

基于Java的UDP协议程序设计初探 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/252/2021_2022__E5_9F_BA_E4_BA_8EJava_c104_252439.htm 在Java中进行网络编程是相对容易的，因为J2SE中的Java.net包已经对各种通信协议很好的进行了封装，本文主要讲述如何基于UDP(用户数据报)协议编写应用程序。通常我们进行网络编程一般都是使用基于socket的TCP/IP编程，毕竟TCP/IP应用非常的广泛，比如我们浏览互联网就是基于HTTP协议、我们发送邮件是通过SMTP协议。它们都是基于TCP/IP的。TCP/IP的传输最重要的是它可以保证数据到达目的地，而UDP则不同他并不保证准确的传输，数据有可能丢失。如果有兴趣的话，读者可以参考《计算机网络》一书。在介绍UDP编程之前有必要介绍一个重要的类InetAddress，用最简单的一句话描述这个类的作用就是：它代表了一个IP地址。这非常重要在互联网中如果知道了IP地址则意味着我们知道了通信的端点。这个类没有构造器但是有几个工厂方法，通过传递不同的参数例如IP,Hostname等来得到一个InetAddress的实例，下面的这个小例子可以得到我机器的IP地址。

```
import Java.net.*;public class TestNet{ public static void main(String[] args) throws Exception { InetAddress ia = InetAddress.getByName("compaq"); String ipAdr = ia.getHostAddress(); System.out.println(ipAdr); } }
```

当然我的机器的名字为compaq,如果你传入localhost的话将会得到127.0.0.1。接下来开始讲述如何使用UDP编程，这非常容易理解，我们应该首先构造一个数据报，然后把它发送出去，同时我们也可以接收数据报。在Java中提供了DatagramPacket

和DatagramSocket两个类来完成这样的任务，前者负责构造数据报后者负责发送和接收。看看DatagramPacket的构造器DatagramPacket(byte[] buf, int length, InetAddress address, int port) DatagramPacket(byte[] buf, int offset, int length, InetAddress address, int port) DatagramPacket(byte[] buf, int offset, int length, SocketAddress address) DatagramPacket(byte[] buf, int length, SocketAddress address) DatagramPacket(byte[] buf, int length) DatagramPacket(byte[] buf, int offset, int length) 其中前面四个是为构造发送的数据报而用的，因为他们有InetAddress或者SocketInetAddress作为接收端点的地址，后面一个则是为了接受数据报用的。同样我们编写一个C/S模型的例子来说明如何使用这两个重要的类，如果对API还不熟悉请参考Java doc。下面的程序在本机构造一个时间服务器，客户端来取得时间。以前曾经写过类似的时间服务器程序，不过这个是基于UDP的编程。

```
import java.io.*;import java.net.*;import java.util.*;public class TimeServer { final private static int DAYTIME_PORT = 13; public static void main(String args[]) throws IOException { DatagramSocket socket = new DatagramSocket(DAYTIME_PORT); while (true) { byte buffer[] = new byte[256]; DatagramPacket packet = new DatagramPacket(buffer, buffer.length); socket.receive(packet); String date = new Date().toString(); buffer = date.getBytes(); // Get response address/port // for client from packet InetAddress address = packet.getAddress(); int port = packet.getPort(); packet = new DatagramPacket(buffer, buffer.length, address, port); socket.send(packet); } }import java.io.*;import java.net.*;public
```

```
class GetTime { final private static int DAYTIME_PORT = 13.  
public static void main(String args[]) throws IOException { if  
(args.length == 0) { System.err.println("Please specify daytime  
host"). System.exit(-1). } String host = args[0]. byte message[] = new  
byte[256]. InetAddress address = InetAddress.getByName(host).  
System.out.println("Checking at: " address). DatagramPacket packet  
= new DatagramPacket(message, message.length, address,  
DAYTIME_PORT). DatagramSocket socket = new  
DatagramSocket(). socket.send(packet). packet =new  
DatagramPacket(message, message.length). socket.receive(packet).  
String time = new String(packet.getData()). System.out.println(The  
time at " host " is: " time). socket.close(). }}  
100Test 下载频道开通  
， 各类考试题目直接下载。 详细请访问 www.100test.com
```