



西北師範大學
NORTHWEST NORMAL UNIVERSITY

计算机网络

应用层 (1)

陈旺虎

chenwh@nwnu.edu.cn



第6章 应用层

6.1 域名系统 DNS

6.2 文件传送协议

6.3 远程终端协议 TELNET

6.4 万维网 WWW

6.5 电子邮件

6.6 动态主机配置协议 DHCP

6.7 简单网络管理协议 SNMP

6.8 应用进程跨越网络的通信

概述

□ 应用层的具体内容

- ▣ 规定应用进程在通信时所遵循的协议
- ▣ 每个应用层协议为了解决某一类应用问题
 - 问题的解决又往往是通过位于不同主机中的多个应用进程之间的通信和协同工作来完成的。

□ 应用层协议多采用C/S方式

6.1 域名系统 DNS

6.1.1 域名系统概述

- 域名系统 (DNS , Domain Name System)
 - ▣ 因特网使用的命名系统，其本质是一种名字系统
 - ▣ 用来把便于人们使用的机器名字转化成IP地址
- 用户使用域名；IP数据报使用IP地址
- 域名解析
 - ▣ 因特网的域名到IP地址的解析
 - ▣ 由若干个域名服务器程序完成，运行该程序的机器称为**域名服务器**

6.1.2 因特网的域名结构

- 因特网采用了层次树状结构的命名方法
 - ▣ 任何一个连接在因特网上的主机或路由器，都有一个唯一的层次结构的名称，即**域名**
 - ▣ 域：名字空间中一个可被管理的划分
 - ▣ 域名的结构由标号序列（. 分隔）构成

… · 三级域名 · 二级域名 · 顶级域名

例如： www.sina.com.cn, mail.nwnu.edu.cn

域名只是个逻辑概念

- 域名只是个逻辑概念，并不代表计算机所在的物理地点
- 域名中的“点”和点分十进制 IP 地址中的“点”并无一一对应的关系

顶级域名 (Top Level Domain, TLD)

(1) 国家顶级域名：如: .cn ， .us ， .uk ， .au...

(2) 通用顶级域名，最早包括：

.com （公司和企业）， .net （网络服务机构）

.org （非赢利性组织）， .edu （美国专用的教育机构）

.gov （美国专用的政府部门）

.mil （美国专用的军事部门）， .int （国际组织）

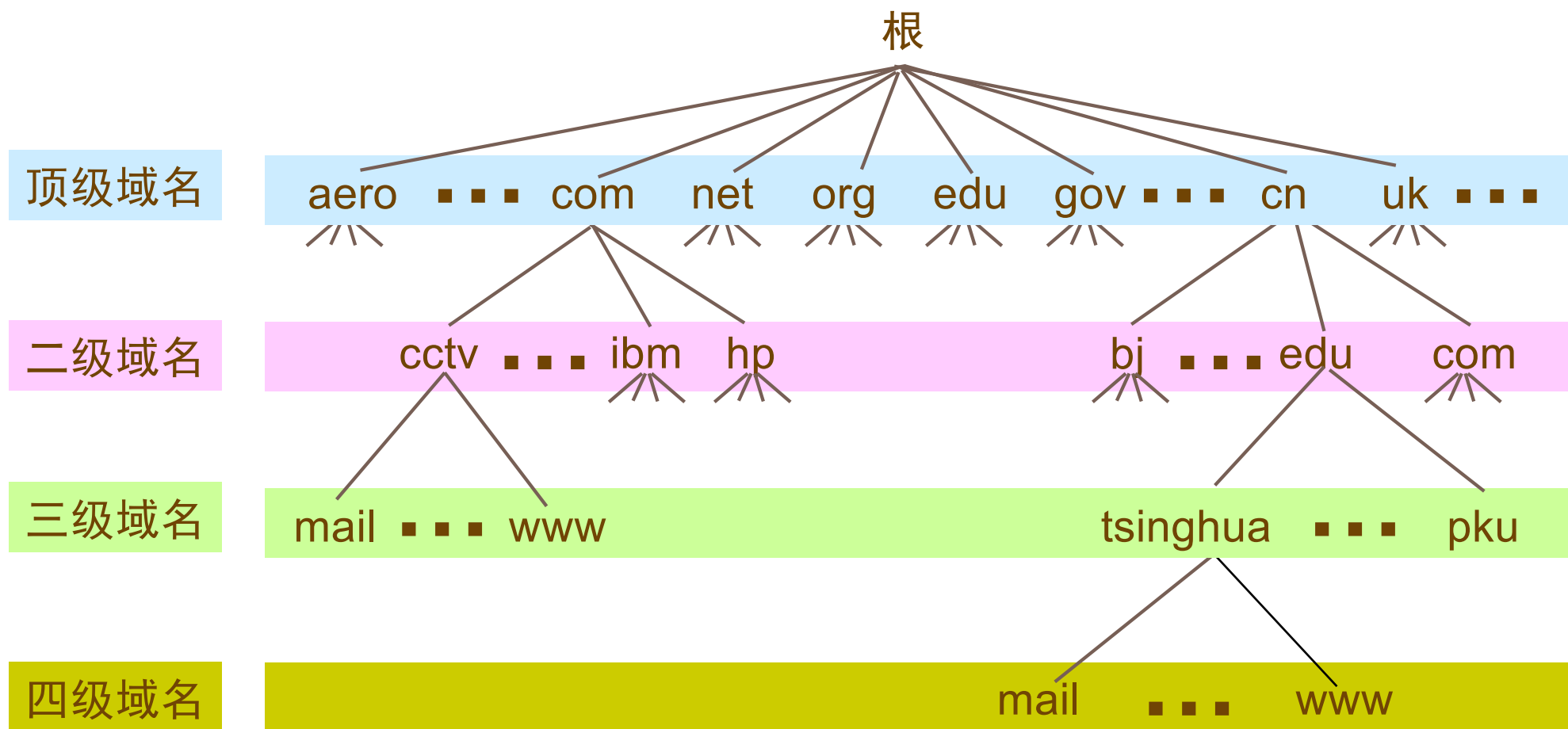
(3) 基础结构域名(infrastructure domain)

- ▣ 只有一个：arpa

- ▣ 用于反向域名解析，因此又称为**反向域名**。

- 国家顶级域名下的二级域名，可由该国自行设定；
- 我国把二级域名分为
 - ▣ 类别域名：ac, com, edu, gov, mil, net, org
 - ▣ 行政区域名：bj, sh, gs, ...
- 注：我国现在允许在cn下直接注册二级域名
- www.sina.cn

