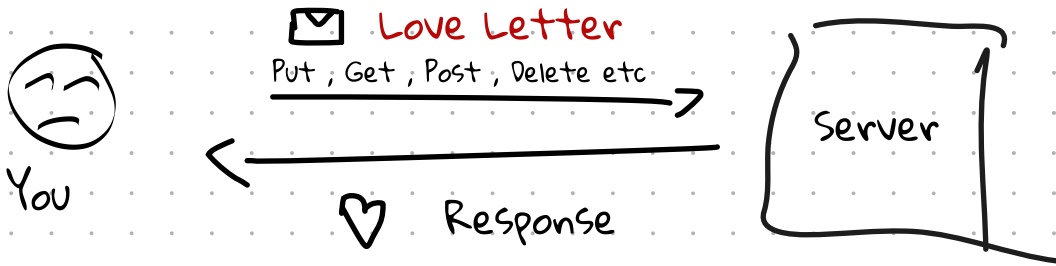


Websockets

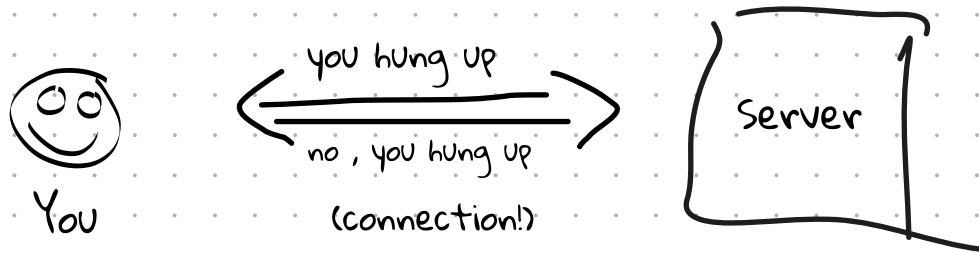
THE PHONE CALL OF THE WEB

Regular **HTTP** is like sending letters.



You send something, wait for reply, done.
Want updates? Send another letter!

Websockets are like a **PHONE CALL**:



Once connected, **EITHER** side can talk WHENEVER they want. No asking permission!

HTTP

```

you: "Hey server, anything new?"
Server : "Nope"      (5 seconds later)
you : "Hey server, anything new?"
Server : "Nope"      (5 seconds later)

"Hey server, anything new?"
Server : "YES! Here's a chat message!"
    
```

websocket

```

WebSocket: sitting quietly...sitting quietly
SERVER: "HEY! New chat message!"
You: "Thanks!"
    
```

But.....
WHY?



how it starts

1 You: "Hey can we upgrade to WebSocket?"
HTTP GET request with special headers

```
Upgrade: websocket      Connection:
Upgrade      Sec-WebSocket-Key:
[random stuff]
```

2 Server: "Sure! Here's my key back"

```
HTTP 101 Switching Protocols
```

3 NOW YOU'RE CONNECTED!
The HTTP connection transforms
into a WebSocket connection

WHAT ACTUALLY HAPPENS

★ it's a TCP connection
stays OPEN

Both can send
"frames" whenever!

Frame = tiny packet
of data could be
text, could be binary



THE CATCH

Wait .. there's a catch?!

WebSockets are
STATEFUL!

Server remembers you're connected Server uses memory
for your connection If server restarts? Connection dies!

Load balancing gets tricky too, proper message queue (Kafka,
NATS) handles distribution and persistence way better than rolling
your own WebSocket server but that's a story for another note