ZBRA

SOLUTIONS

www.zbra.com.br

Stomp

Filipe Sbragio – filipe.sbragio@zbra.com.br
Márcio Hariki – marcio.hariki@zbra.com.br

Outline

- STOMP
- Why use STOMP?
- Communication
- Headers
- Connection
- Heartbeat
- Frames

STOMP

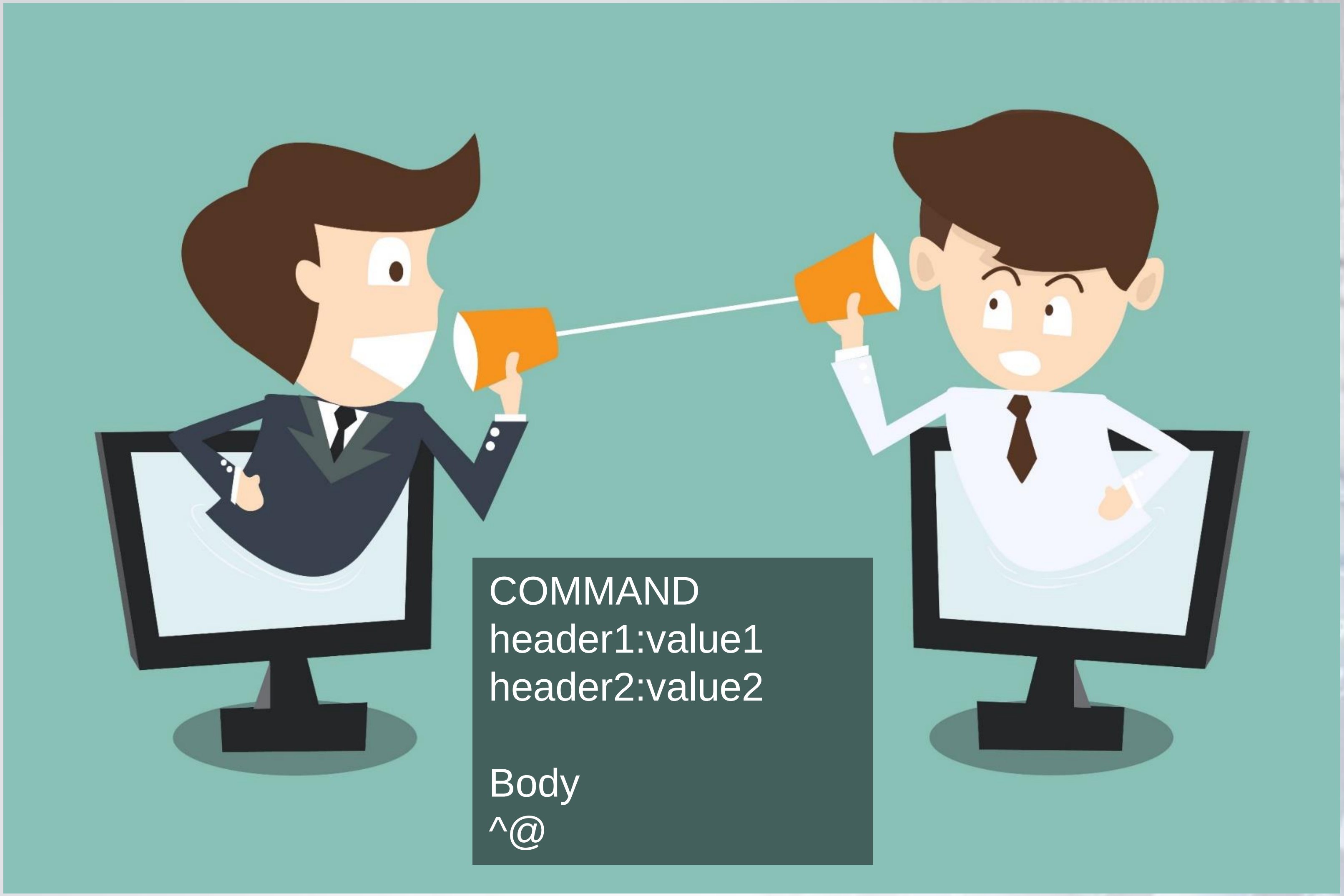
STOMP

*Mandatory impact image

Why use STOMP?

