网络通信及网络分层初识

|  |  |
| --- | --- |
| 编 制 人： | Lizy |
| 发布日期： | 2023.3.3 |
| 页 数： | 13 |
| [一篇文章，只用看三遍，终生不忘网络分层 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/380119935) | |

更改记录

|  |  |
| --- | --- |
| **发布日期** | **更改描述** |
| 20230303 | 第一版发布 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# 前言

网络分层、协议、网络通信是什么？

# 网络、通信、协议

网络是什么？网络是用物理链路将工作站与主机相连在一起，组成数据链路，从而达到资源共享和通信的目的。这里的物理链路不仅仅是指我们能够看到的双绞线、光纤，也可能是无线电波。

网络通信，狭义的讲，可以理解为计算机与计算机之间基于网络进行的数据交换。

就像人与人沟通一样，既然想“聊天”，起码得有一个基本条件，比如都使用汉语或英语，这算是一个比较宽泛的协议了。而计算机通信中，也是约定好交互时的数据格式、数据项的意思，收到后如何处理或回应。

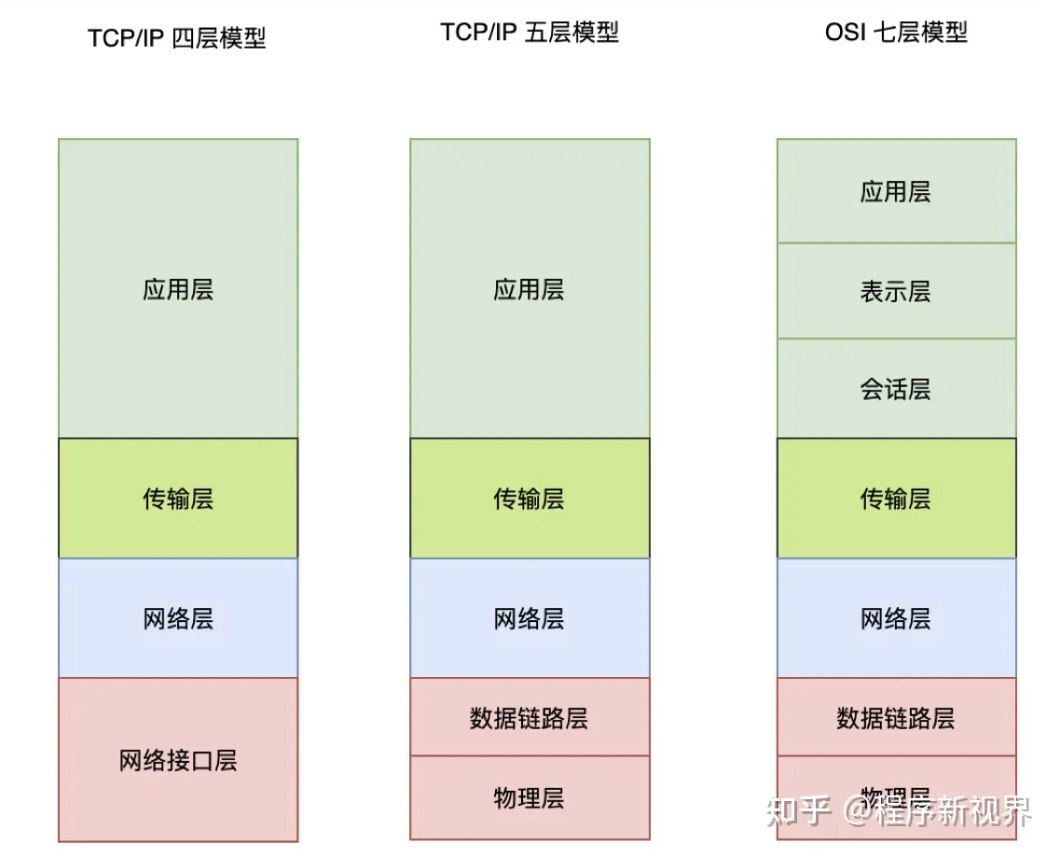
简单来说，互联网协议就是在互联网上传输数据的规则。当然，协议非常多，比如TCP、UDP、IP协议、FTP协议等等。而使用这些协议的最基本要求就是发送方和接收方所使用的协议必须一致，否则就是鸡同鸭讲。