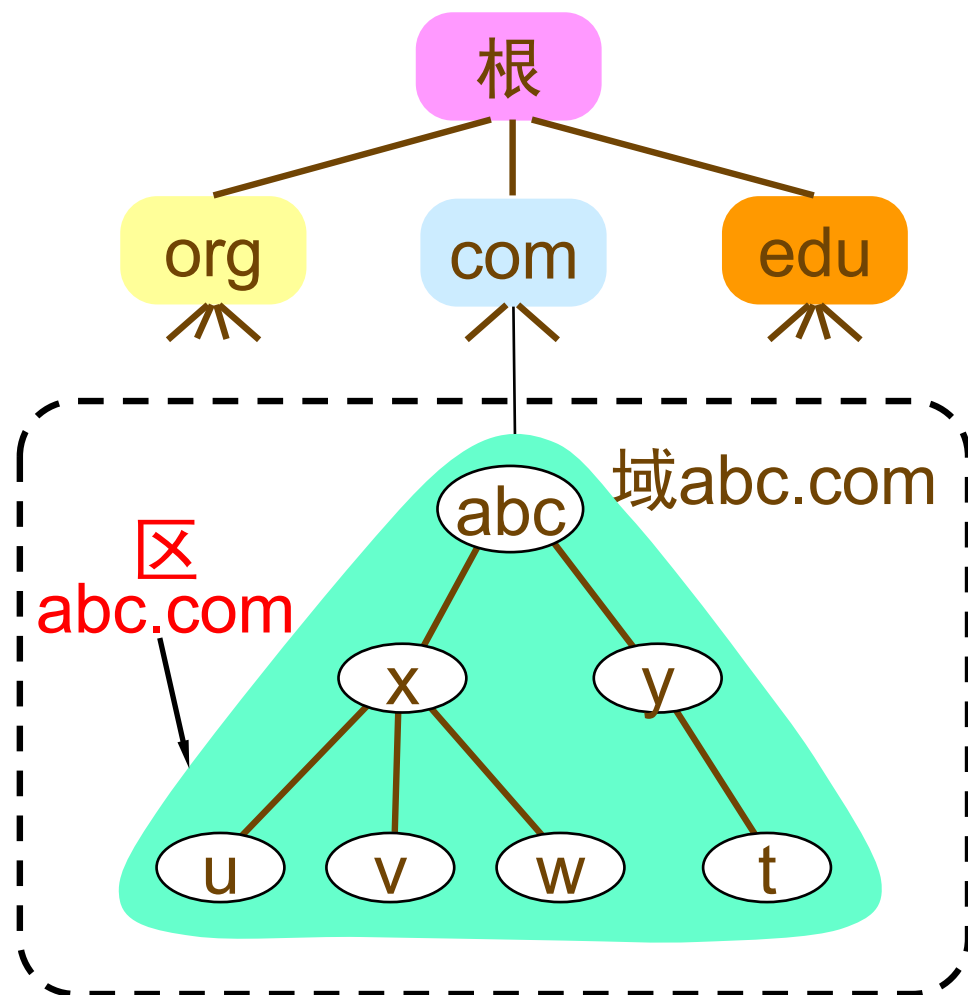
因特网的域名空间



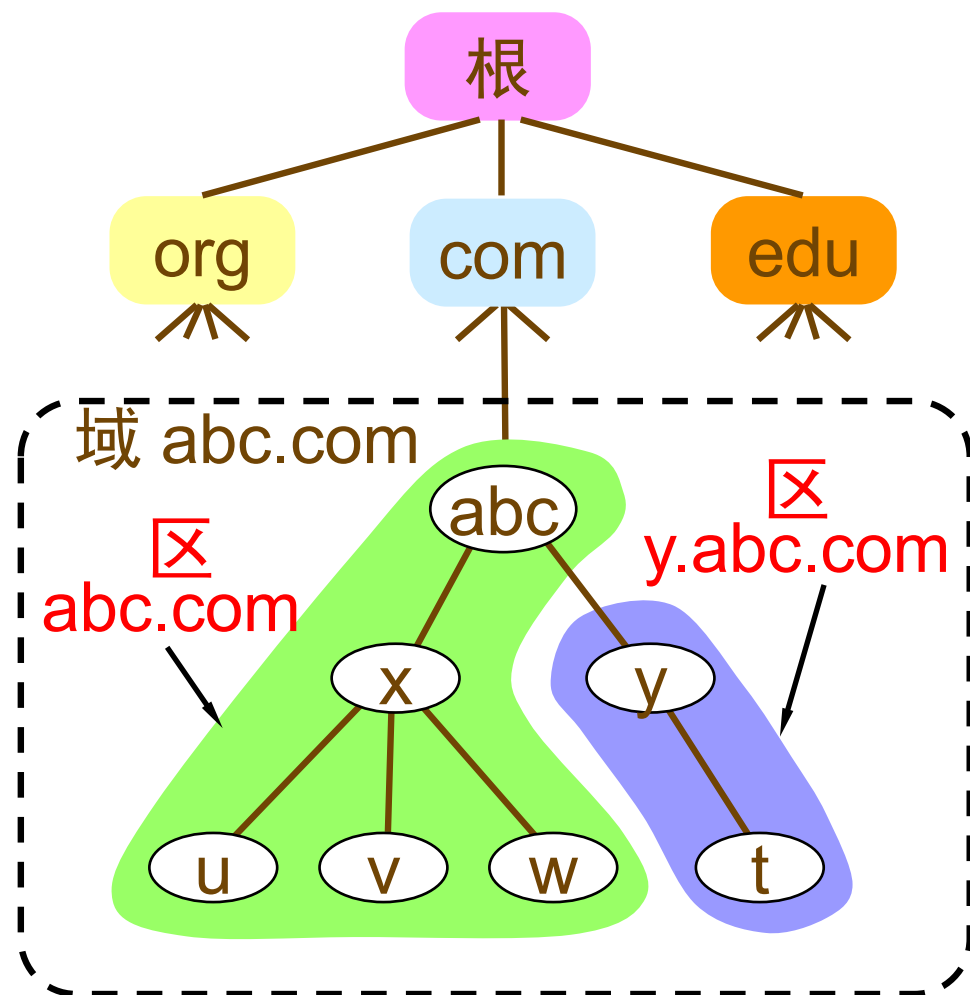
6.1.3 域名服务器

- DNS 服务器的管辖范围不是以“域”为单位，而是以“区”为单位
 - ▣ 一个服务器所负责管辖的范围叫做区(zone)
 - ▣ 各单位根据具体情况来划分自己管辖范围的区。但在一个区中的所有节点必须是能够连通的
 - ▣ 每一个区设置相应的权限域名服务器，用来保存该区中的所有主机的域名到IP地址的映射。

