# 网络编程概述

## 计算机网络

是指将地理位置不同的具有独立功能的多台计算机及其外部设备，通过通信线路连接起来，在网络操作系统，网络管理软件及网络通信协议的管理和协调下，实现资源共享和信息传递的计算机系统。

## 网络编程

就是用来实现网络互连的不同计算机上运行的程序间可以进行数据交换。

## 网络模型

网络模型一般是指OSI（Open System Interconnection开放系统互连）参考模型，TCP/IP参考模型



网络模型7层概述：

1.物理层：

主要定义物理设备标准，如网线的接口类型、光纤的接口类型、各种传输介质的传输速率等。它的主要作用是传输比特流（就是由1、0转化为电流强弱来进行传输,到达目的地后在转化为1、0，也就是我们常说的数模转换与模数转换）。这一层的数据叫做比特。

2. 数据链路层：

主要将从物理层接收的数据进行MAC地址（网卡的地址）的封装与解封装。常把这一层的数据叫做帧。在这一层工作的设备是交换机，数据通过交换机来传输。

3. 网络层：

主要将从下层接收到的数据进行IP地址（例192.168.0.1)的封装与解封装。在这一层工作的设备是路由器，常把这一层的数据叫做数据包。

4. 传输层：

定义了一些传输数据的协议和端口号（WWW端口80等），如：TCP（传输控制协议，传输效率低，可靠性强，用于传输可靠性要求高，数据量大的数据），UDP（用户数据报协议，与TCP特性恰恰相反，用于传输可靠性要求不高，数据量小的数据，如QQ聊天数据就是通过这种方式传输的）。 主要是将从下层接收的数据进行分段和传输，到达目的地址后再进行重组。常常把这一层数据叫做段。

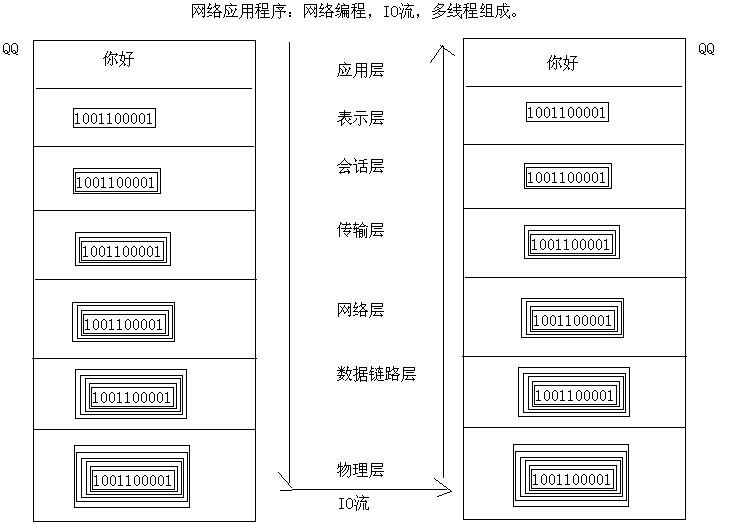
5.会话层：

通过传输层（端口号：传输端口与接收端口）建立数据传输的通路。主要在你的系统之间发起会话或者接受会话请求（设备之间需要互相认识可以是IP也可以是MAC或者是主机名）

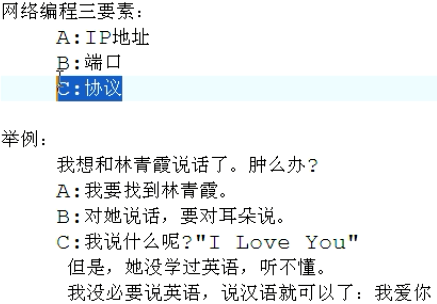
6.表示层：

主要是进行对接收的数据进行解释、加密与解密、压缩与解压缩等（也就是把计算机能够识别的东西转换成人能够能识别的东西（如图片、声音等）。

7.应用层： 主要是一些终端的应用，比如说FTP（各种文件下载），WEB（IE浏览），QQ之类的（可以把它理解成我们在电脑屏幕上可以看到的东西．就是终端应用）。



# 网络编程三要素



## IP地址:InetAddress

网络中设备的标识，不易记忆，可用主机名

计算机只能识别二进制的数据，所以我们的IP地址应该是一个二进制的数据。

但是呢，我们配置的IP地址确不是二进制的，为什么呢?

