

**二、實體層**

**"實體層"，它就是把電腦連接起來的物理手段。它主要規定了網絡的一些電氣特性，作用是負責傳送0和1的電信號。**

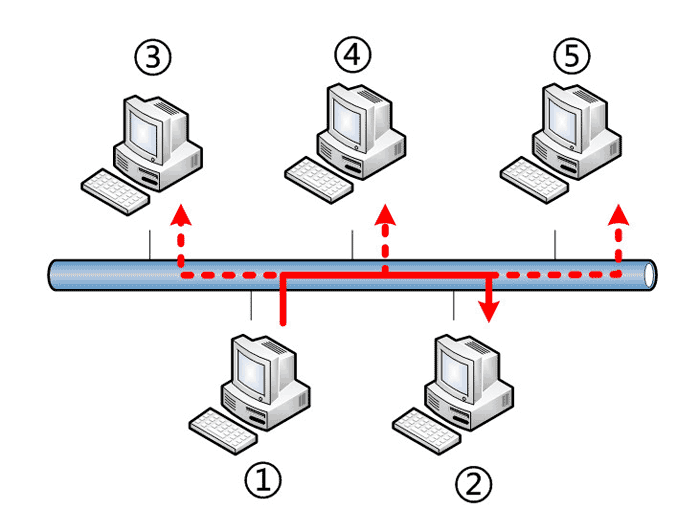
**三、鏈接層**

單純的0和1沒有任何意義，必須規定解讀方式：多少個電信號算一組？每個信號位有何意義？

**這就是"鏈接層"的功能，它在"實體層"的上方，確定了0和1的分組方式。**

以太網規定，一組電信號構成一個數據包，叫做"幀"（Frame）。每一幀分成兩個部分：標頭（Head）和數據（Data）。

每塊網卡出廠的時候，都有一個全世界獨一無二的MAC地址，長度是48個二進制位，通常用12個十六進制數表示。



上圖中，1號計算機向2號計算機發送一個數據包，同一個子網絡的3號、4號、5號計算機都會收到這個包。它們讀取這個包的"標頭"，找到接收方的MAC地址，然後與自身的MAC地址相比較，如果兩者相同，就接受這個包，做進一步處理，否則就丟棄這個包。這種發送方式就叫做"廣播"（broadcasting）。

"網絡層"出現以後，每台計算機有了兩種地址，一種是MAC地址，另一種是網絡地址。兩種地址之間沒有任何聯系，MAC地址是綁定在網卡上的，網絡地址則是管理員分配的，它們只是隨機組合在一起。

網絡地址幫助我們確定計算機所在的子網絡，MAC地址則將數據包送到該子網絡中的目標網卡。因此，從邏輯上可以推斷，必定是先處理網絡地址，然後再處理MAC地址。

**4.2 IP協議**

互聯網上的每一台計算機，都會分配到一個IP地址。這個地址分成兩個部分，前一部分代表網絡，後一部分代表主機。比如，IP地址172.16.254.1，這是一個32位的地址，假定它的網絡部分是前24位（172.16.254），那麼主機部分就是後8位（最後的那個1）。處於同一個子網絡的電腦，它們IP地址的網絡部分必定是相同的，也就是說172.16.254.2應該與172.16.254.1處在同一個子網絡。

但是，問題在於單單從IP地址，我們無法判斷網絡部分。還是以172.16.254.1為例，它的網絡部分，到底是前24位，還是前16位，甚至前28位，從IP地址上是看不出來的。

那麼，怎樣才能從IP地址，判斷兩台計算機是否屬於同一個子網絡呢？這就要用到另一個參數"子網掩碼"（subnet mask）。

所謂"子網掩碼"，就是表示子網絡特徵的一個參數。它在形式上等同於IP地址，也是一個32位二進制數字，它的網絡部分全部為1，主機部分全部為0。比如，IP地址172.16.254.1，如果已知網絡部分是前24位，主機部分是後8位，那麼子網絡掩碼就是11111111.11111111.11111111.00000000，寫成十進制就是255.255.255.0。

