

CS375-Networks
Fall 2007
Marc Smith
Tue, Sep 4

2.2 Using Sockets

Two varieties: TCP/IP and UDP (datagram)

Sockets provide a means for process communication across a network.

Ideal for establishing Client/Server relationships between processes. The steps to establish clients and servers are clearly labeled in the sample code.

2.2.1 TCP Sockets

Steps to set up a server process:

1. Create a ServerSocket object
2. Put the server into a waiting state
3. Set up input and output streams
4. Send and receive data
5. Close the connection (after completion of the dialog)

Steps to set up a client process:

1. Establish a connection to the server
2. Set up input and output streams
3. Send and receive data
4. Close the connection

(over)

2..2.2 Datagram (UDP) Sockets

A *connectionless* dialog between client and server

Steps to set up a server process:

1. Create a DatagramSocket object
2. Create a buffer for incoming datagrams
3. Create a DatagramPacket object for the incoming datagrams
4. Accept an incoming datagram
5. Accept the sender's address and port from the packet
6. Retrieve the data from the buffer
7. Create the response datagram
8. Send the response datagram
9. Close the DatagramSocket (only in exceptional circumstances)

Steps to set up a client process:

1. Create a DatagramSocket object
2. Create the outgoing datagram
3. Send the datagram message
4. Create a buffer for incoming datagrams
5. Create a DatagramPacket object for the incoming datagrams
6. Accept an incoming datagram
7. Retrieve the data from the buffer
8. Close the DatagramSocket