



ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN
Vietnam - Korea University of Information and Communication Technology

BÀI GIẢNG LẬP TRÌNH MẠNG

**PGS.TS.Huỳnh Công Pháp; Nguyễn Anh Tuấn; Lê Tân;
Nguyễn Thanh Cẩm; Hoàng Hữu Đức**

Khoa Khoa học máy tính

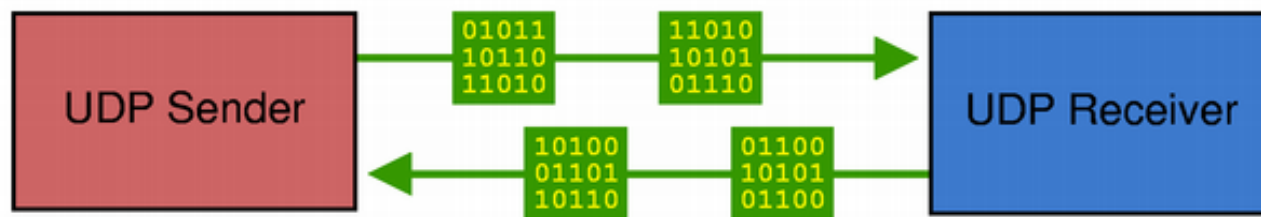


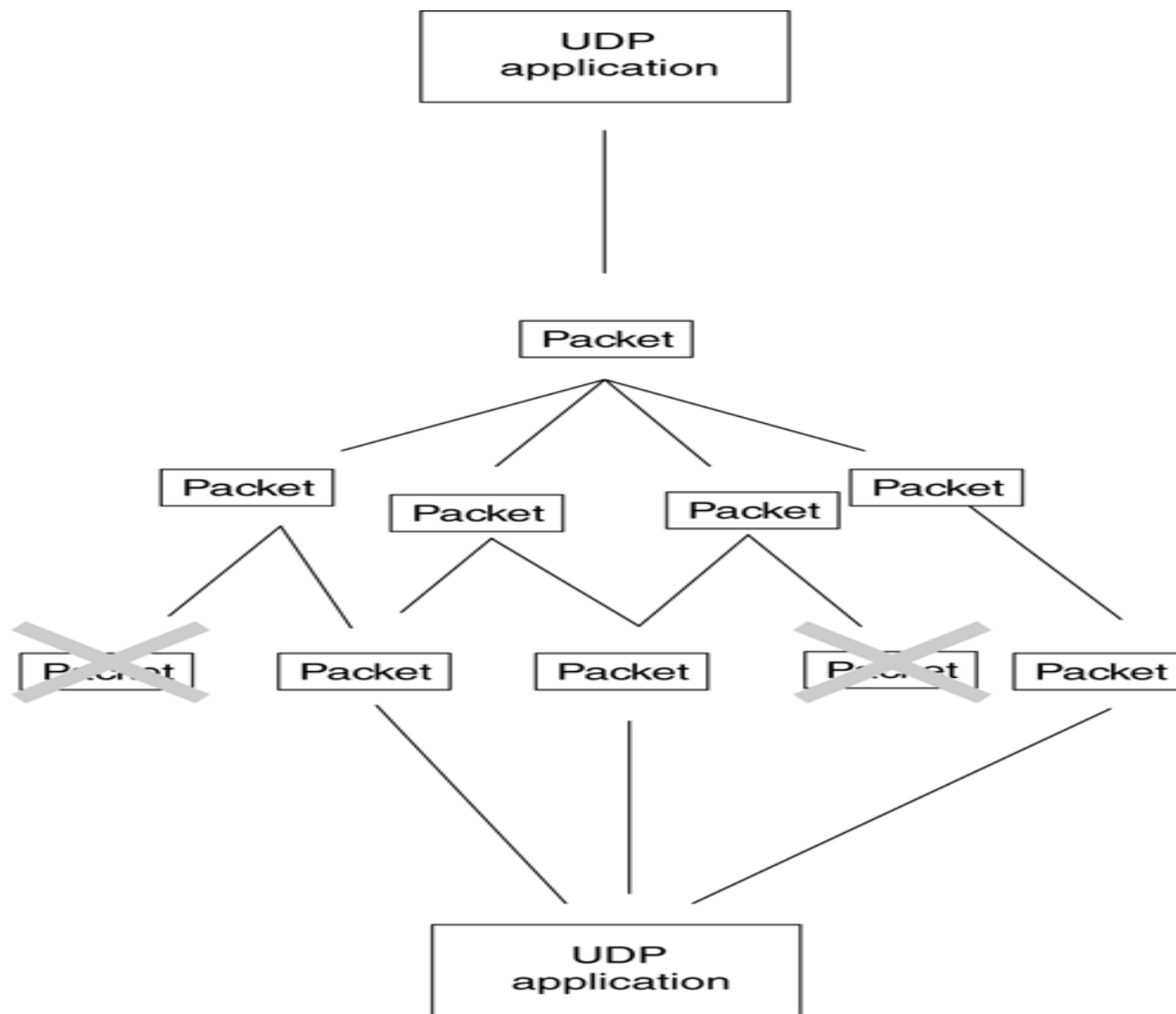
Bài 5. Lập trình với giao thức UDP (User Datagram Protocol)