知道"子網掩碼"，我們就能判斷，任意兩個IP地址是否處在同一個子網絡。方法是將兩個IP地址與子網掩碼分別進行AND運算（兩個數位都為1，運算結果為1，否則為0），然後比較結果是否相同，如果是的話，就表明它們在同一個子網絡中，否則就不是。

比如，已知IP地址172.16.254.1和172.16.254.233的子網掩碼都是255.255.255.0，請問它們是否在同一個子網絡？兩者與子網掩碼分別進行AND運算，結果都是172.16.254.0，因此它們在同一個子網絡。

總結一下，IP協議的作用主要有兩個，一個是為每一台計算機分配IP地址，另一個是確定哪些地址在同一個子網絡。

**4.3 IP數據包**



藍色以太數據包,紅色ip數據包

**4.4 ARP協議**

因為IP數據包是放在以太網數據包裡發送的，所以我們必須同時知道兩個地址，一個是對方的MAC地址，另一個是對方的IP地址。通常情況下，對方的IP地址是已知的（後文會解釋），但是我們不知道它的MAC地址。

所以，我們需要一種機制，能夠從IP地址得到MAC地址。

這裡又可以分成兩種情況。

第一種情況，如果兩台主機不在同一個子網絡，那麼事實上沒有辦法得到對方的MAC地址，只能把數據包傳送到兩個子網絡連接處的"網關"（gateway），讓網關去處理。

第二種情況，如果兩台主機在同一個子網絡，那麼我們可以用ARP協議，得到對方的MAC地址。ARP協議也是發出一個數據包（包含在以太網數據包中），其中包含它所要查詢主機的IP地址，在對方的MAC地址這一欄，填的是FF:FF:FF:FF:FF:FF，表示這是一個"廣播"地址。它所在子網絡的每一台主機，都會收到這個數據包，從中取出IP地址，與自身的IP地址進行比較。如果兩者相同，都做出回復，向對方報告自己的MAC地址，否則就丟棄這個包。

總之，有了ARP協議之後，我們就可以得到同一個子網絡內的主機MAC地址，可以把數據包發送到任意一台主機之上了。

**五、傳輸層**

**5.1 傳輸層的由來**

有了MAC地址和IP地址，我們已經可以在互聯網上任意兩台主機上建立通信。

接下來的問題是，同一台主機上有許多程序都需要用到網絡，比如，你一邊瀏覽網頁，一邊與朋友在線聊天。當一個數據包從互聯網上發來的時候，你怎麼知道，它是表示網頁的內容，還是表示在線聊天的內容？

也就是說，我們還需要一個參數，表示這個數據包到底供哪個程序（進程）使用。這個參數就叫做"端口"（port），它其實是每一個使用網卡的程序的編號。每個數據包都發到主機的特定端口，所以不同的程序就能取到自己所需要的數據。

\*\*\*\*\*

**"傳輸層"的功能，就是建立"端口到端口"的通信。相比之下，"網絡層"的功能是建立"主機到主機"的通信。只要確定主機和端口，我們就能實現程序之間的交流。**因此，Unix系統就把主機+端口，叫做"套接字"（socket）。有了它，就可以進行網絡應用程序開發了。

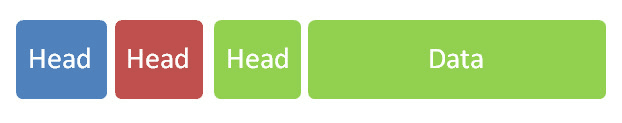
**5.2 UDP協議**

現在，我們必須在數據包中加入端口信息，這就需要新的協議。最簡單的實現叫做UDP協議，它的格式幾乎就是在數據前面，加上端口號。

UDP數據包，也是由"標頭"和"數據"兩部分組成。



"標頭"部分主要定義了發出端口和接收端口，"數據"部分就是具體的內容。然後，把整個UDP數據包放入IP數據包的"數據"部分，而前面說過，IP數據包又是放在以太網數據包之中的，所以整個以太網數據包現在變成了下面這樣：



