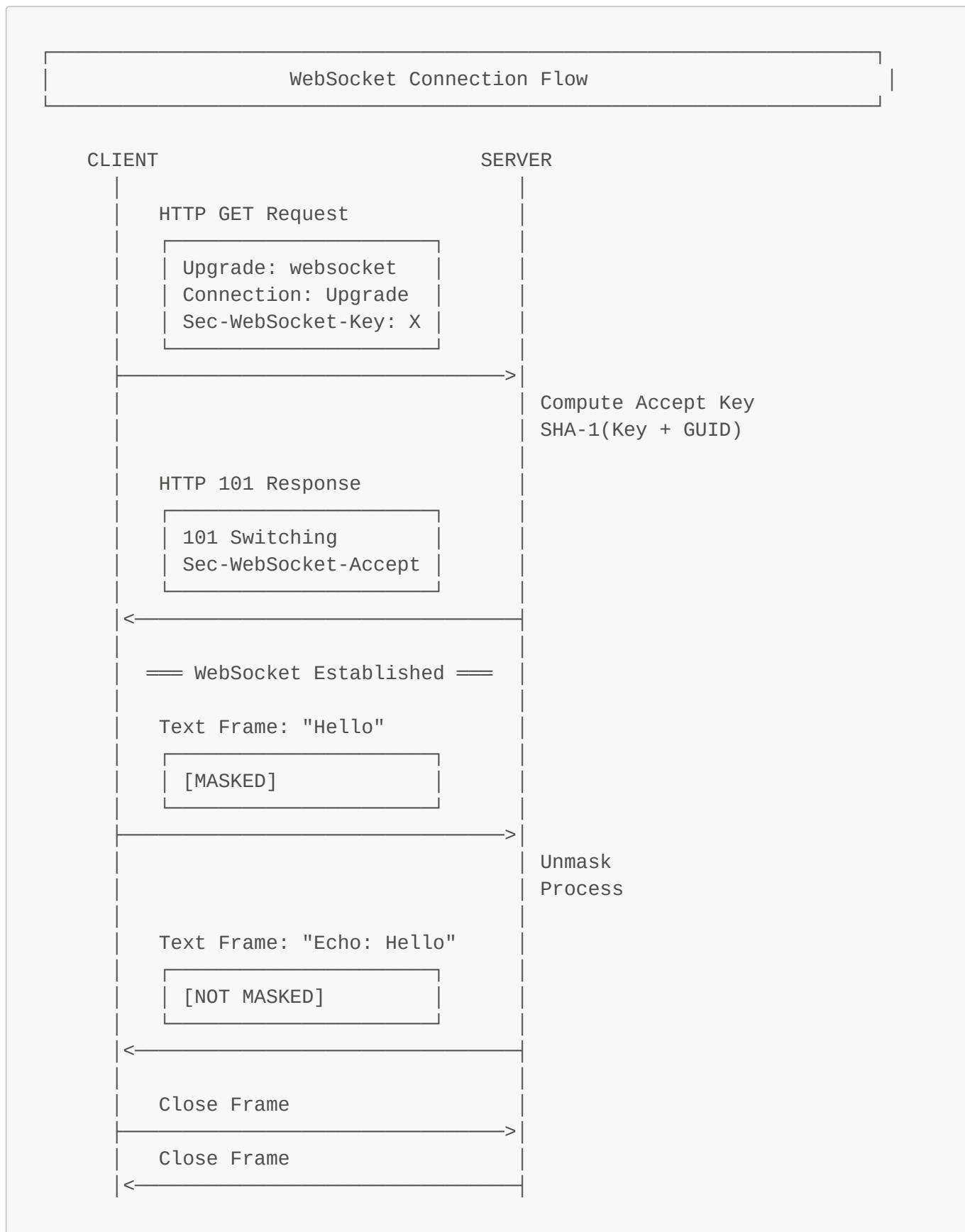


# Visual Protocol Diagrams

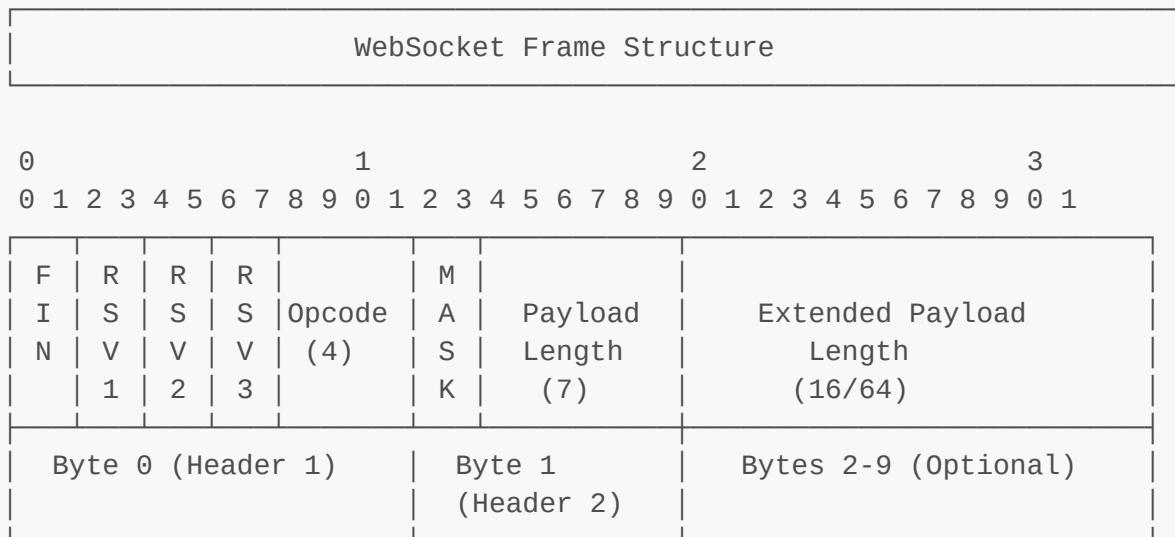
ASCII art diagrams to visualize the WebSocket protocol.

## Connection Flow Diagram



x Connection Closed

## Frame Structure Diagram



Masking Key (4 bytes) - Only if MASK=1

Byte 0 | Byte 1 | Byte 2 | Byte 3 |

Payload Data  
(Variable Length)  
Masked if MASK=1, Unmasked if MASK=0

## Byte-Level Breakdown

## First Byte (Byte 0)

Bit: 7 6 5 4 3 2 1 0



FIN: 1 = final fragment, 0 = more fragments

RSV: Reserved bits (must be 0)

Opcode: 4-bit frame type identifier

### Second Byte (Byte 1)

Bit: 7 6 5 4 3 2 1 0

MASK	LEN						
------	-----	-----	-----	-----	-----	-----	-----