当然我们说几层协议时，一般来说：一台设备上的第X层与另一台设备上的第X层进行通信的规则就是第X层协议。

# 网络分层模式

就像一家公司，有老板、经理、组长、成员一样，不同层面的人有不同的沟通方式，但最终还是从上而下的进行实时。老板与老板的沟通就相当于应用层之间的沟通，而老板的目标又需要经下面经理的支持，就像应用层需要传输层的支持一样。

网络分层有几种方式。标准的七层网络分层，也就是OSI七层模型。TCP/IP五层模型和TCP/IP四层模型事从OSI七层优化而来的。



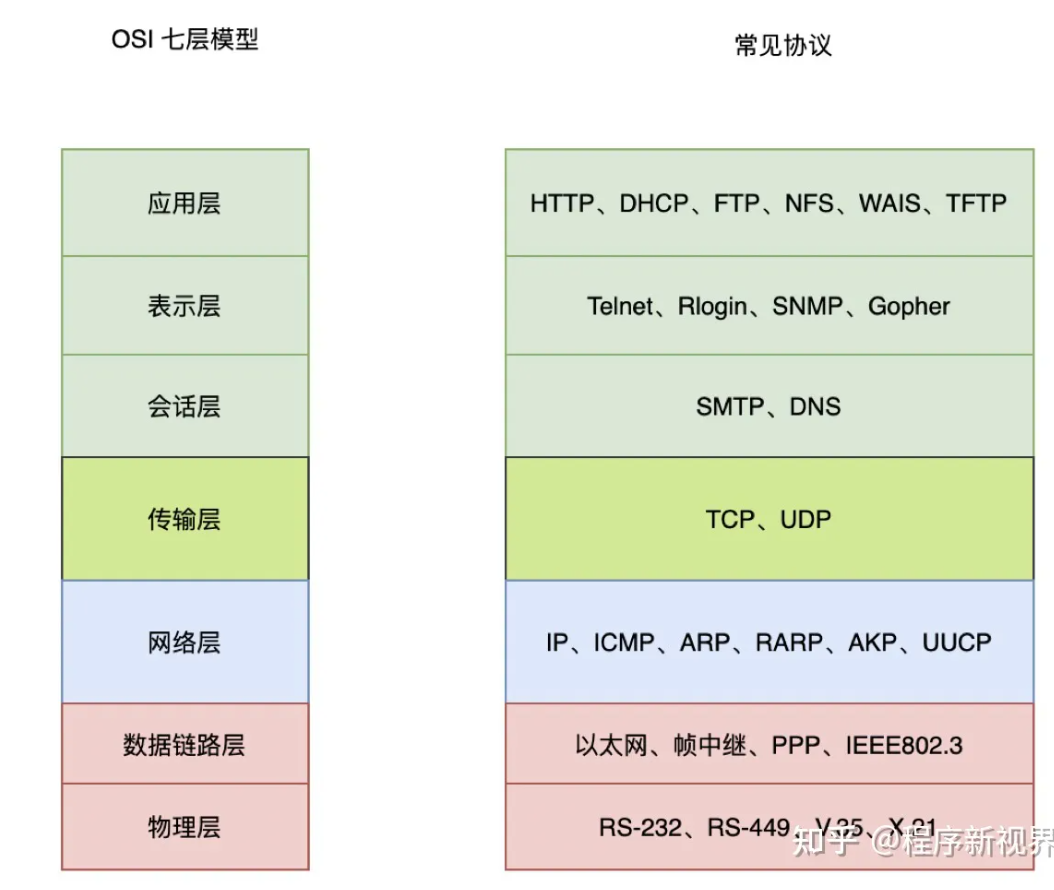
通常，作为用户，使用最多的就是应用层，大多数情况下可能感知不到其他层的存在。

那么，为什么要分层呢？这与设计软件架构的思想差不多，要具有高内聚、低耦合、复用、扩展性等特性。分层后，可以将不同的功能模块化，修改也只用修改对应的模块即可。而且，不同层还可以被复用，只要确保按照这个层的协议来处理就可以了。

