

网络编程概述

- Java是 Internet 上的语言，它从语言级上提供了对网络应用程序的支持，程序员能够很容易开发常见的网络应用程序。
- Java提供的网络类库，可以实现无痛的网络连接，联网的底层细节被隐藏在 Java 的本机安装系统里，由 JVM 进行控制。并且 Java 实现了一个跨平台的网络库，程序员面对的是一个统一的网络编程环境。

网络基础

● 计算机网络：

把分布在不同地理区域的计算机与专门的外部设备用通信线路互连成一个规模大、功能强的网络系统，从而使众多的计算机可以方便地互相传递信息、共享硬件、软件、数据信息等资源。

● 网络编程的目的：

直接或间接地通过网络协议与其它计算机进行通讯。

● 网络编程中有两个主要的问题：

- 如何准确地定位网络上一台或多台主机
- 找到主机后如何可靠高效地进行数据传输。

● 如何实现网络中的主机互相通信：

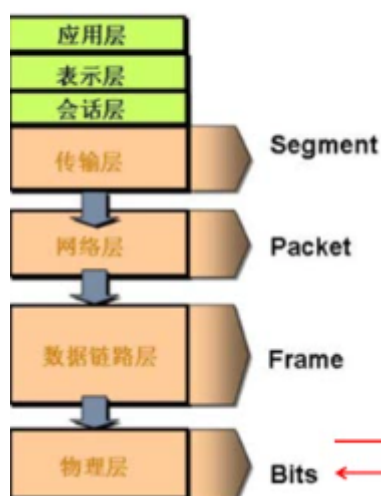
➢ 通信双方地址

➢ 一定的规则（有两套参考模型）

- ✓ OSI参考模型：模型过于理想化，未能在因特网上进行广泛推广
- ✓ TCP/IP参考模型(或TCP/IP协议)：事实上的国际标准。

OSI参考模型	TCP/IP参考模型	TCP/IP参考模型 各层对应协议
应用层	应用层	HTTP、ftp、 telnet、DNS...
表示层		
会话层		
传输层	传输层	TCP、UDP、...
网络层	网络层	IP、ICMP、ARP...
数据链路层	物理+数据链路层	Link
物理层		

数据封装



数据拆封



通讯要素1: IP 和 端口号

IP 地址: InetAddress

- 唯一的标识 Internet 上的计算机
- 本地回环地址(hostAddress): 127.0.0.1 主机名(hostName): localhost
- 不易记忆

端口号标识正在计算机上运行的进程（程序）

- 不同的进程有不同的端口号
- 被规定为一个 16 位的整数 0~65535。其中，0~1023被预先定义的服务通信占用（如MySQL占用端口3306，http占用端口80等）。除非我们需要访问这些特定服务，否则，就**应该使用 1024~65535 这些端口中的某一个进行通信，以免发生端口冲突。**

端口号与IP地址的组合得出一个网络套接字。