区的不同划分方法举例

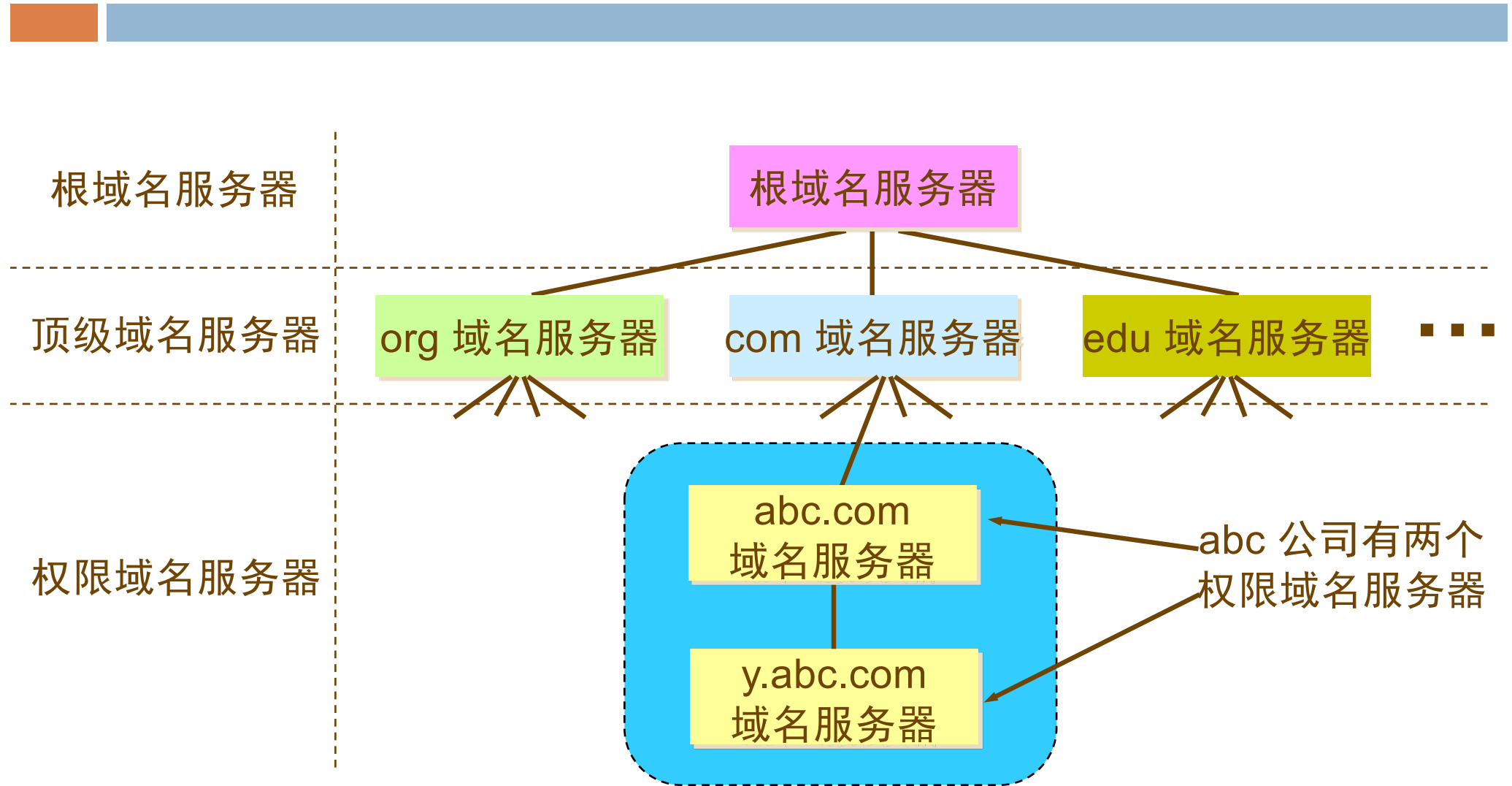


(a) 区 = 域



(b) 区 < 域

树状结构的 DNS 域名服务器



域名服务器的四种类型



- 根域名服务器
- 顶级域名服务器
- 权限域名服务器
- 本地域名服务器

(1) 根域名服务器

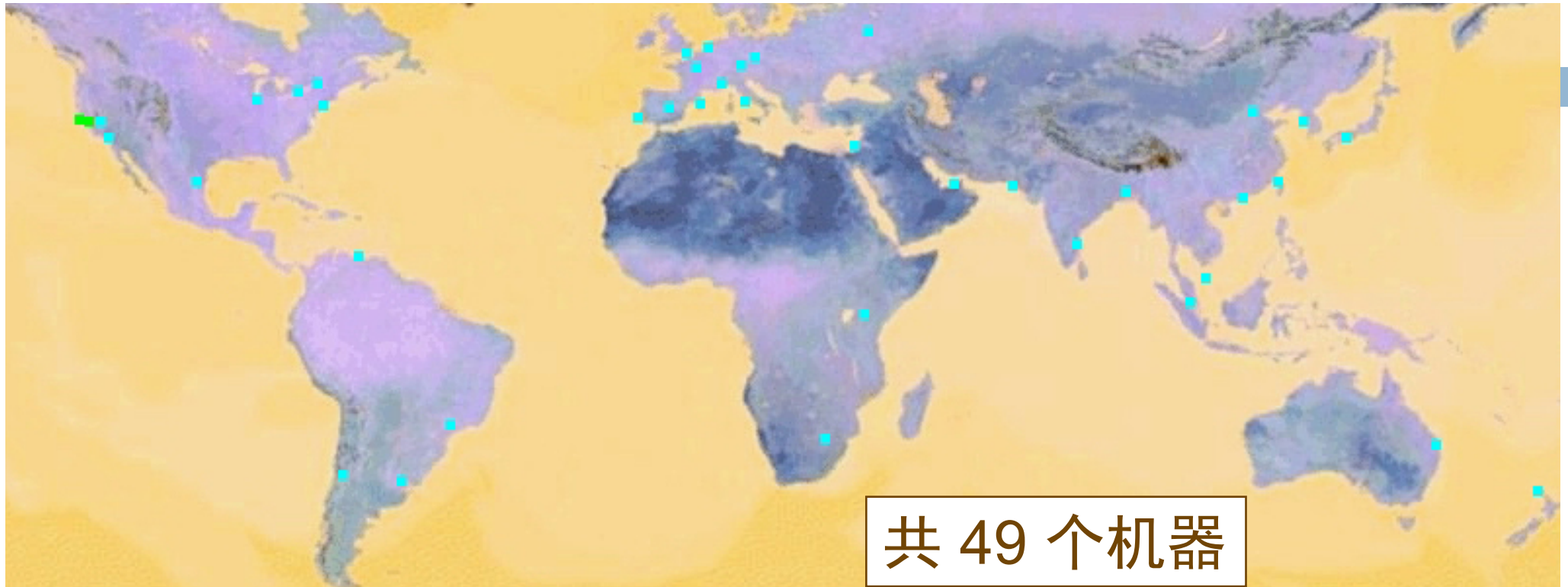
——最高层次的域名服务器——

- 根DNS服务器是最重要的DNS服务器
 - ▣ 所有的根DNS服务器都知道所有的顶级DNS服务器的域名和IP地址
 - ▣ 不管是哪一个本地域名服务器，若要对因特网上任何一个域名进行解析，只要自己无法解析，就首先求助于根域名服务器
- 在因特网上共有13 个不同IP地址的根域名服务器

根域名服务器共有13套装置 (不是 13 个机器)

- 这些根域名服务器相应的域名分别是
 - a.rootservers.net
 - b.rootservers.net
 - ...
 - m.rootservers.net
- 全世界已经安装了一百多个根域名服务器机器
- 使世界上大部分DNS都能就近找到一个根域名服务器。

举例：根域名服务器 f 的地点分布图



- 根域名服务器并不直接把域名转换成IP地址。
- 在使用迭代查询时，根域名服务器把下一步应当找的顶级域名服务器的IP地址告诉本地域名服务器。

(2) 顶级域名服务器

- 负责管理在该顶级DNS服务器注册的所有二级域名
- 当收到DNS查询请求时，就给出相应的回答
 - ▣ 可能是最后的结果，也可能是下一步应当找的DNS服务器的IP地址

(3) 权限域名服务器

- 是负责一个区的DNS服务器
- 当一个权限DNS服务器还不能给出最后的查询回答时，就会告诉发出查询请求的DNS客户，下一步应当找哪一个DNS服务器

(4) 本地域名服务器

- 本地域名服务器对域名系统非常重要
- 作用
 - ▣ 当一个主机发出DNS查询请求时，这个查询请求报文就发送给本地DNS服务器
 - ▣ 每一个ISP、大学（系）都可以拥有DNS服务器
- 这种域名服务器有时也称为默认域名服务器

提高域名服务器的可靠性

□ 主域名服务器&辅助域名服务器

- ▣ 当主域名服务器出故障时，辅助域名服务器可以保证 DNS 的查询工作不会中断。

□ 域名服务器数据一致性的保障

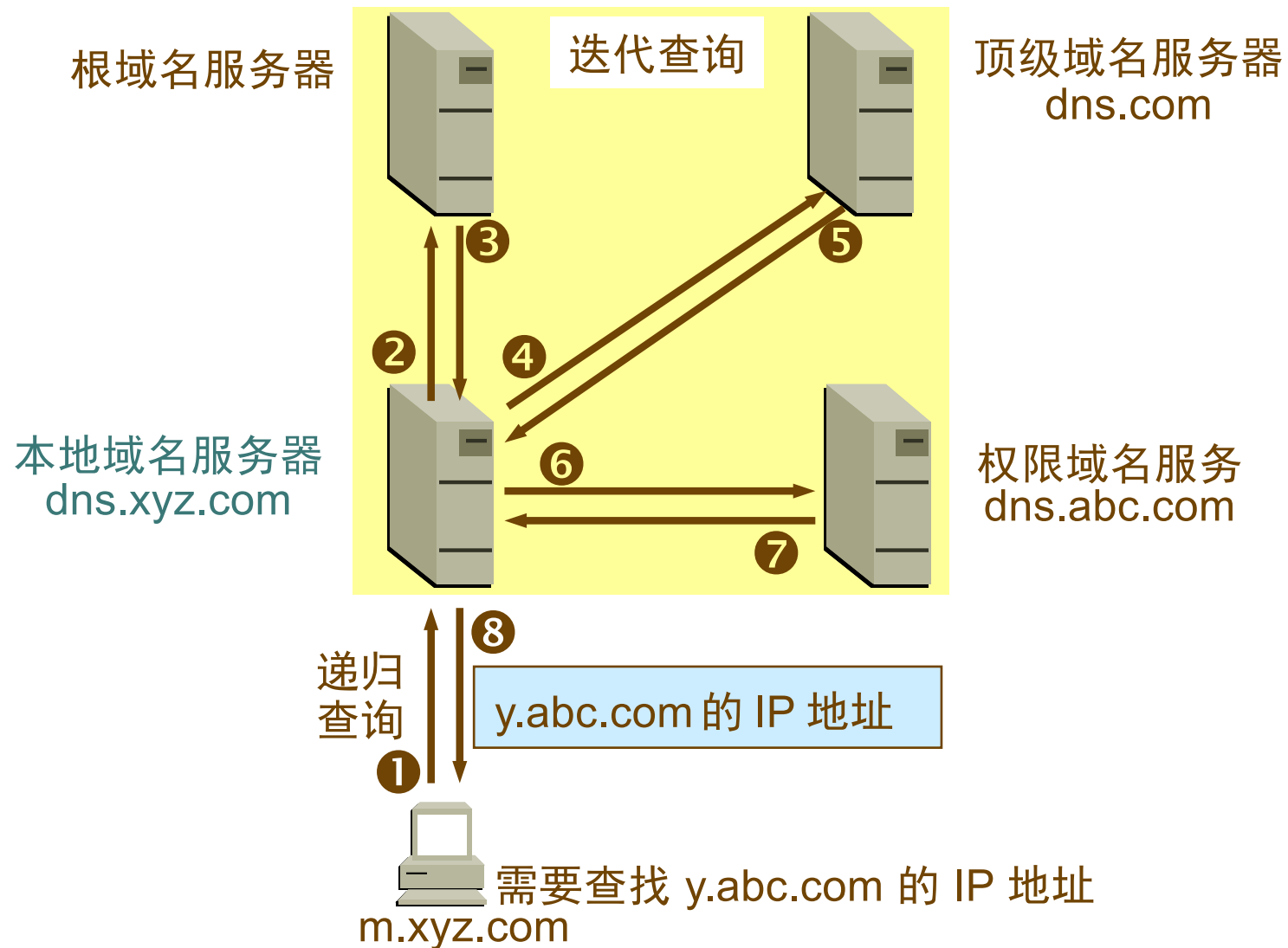
- ▣ 主域名服务器定期把数据复制到辅助域名服务器中
- ▣ 而更改数据只能在主域名服务器中进行。