UDP數據包非常簡單，"標頭"部分一共只有8個字節，總長度不超過65,535字節，正好放進一個IP數據包。

**5.3 TCP協議**

UDP協議的優點是比較簡單，容易實現，但是缺點是可靠性較差，一旦數據包發出，無法知道對方是否收到。

為瞭解決這個問題，提高網絡可靠性，TCP協議就誕生了。這個協議非常復雜，但可以近似認為，它就是有確認機制的UDP協議，每發出一個數據包都要求確認。如果有一個數據包遺失，就收不到確認，發出方就知道有必要重發這個數據包了。

因此，TCP協議能夠確保數據不會遺失。它的缺點是過程復雜、實現困難、消耗較多的資源。

TCP數據包和UDP數據包一樣，都是內嵌在IP數據包的"數據"部分。TCP數據包沒有長度限制，理論上可以無限長，但是為了保證網絡的效率，通常TCP數據包的長度不會超過IP數據包的長度，以確保單個TCP數據包不必再分割。

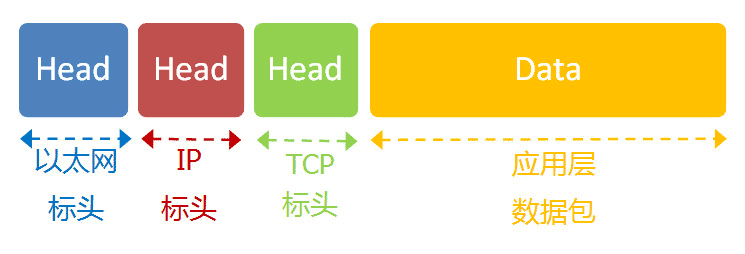
**六、應用層**

應用程序收到"傳輸層"的數據，接下來就要進行解讀。由於互聯網是開放架構，數據來源五花八門，必須事先規定好格式，否則根本無法解讀。

**"應用層"的作用，就是規定應用程序的數據格式。**

舉例來說，TCP協議可以為各種各樣的程序傳遞數據，比如Email、WWW、FTP等等。那麼，必須有不同協議規定電子郵件、網頁、FTP數據的格式，這些應用程序協議就構成了"應用層"。

這是最高的一層，直接面對用戶。它的數據就放在TCP數據包的"數據"部分。因此，現在的以太網的數據包就變成下面這樣。



藍色以太數據包,紅色ip數據包

這是從系統的角度，解釋互聯網是如何構成的。[下一篇](https://www.ruanyifeng.com/blog/2012/06/internet_protocol_suite_part_ii.html)，我反過來，從用戶的角度，自上而下看看這個結構是如何發揮作用，完成一次網絡數據交換的。

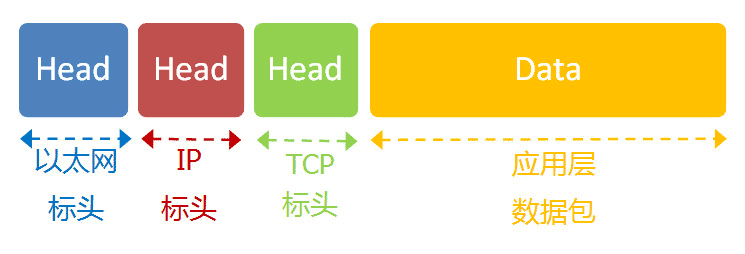
**互聯網協議入門（二）**

[上一篇文章](https://www.ruanyifeng.com/blog/2012/05/internet_protocol_suite_part_i.html)分析了互聯網的總體構思，從下至上，每一層協議的設計思想。