MASK: 1 = payload is masked, 0 = not masked

LEN: 7-bit payload length (or indicator)

## Length Encoding Examples

### Small Message (0-125 bytes)

Example: "Hi" (2 bytes)

0x81	0x82	Mask (4B)	"Hi"
FIN+TXT	MASK+2	0x12345678	Masked

### Medium Message (126-65,535 bytes)

Example: 200 bytes

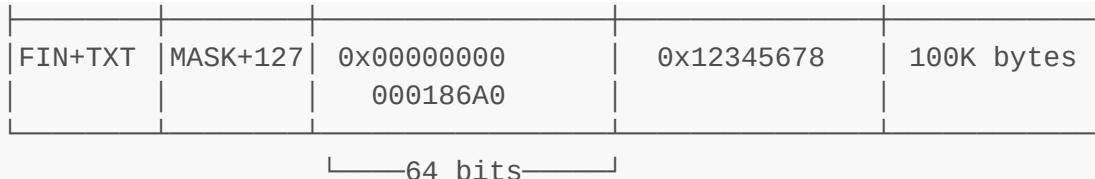
0x81	0xFE	Length	Mask (4B)	Payload
FIN+TXT	MASK+126	0x00C8	0x12345678	200 bytes

└16 bits┘

### Large Message (> 65,535 bytes)

Example: 100,000 bytes

0x81	0xFF	Length	Mask (4B)	Payload
------	------	--------	-----------	---------



## 🎭 Masking Algorithm Visual

### XOR Masking Process

Original Message: "HELLO"

Masking Key: [0x37, 0xFA, 0x21, 0x3D]