域名的解析过程

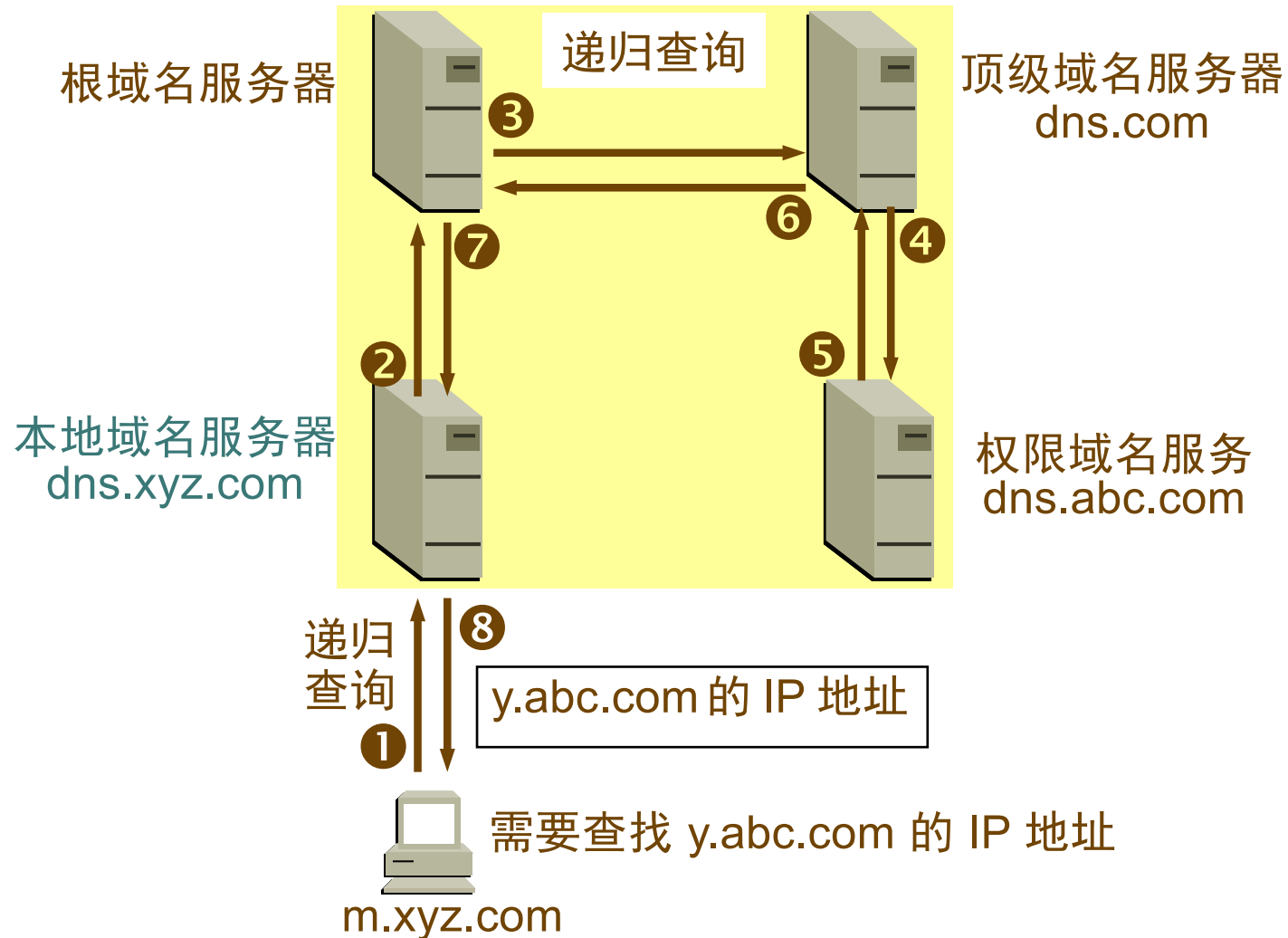


- 主机向本地域名服务器的查询一般都是采用递归查询
- 本地域名服务器向根域名服务器的查询通常是采用迭代查询

本地域名服务器采用迭代查询



本地域名服务器采用递归查询 (比较少用)



名字的高速缓存

- 每个域名服务器都维护一个高速缓存
 - ▣ 存放最近用过的名字以及从何处获得名字映射信息的记录。
 - ▣ 可大大减轻根域名服务器的负荷，使因特网上的DNS 查询请求和回答报文的数量大为减少。
- 域名服务器应为每项内容设置计时器，并处理超过合理时间的项
- 当权限域名服务器回答一个查询请求时，在响应中都指明绑定有效存在的时间值

6.2 文件传送协议

6.2.1 FTP概述

- 文件传送协议 FTP (File Transfer Protocol)
 - ▣ FTP 提供交互式的访问，允许客户指明文件的类型与格式，允许文件具有存取权限
- 网络环境下复制文件的复杂性
 - ▣ 计算机存储数据的格式不同
 - ▣ 文件的目录结构和文件命名的规定不同
 - ▣ 对于相同的文件存取功能，操作系统使用的命令不同
 - ▣ 访问控制方法不同
- FTP 屏蔽了各计算机系统的细节，因而适合于在异构网络中任意计算机之间传送文件

6.2.2 FTP 的基本工作原理

- 只提供文件传送的一些基本的服务
- FTP 的主要功能
 - ▣ 减少或消除在不同操作系统下处理文件的不兼容性。
- FTP 使用客户服务器方式
 - ▣ FTP 的服务器进程由两大部分组成：一个**主进程**，负责接受新的请求；另外有**若干个从属进程**，负责处理单个请求

主进程的工作步骤

- 打开熟知端口（21），等待客户进程发出连接请求
- 启动从属进程
 - ▣ 处理客户进程发来的请求；
 - ▣ 从属进程对客户进程的请求处理完毕后即终止
- 回到等待状态
 - ▣ 继续接受其他客户进程发来的请求
 - ▣ 主进程与从属进程的处理是并发地进行

两个连接

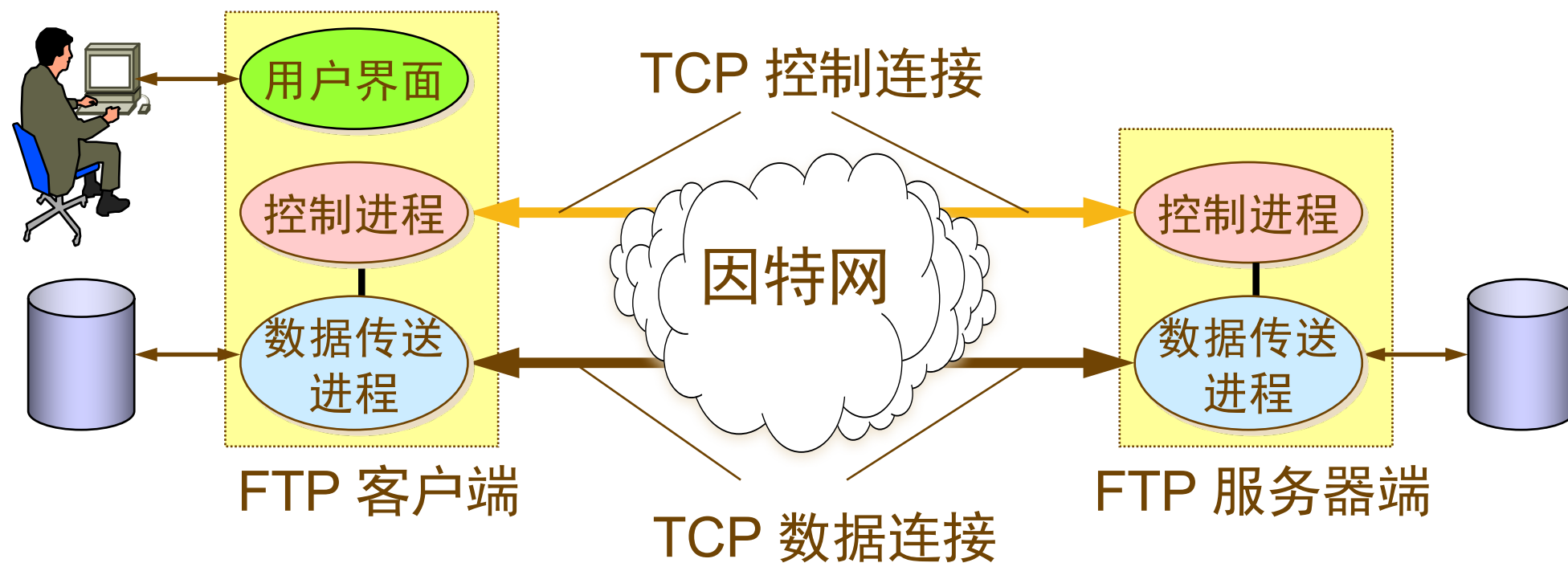
□ 控制连接

- ▣ 在整个会话期间一直保持打开，FTP客户发出的传送请求通过控制连接发送给服务器端的控制进程
- ▣ 控制连接不用来传送文件

□ 数据连接

- ▣ 实际用于传输文件
- ▣ 控制进程在接收到FTP客户发送来的文件传输请求后，创建“数据传送进程”和“数据连接”
- ▣ 数据传送进程实际完成文件的传送，在传送完毕后关闭“数据传送连接”并结束运行