這是從設計者的角度看問題，今天我想切換到用戶的角度，看看用戶是如何從上至下，與這些協議互動的。

先對前面的內容，做一個小結。

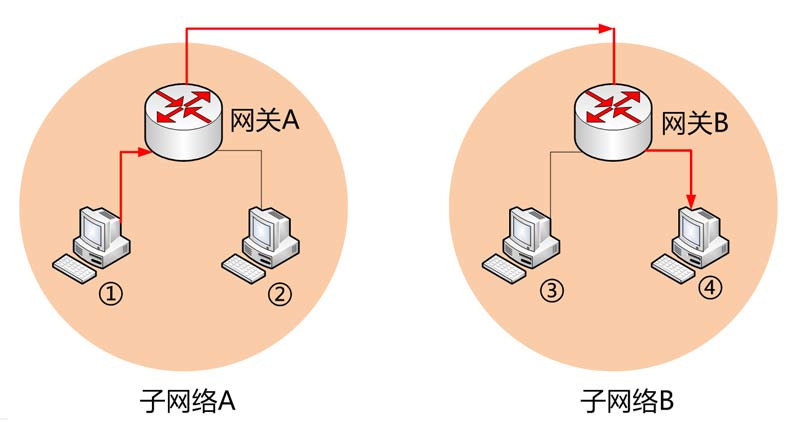
我們已經知道，網絡通信就是交換數據包。電腦A向電腦B發送一個數據包，後者收到了，回復一個數據包，從而實現兩台電腦之間的通信。數據包的結構，基本上是下面這樣：

發送這個包，需要知道兩個地址：

　　\* 對方的MAC地址

　　\* 對方的IP地址

有了這兩個地址，數據包才能准確送到接收者手中。但是，前面說過，MAC地址有侷限性，如果兩台電腦不在同一個子網絡，就無法知道對方的MAC地址，必須通過網關（gateway）轉發。



上圖中，1號電腦要向4號電腦發送一個數據包。它先判斷4號電腦是否在同一個子網絡，結果發現不是（後文介紹判斷方法），於是就把這個數據包發到網關A。網關A通過路由協議，發現4號電腦位於子網絡B，又把數據包發給網關B，網關B再轉發到4號電腦。

1號電腦把數據包發到網關A，必須知道網關A的MAC地址。所以，數據包的目標地址，實際上分成兩種情況：

|  |  |
| --- | --- |
| 場景 | 數據包地址 |
| 同一個子網絡 | 對方的MAC地址，對方的IP地址 |
| 非同一個子網絡 | 網關的MAC地址，對方的IP地址 |