H	E	L	L	O
0x48	0x45	0x4C	0x4C	0x4F
XOR	XOR	XOR	XOR	XOR
[0x37]	[0xFA]	[0x21]	[0x3D]	[0x37] ← Mask repeats
↓	↓	↓	↓	↓
0x7F	0xBF	0x6D	0x71	0x78 ← Masked bytes

Unmasking (same process):

0x7F	0xBF	0x6D	0x71	0x78
XOR	XOR	XOR	XOR	XOR
[0x37]	[0xFA]	[0x21]	[0x3D]	[0x37]
↓	↓	↓	↓	↓
0x48	0x45	0x4C	0x4C	0x4F
H	E	L	L	O ← Original restored

## 🔒 Handshake Key Computation

### Sec-WebSocket-Accept Calculation

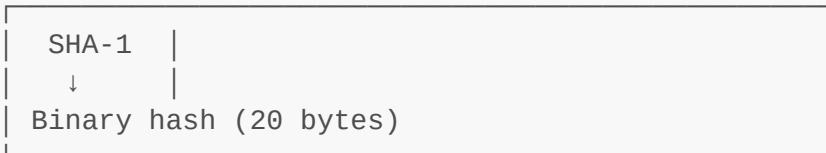
Step 1: Receive Client Key

Sec-WebSocket-Key: dGhIHNhbXBsZSBub25jZQ==

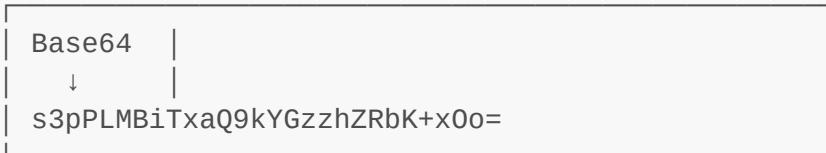
Step 2: Concatenate with Magic String

dGhIHNhbXBsZSBub25jZQ==258EAFA5-E914-47DA-... .