FTP 使用的两个TCP连接



两个不同的端口号

- 客户进程发出建立连接请求时，寻找连接服务器进程的熟知端口(21)
 - ▣ 同时，告诉服务器进程自己的另一个端口号码，用于建立数据传送连接。
- 服务器进程用自己传送数据的熟知端口(20)与客户进程所提供的端口号码建立数据传送连接
- 由于 FTP 使用了两个不同的端口号，所以数据连接与控制连接不会发生混乱

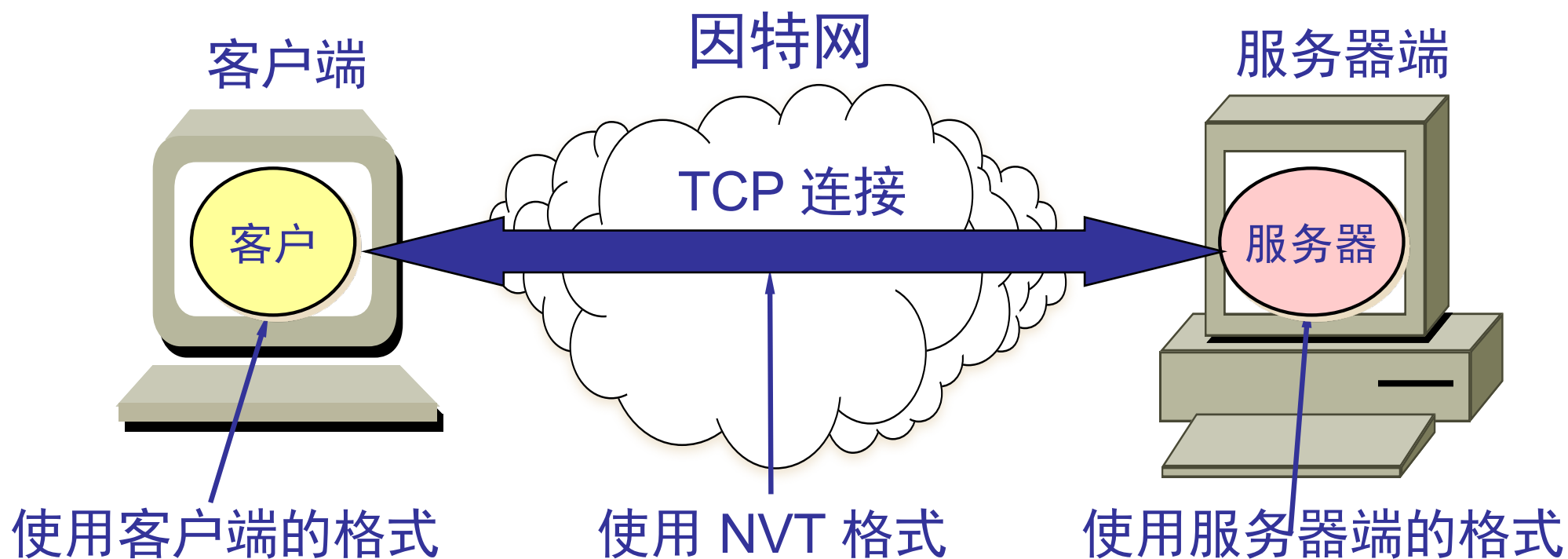
6.2.3 简单文件传送协议 TFTP (Trivial File Transfer Protocol)

- 是一个很小的且易于实现的文件传送协议
- 使用客户服务器方式，UDP数据报
 - ▣ 因此，TFTP 需要有自己的差错改正措施
- TFTP特点
 - ▣ 只支持文件传输而不支持交互
 - ▣ 没有一个庞大的命令集，没有列目录的功能，也不能对用户进行身份鉴别
 - ▣ `tftp [-i] [Host] [{get | put}] [Source] [Destination]`

6.3 远程终端协议 TELNET

- 用户用TELNET可通过TCP连接登录到远地的另一个主机上
- TELNET能将用户的击键传到远地主机，同时也能将远地主机的输出通过TCP连接返回到用户屏幕
 - ▣ 这种服务是透明的，因为用户感觉到好像键盘和显示器是直接连在远地主机上
- TELNET 也使用C/S方式
 - ▣ 和FTP的情况相似，服务器中的主进程等待新的请求，并产生从属进程来处理每一个连接

TELNET 使用网络虚拟终端 NVT 格式



网络虚拟终端 NVT 格式

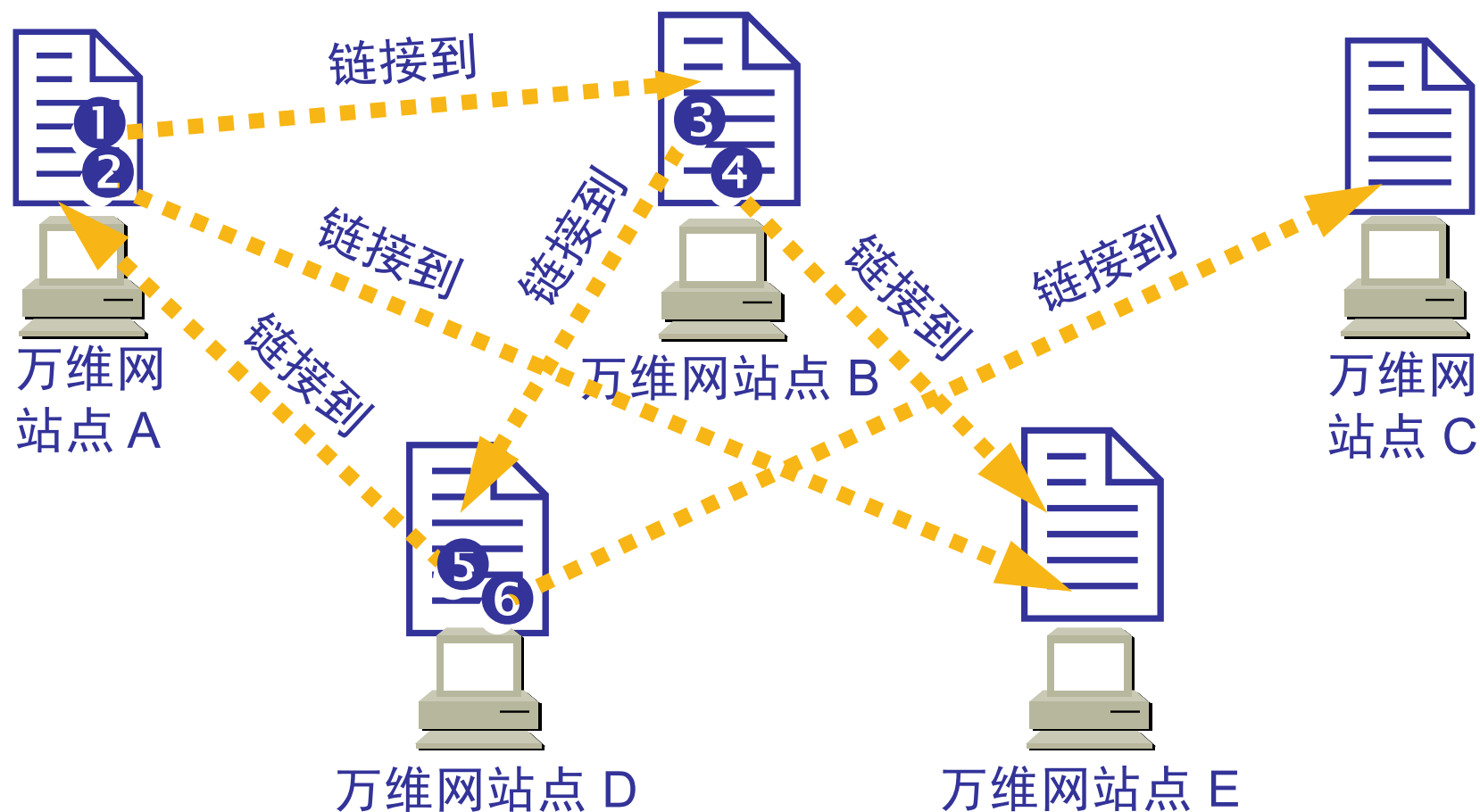
- 客户软件把用户的击键和命令转换成NVT 格式，并送交服务器
- 服务器软件把收到的数据和命令，从NVT 格式转换成远地系统所需的格式
- 向用户返回数据时，服务器把远地系统的格式转换为NVT格式，本地客户再从NVT 格式转换到本地系统所需的格式

