

# Introduction to Java for C++ Programmers

Networking Basics

By: Mahboob Ali

# Network

- A network is a system of computers connected together so they can share resources and communicate with each other.
- **Networking** refers to how the connected computers communicate.

# Intranet

- Computers can also communicate across a private network called **Intranet**.
- Still common in today's businesses.
- Networking base comes from intranets.
- For examples, printers shared in a business environment over the business's intranet.
- Business can have multiple intranets one for each department.

# Host

- You can also use networking when you want applications running on the same machine to communicate with each other.
- In networking a machine is usually referred as a **host**.

# Client/ Server

- A common network configuration that one probably heard of is **client/server**, meaning that one (or more) hosts on the network are acting as servers, and the other hosts are clients that connect to the server.
- The browser is a client, and when you type in a web address, it connects to a server that has the files for the website address. In simple that's how internet works.

- You can have client/server interaction on the same host, for example, the MySQL database comes with a workbench you can use to perform database operations. The **workbench** is a client, and it connects to the MySQL database server.
- Quite often what web developers do when they are working on a website, is possibly to run an apache or IIS server on computer and connect to it using a web browser on the computer.

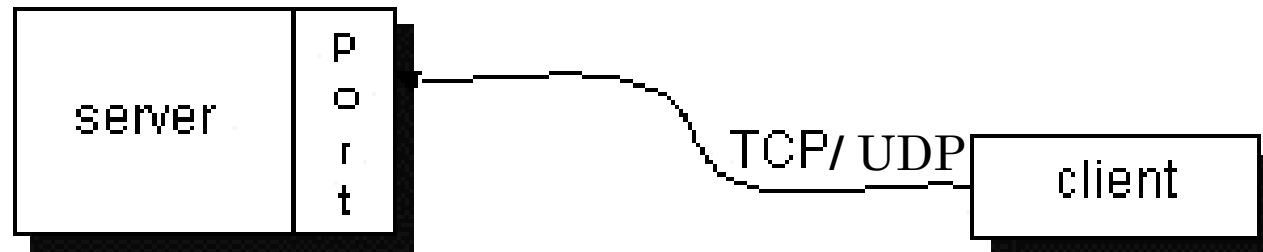
# Network Communications

- Computers on a network (including internet), communicates with each other using a **transport protocol**,
  1. TCP (Transmission Control Protocol)
  2. UDP (User Datagram Protocol)

- Generally, a computer will have one **physical connection** to a network, and anything sent to the computer from other hosts on the network will arrive through that connection.
- Sometimes the same computer will be running multiple applications that want data from the network. For example, at any one time, you might have browser open, a chat application open, streaming music etc.



- Once data arrives at one of the physical connection to the network, how does it get to the targeted application?
- *Answer:* Ports



- Each application that needs data from the network is assigned a port.

# TCP and UDP

- TCP (*Transmission Control Protocol*): is a connection-based protocol that provides a reliable flow of data between two computers.
- UDP (*User Datagram Protocol*): is a protocol that sends independent packets of data, called datagrams, from one computer to another with no guarantees about arrival. UDP is not connection-based like TCP.

# TCP (Transmission Control Protocol)

- **Definition:** A connection based protocol, provides a reliable flow of data between two computers.
- Like making a Telephone call.
- Provides point-to-point channel for applications that require reliable communications.
- Like HTTP, FTP and telnet.
- The key of a successful transfer is in the order of data.
- **Problem:**
  - Prior to information transmission, a connection must be established.
  - Connection must be rigidly held during the whole communication.
  - The connection must be closed explicitly at the end.
  - Holding the bandwidth even if there is no communication happening.

# Communication steps TCP

- When communicating TCP/IP, the sequence of events is as follows,
  1. The client opens a connection to the server.
  2. The client sends a request to the server.
  3. The server sends a response to the client.
  4. The client closes the connection to the server.
- Steps 2 and 3 may be repeated multiple times before the connection is closed.

# UDP (User Datagram Protocol)

- **Definition:** is a non-connection based protocol that sends independent packets of data, called *datagrams*.
- No guarantee of data arrival.
- Like sending a letter through postal service. i.e. the order of delivery is not important and is not guaranteed as well each message is independent of each other.
- Example: clock server that sends current time to its client when requested to do so.