Step 3: Compute SHA-1 Hash



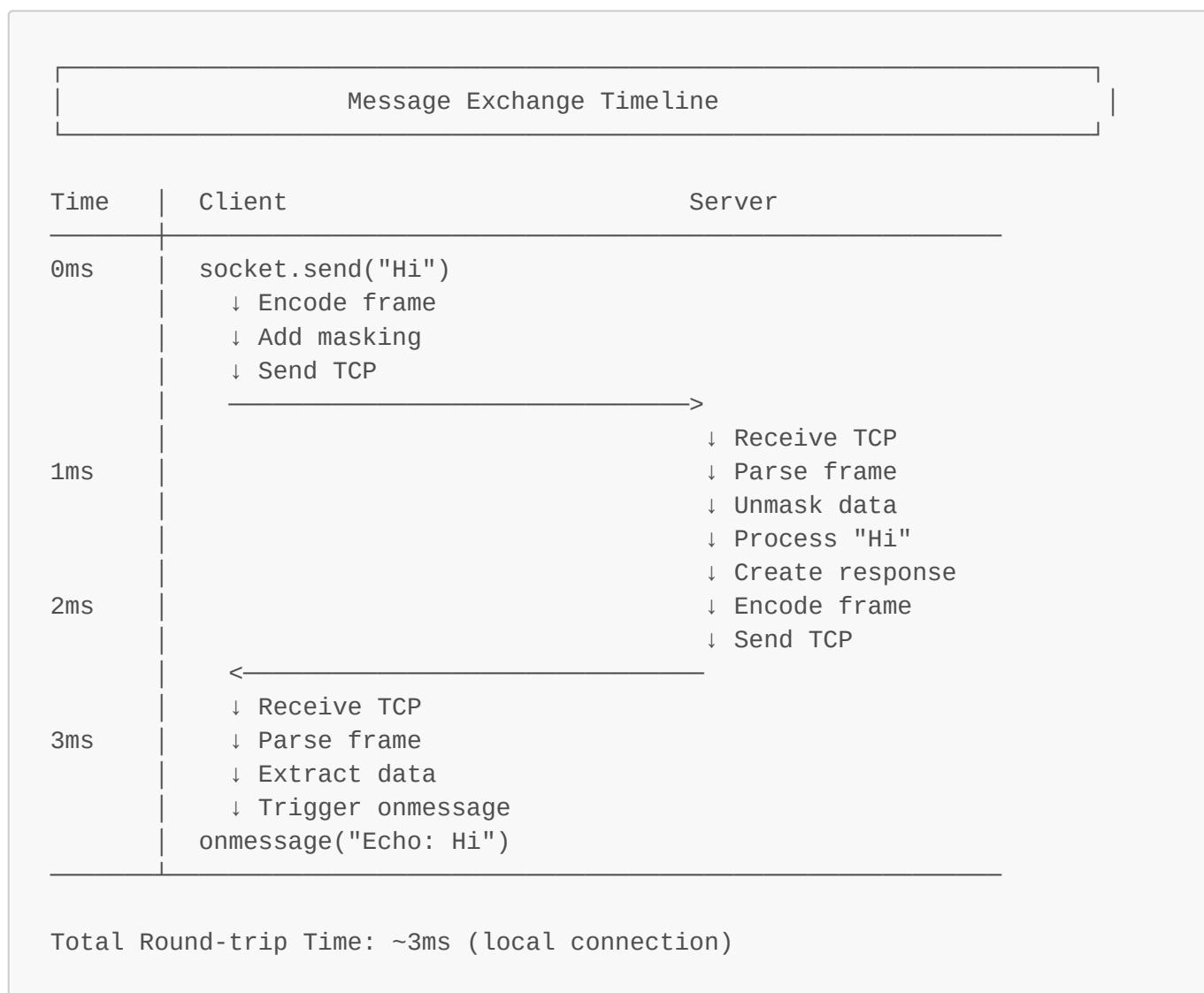
Step 4: Base64 Encode