6.4 万维网 WWW

6.4.1 概述

- 万维网 WWW (World Wide Web)
 - 并非某种特殊的计算机网络。
 - 万维网是一个大规模的、联机式的信息储藏所
- 访问方式：链接

万维网提供分布式服务



超媒体与超文本

- 万维网是分布式超媒体(hypermedia)系统，它是超文本(hypertext)系统的扩充
- 超文本
 - ▣ 一个超文本由多个信息源链接成，利用一个链接可使用户找到另一个文档
 - ▣ 这些文档可以位于世界上任何一个接在因特网上的超文本系统中
 - ▣ 超文本是万维网的基础
- 超媒体
 - ▣ 超媒体与超文本的区别是文档内容不同。
 - ▣ 超文本文档仅包含文本信息，而超媒体文档还包含其他表示方式的信息，如图形、图像、声音、动画，甚至活动视频图像。

万维网的工作方式

- C/S方式
- 客户端：浏览器
- 服务器：万维网文档所驻留的计算机运行服务器程序，因此这个计算机也称为万维网服务器
- 页面：在一个客户程序主窗口上显示出的万维网文档

万维网必须解决的问题

- 怎样标志分布在整个因特网上的万维网文档？
 - ▣ 统一资源定位符（URL）
 - ▣ 每一个文档在因特网范围内的唯一标识
- 用何种协议实现万维网上各种超链的连接？
 - ▣ 超文本传送协议（HTTP）
- 怎样使各种万维网文档在因特网上的各种计算机上显示出来，并显示链接？
 - ▣ 超文本标记语言（HTML）
- 怎样使用户能够很方便地找到所需的信息？
 - ▣ 搜索引擎

6.4.2 统一资源定位符 URL

- URL：因特网上的资源的位置和访问方法的一种表示
- URL 的一般形式

<协议>://<主机>:<端口>/<路径>

ftp —— 文件传送协议 FTP

http —— 超文本传送协议 HTTP

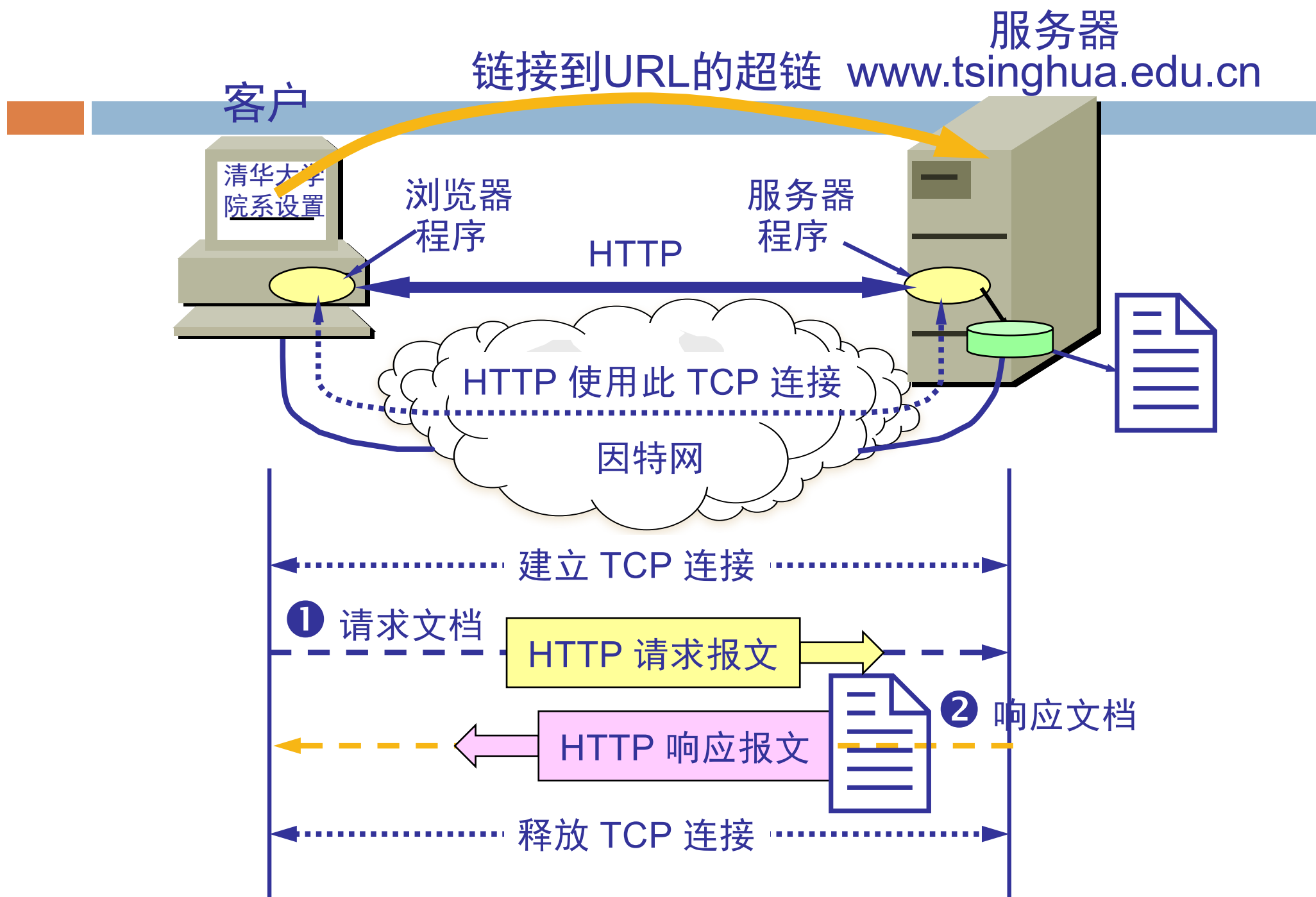
News —— 网络新闻组协议

news://msnews.microsoft.com

6.4.3 超文本传送协议 HTTP

- HTTP是面向事务的应用层协议
 - ▣ 是万维网上能够可靠地交换文件（文本、声音、图像等）的重要基础。
 - ▣ 面向事务——信息交换的过程是一个事务，即具有原子性

万维网的工作过程



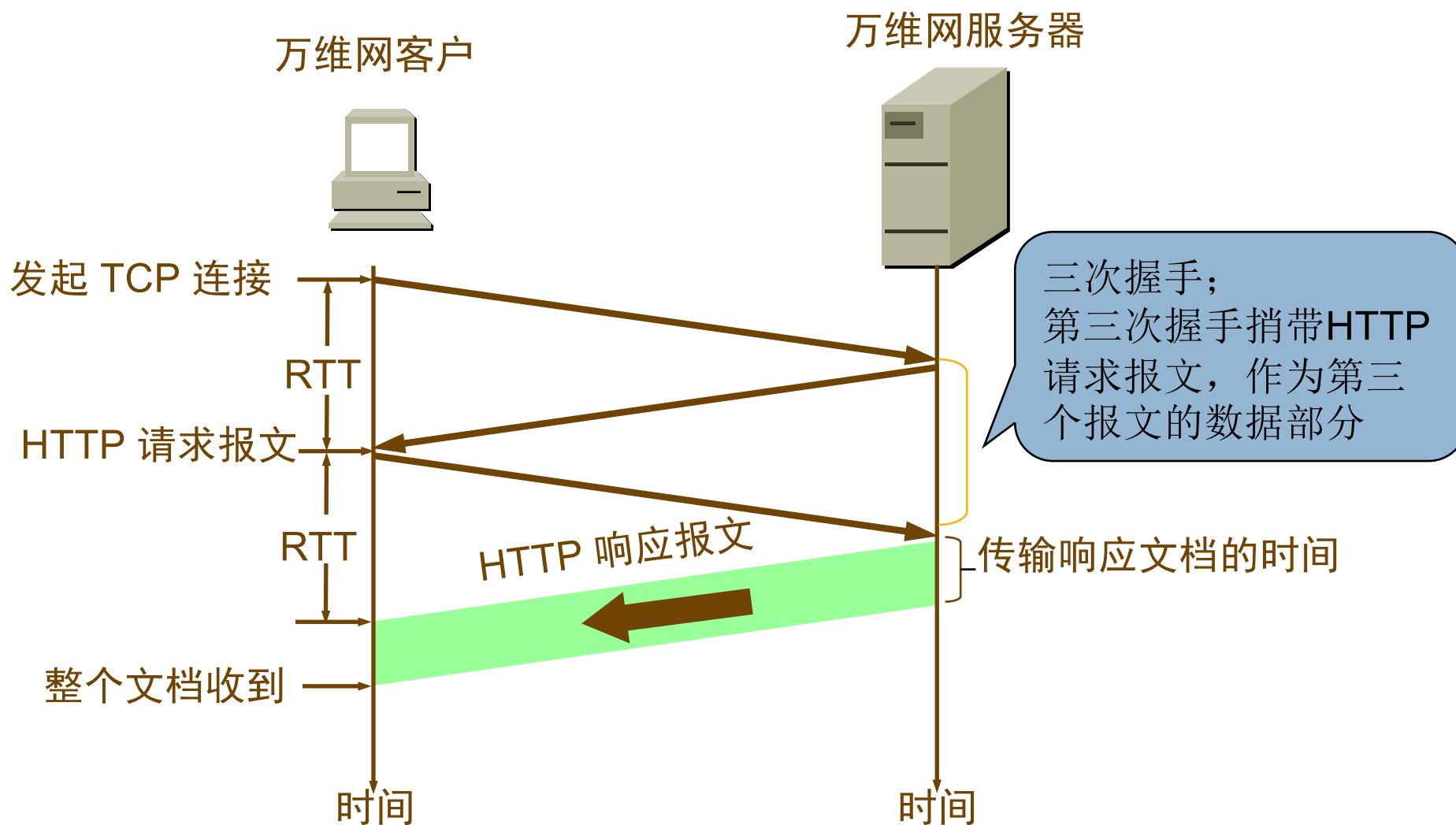
用户点击鼠标后所发生的事件

1. 浏览器分析超链指向的页面的URL
2. 浏览器向DNS请求解析www.tsinghua.edu.cn 的IP 地址
3. 域名系统DNS 解析出清华大学服务器的 IP 地址
4. 浏览器与服务器建立TCP 连接
5. 浏览器发出取文件命令：`GET /chn/yxsx/index.htm`
6. 服务器给出响应，把文件index.htm发给浏览器
7. TCP连接释放
8. 浏览器显示“清华大学院系设置” 文件index.htm 中的所有文本