IP：192.168.1.100

换算：11000000 10101000 00000001 01100100

假如真是：11000000 10101000 00000001 01100100的话。

我们如果每次再上课的时候要配置该IP地址，记忆起来就比较的麻烦。

所以，为了方便表示IP地址，我们就把IP地址的每一个字节上的数据换算成十进制，然后用.分开来表示：

"点分十进制"

### IP地址的组成：

网络号段+主机号段

A类：第一号段为网络号段+后三段的主机号段

一个网络号：256\*256\*256 = 16777216

B类：前二号段为网络号段+后二段的主机号段

一个网络号：256\*256 = 65536

C类：前三号段为网络号段+后一段的主机号段

一个网络号：256

### IP地址的分类：

A类 1.0.0.1---127.255.255.254 (1)10.X.X.X是私有地址(私有地址就是在互联网上不使用，而被用在局域网络中的地址) (2)127.X.X.X是保留地址，用做循环测试用的。

B类 128.0.0.1---191.255.255.254 172.16.0.0---172.31.255.255是私有地址。169.254.X.X是保留地址。

C类 192.0.0.1---223.255.255.254 192.168.X.X是私有地址

D类 224.0.0.1---239.255.255.254

E类 240.0.0.1---247.255.255.254

### DOS命令及特殊ip：

ipconfig 查看本机ip地址

ping 后面跟ip地址。测试本机与指定的ip地址间的通信是否有问题

127.0.0.1 回环地址(表示本机)

x.x.x.255 广播地址

x.x.x.0 网络地址

### InetAddress类

没有构造方法，那么如何使类提供的功能呢？

要掌握的功能

获取任意主机：getByName

主机名：getHostName

主机Ip地址：getHostAddress

**import** java.net.InetAddress;

/\*

\* 如果一个类没有构造方法：

\* A:成员全部是静态的(Math,Arrays,Collections)

\* B:单例设计模式(Runtime)

\* C:类中有静态方法返回该类的对象(InetAddress)

\* class Demo {

\* private Demo(){}

\*

\* public static Demo getXxx() {

\* return new Demo();

\* }

\* }

\*

\* 看InetAddress的成员方法：

\* public static InetAddress getByName(String host):根据主机名或者IP地址的字符串表示得到IP地址对象

\*/

**public** **class** TestInetAddress {

**public** **static** **void** main(String[] args) **throws** Exception {

InetAddress inetAddress = InetAddress.*getByName*("192.168.1.2");

String name = inetAddress.getHostName();

String ip = inetAddress.getHostAddress();

System.*out*.println(name + "----" + ip);

}

}

## 端口号

用于标识进程的逻辑地址，不同进程的标识

物理端口 网卡口

逻辑端口 我们指的就是逻辑端口

A:每个网络程序都会至少有一个逻辑端口

B:用于标识进程的逻辑地址，不同进程的标识

C:有效端口：0~65535，其中0~1024系统使用或保留端口。

通过360可以查看端口号

## 传输协议

通讯的规则

常见协议：TCP，UDP

### UDP

将数据源和目的封装成数据包中，不需要建立连接；

每个数据报的大小在限制在64k；

因无连接，是不可靠协议；

不需要建立连接，速度快

### TCP

建立连接，形成传输数据的通道；

在连接中进行大数据量传输；

通过三次握手完成连接，是可靠协议；

必须建立连接，效率会稍低

# Socket

## Socket套接字：

网络上具有唯一标识的IP地址和端口号组合在一起才能构成唯一能识别的标识符套接字。

## Socket原理机制：

通信的两端都有Socket。

网络通信其实就是Socket间的通信。

数据在两个Socket间通过IO传输

## Socket机制图解