Communication

Communication

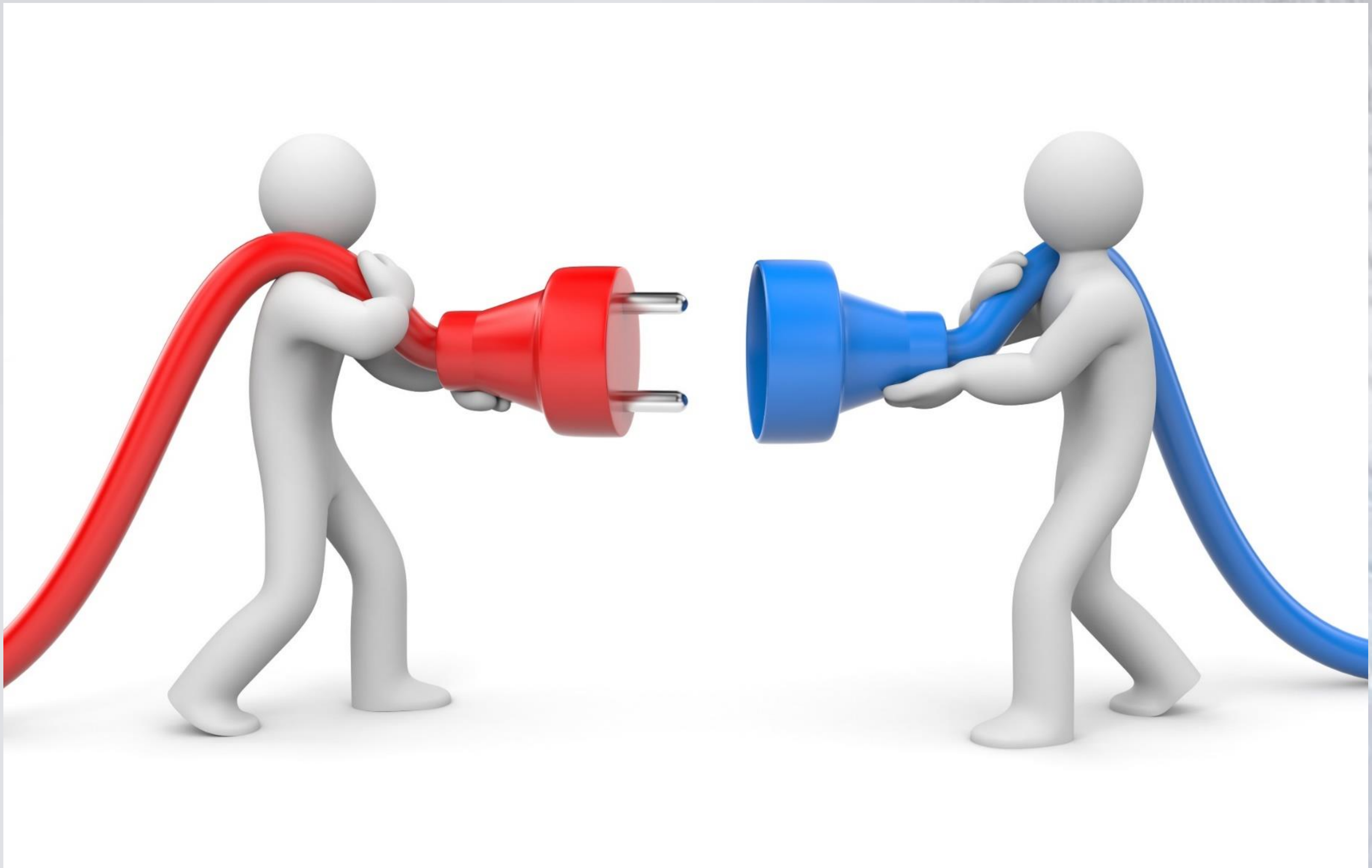


```
COMMAND
header1:value1
header2:value2

Body
^@
```