發送數據包之前，電腦必須判斷對方是否在同一個子網絡，然後選擇相應的MAC地址。接下來，我們就來看，實際使用中，這個過程是怎麼完成的。

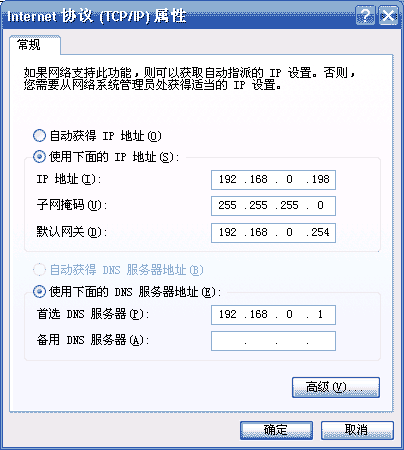
**八、用戶的上網設置**

**8.1 靜態IP地址**

通常你必須做一些設置。有時，管理員（或者ISP）會告訴你下面四個參數，你把它們填入操作系統，計算機就能連上網了：

　　\* 本機的IP地址  
　　\* 子網掩碼  
　　\* 網關的IP地址 gateway  
　　\* DNS的IP地址

下圖是Windows系統的設置窗口。 tcp/ip屬性



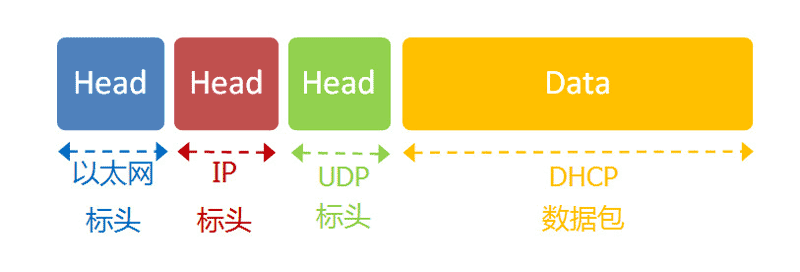
這四個參數缺一不可，後文會解釋為什麼需要知道它們才能上網。由於它們是給定的，計算機每次開機，都會分到同樣的IP地址，所以這種情況被稱作"靜態IP地址上網"。

但是，這樣的設置很專業，普通用戶望而生畏，而且如果一台電腦的IP地址保持不變，其他電腦就不能使用這個地址，不夠靈活。出於這兩個原因，大多數用戶使用"動態IP地址上網"。

**8.2 動態IP地址**

所謂"動態IP地址"，指計算機開機後，會自動分配到一個IP地址，不用人為設定。它使用的協議叫做[DHCP協議](https://zh.wikipedia.org/zh/DHCP)。

**動態主機設定協定**（英語：**D**ynamic **H**ost **C**onfiguration **P**rotocol，縮寫：**DHCP**），又稱**動態主機組態協定**，是一個用於[IP](https://zh.wikipedia.org/wiki/%E7%BD%91%E9%99%85%E5%8D%8F%E8%AE%AE)網絡的[網絡協議](https://zh.wikipedia.org/wiki/%E7%BD%91%E7%BB%9C%E5%8D%8F%E8%AE%AE" \o "網絡協議)，位於[OSI模型](https://zh.wikipedia.org/wiki/OSI%E6%A8%A1%E5%9E%8B)的[應用層](https://zh.wikipedia.org/wiki/%E5%BA%94%E7%94%A8%E5%B1%82)，使用[UDP](https://zh.wikipedia.org/wiki/%E7%94%A8%E6%88%B7%E6%95%B0%E6%8D%AE%E6%8A%A5%E5%8D%8F%E8%AE%AE)協議工作，主要有兩個用途：

* 用於內部網路或網絡服務供應商自動分配[IP地址](https://zh.wikipedia.org/wiki/IP%E5%9C%B0%E5%9D%80)給用戶
* 用於內部網路管理員對所有電腦作中央管理
* 這個協議規定，每一個子網絡中，有一台計算機負責管理本網絡的所有IP地址，它叫做"DHCP服務器"。新的計算機加入網絡，必須向"DHCP服務器"發送一個"DHCP請求"數據包，申請IP地址和相關的網絡參數。
* 前面說過，如果兩台計算機在同一個子網絡，必須知道對方的MAC地址和IP地址，才能發送數據包。但是，新加入的計算機不知道這兩個地址，怎麼發送數據包呢？
* DHCP協議做了一些巧妙的規定。
* **8.3 DHCP協議**
* 首先，它是一種應用層協議，建立在UDP協議之上，所以整個數據包是這樣的：
* 
* （1）最前面的"以太網標頭"，設置發出方（本機）的MAC地址和接收方（DHCP服務器）的MAC地址。前者就是本機網卡的MAC地址，後者這時不知道，就填入一個廣播地址：FF-FF-FF-FF-FF-FF。
* （2）後面的"IP標頭"，設置發出方的IP地址和接收方的IP地址。這時，對於這兩者，本機都不知道。於是，發出方的IP地址就設為0.0.0.0，接收方的IP地址設為255.255.255.255。
* （3）最後的"UDP標頭"，設置發出方的端口和接收方的端口。這一部分是DHCP協議規定好的，發出方是68端口，接收方是67端口。

這個數據包構造完成後，就可以發出了。以太網是廣播發送，同一個子網絡的每台計算機都收到了這個包。因為接收方的MAC地址是FF-FF-FF-FF-FF-FF，看不出是發給誰的，所以每台收到這個包的計算機，還必須分析這個包的IP地址，才能確定是不是發給自己的。當看到發出方IP地址是0.0.0.0，接收方是255.255.255.255，於是DHCP服務器知道"這個包是發給我的"，而其他計算機就可以丟棄這個包。