Giao thức UDP

- UDP (User Datagram Protocol) là giao thức phi kết nối
 - Không thiết lập kết nối trước khi truyền (connectionless)
 - Các gói dữ liệu được gửi độc lập (gọi là datagrams)
 - Không bảo đảm sự toàn vẹn dữ liệu và trật tự.
 - UDP không cung cấp cơ chế báo nhận (acknowledgment),
- Cơ chế hoạt động tương tự như gửi một lá thư thông qua dịch vụ bưu điện

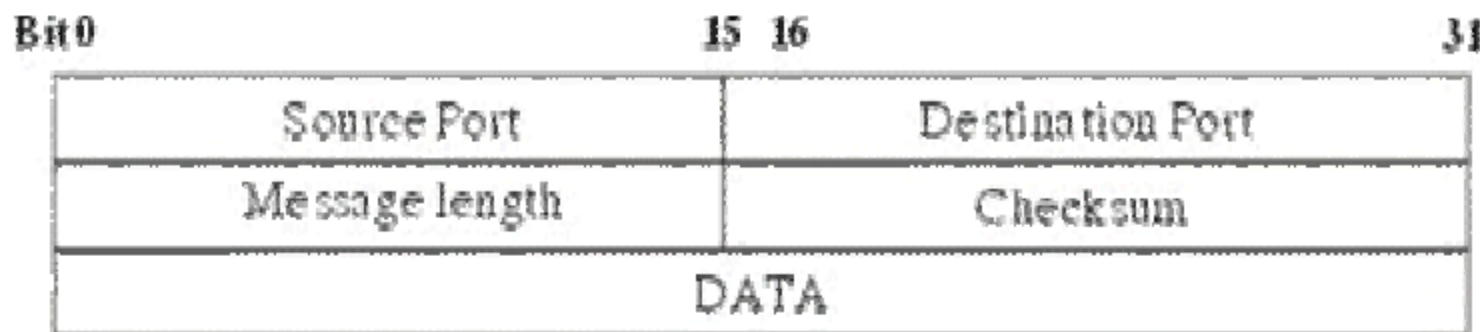






Khuôn dạng UDP datagram

- UDP datagram có các tham số đơn giản hơn nhiều so với TCP segment.



- UDP cũng cung cấp cơ chế gán và quản lý các số hiệu cổng (port number) để định danh duy nhất cho các ứng dụng chạy trên một trạm của mạng.
- Do ít chức năng phức tạp nên UDP thường có xu thế hoạt động nhanh hơn so với TCP.
- UDP thường dùng cho các ứng dụng không đòi hỏi độ tin cậy cao trong khi truyền.