HTTP 的主要特点

- HTTP是面向事务的客户服务器协议
- HTTP 1.0 协议是无状态的(stateless)
 - ▣ 每一次请求和响应都是独立的；
 - ▣ 对同一资源的多次访问过程是独立的，不记录用户的行为状态；
- HTTP协议本身也是无连接的，虽然它使用了面向连接的TCP 向上提供的服务

请求一个万维网文档所需的时间



持续连接

(persistent connection)

- HTTP/1.1 协议使用持续连接
 - ▣ 万维网服务器在发送响应后仍然在一段时间内保持这条连接，使同一个客户（浏览器）和该服务器可以继续在这条连接上传送后续的 HTTP 请求报文和响应报文。
- 目前浏览器的默认设置就是使用HTTP/1.1

持续连接的两种工作方式

□ 非流水线方式

- ▣ 客户在收到前一个响应后才能发出下一个请求

□ 流水线方式

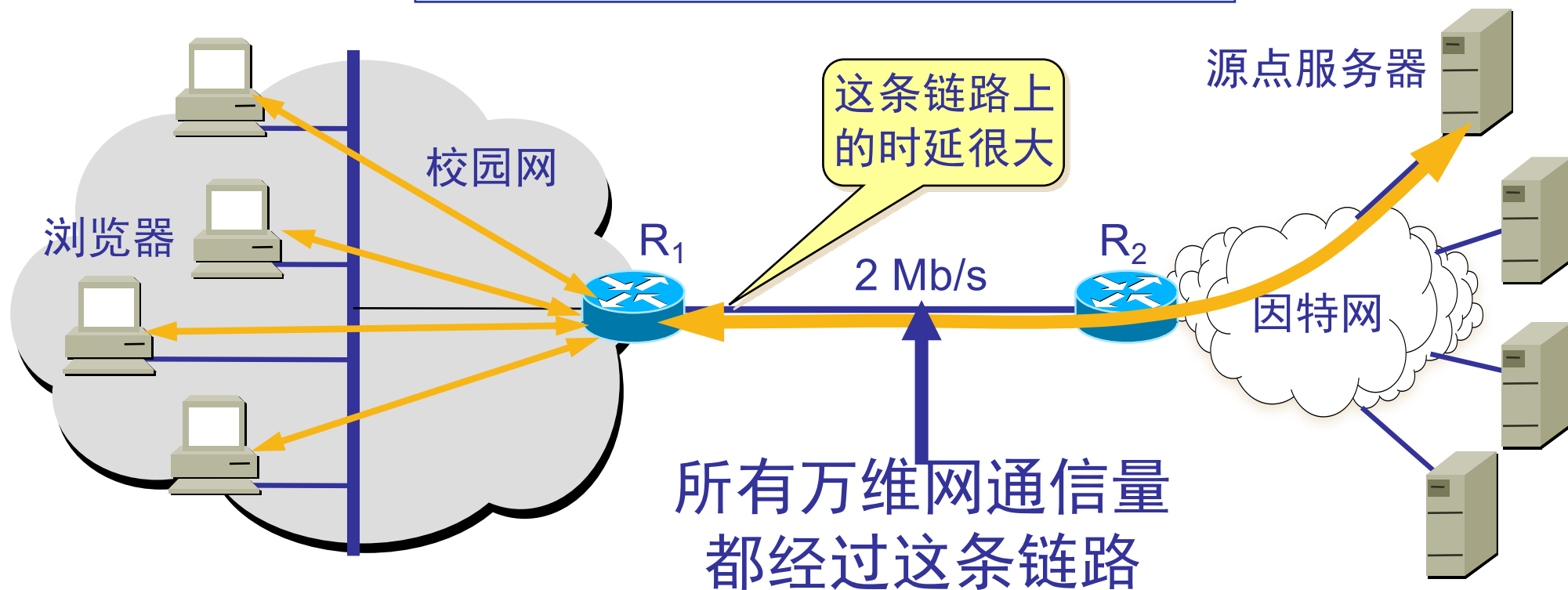
- ▣ 客户在收到 HTTP 的响应报文之前就能够接着发送新的请求报文

代理服务器(proxy server)

- 代理服务器，proxy server，又称为万维网高速缓存
 - ▣ 代表浏览器发出HTTP请求
 - ▣ 缓存最近的一些请求和响应
 - ▣ 当相同请求到达时，代理服务器返回暂存的响应，不需要按URL 的地址再去因特网访问该资源