接下來，DHCP服務器讀出這個包的數據內容，分配好IP地址，發送回去一個"DHCP響應"數據包。這個響應包的結構也是類似的，以太網標頭的MAC地址是雙方的網卡地址，IP標頭的IP地址是DHCP服務器的IP地址（發出方）和255.255.255.255（接收方），UDP標頭的端口是67（發出方）和68（接收方），分配給請求端的IP地址和本網絡的具體參數則包含在Data部分。

新加入的計算機收到這個響應包，於是就知道了自己的IP地址、子網掩碼、網關地址、DNS服務器等等參數。

**8.4 上網設置：小結**

這個部分，需要記住的就是一點：不管是"靜態IP地址"還是"動態IP地址"，電腦上網的首要步驟，是確定四個參數。這四個值很重要，值得重復一遍：

　　\* 本機的IP地址  
　　\* 子網掩碼  
　　\* 網關的IP地址  
　　\* DNS的IP地址

有了這幾個數值，電腦就可以上網"沖浪"了。接下來，我們來看一個實例，當用戶訪問網頁的時候，互聯網協議是怎麼運作的。

**九、一個實例：訪問網頁**

**9.1 本機參數**

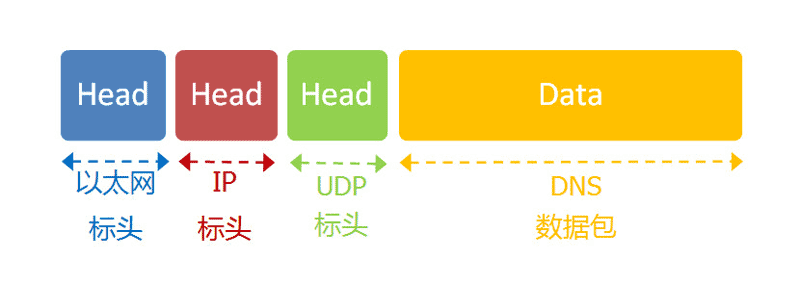
我們假定，經過上一節的步驟，用戶設置好了自己的網絡參數：

　　\* 本機的IP地址：192.168.1.100  
　　\* 子網掩碼：255.255.255.0  
　　\* 網關的IP地址：192.168.1.1  
　　\* DNS的IP地址：8.8.8.8

然後他打開瀏覽器，想要訪問Google，在地址欄輸入了網址：www.google.com。

**9.2 DNS協議**

[DNS協議](https://en.wikipedia.org/wiki/Domain_Name_System)可以幫助我們，將這個網址(www.google.com)轉換成IP地址。已知DNS服務器為8.8.8.8，於是我們向這個地址發送一個DNS數據包（53端口）。



然後，DNS服務器做出響應，告訴我們Google的IP地址是172.194.72.105。於是，我們知道了對方的IP地址。

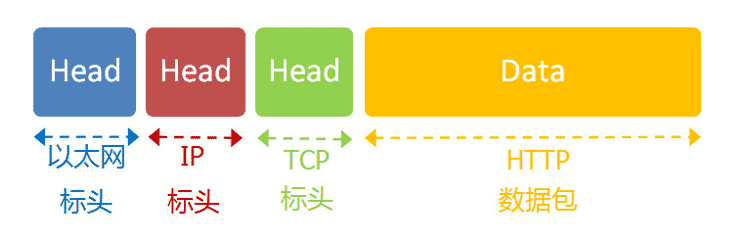
**9.3 子網掩碼**

已知子網掩碼是255.255.255.0，本機用它對自己的IP地址192.168.1.100，做一個二進制的AND運算（兩個數位都為1，結果為1，否則為0），計算結果為192.168.1.0；然後對Google的IP地址172.194.72.105也做一個AND運算，計算結果為172.194.72.0。這兩個結果不相等，所以結論是，Google與本機不在同一個子網絡。