Các bước tạo ứng dụng UDP Socket

- ứng dụng UDP socket không thiết lập kết nối

- Client

- socket, tạo một điểm cuối truyền thông phía client
 - sendto and recvfrom, truyền dữ liệu

- Server

- socket, Tạo một điểm cuối truyền thông phía server
 - bind, gán một cổng đến kết nối
 - sendto and recvfrom, truyền dữ liệu

• server : socket → bind → recvfrom → sendto → close

• client : socket → sendto → recvfrom → close

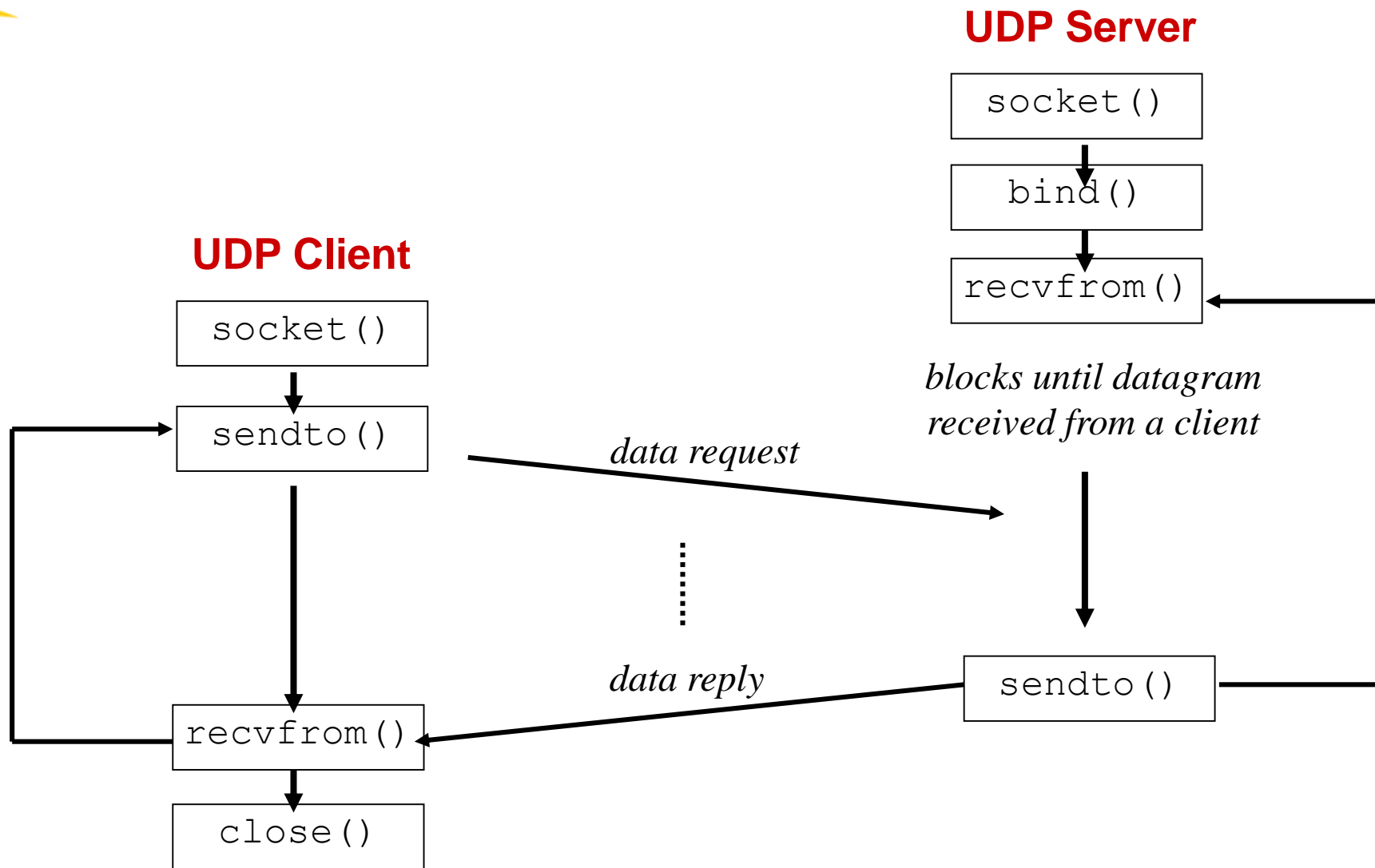




Lưu ý

- Client không thiết lập kết nối đến server
 - Kết nối không cần thiết
- Server không chấp nhận kết nối
 - chờ và lắng nghe không cần thiết
 - chấp nhận kết nối không tồn tại

