

## § 13.4 UDP数据报

- 基于UDP的通信和基于TCP的通信不同，基于UDP的信息传递更快，但不提供可靠性保证。
- 基于UDP通信的基本模式是：
  - 1. 将数据打包，称为数据包（好比将信件装入信封一样），然后将数据包发往目的地。
  - 2. 接受别人发来的数据包（好比接收信封一样），然后查看数据包中的内容。

## § 13.4.1 发送数据包

- 1. 用DatagramPacket类将数据打包，即用DatagramPacket类创建一个对象，称为数据包。  
用DatagramPacket的以下两个构造方法创建待发送的数据包：
  - `DatagramPacket(byte data[], int length, InetAddress address, int port)`
  - `DatagramPack(byte data[], int offset, int length, InetAddress address, int port)`
- 2. 用DatagramSocket类的不带参数的构造方法：`DatagramSocket()` 创建一个对象，该对象负责发送数据包。例如：
  - `DatagramSocket mail_out=new DatagramSocket();`
  - `mail_out.send(data_pack);`



## § 13.4.2 接收数据包

- 1. 首先用DatagramSocket的另一个构造方法DatagramSocket(int port) 创建一个对象, 其中的参数必须和待接收的数据包的端口号相同。例如, 如果发送方发送的数据包的端口是5666, 那么如下创建DatagramSocket对象:
  - DatagramSocket mail\_in=new DatagramSocket(5666);
- 2. 然后对象mail\_in使用方法receive(DatagramPacket pack)接受数据包。
- 3. 用 DatagramPack类的另外一个构造方法: DatagramPack(byte data[], int length) 创建一个数据包, 用于接收数据包, 例如:

```
byte data[]=new byte[100];
```

```
int length=90;
```

```
DatagramPacket pack=new DatagramPacket(data, length);
```

```
mail_in.receive(pack);
```

- 该数据包pack将接收长度是length字节的数据放入data。

## § 13.4.2 接收数据包-例题

- 例子5中：张山和李四使用用户数据报（可用本地机模拟）互相发送和接收数据包，程序运行时“张三”所在主机在命令行输入数据发送给“李四”所在主机，将接收到的数据显示在命令行的右侧（效果如图13.8）；同样，“李四”所在主机在命令行输入数据发送给“张三”所在主机，将接收到的数据显示在命令行的右侧（效果如图13.9）。

“张三”主机 (ZhanSan.java ,  
ReceiveLetterForZhang.java )

```
输入发送给李四的信息:      收到:how are you  
I am fine  
继续输入发送给李四的信息:
```

图 13.8 “张三”主机

“李四”主机 (LiSi.java ,  
ReceiveLetterForLi.java )

```
输入发送给张三的信息:how are you  
继续输入发送给张三的信息:      收到:I am fine
```

图 13.9 “李四”主机