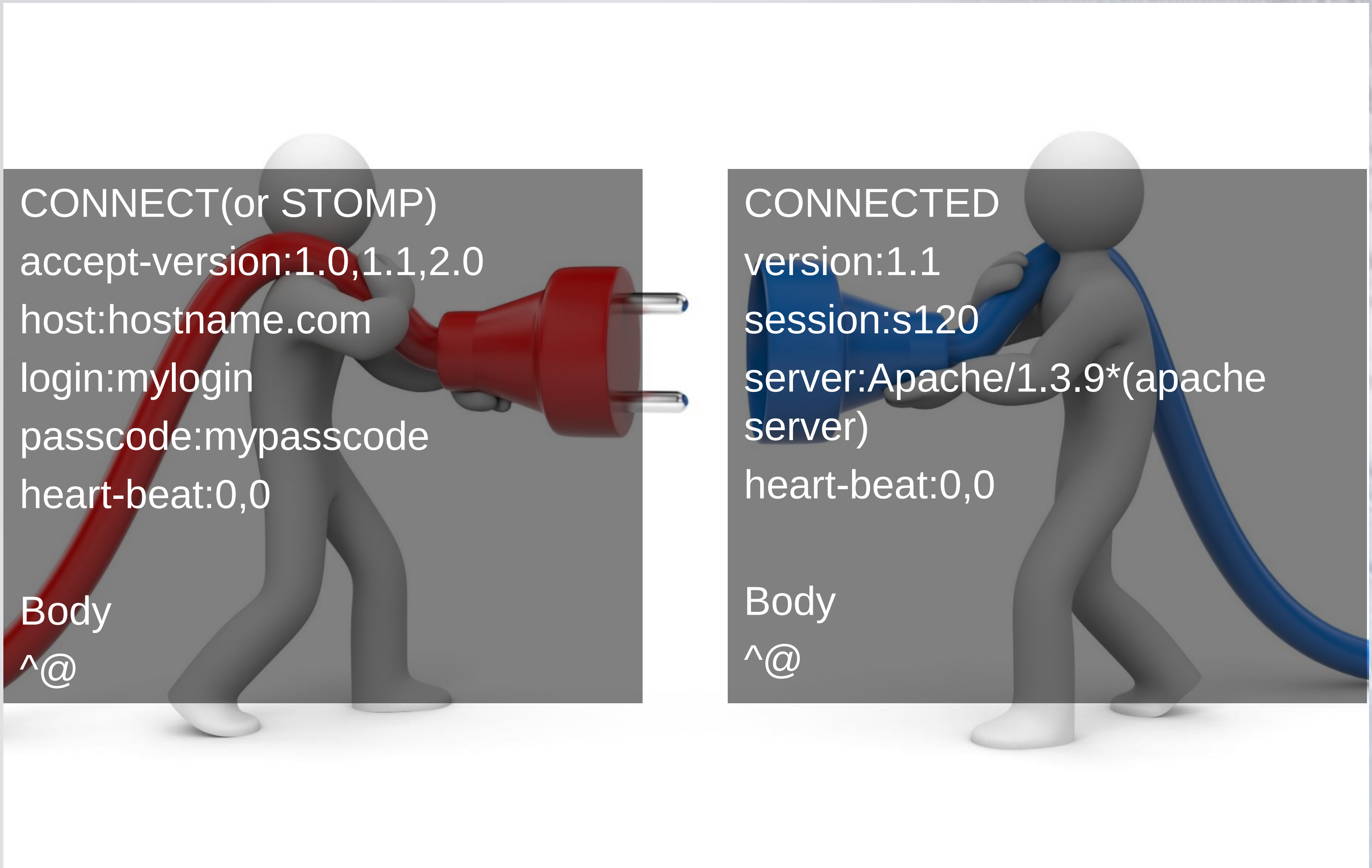
Headers

```
COMMAND
content-length:50
content-type:text/html;charset=utf-16
receipt:arbitraryString
receipt:y-u-repeat-header?

Body
^@
```

Connection



Connection

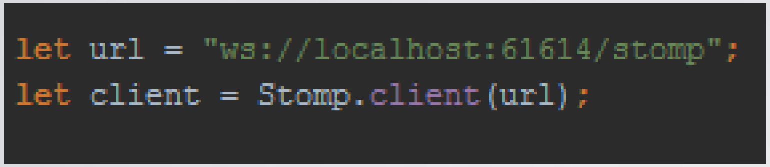


```
CONNECT(or STOMP)
accept-version:1.0,1.1,2.0
host:hostname.com
login:mylogin
passcode:mypasscode
heart-beat:0,0

Body
^@
```

```
CONNECTED
version:1.1
session:s120
server:Apache/1.3.9*(apache server)
heart-beat:0,0

Body
^@
```

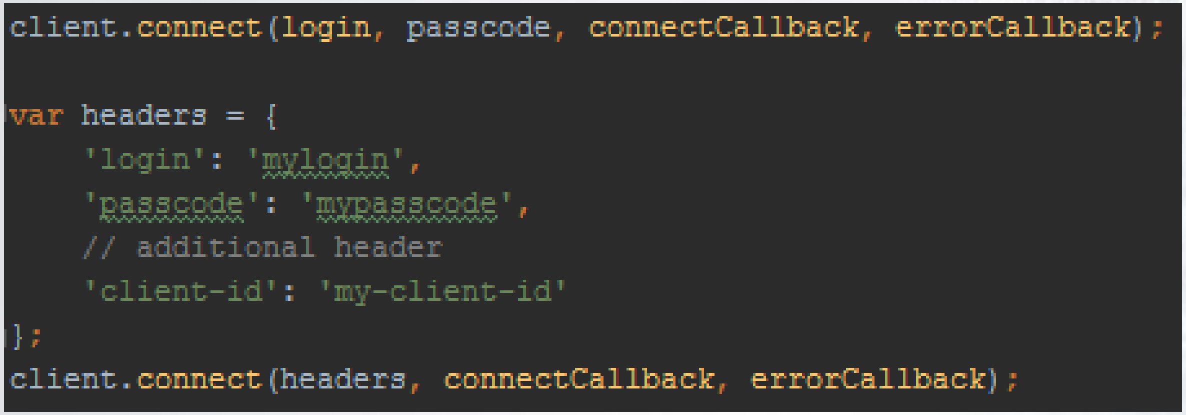
Connection (StompJS)