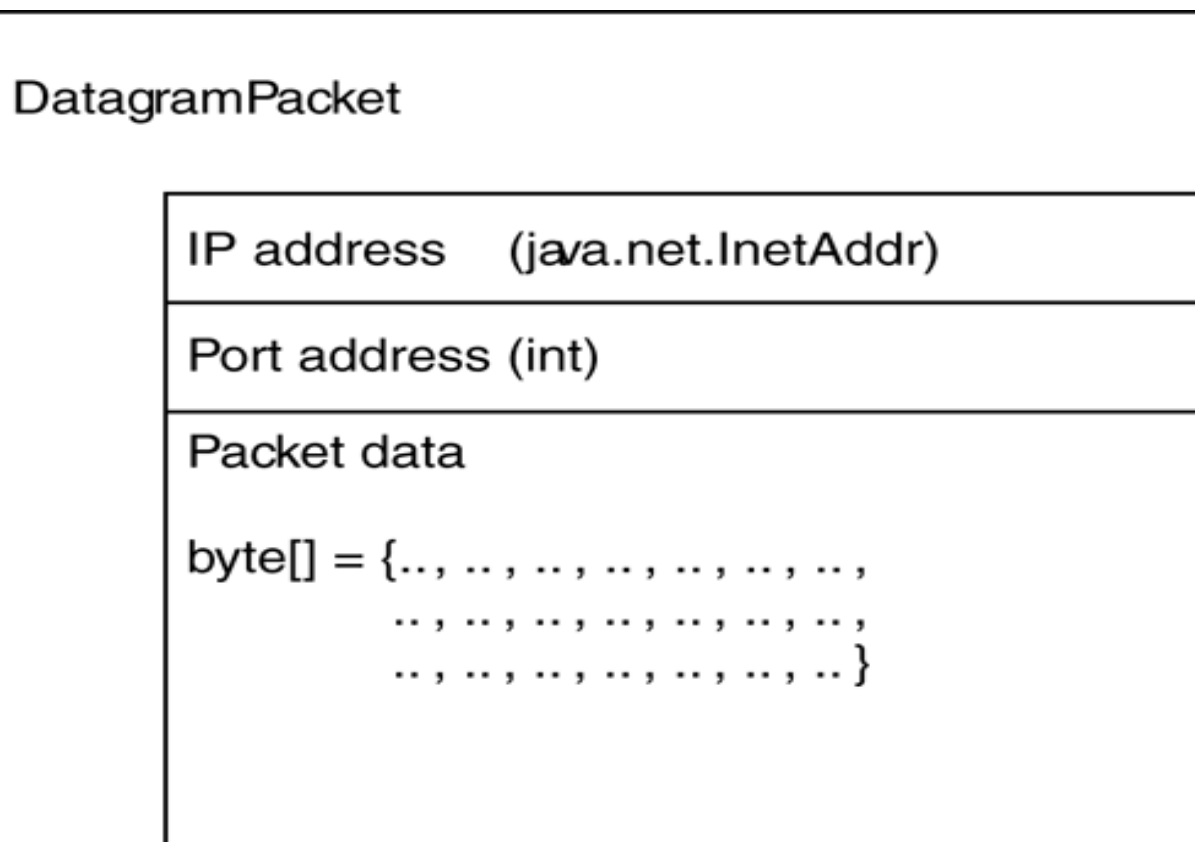
Các bước tạo ứng dụng UDP Socket





Tạo ứng dụng UDP socket với Java

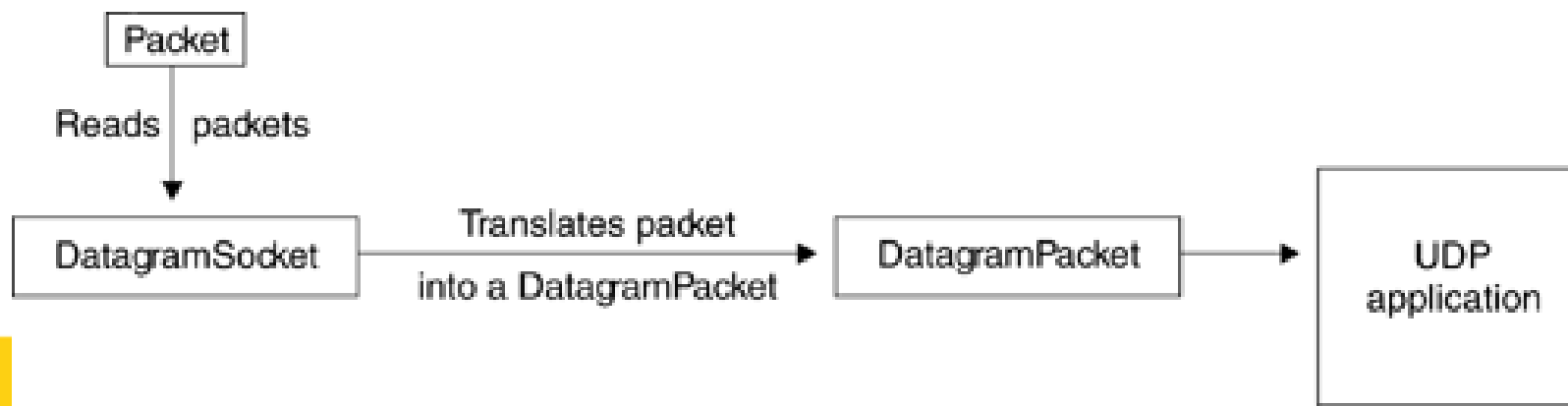
- Sử dụng lớp **DatagramPacket** biểu diễn một packet dữ liệu truyền bằng giao thức UDP.
- Packets chứa dãy các bytes, và các thông tin địa chỉ như IP và số cổng.





Tạo ứng dụng UDP socket với Java

- Tạo DatagramSocket
 - Một **DatagramSocket** có thể dùng để gửi và nhận các packets dữ liệu. Mỗi DatagramSocket gắn đến một cổng, được dùng để đánh địa chỉ cho packet.
- Lắng nghe UDP Packets
 - Trước khi một ứng dụng có thể đọc được các UDP packets được gửi từ máy khác, nó phải gắn một socket đến một cổng UDP sử dụng DatagramSocket và tạo một DatagramPacket mà sẽ dùng như một bộ chứa dữ liệu UDP packet gửi đến.





