

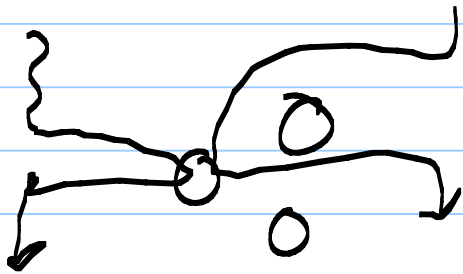
Interprocess Communication

Note Title

Threads

indep. control flow

shared object heap

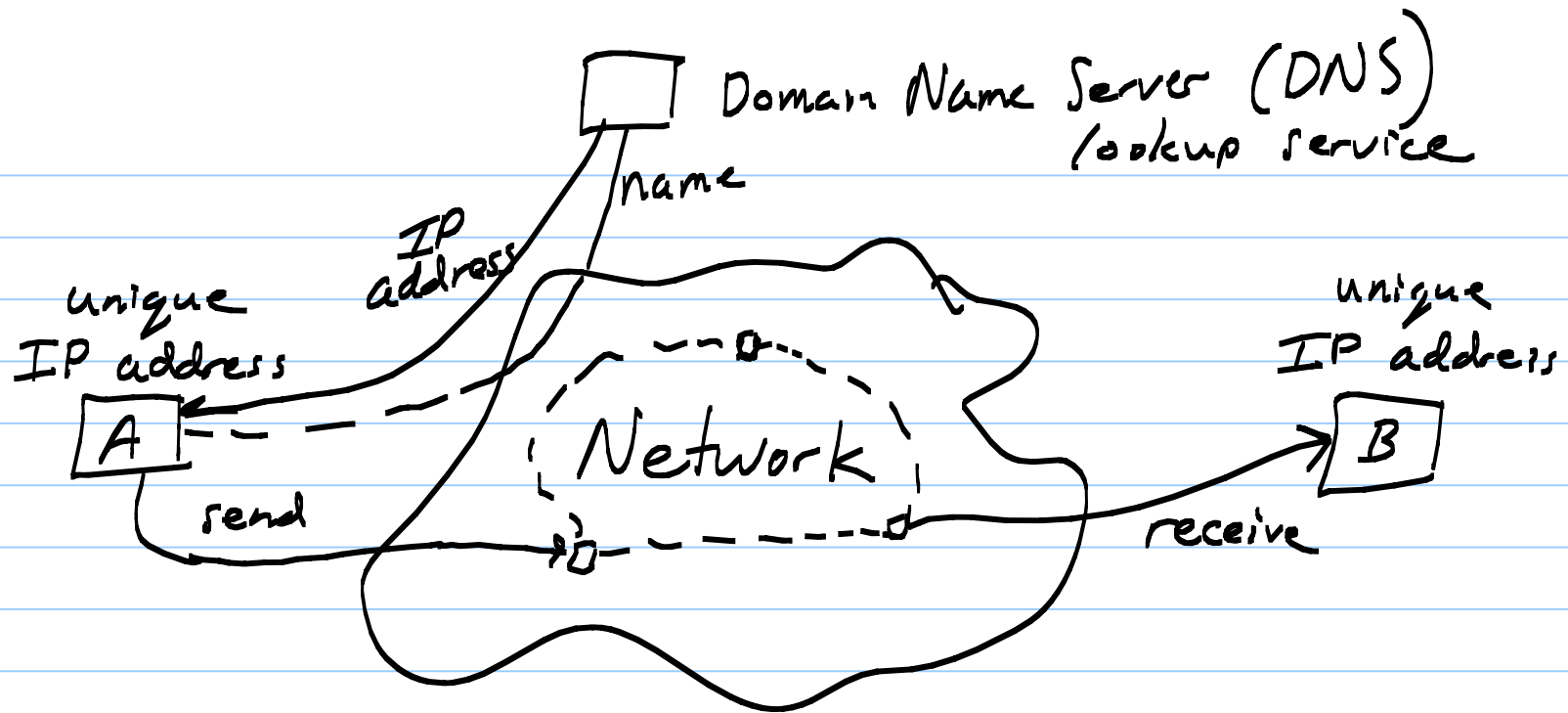


Interprocess Comm.

indep. control flow

message passing





Internet Protocol (IP)

- datagrams (best effort)
(can reorder)

checksum prevent garbled msgs.