后面，主要针对TCP/IP五层模型来逐个讲解。

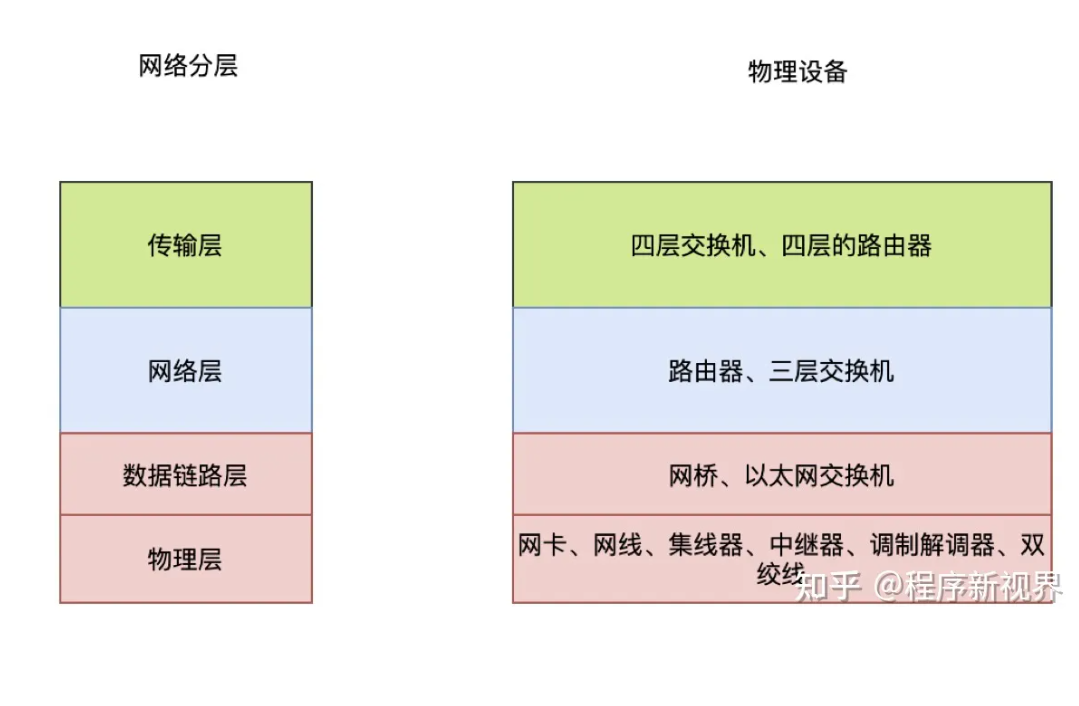
## 网络分层与协议

上面了解了协议和网络分层，这里通过一张图来对照一下常见的协议属于哪个层。



## 网络分层与物理设备

通过一张图，看一下不同网络分层通常对应的硬件设备：



## 物理层

下面看看每层的作用，通过功能的相关性来记忆网络分层是非常好的记忆方法。首先看一下物理层。

这一层就是把各个网络设备连接起来，让其可以传输0 1 0 1 的电信号。

物理层，顾名思义，用物理的手段将电脑连接起来，基本是用双绞线、光纤、无线电波的方式来实现物理层。网络设备连接起来后，就可以基于它来发送高低电平进行通信，高电压对应数字1，低电平对应数字0。0 1 信号本身没有任何现实意义，所以需要用另一层来规定不同0 1 组合的意义。

## 数据链路层

单纯的电信号0 1 没有任何意义，必须规定电信号多少位一组，每组什么意思。数据链路层的功能就是通过规定一套协议来定义电信号的分组方式，以及规定不同组代表什么意思，从而双方计算机都能够进行识别，这个协议就是“以太网协议”。

以太网协议规定，一组电信号构成一个数据包，我们把这个数据包称之为帧。每一个帧由包头（Head）和数据（Data）两部分组成。



其中Head包含数据包的一些说明信息，包括发送者、接收者、数据类型；Data则是数据包的具体内容。

Head部分包含固定的18个字节：

* 发送者/源地址，6个字节；
* 接收者/目标地址，6个字节；
* 数据类型，6个字节。

Data部分的长度，最短为46字节，最长为1500字节。因此，整个“帧”最短为64字节，最长为1518字节。如果数据很长，就必须分割成多个帧进行传输。

有了数据包的定义，那么计算机是如何标识谁是谁呢？以及如何知道对方的地址呢？这就涉及到MAC地址和广播。

### MAC地址（物理地址）

以太网规定，连入网络的所有设备都必须具有网卡。数据包的发送地址和接收地址指的就是网卡的地址，即MAC地址。

MAC地址作为网络中计算机设备的唯一标识，从计算机在厂商生产出来就被十六进制的数标识为MAC地址，MAC地址理论上是独一无二的。这也是为什么很多软件校验设备唯一性时，会用到MAC地址。

