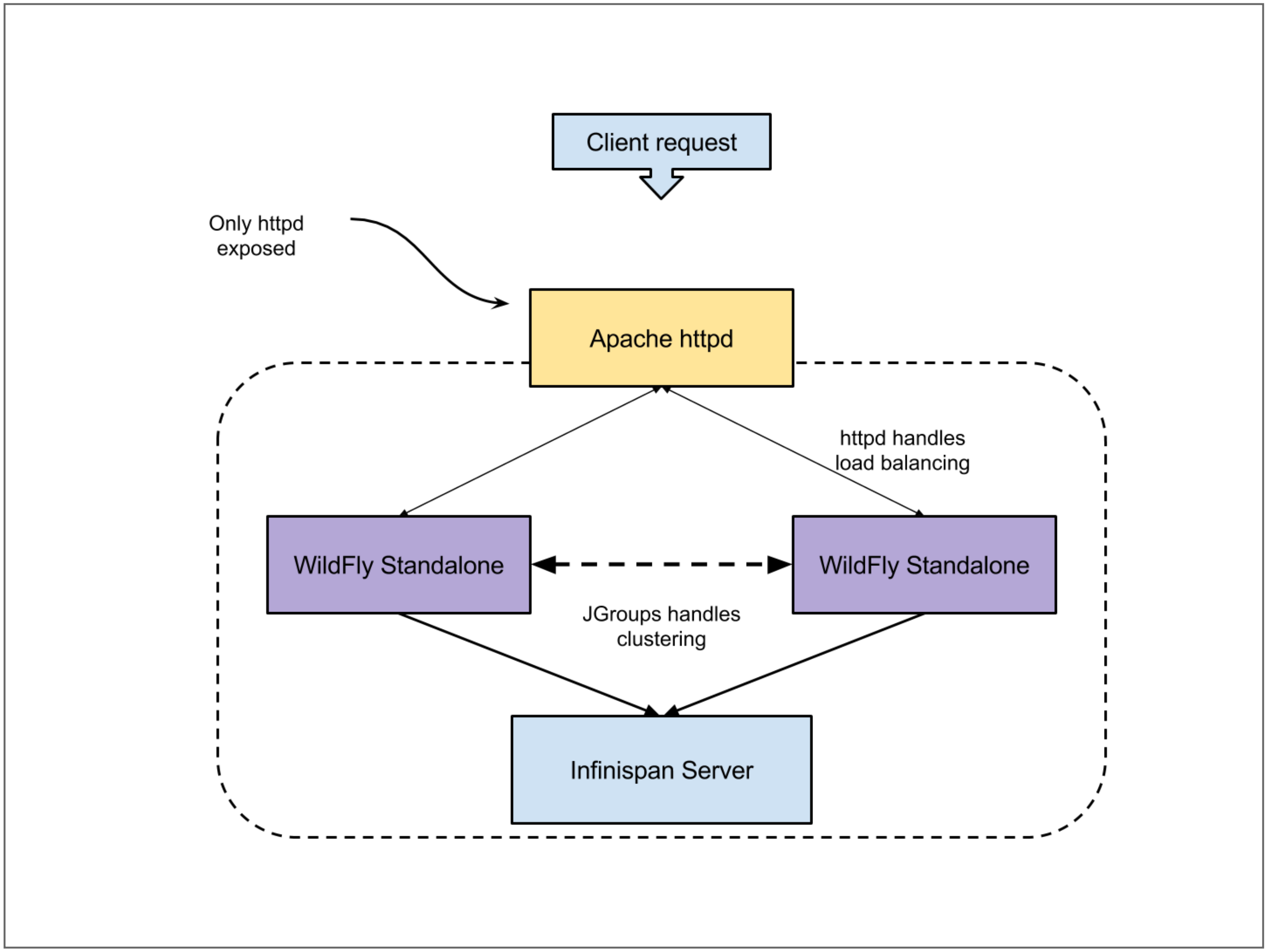
- **tcpkali**
- **hellsockets**

МАСШТАБИРОВАНИЕ



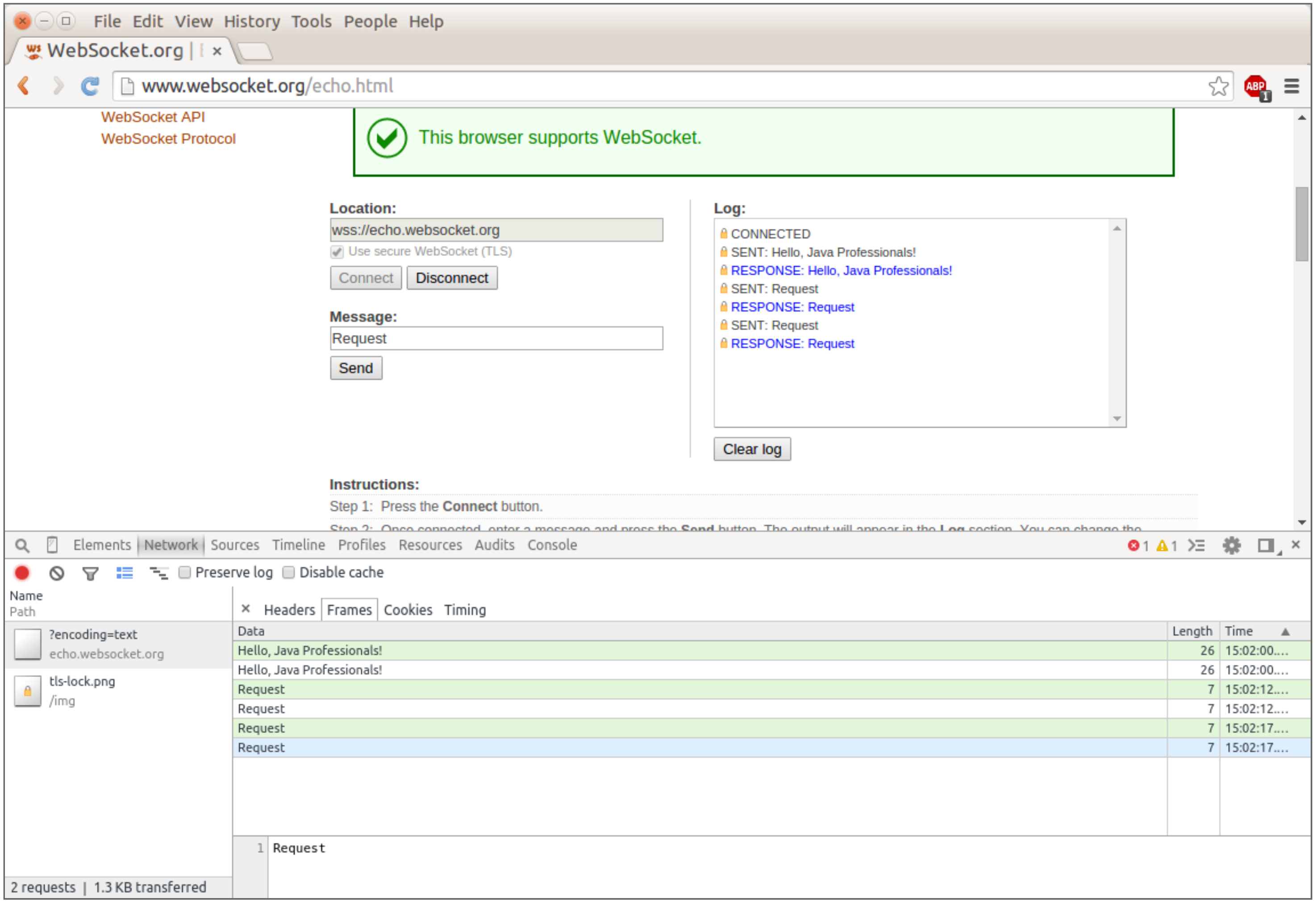
Горизонтальное масштабирование websocket-ов на Ruby

МАСШТАБИРОВАНИЕ



[1], [2]

ДЕБАЖИТЬ В CHROME?


File Edit View History Tools People Help
WebSocket.org
www.websocket.org/echo.html
WebSocket API
WebSocket Protocol
Connect
Disconnect
Send
Clear log
Elements Network Sources Timeline Profiles Resources Audits Console
1 1
□ Preserve log □ Disable cache
Name
Path
× Headers Frames Cookies Timing

This browser supports WebSocket.

Location:
wss://echo.websocket.org
☑ Use secure WebSocket (TLS)

Message:
Request

Log:

```
CONNECTED
SENT: Hello, Java Professionals!
RESPONSE: Hello, Java Professionals!
SENT: Request
RESPONSE: Request
SENT: Request
RESPONSE: Request
```

Instructions:
Step 1: Press the **Connect** button.
Step 2: Once connected, enter a message and press the **Send** button. The output will appear in the **Log** section. You can change the

?encoding=text
echo.websocket.org

tls-lock.png
/img

Data	Length	Time
Hello, Java Professionals!	26	15:02:00....
Hello, Java Professionals!	26	15:02:00....
Request	7	15:02:12....
Request	7	15:02:12....
Request	7	15:02:17....
Request	7	15:02:17....

```
Request
```

2 requests | 1.3 KB transferred

ДЕБАЖИТЬ В FIREFOX?

Bug 885508

FIREBUG?

Issue 6330

ibragimov.by / ruslan@ibragimov.by