Stomp Client



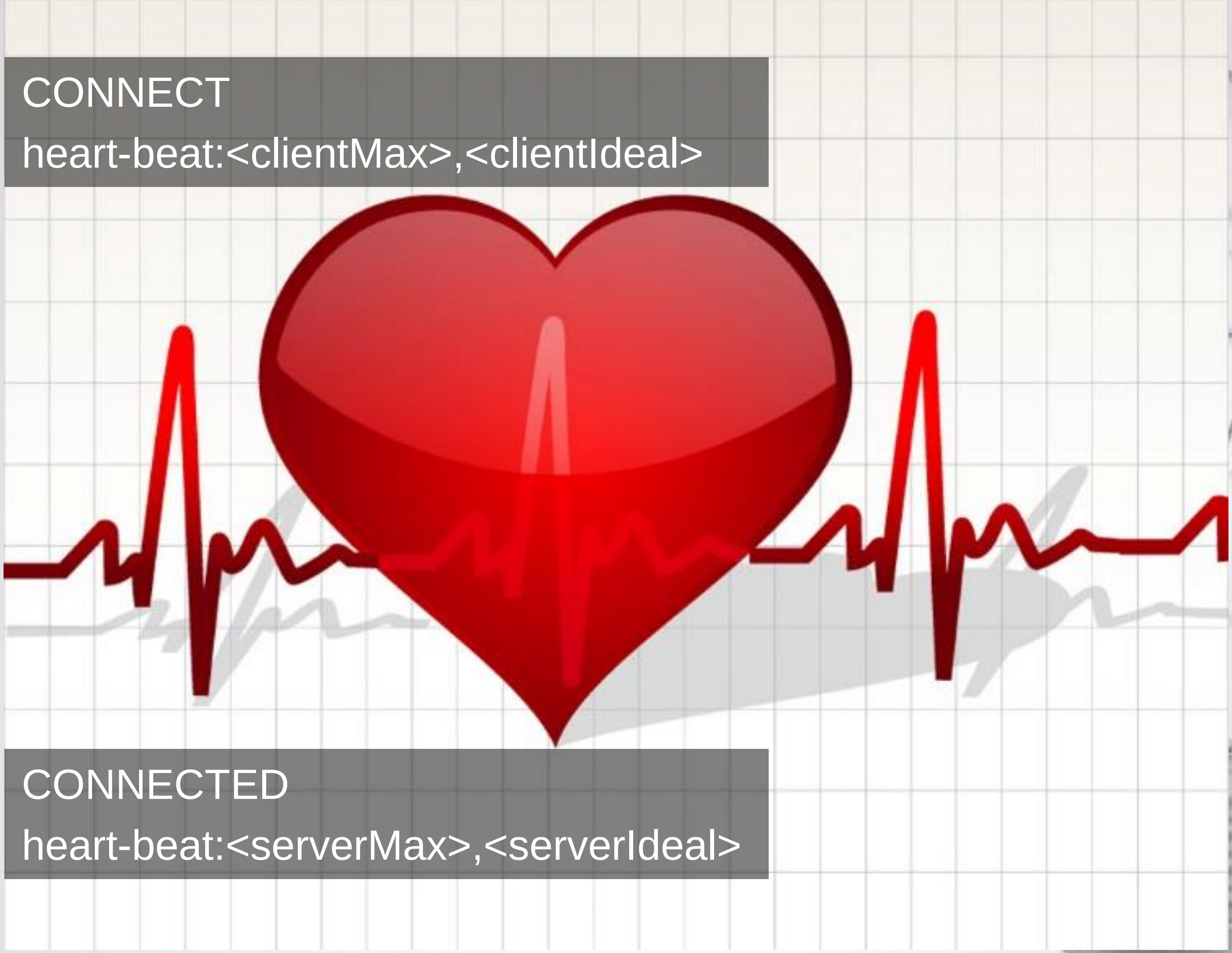
```
let url = "ws://localhost:61614/stomp";
let client = Stomp.client(url);
```

Connection



```
client.connect(login, passcode, connectCallback, errorCallback);

var headers = {
    'login': 'mylogin',
    'passcode': 'mypasscode',
    // additional header
    'client-id': 'my-client-id'
};
client.connect(headers, connectCallback, errorCallback);
```