Tạo ứng dụng UDP socket với Java

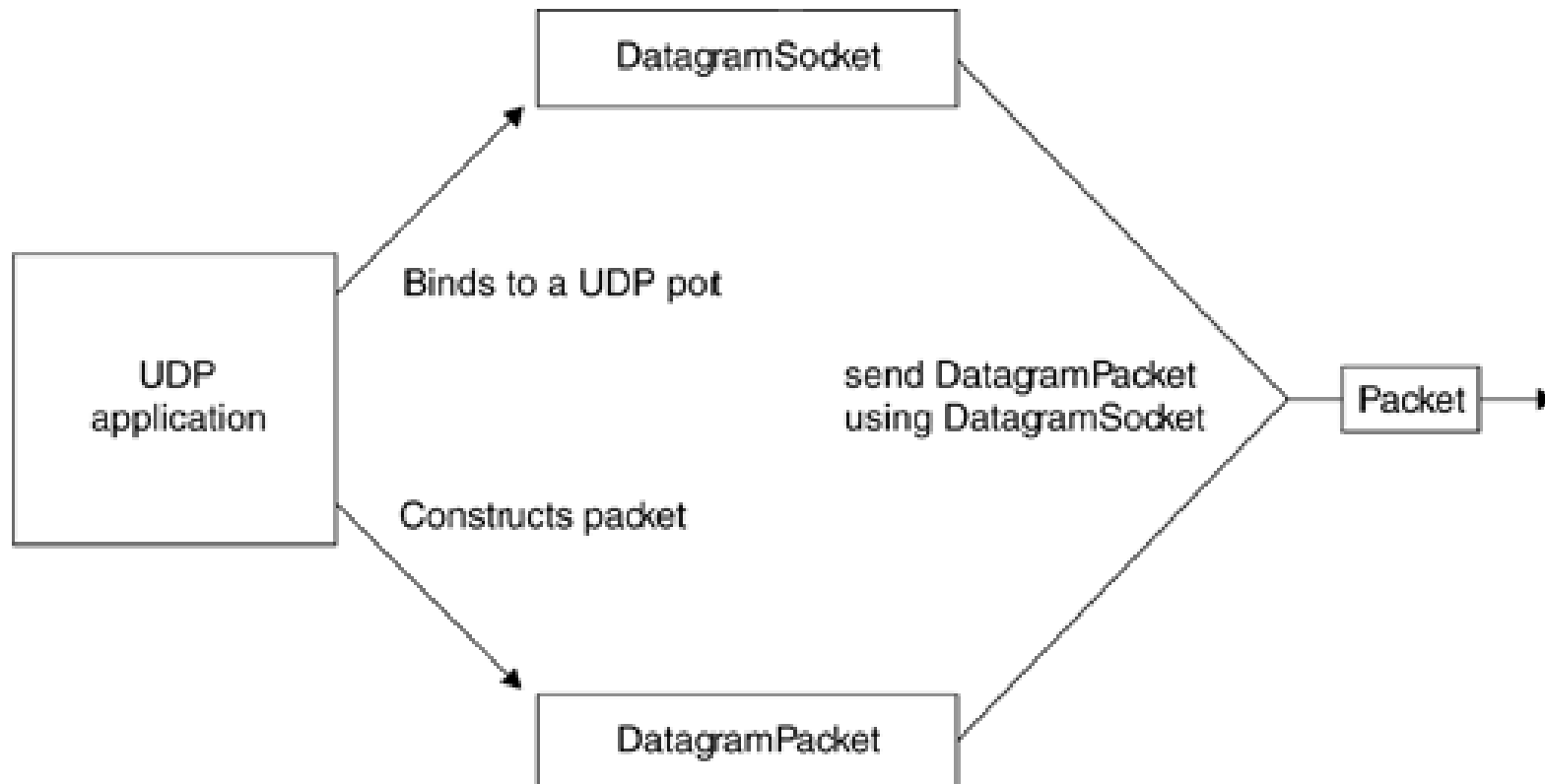
- Đoạn mã Java nhận UDP packet

```
DatagramPacket packet = new DatagramPacket (new byte[256], 256);  
DatagramSocket socket = new DatagramSocket(2000);  
boolean finished = false;  
while (! finished )  
{  
    socket.receive (packet);  
    // process the packet  
}  
socket.close();
```