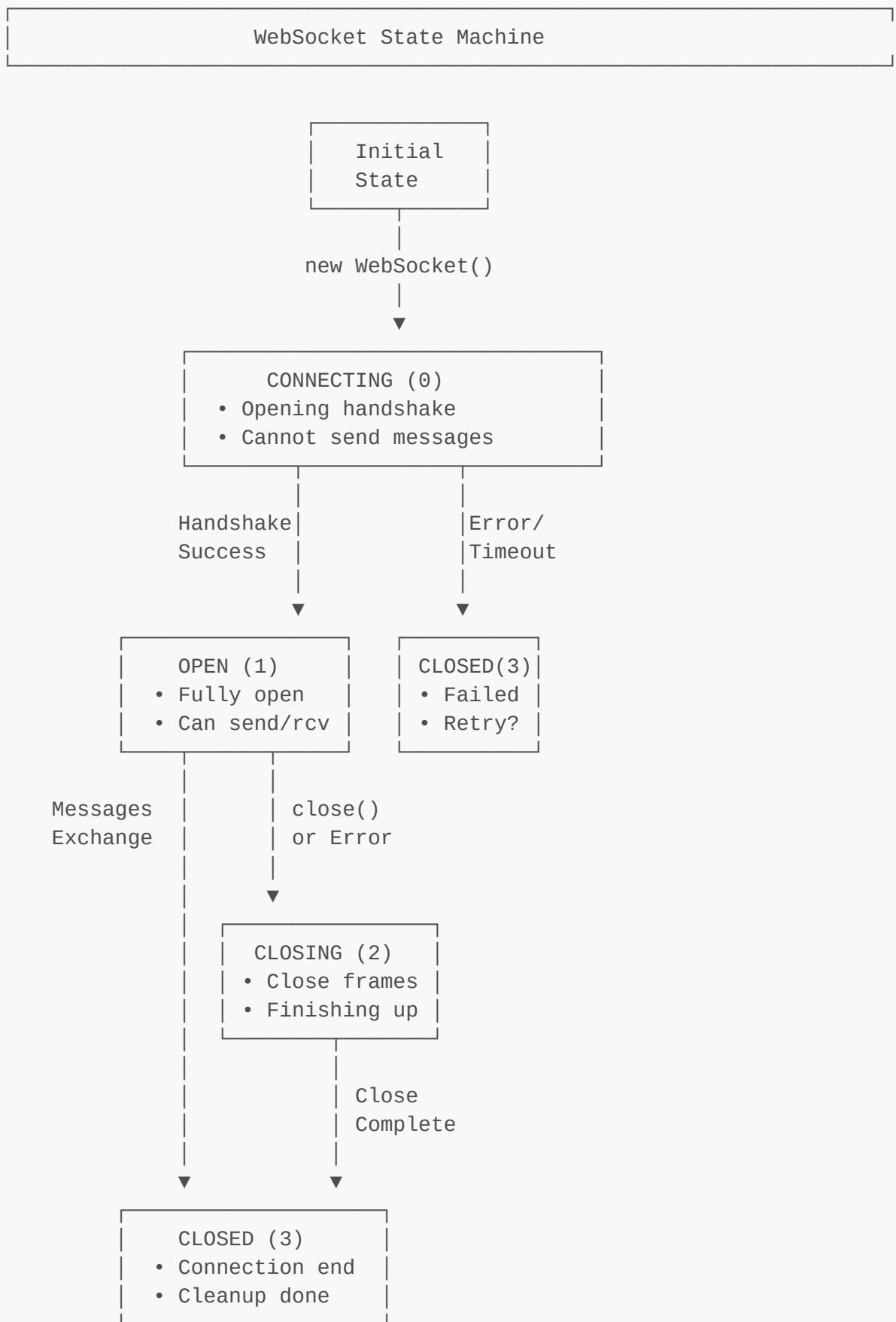
Step 5: Send Response



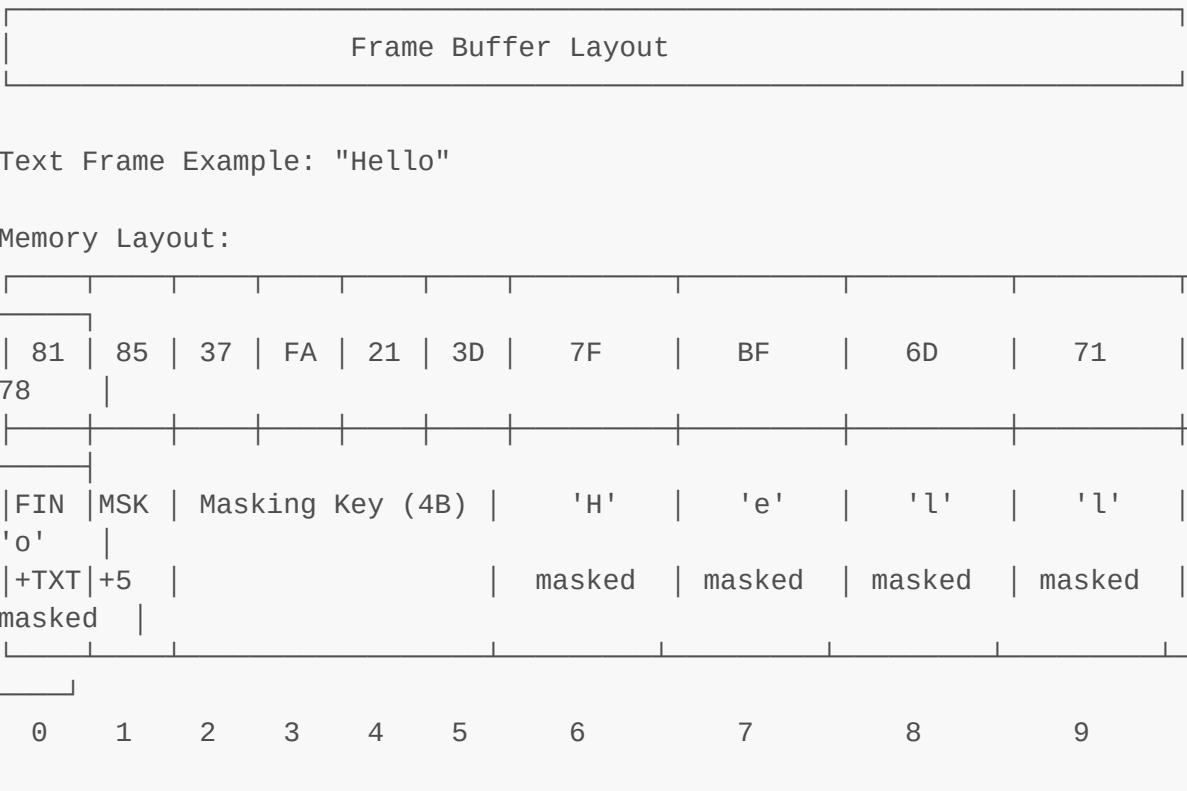