Heartbeat



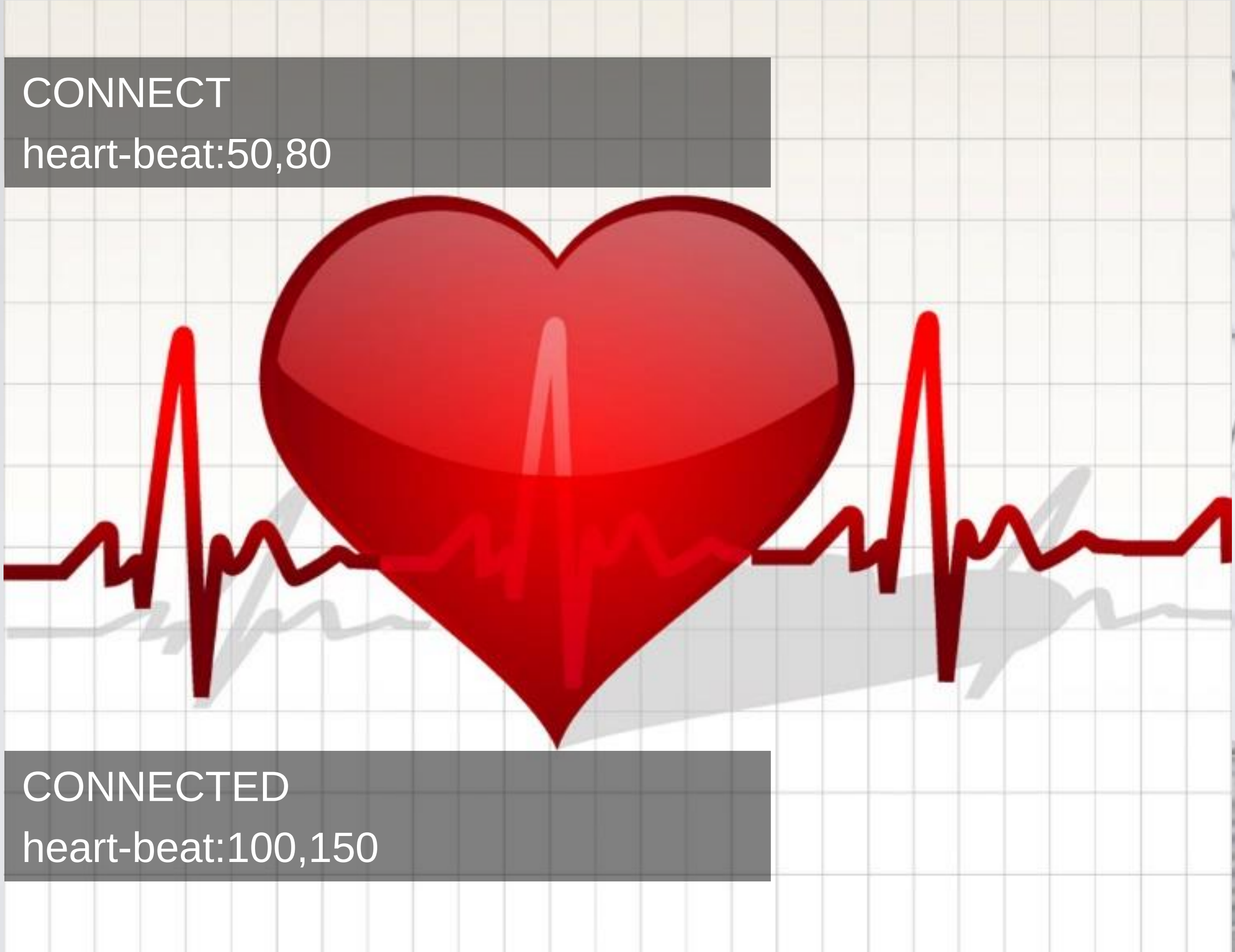
CONNECT

heart-beat:<clientMax>,<clientIdeal>

CONNECTED

heart-beat:<serverMax>,<serverIdeal>

Heartbeat



CONNECT

heart-beat:50,80

CONNECTED

heart-beat:100,150

Client Frames

```
SEND
destination:/zbra-chat/messages/1
content-length:50
content-type:application/json
irrelevant-header:isIrrelevant

{ user: { username: ‘zbra054', email: ‘zbra054@haha.com' }, text: ‘Hello! ' }
^@
```

Client Frames

```
SEND
destination:/zbra-chat/messages/1
content-length:50
content-type:application/json
irrelevant-header:isIrrelevant

{ user: { username: ‘zbra054', email: ‘zbra054@haha.com' }, text: ‘Hello! ' }
^@
```

```
SUBSCRIBE
Id:0
destination:/queue/messages/1
ack:client

^@
```

Client Frames

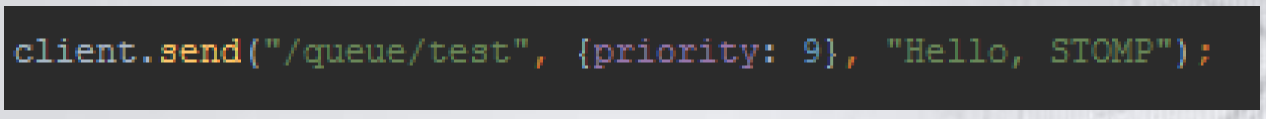
```
SEND
destination:/zbra-chat/messages/1
content-length:50
content-type:application/json
irrelevant-header:isIrrelevant

{ user: { username: ‘zbra054', email: ‘zbra054@haha.com' }, text: ‘Hello! ' }
^@
```

```
SUBSCRIBE
Id:0
destination:/queue/messages/1
ack:client

^@
```

```
UNSUBSCRIBE
id:0

^@
```

