

TCP/IP Networking and Socket I/O

14 Mar 2008

CMPT166

Dr. Sean Ho

Trinity Western University

Internet Protocol transmissions

- The **Internet Protocol** (IP) is the standard by which all communications over the Internet are done
 - **Host** / IP address
 - **Port** number: 0-65535 (0-1023 reserved)
- Two kinds of IP **packets**:
 - **TCP: connection-oriented**
 - ◆ Handshake, more **overhead**
 - **UDP: connectionless**
 - ◆ Packets might **disappear**, or be out of **order**, or get **duplicated**

Making a TCP Server in Java

- `java.net.ServerSocket` object
 - ◆ `server = new ServerSocket(port, maxclients);`
 - `maxclients` is **queue** length
 - **BindException** raised if invalid/used port
- **Bind** socket (start **listening**) (blocking):
 - ◆ `connection = server.accept();`
 - Returns a `java.net.Socket` object
- Communicate via **streams**:
 - ◆ `connection.getInputStream();`
 - ◆ `new ObjectInputStream(connection.getInputStream());`

Servers and multithreading

- (More on threading in ch23)
- **Master** thread listens on port
- When a **client** connects, **fork** off a thread
 - Thread handles **communication** with that client
- Master thread continues **listening** for other connections

- **Overhead** in forking new threads: keep a **pool** of threads available, reuse dormant threads

TCP client in Java

- Connect to server:
 - ◆ `connection = new Socket(host, port);`
 - `UnknownHostException` may be raised
- Communicate via `streams`:
 - ◆ `connection.getInputStream();`
 - ◆ `new ObjectInputStream(connection.getInputStream());`