

计算机网络概览

作者: [Aizen](#)

1 前言

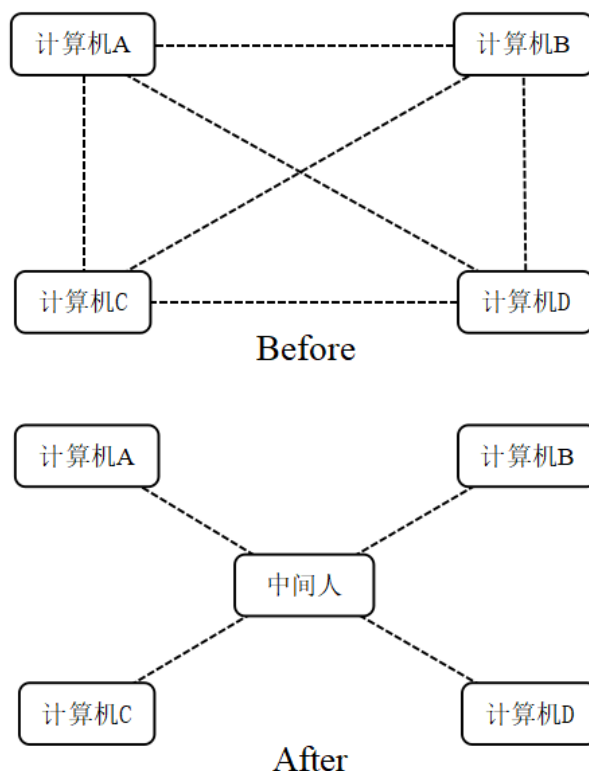
我知道很多人似乎没有去听计网的课程，所以我这里编写一个《计算机网络概览》让各位对计算机网络有一个总体的概念，这个概念主要在于解释为什么计网有这些东西，或者这些东西有什么用。

2 计网的由来

早期的时候，创建一个独立的计算机已经算是一件很简单的事情了，困难的是计算机之间如何连接起来。为了计算机之间的数据传输，计算机厂商给计算机开了一个接口，可以使用一根连线连接两个计算机接口，将它们连接在一起，数据就可以通过这条连线，从计算机A传输到计算机B。

我们知道，数据是01的，称为“数据流”，计算机A输出的时候是按照数据流输出，计算机B读取的时候也是数据流读取，但是读取了数据然后呢？怎么保证数据的表示的什么。例如，计算机A这里的数据流“0010 0010”代表“lbwnb”，那么计算机B读取的时候，只能知道数据流是“0010 0010”，并不知道数据到底表示什么，这时候就出现了“协议”（Protocol），代表我输出的东西就是按照某种数据规则来输出的，那么只要计算机A和计算机B都遵循这个协议，具体的数据信息就能够准确地传输了。

既然可以将两个计算机连接起来，那么就可以使用多个线将多个主机连接起来，但是有一个问题：连接两个计算机还可以，1条线；三个计算机也能接受，3条线；四个也能接受；但是一多，我去， $n(n-1)/2$ 条，这算是平方增长了。那么就有人提出：“不用，我专门搞一个计算机，其他计算机连接到这个计算机，这个计算机就主要用来数据传输就行，其他所有计算机，只连接这一个就行”。这样的想法，让原本计算机之间的“1-1”连接方式，转移成了“1-中间人-1”“中间人-n”的模式。后来为了让“中间计算机”专门做这件事情，就让它专门搞信息转换，这个主机逐渐进化，最后发展成了“路由器”。



原本，这些连接是在某一个大学里面搞的，然后只要大学里面设置了这种计算机结构，就可以实现学校内的计算机之间的通信。

连接了之后，然后呢？传输，传输什么？传输很多东西，只要能够传输“数据流”，然后按照某种规律读取这些数据流，就能够传输信息。发散你的思维，你觉得学生A在校园的北边，学生B在校园的南边，这两个人能干嘛？发消息对吧；再者，多个学生在其他地方，一个计算机在图书馆，能干嘛？把书的内容放到图书馆的计算机，大家都可以直接从这个计算机下载内容。这样，计算机网络雏形就诞生了！