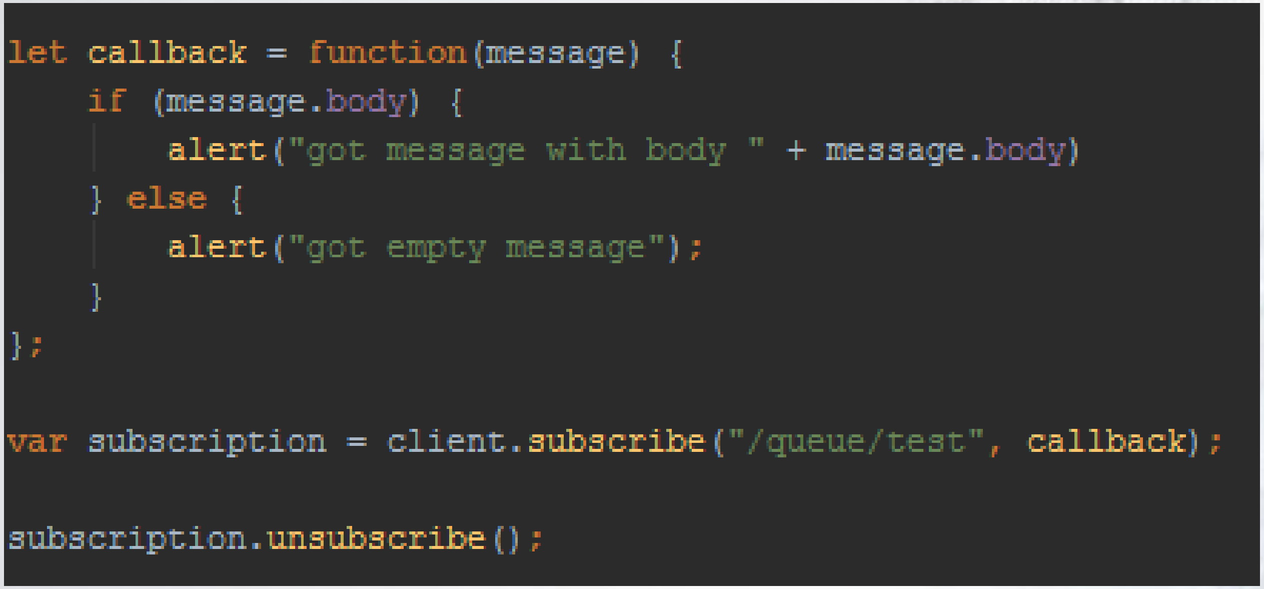
Client Frames

SEND



```
client.send("/queue/test", {priority: 9}, "Hello, STOMP");
```

SUBSCRIBE & UNSUBSCRIBE



```
let callback = function(message) {
    if (message.body) {
        alert("got message with body " + message.body)
    } else {
        alert("got empty message");
    }
};

var subscription = client.subscribe("/queue/test", callback);

subscription.unsubscribe();
```

Client Frames

```
ACK
id:12345
transaction:tx1


^@
```

Client Frames

```
ACK
id:12345
transaction:tx1

^@
```

```
NACK
id:12345
transaction:tx1

^@
```

Client Frames

```
ACK
id:12345
transaction:tx1

^@
```

```
NACK
id:12345
transaction:tx1

^@
```

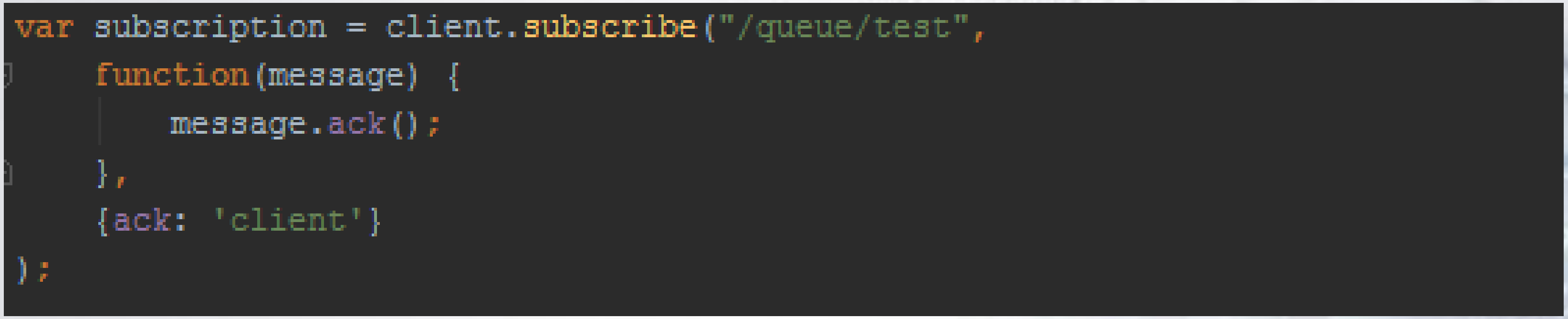
```
DISCONNECT
receipt:xd

^@
```

Client Frames

```
ACK
id:12345
transaction:tx1

^@
```

```
NACK
id:12345
transaction:tx1

^@
```

```
DISCONNECT
receipt:xd

^@
```



```
var subscription = client.subscribe("/queue/test",
    function(message) {
        message.ack();
    },
    {ack: 'client'}
);
```