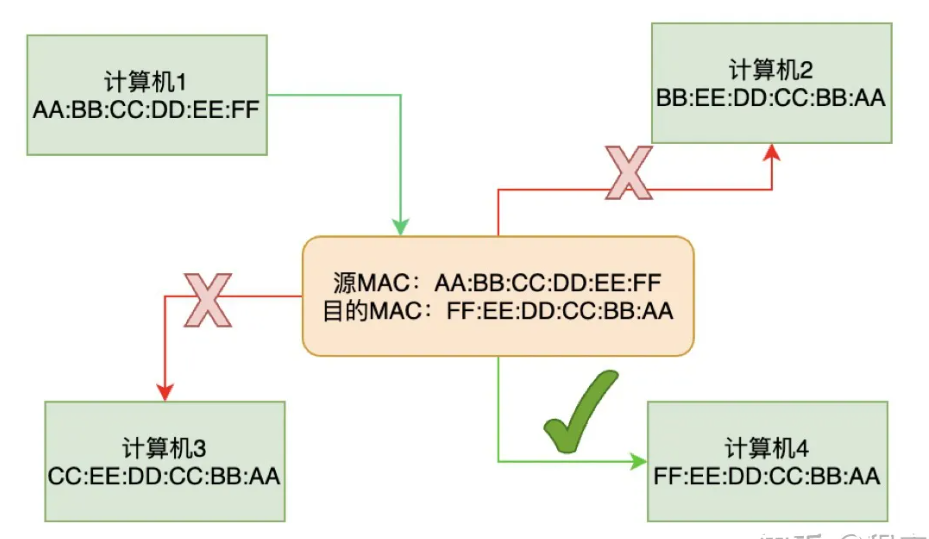
MAC地址长度为48位2进制，通常由12位16进制数表示（前六位是厂商编号，后六位是流水线号）。



有了MAC地址，那么两台设备之间是如何找到对方进行通信的呢？

### 广播

有了MAC地址，在同一网络内的两台主机就可以通信了。



在同一子网中，计算机1要想计算机4发送一个数据包，数据包中包含接收方的MAC地址。计算机1向本网络中的所有计算机都发送（以广播的方式），这时同一子网中的每台计算机（包括2、3）都会收到这个数据包。然后每台计算机都会把数据包的MAC地址取出来，与自身的MAC地址进行对比，如果两者相同，则接受这个数据包，否则就丢弃这个数据包。计算4发现与自身MAC地址匹配，于是进行响应。

## 网络层

以太网通过广播这种原始的形式，解决了两台计算机的通信问题。但是明显，它不是把数据包准确的送达接收方，而是向网络中所有的计算机发送数据包。

而我们所处的网络是由无数个子网络构成的。即便忽略掉子网的存在，如果所有的通信都采用以太网的广播方式，那么一台计算机发送的数据包全世界都会收到，整个网络是会崩溃的。

而现实的网络又是由无数个子网络构成的，以太网采用的广播方式发送数据包，效率低且发送的数据只能局限在发送者所在的子网络中。

此时就需要另想办法，如果是同一个子网络，就采用广播的方式发送，如果不在同一子网中，那么就可以通过网关和路由器向不同广播域/子网分发数据包。所以就有了网络层，它是处理分组在网络中的活动，比如分组选路。

网络层引入一套新的协议用来区分不同广播域/子网，于是有了IP协议。

### IP协议

网络地址协议，叫做IP协议。相对于上面的MAC地址，IP地址可以理解为逻辑地址，也就是MAC地址是物理地址，是固定的。IP地址是动态分配的，不固定。现在广泛采用的是IPv4地址，同时IPv6也在不断发展壮大。