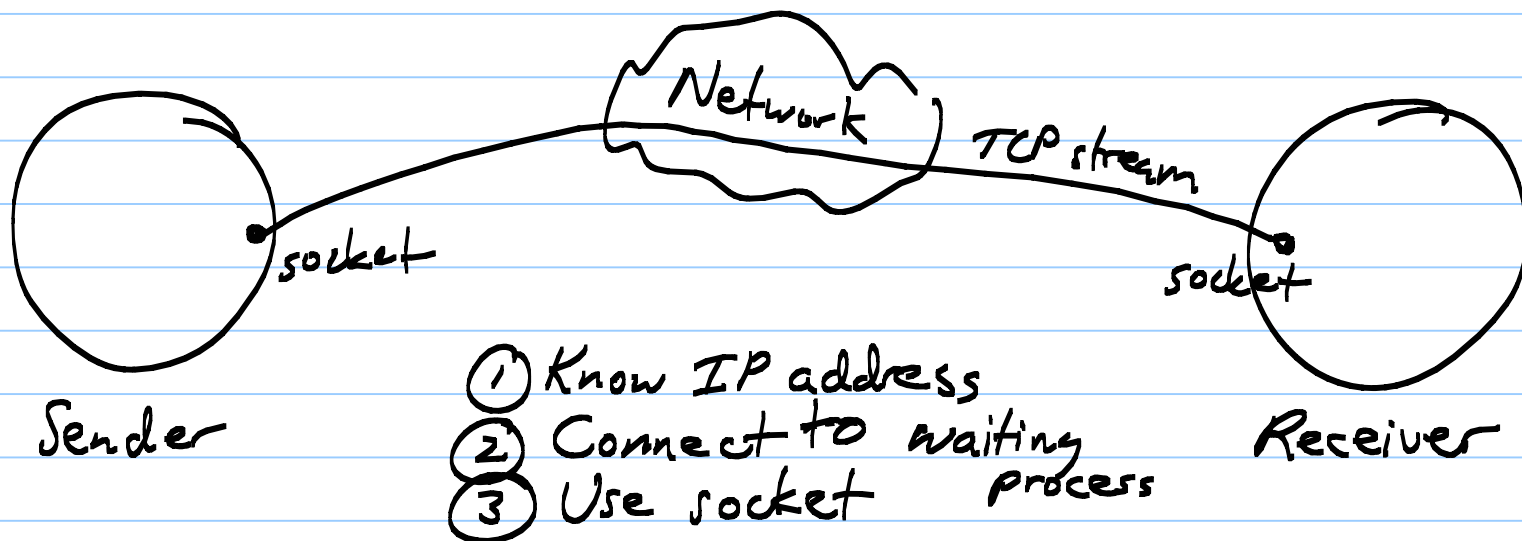
Transmission Control Protocol (TCP)