3 计网的知识体系发展

随着技术的发展，其实如今网络和开始的网络结构没有什么区别，但是随着人们对计网的研究和分析，计网也成为了一个复杂的学科。

早期，网络没有一个完全的规范，大家都按照各自的规范传东西，而且每个计算机厂商都有各自的规范，不同品牌之间的计算机传输个数据，假如不支持同一个“协议”，那就没辙了。所以，那时候有一个协会，通过科学的验证和分析（哈哈，这里其实有一个笑话），将网络分为了不同的层级，也就是著名的“OSI网络体系”（Open System Inerconnection）。将整个计算机网络分为了不同的层级，每个层级都可单独发展，互不干涉。其中分为了：物理层、数据链路层、网络层、传输层、会话层、表示层、应用层。每一个层级都有指定的协议，只要遵循指定的协议，就能够实现同层之间的正常通信。

我来分析各个部分，来告诉你们为什么这样分。

3.1 物理层

物理层，是最偏向于数据传输的一层。就是刚开始所说的“连接两个计算机的线”所在的一层，它就只负责传输01数据流，而不在乎数据是什么，存在什么问题。到了现在的21世纪，已经可以实现无限传输了，两个主机也不需要什么线来连接了。还有早些年一直能听到的“光纤技术”，就是通过光的频率来传输数据。

我举一个例子，你可以这样理解，在韩国电影《寄生虫》的最后，穷人丈夫被迫潜逃在富人的地下室里，他碰巧能够在地下室控制灯泡的闪烁频率，他的儿子偶尔去山上，望向富人。后来他每天都尝试通过控制灯泡的闪烁频率提供01流来告诉他的儿子。这其实就很像所谓的光纤技术，但是实际上更加复杂一些。