通过IP协议发送的数据，就叫做IP数据包，也分为“包头”和“数据”两部分：“包头”部分主要包括版本、长度、IP地址等信息；“数据”部分则是IP数据包的具体内容。IP数据包的“包头”部分的长度为20到60字节，整个数据包的总长度最大为65535字节。

IPv4地址是由32位二进制数组成，一般把它分为4段的十进制表示，地址范围为：0.0.0.0 ~ 255.255.255.255。

IP地址分为两个部分：网络部分（标识子网）和主机部分（标识主机）。网络部分和主机部分所占用的二进制位数是不固定的。

IP地址段只是标识了IP地址的种类，从网络部分或主机部分都无法辨识一个IP所处的子网。如果两个IP的网络部分相同，则说明他们处于同一子网中。例如192.168.33.1和192.168.33.2，如果他们的网络部分为24位，主机部分为8位，网络部分都为192.168.33，处于同一子网中。

但如果像192.16.10.1和192.16.10.2，并不知道网络部分和主机部分各几位，就不能确定是否处于同一子网中。于是有了子网掩码。

### 子网掩码

子网掩码就是用来标识同一局域网中的IP地址信息的。子网掩码也是由32个二进制位组成的，但只能用0或1表示，如11111111.11111111.11111111.00000000，写成十进制就是255.255.255.0。

假设192.168.33.1和192.168.32.2的子网掩码都是255.255.255.0，把IP与子网掩码相与（&），结果都是为192.168.33.0，那么说明他们处于同一子网中。

### ARP协议

有了IP协议，可以判断计算机属于哪个子网了。在链路层时我们通过MAC地址进行通信的，现在只有目标的IP地址，如何才能拿到他的MAC地址呢？这就又要用到ARP协议。

ARP协议可以通过广播的方式发送数据包，获取目标主机的MAC地址。具体实现为：发送端主机通过广播的方式发送数据包，所有主机接收后拆开包，如果发现目标IP为自己就回应，返回MAC地址。

### 数据包格式

发送端MAC地址，目标MAC地址，发送端IP地址，目标IP地址，数据



## 传输层

经过数据链路层和网络层的支持，我们已经可以正常在两台计算机之间进行通讯了，但是计算机会同时运行着许多程序，比如同时开着QQ与WX，那么怎么区分消息是QQ还是WX呢？

通常我们会看到计算机上运行的不同程序都会分配不同的端口，而传输层的功能就是建立端口到端口的通信，使得数据能够正确的传送给不同的应用程序。

端口是每一个使用网卡程序的编号，每一个端口号标志一个可执行程序。端口号是0-65535之间的一个整数，正好16个二进制位，0-1023为系统占用端口，其他应用程序只能选用大于1023的端口。

至此，用IP+端口，已经能实现唯一确定互联网上一个程序，进而实现网络间的程序通信了。

传输层最常见的两大协议是TCP（Transmission Control Protocol，传输控制协议）协议和UDP（User Data Protocol ，用户数据包协议）协议。

UDP协议就是在数据前面加上端口号，“包头”部分主要定义了发送端口和接收端口，一共只有8个字节。“数据”部分就是具体的内容。总长度不超过65535字节，正好放进去一个IP数据包。

UDP协议简单且容易实现，这是它的优点。但同样也有缺点，就是进行通信时不知道对方是否接收到数据了。此时需要再定义一套规则，让其可以和对方确认，那么TCP出现了。

TCP数据包没有长度限制，理论上可以无限长，可是为了保证网络的效率，通常TCP数据长度不会超过IP数据包的长度，也就是小于65535字节，这也确保单个TCP数据包不必再分割。像我们通常说的TCP三次握手和四次握手，就是传输层完成的。关于TCP实现部分的内容比较多，这里就不再扩展了。