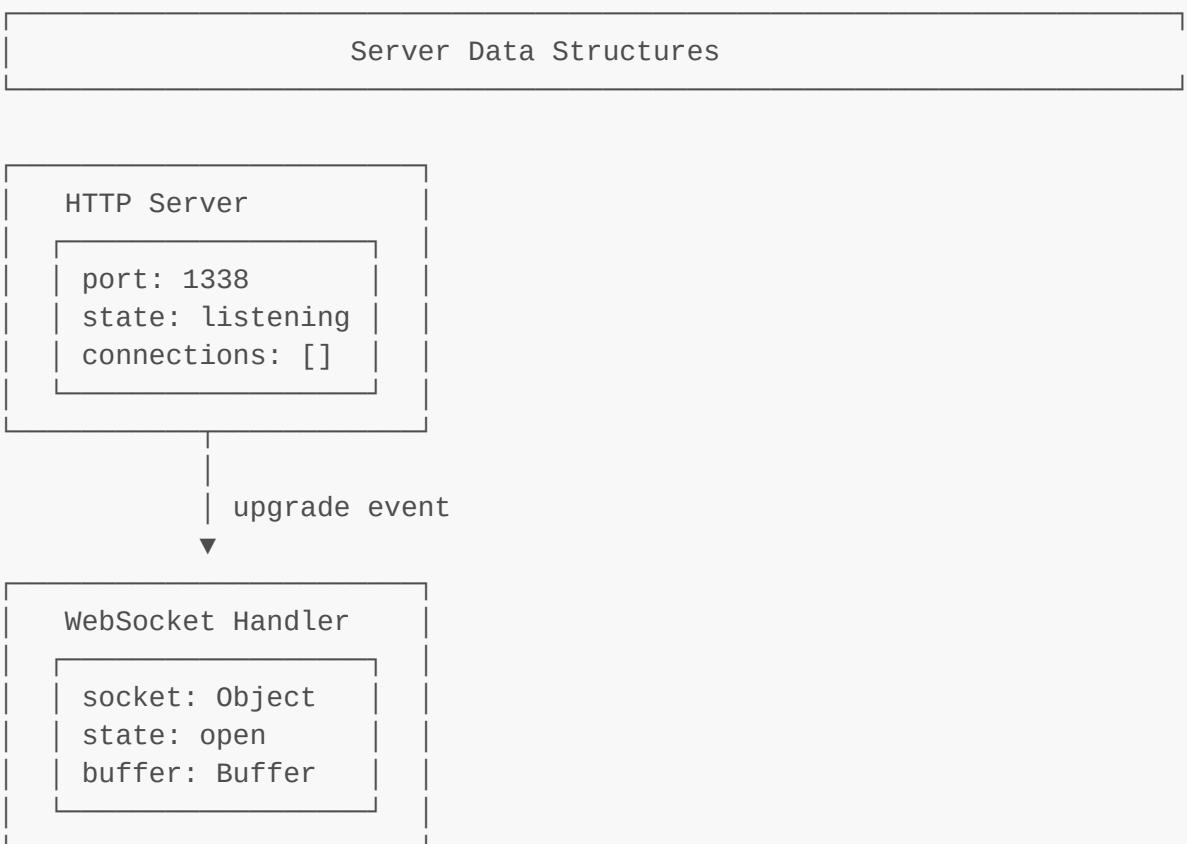
## 🚀 Message Flow Timeline