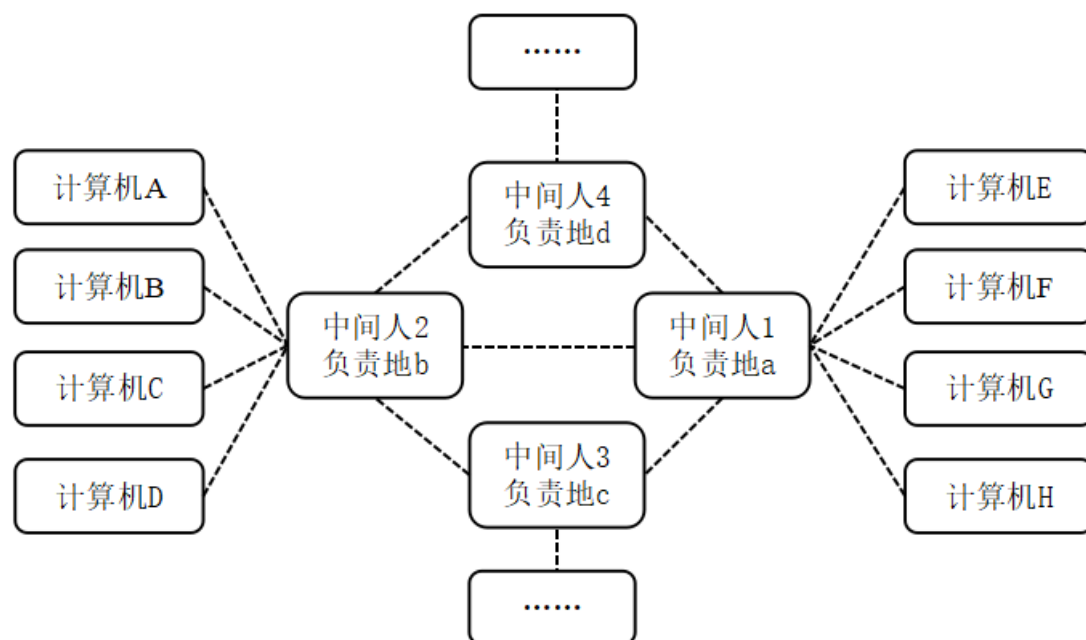
3.2 数据链路层

继续沿用之前的例子“0010 0010”，计算机A传输给计算机B的时候，那根线可能存在一些瑕疵，出现误差，导致计算机B得到的数据变成了“0011 0010”，这时候数据出现错误了，咋办？当时人们发明了一个校验码的方式，数据发送的时候还会把校验码一起发送。例如我这里用一个简单的校验算法：“数据中有奇数个1，校验码就为1；偶数个1，校验码就为0”，那么“0010 0010”的校验码就是0。计算机A会将校验码和数据流一起发送，只要他们都支持同一个数据链路层的协议，那么计算机AB都知道校验码存放在哪即可。假设校验码存储在第一位，则数据流变成了“0 0010 0010”，那么计算机A传输到计算机B，出现错误了，计算机B就知道了。数据链路层就干这事，只要保证了两个主机之间的数据传输正常。

3.3 网络层

还是按照早期计网的发展的例子，在一个学校里面，发展了网络，结果非常的便利，其他公司都想用，国家也想用，然后就铺天盖地地设置网络，当时美国就是第一个搭建网络的国家。

而随着计网的不断发展，搭建方式已经不是简单的“中间人-n”了，而是“n-中间人-中间人-n”，“中间人”之间相互连接，而每个“中间人”负责某一个地域，这个地域的人连接这个“中间人”，就能实现跨域的计算机连接了。



假设现在搭建了国家网络，P1在拉斯维加斯，P2在华盛顿，只要他们连接了这个网络，就能够传输信息。但是呢，P1发送信息，怎么告诉中间人要发送给P2呢？假如P1和P2是一个“中间人”还可以，因为“中间人”知道哪些电脑连接了它，所以直接传输给P2就行。但是现在P1和P2不是同一个“中间人”，怎么办呢？两个方法，一个是每个“中间人”记录其他“中间人”的连接信息，这样每一个“中间人”就知道哪些计算机连接网络了，那么就可以很好的定位所有计算机了；另一个是，用某一种方式标志P2的“中间人”，当P1发送的时候，就按照类似的写信的方式，写下P2的“中间人”，然后加上P2的信息，这样，P1的“中间人”就知道发送给P2的“中间人”了，然后P2的“中间人”再发送数据给P2。

很明显的是，第一个方案，假如每个“中间人”都这样搞，会导致数据爆炸的，所以偏向以前邮差的方式，使用了第二个方案。现在标志每一个计算机的物理地址都是用的IP地址

（Internet Protocol Address，互联网协议地址），标志每一个计算机的身份是用的MAC地址（Media Access Control Address）。IP地址能够知道你所在的“中间人”，而MAC地址是在电脑的网卡上的，电脑一生产就有的，能够唯一标识一台计算机。

但是还有一个问题，拉斯维加斯的“中间人”怎么将数据传输到华盛顿的“中间人”，直接传输吗？不是的，多个中间人之间并不是之前的“两两连接”，而是一个网状的，具体怎么传输这就是网络层要做的事情。简单来说，网络层的事情就是如何根据IP地址，实现中间人中传输，然后再将数据传输到指定计算机中。

3.4 传输层

看似前面的都没什么可扩展的了，但是前面只能保证数据从计算机A传输到计算机B，但是假如出错了怎么办？再者，计算机A怎么知道计算机B得到数据没有？再者，计算机A怎么知道计算机B在线？这就是传输层所做的事情，提供一个传输协议，根据传输协议来管理整个传输的可靠性。

说到这里，终于可以提到了我最喜欢的UDP（User Datagram Protocol，用户数据报协议）和TCP（Transmission Control Protocol，传输控制协议）两个传输层协议了。

先说一下UDP，这个很简单，只要计算机A数据发送了就结束了，压根不管计算机B收到没有，所谓的“提了裤子就走人”，对应的是不可靠的传输。

再说一下TCP，TCP可牛逼了，首先需要“TCP三次握手”确定对方是否在线并且同意连接，只要对方不在线或者不同意，计算机A发送了也没用；然后数据的传输方面，计算机A发送给计算机B，然后计算机B收到了还会回复计算机A说我收到了；接着在传输结束的时候，还会“TCP四次挥手”断开这次连接。（具体的握手和挥手细节可以下去了解）