# IPV4 and IPV6

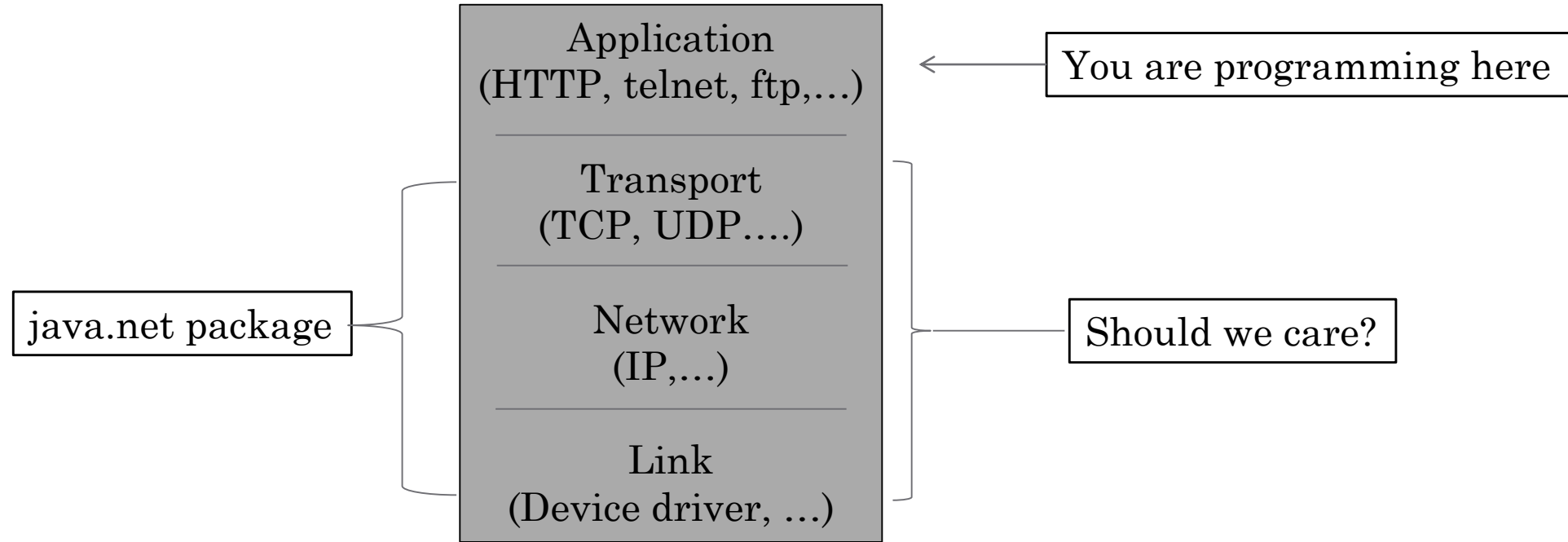
- **IPv4:** Internet Protocol Version 4, uses 32-bit address scheme, allows over 4 billion unique addresses.
- Addresses four integers, separate by dots.
- Not enough in today's time, as we have computers, tablets, game consoles, smart TVs, smart phones, smart appliances etc, all connected to internet, and each of device needs a unique IP address.
- **IPv6:** Internet Protocol version 6, got birth. Uses 128-bit address scheme, lots and lots of unique IP addresses.
- Written in hexadecimal and separated by colons.

[www.whatismyip.com](http://www.whatismyip.com)

- Data that is transmitted over the internet also carries information regarding which computer and port it will deliver to.
- 32-bit IP address identifies which computer to deliver the message.
- 16-bit number identify the Port, which TCP and UDP uses to deliver the data to the correct application.

- Port numbers range from 0 to 65,535 because ports are represented by 16-bit numbers.
- **Well-known:** port numbers ranging from 0 - 1023 are restricted; reserved for use by well-known services such as HTTP and FTP and other system services.
- Your applications should not attempt to bind to them.





- Why do we need to know about TCP, UDP, IP, Ports, ... ?
- To make better decision about which Java Classes to use in the program.

# Java Networking

- *java.net* package contains the classes you will use to establish connection between computers and then send messages between them.
- The package contains two sets of APIs,
  - Low-level API. (socket programming).
  - High-level API. (web-oriented, URL's, URI's)
- Java makes network coding much easier, by letting developers write code using abstract concepts and taking care of implementation detail under cover.