- reliable stream on top of IP
 - sequence #s on msgs
 - retransmit missing msgs

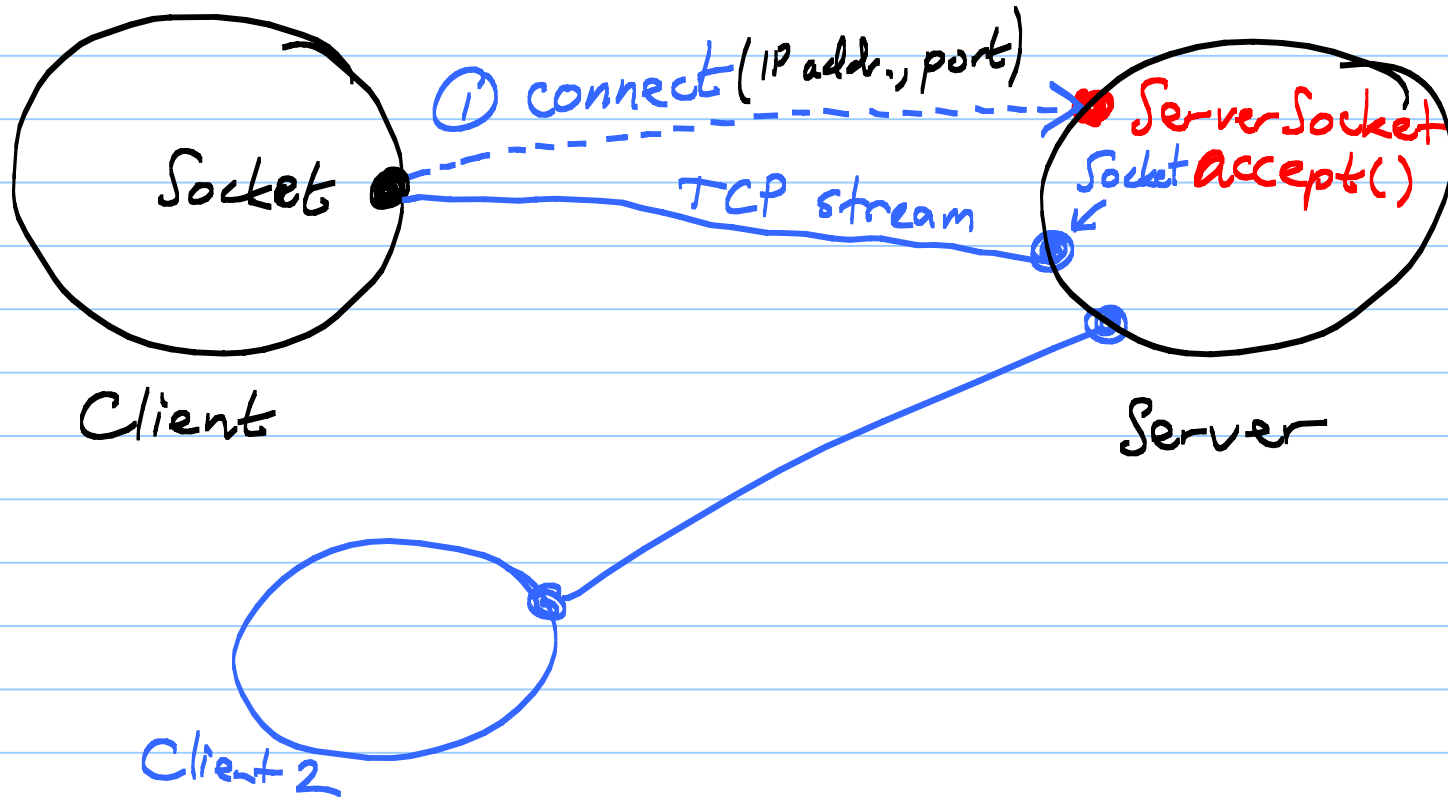
Java support for TCP Streams

```
fos = new FileOutputStream("foo.txt");  
oos = new ObjectOutputStream(fos);  
oos.writeObject(foo);
```

Our setup:



"localhost" — IP address
of machine you're
running on



Use port #s in
10,000 - 30,000

Client

```
Socket s = new Socket ("localhost",  
10450);
```

```
DataOutputStream dos =  
new DataOutputStream(  
s.getOutputStream());
```

```
dos.writeInt(4);
```

```
// create DataInputStream  
to read result from  
server
```

Server

```
ServerSocket ss = new ServerSocket(10450)  
while (---) {  
    Socket s = ss.accept();
```

```
    DataInputStream dis =  
    new DataInputStream(  
    s.getInputStream());
```

```
    int square = Math.pow(  
    dis.readInt(), 2);
```

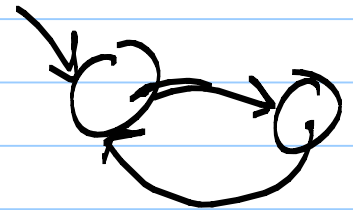
```
    // create DataOutputStream  
    wrapping the socket's output  
    stream + send square to client
```

Warnings: If you use object streams for comm.

- Each object is put in the stream only once

⇒ either create new msg objects
or reset the output stream

- Receive a copy of the object



Multithreaded
Server