Transaction Frames

```
BEGIN
transaction:tx1

^@
```

Transaction Frames

```
BEGIN
transaction:tx1

^@
```

```
COMMIT
transaction:tx1

^@
```

Transaction Frames

```
BEGIN
transaction:tx1

^@
```

```
COMMIT
transaction:tx1

^@
```

```
ABORT
transaction:tx1

^@
```

Transaction Frames

BEGIN
transaction:tx1

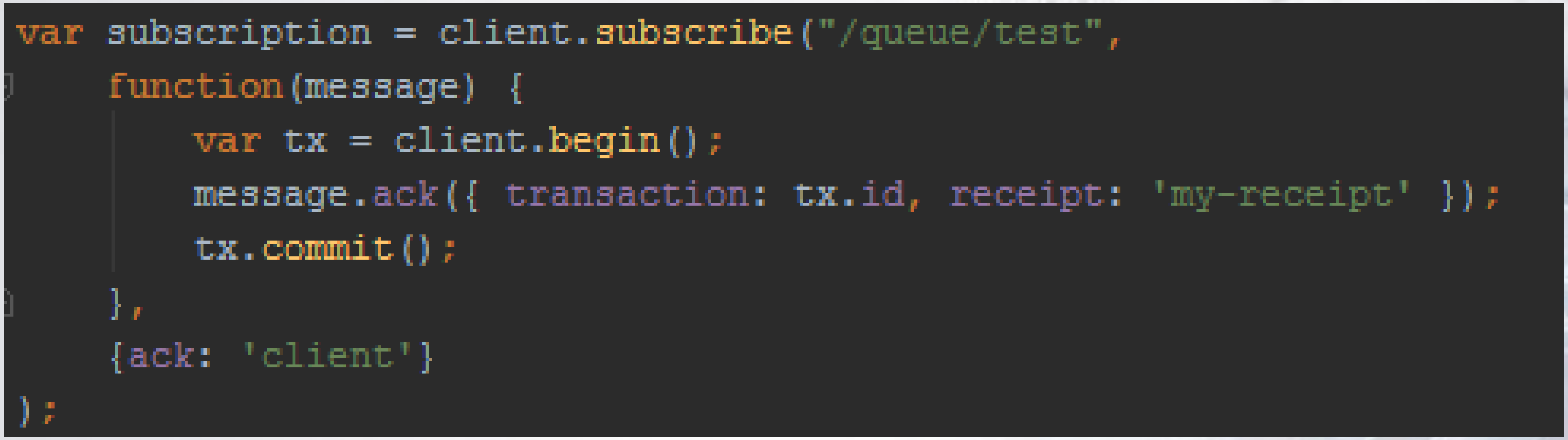
^@

COMMIT
transaction:tx1

^@

ABORT
transaction:tx1

^@



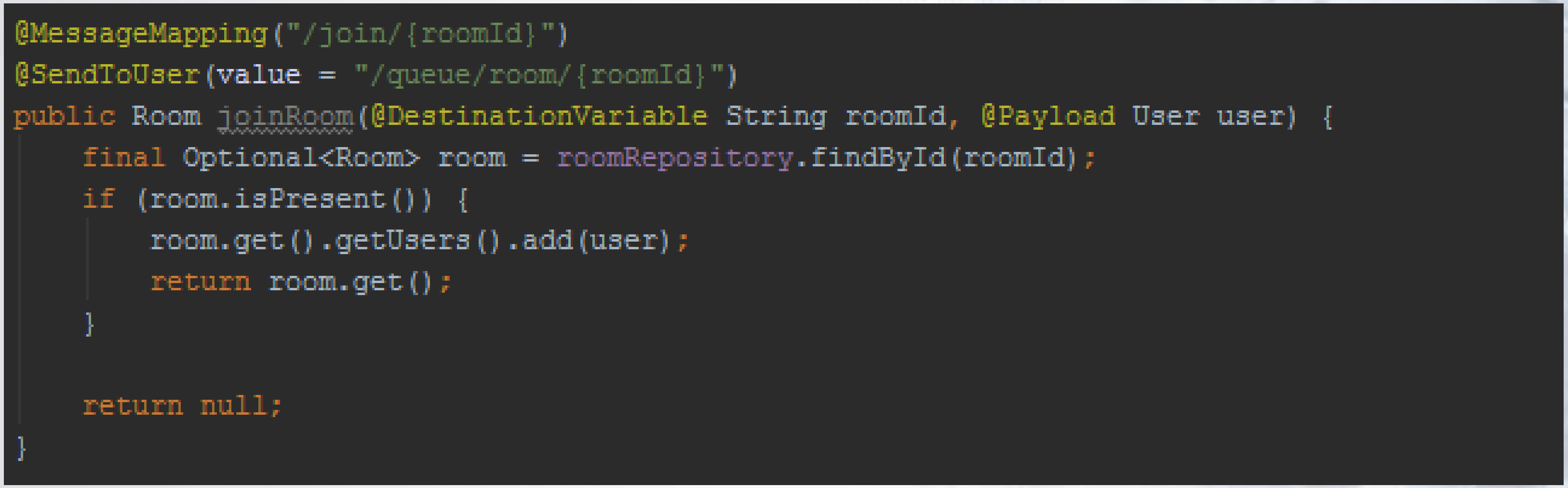
```
var subscription = client.subscribe("/queue/test",
    function(message) {
        var tx = client.begin();
        message.ack({ transaction: tx.id, receipt: 'my-receipt' });
        tx.commit();
    },
    {ack: 'client'}
);
```