但是吧，TCP这么多操作很消耗资源，再者，这些额外的通信规则也会导致原本的数据流暴涨，除了校验码还需要加一堆额外的信息。所以为啥还会保留UDP，读者就知道了吧，有些时候，想吃快餐就去快餐店，想吃西餐就吃西餐店，主要看自己是否有这个需求。

3.5 会话层、表示层、应用层

这三个部分我一起说明，因为在实际的应用中，这三个其实没啥区别的.....而且我在学习这部分的时候，自己能力有限，没能准确知道他们的出现原因，我这里简单说明一下每一层的作用。

会话层：主要掌握对应的连接，管理对应的会话连接，例如，你看的电影，上次看到这里了，然后下次来的时候就从这里开始请求，这就是会话层在管理。还有.....我去，我举不出例子了。

表示层：这里其实没有什么特别的意义，我个人理解是，例如一个照片和一首歌，他们都是01数据流，但是如何这些数据流正常显示呢？就需要按照某种规则来读取，几乎是我们之前提到的“协议”的产生。

应用层：好了，这个才是主角，著名的HTTP（HyperText Transfer Protocol，超文本传输协议）就是这个层的“协议王”。就拿之前的例子“0010 0010”代表“lbwnb”，那么计算机B拿到数据“lbwnb”，这就是简单的信息传输。但是HTTP的诞生，是历史发展的。

以前，多个学生在学校连接了学校的图书馆的计算机，能够获取图书馆的图书，但是需要这样操作：P1需要自己的电脑，P2需要在图书馆的计算机，P1和P2打电话说：“我要这个图书《计算机网络》，你往已经连接的计算机B传输就可以了。”是不是复杂的很？

这时候HTTP的作用就体现出来了，现在《计算机网络》这本书在计算机A的文件夹地址“计算机知识/软件工程/《计算机网络》”；计算机B先和计算机A建立TCP连接，能够确保计算机A和计算机B的通信；接着，计算机B发送信息“计算机知识/软件工程/《计算机网络》”给计算机A，告诉自己要这个书；计算机A收到这个信息，再将对应的《计算机网络》这个数据给计算机B；当数据传输完毕时，在通过TCP断开连接，释放网络资源。

这样，只要将“计算机A收到信息和发送”的过程设置为一个程序，只要收到指定的“地址”，就会向发送端发送“对应地址的书”给计算B。这样形成了一个客户端/服务端模式

（Client/Serve模式，C/S模式），服务端就是图书馆的电脑，客户端就是学生的电脑。这种模式使用的协议一般就是HTTP。即按照HTTP的文本规则，客户端将需要的资源告诉给服务端，服务端再根据HTTP规则返回数据给客户端。

4 总结

其实，在OSI模型公开以来，“会话层、表示层、应用层”都被规约成“应用层”，所以几乎没有什么区别，因为这都是可以在一个协议中说明的，例如HTTP协议里面可以标志上次计算机B和计算机A的连接信息，可以快速建立连接；例如HTTP里存放了文件的类型和大小，计算机B根据对应的内容保存和读取文件即可。

因为“会话层、表示层、应用层”被规约了，所以现在主要使用的是五层网络体系（即规约后的层）来发展每个计算机部分。

可以看到，每一层是层层递进的：物理层管理01数据流传输，保证数据流传输；数据链路层管理数据流的完整性，保证数据流完整性；网络层管理数据的传输走向，保证数据到达；传输层管理两个主机之间传输方式，保证传输可用；应用层管理主机之间的数据交流方式，保证正常交流。其中每一层都会向后面服务，每一层都基于前面。

在学习计算机网络的过程中，就是按照每一层每一层的来学习。额外的，我没有提到“端到端”“IP到IP”“MAC到MAC”这样来讲述，是因为我觉得这种说法应该基于的操作系统来讲才会更加明白，但是如此回顾下来，似乎没有操作系统知识也能够很好的分层，所以索性不说明了。