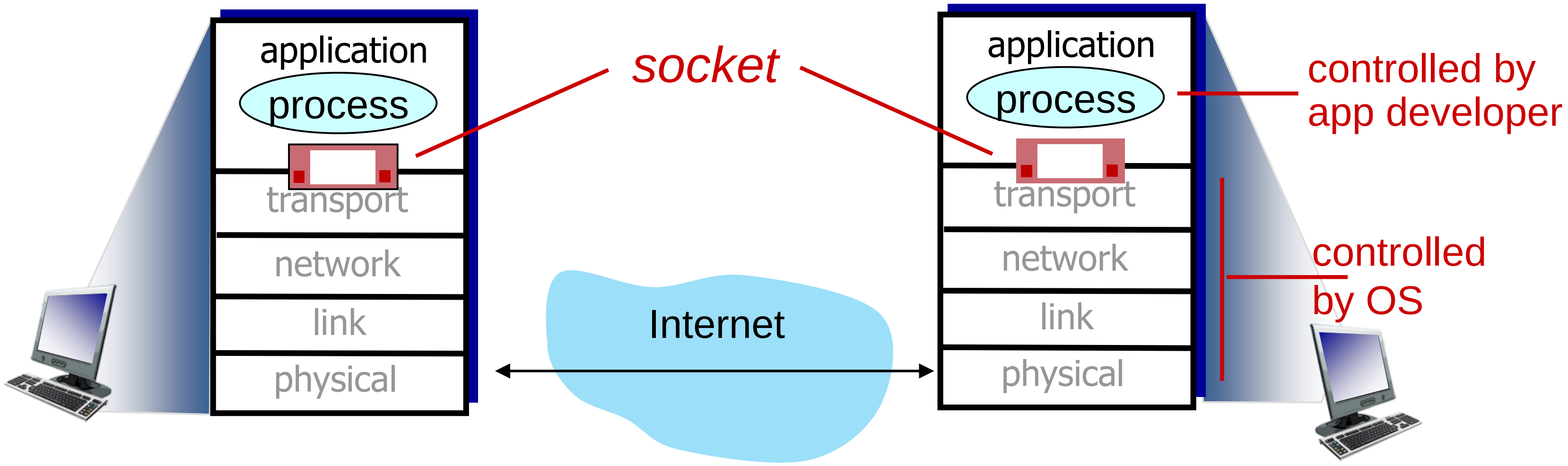
Computer Networks

Team Project
Server/Client Socket Programming

Yoojoong Kim, Ph.D.
Assistant Professor
School of Computer Science and Information Engineering
The Catholic University of Korea, Republic of Korea

Server/Client Socket Programming

- **Socket programming**
  - A way of connecting two nodes on a network to communicate with each other. One socket (node) listens on a particular port at an IP, while the other socket reaches out to the other to form a connection. The **server** forms the listener socket while the **client** reaches out to the server.



Server/Client Socket Programming

- **Socket programming**

  Two socket types for two transport services:

  - UDP (User Datagram Protocol): unreliable datagram
  - TCP (Transmission Control Protocol): reliable, byte stream-oriented

Server/Client Socket Programming

- **Socket programming** with UDP

UDP: no "connection" between client & server

- no handshaking before sending data
- sender explicitly attaches IP destination address and port number to each packet
- receiver extracts sender IP address and port number from received packet

UDP: transmitted data may be lost or received out-of-order

Application viewpoint:

- UDP provides unreliable transfer of groups of bytes ("datagrams") between client and server

Server/Client Socket Programming

- **Socket programming** with TCP

  Client must contact server

  – server process must first be running

  – server must have created socket (door) that welcomes client's contact

  Client contacts server by:

  – Creating TCP socket, specifying IP address, port number of server process

  – **when client creates socket**: client TCP establishes connection to server TCP

  When contacted by client, **server TCP creates new socket** for server process to communicate with that particular client

  – allows server to talk with multiple clients

  – source port numbers used to distinguish clients

  Application viewpoint:

  – TCP provides reliable, in-order byte-stream transfer ("pipe") between client and server

Server/Client Socket Programming

- **Socket programming**
  - Objective: Develop your own **client/server applications** that communicate using sockets.
  - Socket can be used as buffer to deliver sequences of bytes that contain any information of hosts.
  - We can organize byte (or bit) patterns of the buffer to determine some rules between server and client.
  - The rules enable to work the function of server and client.

Server/Client Socket Programming

- **Team Project**
  - Make your own server/client application based on socket programming.
  - Any application is fine, but it should have other functions except socket programming.
  - Both TCP and UDP are allowed.
  - Use your best computer language.

- Team Organization
  - A team should be 1~5 students.
  - Project leader can choose team members.
  - Single developer is fine.
  - Otherwise, I will randomly organize 4 students for one team.

- Evaluation
  - Technical quality: 20%
  - Peer evaluation: 10%