Server Frames

```
MESSAGE
subscription:0
message-id:007
destination:/queue/messages/1
content-type:application/json

{ user: { username: ‘zbra054', email: ‘zbra054@haha.com' }, text: ‘Hello! ' }
^@
```

Server Frames

```
MESSAGE
subscription:0
message-id:007
destination:/queue/messages/1
content-type:application/json

{ user: { username: ‘zbra054', email: ‘zbra054@haha.com' }, text: ‘Hello! ' }
^@
```



```java
@MessageMapping("/join/{roomId}")
@SendToUser(value = "/queue/room/{roomId}")
public Room joinRoom(@DestinationVariable String roomId, @Payload User user) {
    final Optional<Room> room = roomRepository.findById(roomId);
    if (room.isPresent()) {
        room.get().getUsers().add(user);
        return room.get();
    }

    return null;
}
```

Server Frames

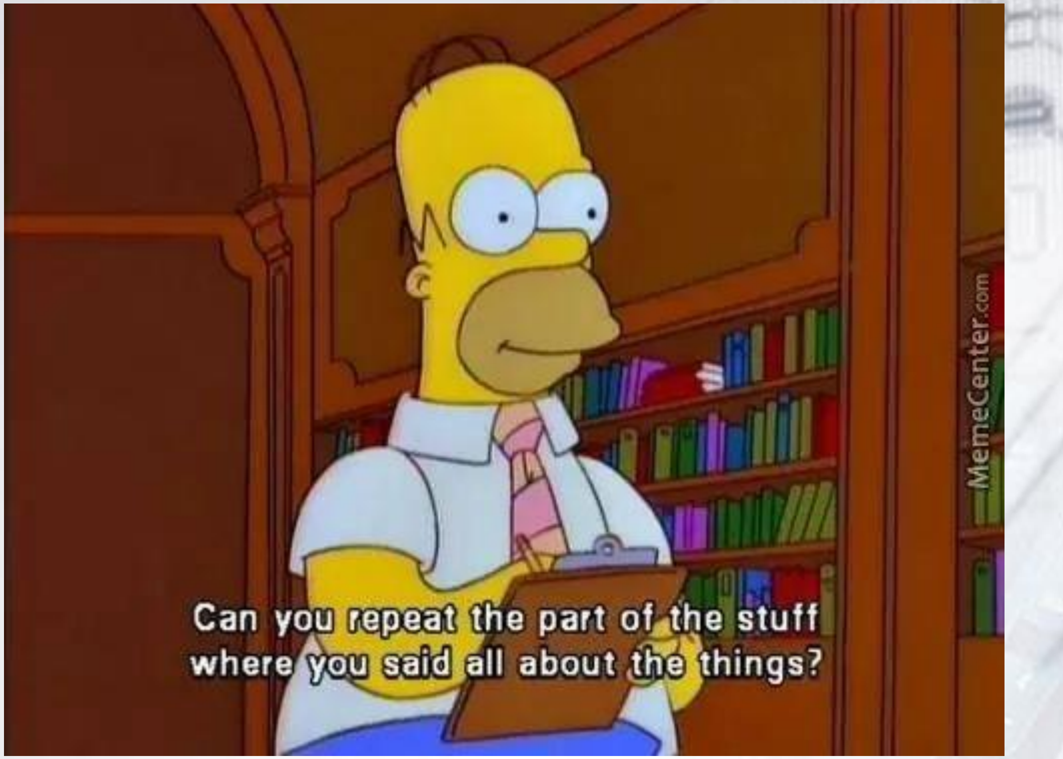
```
RECEIPT
receipt-id:xd


^@
```

Server Frames

```
RECEIPT
receipt-id:xd

^@
```

```
ERROR
receipt-id:xd
content-type:application/json
content-length:171
message:message could not be processed

{ faultReason: ‘User zbra054 has been removed, try using zbra54’ }
^@
```

Questions


Can you repeat the part of the stuff
where you said all about the things?
MemeCenter.com

www.zbra.com.br

Filipe Sbragio - filipe.sbragio@zbra.com.br

Márcio Hariki - marcio.hariki@zbra.com.br