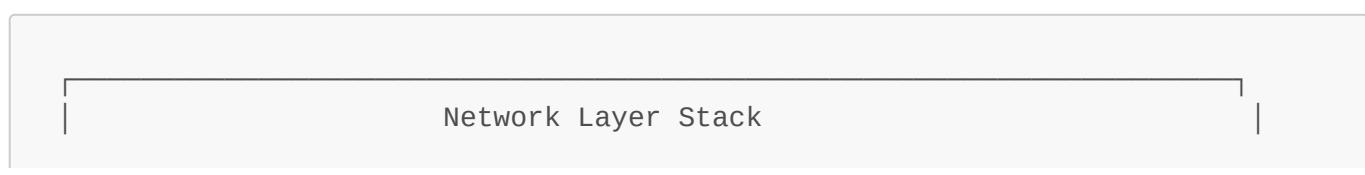
## 🔄 State Machine Diagram

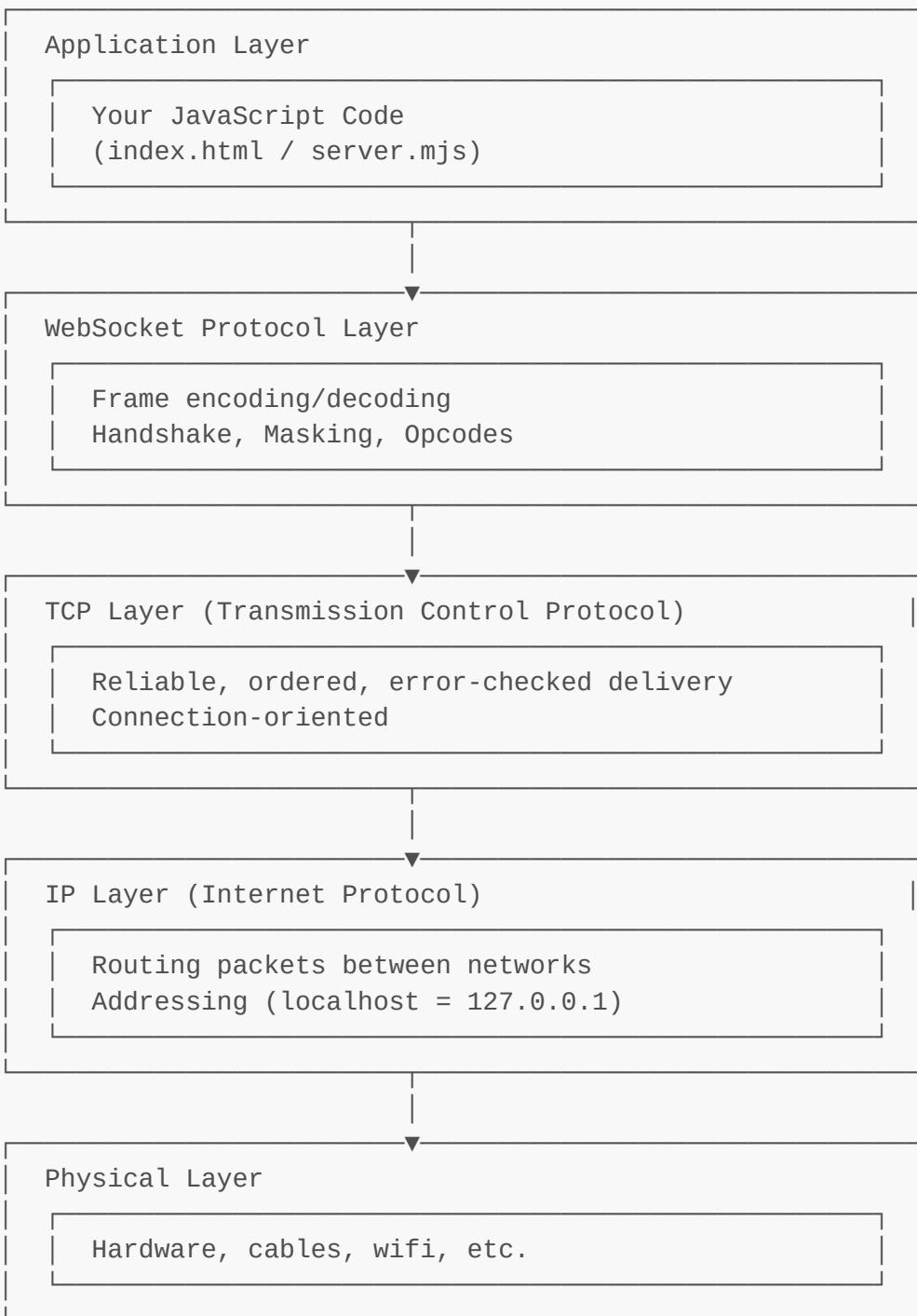


## ✓ Data Structure Layouts



## 🌐 Network Stack





These visual diagrams help understand the WebSocket protocol at every level! 