因此，我們要向Google發送數據包，必須通過網關192.168.1.1轉發，也就是說，接收方的MAC地址將是網關的MAC地址。

**9.4 應用層協議**

瀏覽網頁用的是HTTP協議，它的整個數據包構造是這樣的：



HTTP部分的內容，類似於下面這樣：

　　GET / HTTP/1.1  
　　Host: www.google.com  
　　Connection: keep-alive  
　　User-Agent: Mozilla/5.0 (Windows NT 6.1) ......  
　　Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8  
　　Accept-Encoding: gzip,deflate,sdch  
　　Accept-Language: zh-CN,zh;q=0.8  
　　Accept-Charset: GBK,utf-8;q=0.7,\*;q=0.3  
　　Cookie: ... ...

我們假定這個部分的長度為4960字節，它會被嵌在TCP數據包之中。

**9.5 TCP協議**

TCP數據包需要設置端口，接收方（Google）的HTTP端口默認是80，發送方（本機）的端口是一個隨機生成的1024-65535之間的整數，假定為51775。

TCP數據包的標頭長度為20字節，加上嵌入HTTP的數據包，總長度變為4980字節。

**9.6 IP協議**

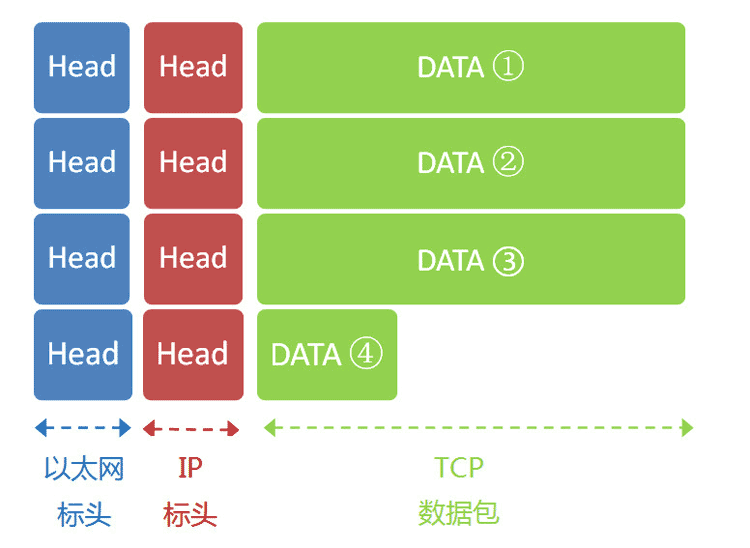
然後，TCP數據包再嵌入IP數據包。IP數據包需要設置雙方的IP地址，這是已知的，發送方是192.168.1.100（本機），接收方是172.194.72.105（Google）。

IP數據包的標頭長度為20字節，加上嵌入的TCP數據包，總長度變為5000字節。

**9.7 以太網協議**

最後，IP數據包嵌入以太網數據包。以太網數據包需要設置雙方的MAC地址，發送方為本機的網卡MAC地址，接收方為網關192.168.1.1的MAC地址（通過ARP協議得到）。

以太網數據包的數據部分，最大長度為1500字節，而現在的IP數據包長度為5000字節。因此，IP數據包必須分割成四個包。因為每個包都有自己的IP標頭（20字節），所以四個包的IP數據包的長度分別為1500、1500、1500、560。



**9.8 服務器端響應**

經過多個網關的轉發，Google的服務器172.194.72.105，收到了這四個以太網數據包。

根據IP標頭的序號，Google將四個包拼起來，取出完整的TCP數據包，然後讀出裡面的"HTTP請求"，接著做出"HTTP響應"，再用TCP協議發回來。

本機收到HTTP響應以後，就可以將網頁顯示出來，完成一次網絡通信。



這個例子就到此為止，雖然經過了簡化，但它大致上反映了互聯網協議的整個通信過程。