TCP的优点是能够确保数据不会遗失，安全可靠的传输。缺点是过程复杂、实现困难、消耗较多的资源。

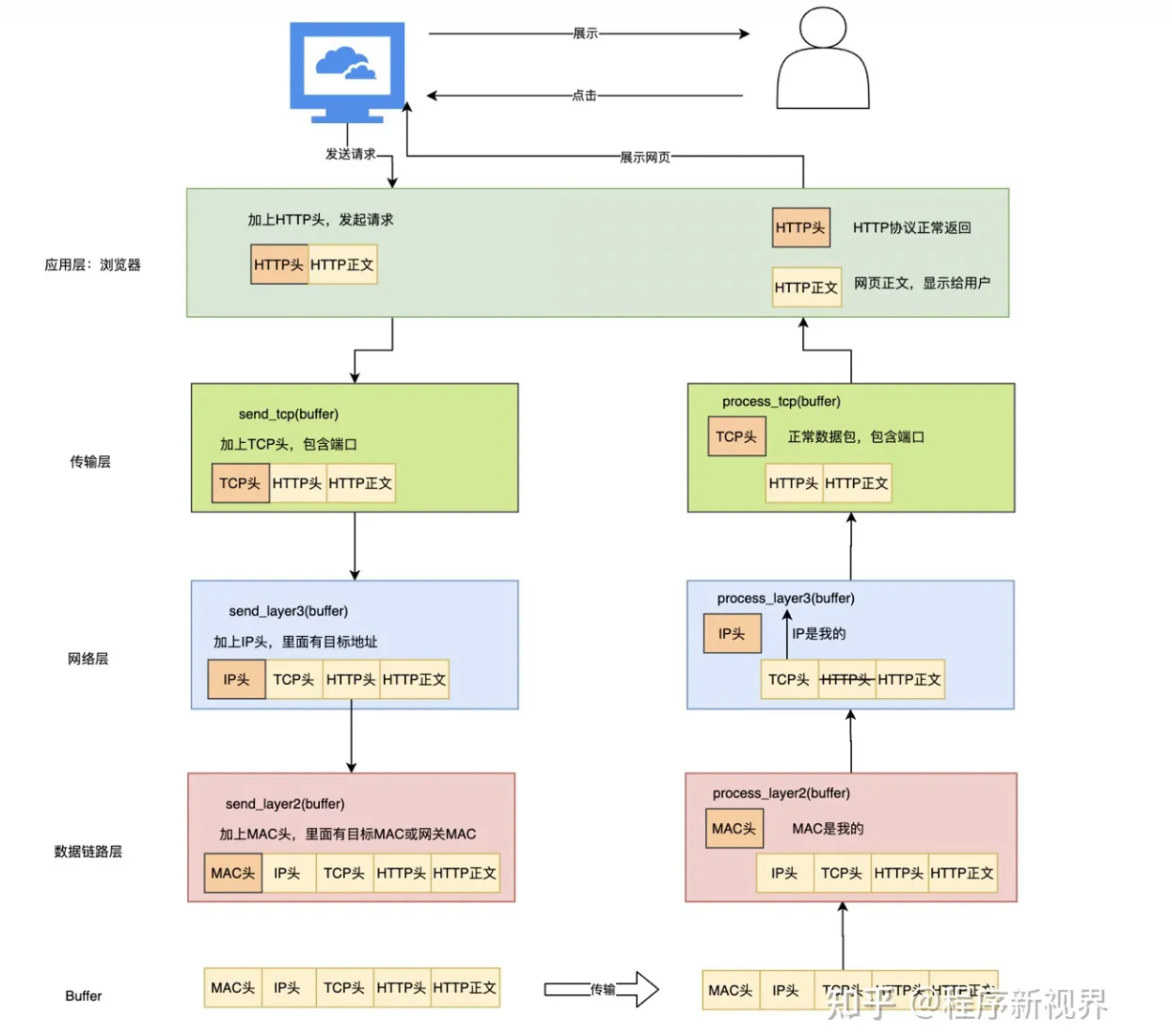
## 应用层

应用层的功能就是规定了应用程序的数据格式。我们经常用到的电子邮件、HTTP协议以及FTP协议数据的格式，就是在应用层定义的。

应用层的协议有：FTP（文件传送协议、21端口）、Telnet（远程登录协议）、DSN（域名解析协议）、SMTP（邮件传送协议、25端口）、POP3协议（邮局协议）、HTTP协议（Hyper Text Transfer Protocol、80端口）。

## 正反俄罗斯套娃

以HTTP请求为例，梳理整个网络分层中各层的职责和处理流程。在整个过程中，就像俄罗斯套娃一样，一层层的包装起来，然后再一层层的打开。



# 小结

通过这篇文章，我们从最下方的物理层到最上方的应用层，按照故事性的形式进行了讲解。