1、客户端浏览器通过**DNS解析到www.baidu.com的IP地址202.108.22.5，**通过这个IP地址找到客户端到服务器的路径。客户端浏览器发起一个HTTP会话到202.108.22.5，然后通过**TCP进行封装数据包，输入到网络层。**   
2、在客户端的传输层，把HTTP会话请求分成报文段，添加源和目的端口，如服务器使用80端口监听客户端的请求，客户端由系统随机选择一个端口如5000，与服务器进行交换，服务器把相应的请求返回给客户端的5000端口。然后使用IP层的IP地址查找目的端。 **(建立tcp连接)**  
3、客户端的网络层不用关心应用层或者传输层的东西，**主要做的是通过查找路由表确定如何到达服务器**，期间可能经过多个路由器，这些都是由路由器来完成的工作，我不作过多的描述，无非就是通过查找路由表决定**通过那个路径到达服务器。** （**IP的重要功能是寻址和路由**） **4、客户端的链路层，包通过链路层发送到路由**器，通过邻居协议**查找给定IP地址的MAC地址，**然后发送ARP请求查找目的地址，如果得到回应后就可以使用ARP的请求应答交换的IP数据包现在就可以传输了，然后发送**IP数据包到达服务器的地址。（IP地址转为MAC）**

## ****事件顺序****

(1) 浏览器获取输入的域名www.baidu.com   
(2) 浏览器向DNS请求解析www.baidu.com的IP地址   
(3) 域名系统DNS解析出百度服务器的IP地址

(4) 浏览器发出HTTP请求，请求百度首页   
(5) 浏览器与该服务器建立TCP连接(默认端口号80)

(6)IP的重要功能是寻址和路由

（7）定义了通过物理网络输出数据所需的协议与硬件要求

服务器通过HTTP响应把首页文件发送给浏览器   
  
(8) 浏览器将首页文件进行解析，并将Web页显示给用户。

(9) TCP连接释放

## ****涉及到的协议****

(1) 应用层：HTTP**(WWW访问协议)**，DNS**(域名解析服务)**   
DNS解析域名为目的IP，通过IP找到服务器路径，客户端向服务器发起HTTP会话，然后通过运输层TCP协议封装数据包，在TCP协议基础上进行传输

(2) 传输层：TCP**(为HTTP提供可靠的数据传输)**，UDP**(DNS使用UDP传输)**  
HTTP会话会被分成报文段，添加源、目的端口；TCP协议进行主要工作

(3)网络层：IP**(IP数据数据包传输和路由选择)**，ICMP**(提供网络传输过程中的差错检测)**，ARP**(将本机的默认网关IP地址映射成物理MAC地址)**为数据包选择路由，IP协议进行主要工作，相邻结点的可靠传输，ARP协议将IP地址转成MAC地址。

>>>**点击网址后，应用层的DNS协议会将网址解析为IP地址；**

DNS查找过程：

1.        浏览器会**检查缓存中有没有这个域名对应的解析过的IP地址**，如果缓存中有，这个解析过程就将结束。

2.        如果用户的浏览器缓存中没有，浏览器会查找操作系统缓存**（hosts文件）**中是否有这个域名对应的DNS解析结果。

3.        若还没有，此时会发送一个数据包给DNS服务器，**DNS服务器找到后将解析所得IP地址返回给用户**。

>>>**在应用层，浏览器会给web服务器发送一个HTTP请求；**

请求头为：GET http://www.baidu.com/HTTP/1.1

>>>**在传输层，（上层的传输数据流分段）HTTP数据包会嵌入在TCP报文段中；**

TCP报文段需要设置端口**，接收方（百度）的HTTP端口默认是80**，**本机的端口是一个1024-65535之间的随机整数，这里假设为1025**，**这样TCP报文段由TCP首部（包含**发送方和接收方的端口信息**）+ HTTP数据包组成。**

>>>**在网络层中，TCP报文段再嵌入IP数据包中；**

IP数据包需要知道双方的IP地址，**本机IP地址假定为192.168.1.5，接受方IP地址为220.181.111.147（百度），这样IP数据包由IP头部（IP地址信息）+TCP报文段组成。**

>>> **在网络接口层，IP数据包嵌入到数据帧（以太网数据包）中在网络上传送；**

**数据帧中包含源MAC地址和目的MAC地址（通过ARP地址解析协议得到的）。这样数据帧由头部（MAC地址）+IP数据包组成。**

>>>**数据包经过多个网关的转发到达百度服务器，请求对应端口的服务；**

服务接收到发送过来的以太网数据包开始解析请求信息，从以太网数据包中提取**IP数据包—>TCP报文段—>HTTP数据包**，并组装为有效数据交与对应线程池中分配的线程进行处理**，在这个过程中，生成相应request、response对象。**

>>>**请求处理完成之后，服务器发回一个HTTP响应；**

请求处理程序会阅读请求及它的参数和cookies。它会读取也可能更新一些数据，并将数据存储在服务器上。处理完毕后，数据通过response对象给客户输出信息，输出信息也需要拼接HTTP协议头部分，关闭后断开连接。断开后，服务器端自动注销request、response对象，并将释放对应线程的使用标识（一般一个请求单独由一个线程处理，部分特殊情况有一个线程处理多个请求的情况）。

响应头为：HTTP/1.1200 OK

>>>**浏览器以同样的过程读取到HTTP响应的内容（HTTP响应数据包），然后浏览器对接收到的HTML页面进行解析，把网页显示出来呈现给用户。**

**客户端接收到返回数据，去掉对应头信息，形成也可以被浏览器认识的页面HTML字符串信息，交与浏览器翻译为对应页面规则信息展示为页面内容。**

1. 由域名→IP 地址  
   寻找 IP 地址的过程依次经过了浏览器缓存、系统缓存、hosts 文件、路由器缓存、 递归搜索根域名服务器。
2. 建立 TCP/IP 连接（三次握手具体过程）
3. 由浏览器发送一个 HTTP 请求
4. 经过路由器的转发，通过服务器的防火墙，该 HTTP 请求到达了服务器
5. 服务器处理该 HTTP 请求，返回一个 HTML 文件
6. 浏览器解析该 HTML 文件，并且显示在浏览器端
7. 这里需要注意：
   * HTTP 协议是一种基于 TCP/IP 的应用层协议，进行 HTTP 数据请求必须先建立 TCP/IP 连接
   * 可以这样理解：HTTP 是轿车，提供了封装或者显示数据的具体形式；Socket 是发动机，提供了网络通信的能力。
   * 两个计算机之间的交流无非是两个端口之间的数据通信 , 具体的数据会以什么样的形式展现是以不同的应用层协议来定义的。