



# Using Sockets in Java

# TCP/IP

- A protocol is a set of rules that determine how things communicate with each other
- The software which manages Internet communication follows a suite of protocols called *TCP/IP*
- The *Internet Protocol* (IP) determines the format of the information as it is transferred
- The *Transmission Control Protocol* (TCP) dictates how messages are reassembled and handles lost information

# IP

- Internet Protocol: RFC 791
- Base protocol for data transmission
- It is connectionless and unreliable
- It is routable
- Defines the addressing of the nodes using IP addresses

# IP and Internet Addresses

- Each computer on the Internet has a unique *IP address*, such as:

130.136.1.110

- Most computers also have a unique Internet name, which also is referred to as an *Internet address*:

www.cs.unibo.it

- The first part indicates a particular computer (*www*)
- The rest is the *domain name*, indicating the organization (*cs.unibo.it*)

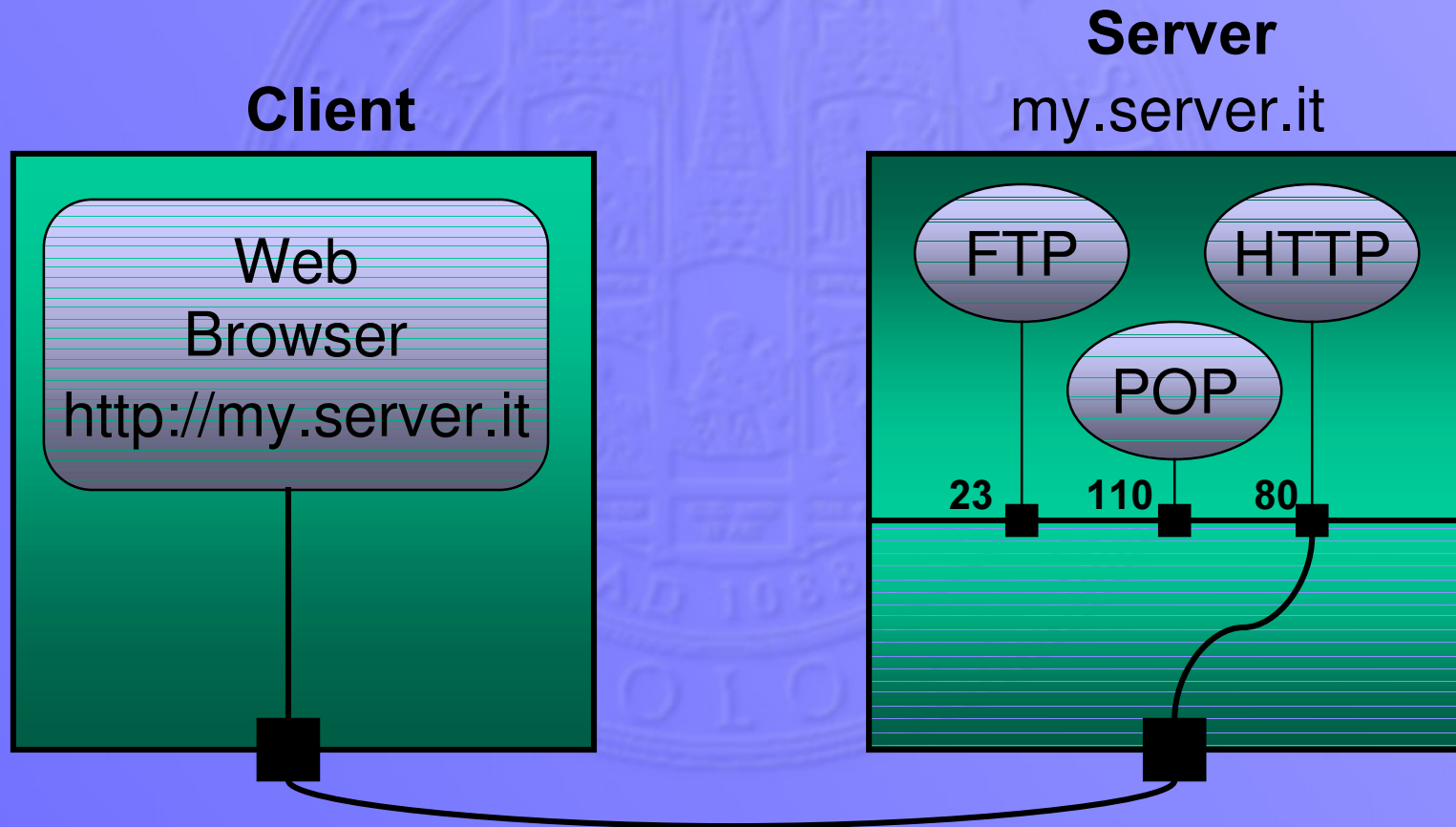
# TCP

- Transmission Control Protocol:  
RFC 793
- It's a reliable connection-oriented protocol
- Reliability is achieved using packets indexing and generating “ack” messages for each received packet

# TCP

- Creating a connection is an asymmetrical process; once the connection is established the protocol becomes symmetric
- Ports are used to initiate a connection

# Connection Sequence





# TCP Standard Ports

► Below 1024, assigned by the IANA

21	FTP
23	Telnet
80	HTTP
110	POP3
119	NNTP



# TCP Connections in Java

- The `Socket` class is used for handling TCP connections in Java
- A socket can be used as a data source for a stream
- The `SocketServer` class is used to establish a connection

# Waiting for a Connection

```
ServerSocket ss =  
new ServerSocket(1234);  
Socket s = ss.accept();
```

# Connecting to a Remote Host

```
Socket s = new  
Socket("my.host.it", 1234);
```