Gửi UDP packets

- **DatagramSocket** cũng dùng để gửi các packets dữ liệu
- Khi gửi một packet, ứng dụng phải tạo một DatagramPacket, thiết lập địa chỉ và thông tin cổng, và viết dữ liệu muốn gửi dưới dạng dãy các bytes





Đoạn mã Java gửi UDP packets

```
DatagramSocket socket = new DatagramSocket(1500);
DatagramPacket packet = new DatagramPacket ("hello".getBytes());
packet.setAddress ( InetAddress.getByName ( "localhost" ) );
packet.setPort ( 2000 );
boolean finished = false;
while (!finished )
{
    // Write data to packet buffer
    .....
    socket.send (packet);
    // Do something else, like read other packets, or check to
    // see if no more packets to send
    .....
}
socket.close();
```



Ví dụ chương trình Java: UDP Server

```
• import java.io.*;
• import java.net.*;
• class UDPServer
• {
•     public static void main(String args[]) throws Exception
•     {
•         DatagramSocket serverSocket = new DatagramSocket(9876);
•         byte[] receiveData = new byte[1024];
•         byte[] sendData = new byte[1024];
•         while(true)
•         {
•             DatagramPacket receivePacket =
•                 new DatagramPacket(receiveData, receiveData.length);
•             serverSocket.receive(receivePacket);
•             InetAddress IPAddress = receivePacket.getAddress();
•             int port = receivePacket.getPort();
•             sendData = "Text from Server".getBytes();
•             DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, IPAddress, port);
•             serverSocket.send(sendPacket);
•         }
•     }
• }
```



Ví dụ chương trình Java: UDP Client

```
import java.io.*;  
import java.net.*;  
class UDPClient  
{  
    public static void main(String args[]) throws Exception  
    {  
        DatagramSocket clientSocket = new DatagramSocket();  
        InetAddress IPAddress = InetAddress.getByName("localhost");  
        byte[] sendData = new byte[1024];  
        byte[] receiveData = new byte[1024];  
        sendData = "Text from Client".getBytes();  
        DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, IPAddress,  
        9876);  
        clientSocket.send(sendPacket);// gửi dữ liệu  
        DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);  
        clientSocket.receive(receivePacket);// nhận dữ liệu  
        clientSocket.close();  
    }  
}
```



Bài tập lập trình UDP Socket

- Xây dựng chương trình ExchangeRate
 - ExchangeRateServer
 - chạy trên máy chủ và đón nhận dữ liệu do máy khách gửi đến cổng 2345. Khi nhận được yêu cầu, máy chủ sẽ gửi trả các thông báo về tỉ giá kèm theo ngày giờ mới nhất về cho máy khách. Có thể dùng các hàng random để lấy giá trị ngẫu nhiên của 3 thị trường là Tokyo, Newyork, và Hồng Kông
 - ExchangeRateTable
 - Chạy trên máy khách, nó chịu trách nhiệm mỗi giây yêu cầu đến máy chủ để cập nhật thông tin về tỉ giá của thị trường chứng khoán.



Bài tập lập trình UDP Socket

- Hãy tạo chương trình Mail Server.
 - Mỗi khi người dùng tạo account mới tại các máy client, thì chương trình server sẽ tạo một thư mục tương ứng ở máy server, đồng thời tạo một file có tên new_email.txt trong thư mục đó có nội dung như sau :
 - “Thank you for using this service. we hope that you will feel comfortabl.....”
 - Mỗi khi người dùng gửi một email, chương trình server nhận được email đó, xác định gửi đến account nào, tại thư mục của account đó chương trình server sẽ tạo một file có nội dung là nội dung email đó.
 - Mỗi khi người dùng ở máy client đăng nhập vào một account, thì chương trình server phải mở thư mục tương ứng với account đó ở server và gửi tất cả các tên file của thư mục đó về client



Thank your listening