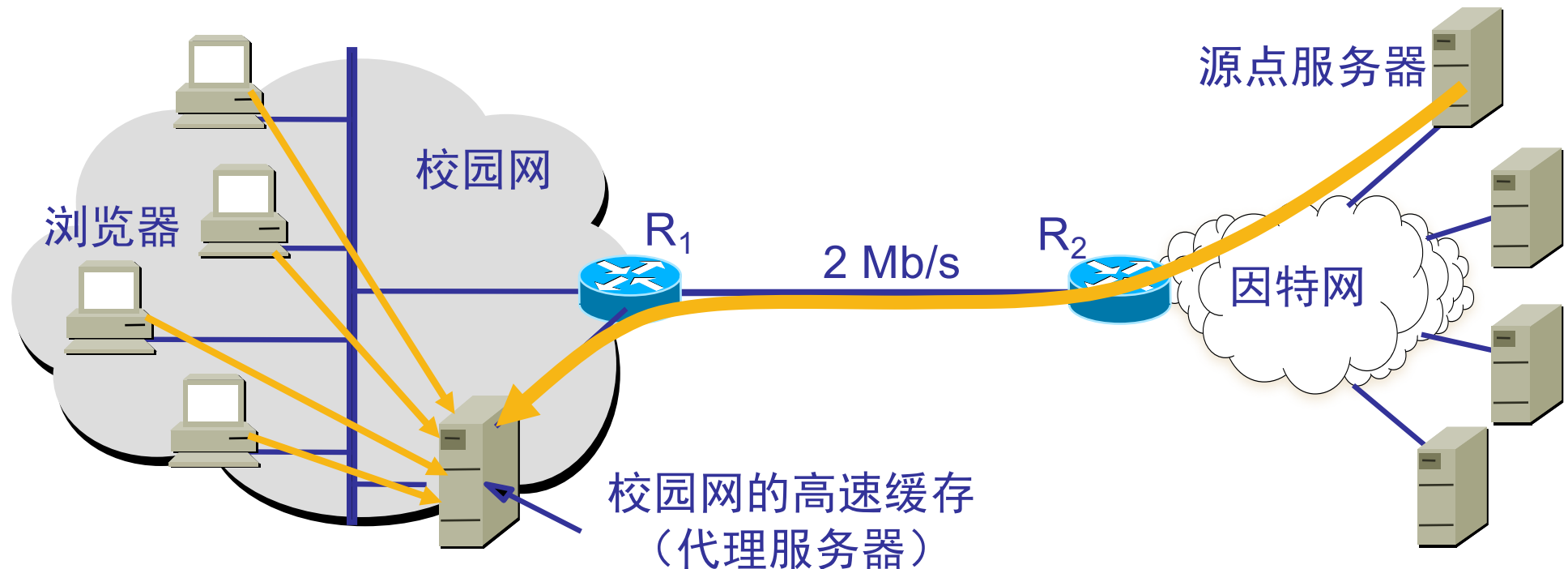
使用高速缓存可减少 访问因特网服务器的时延

没有使用高速缓存的情况



使用高速缓存的情况

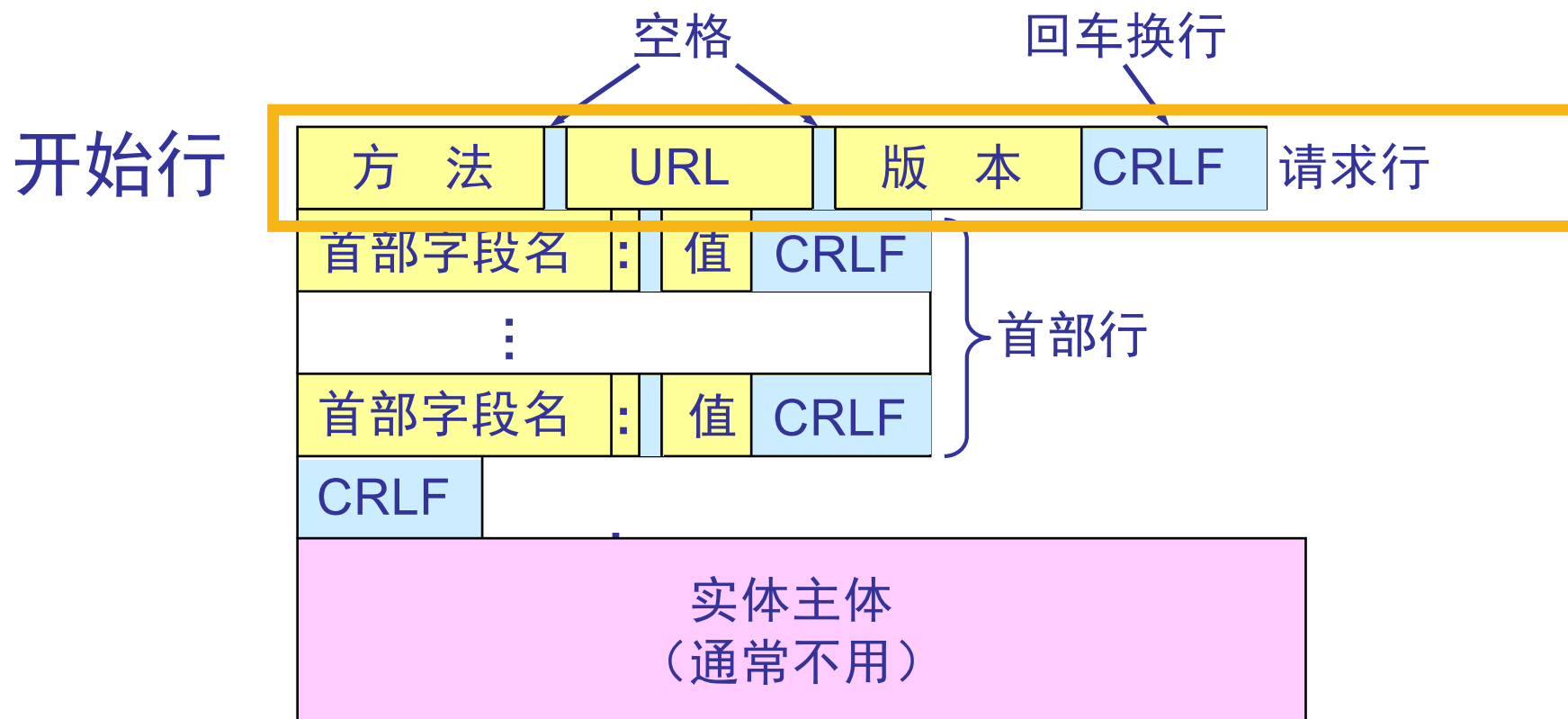
浏览器先与校园网的高速缓存建立 TCP 连接，并向高速缓存发出 HTTP 请求报文；若有缓存？ 否则？



3. HTTP的报文结构

- 请求报文——从客户向服务器发送请求报文
- 响应报文——从服务器到客户的回答
- HTTP报文是面向文本的(text-oriented)，因此在报文中的每一个字段都是ASCII 码串

HTTP 的报文结构（请求报文）



报文由三个部分组成，即**开始行**、**首部行**和**实体主体**。
在请求报文中，开始行就是请求行。

HTTP 请求报文的一些方法

方法（操作）	意义
OPTION	请求一些选项的信息
GET	请求读取由 URL 所标志的信息
HEAD	请求读取由 URL 所标志的信息的首部
POST	给服务器添加信息
PUT	在指明的 URL 下存储一个文档
DELETE	删除指明的 URL 所标志的资源
TRACE	用来进行环回测试的请求报文
CONNECT	用于代理服务器



□ 首部行

- ▣ 说明浏览器、服务器或者报文主体的一些信息；

□ 实体主体

- ▣ 可看作是所发送实体的内容；

□ 实体

- ▣ 实体标题域和实体主体两个部分组成。前者定义了一些元信息，例如：实体的编码类型、长度等

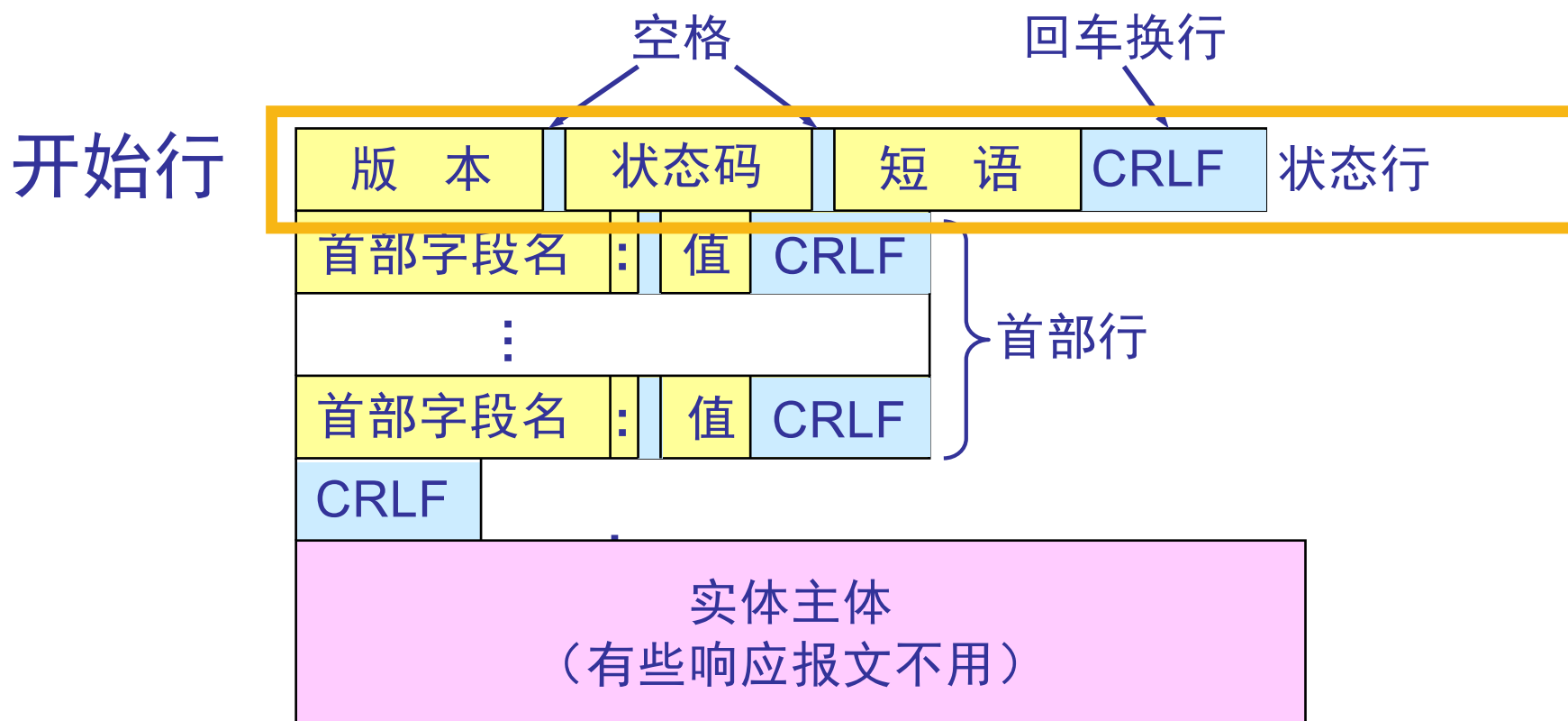
举例：请求报文

- **POST /Music/SearchServlet HTTP/1.1**
- **Referer: <http://localhost/musicSearch.html>** （请求发起的页面）
- **Connection: Keep-Alive**
- **User-Agent: Mozilla/4.72 [en] (WinNT 5.0; U)**
- **Host: localhost:8080**
- **Cookie: UserID=spot**
- **Accept: image/gif, image/x-xbitmap, image/jpeg, */***
- **Accept-Language: en**
- **Accept-Charset: iso-8859-1,*,utf-8**

- **Content-type: application/x-www-form-urlencoded**
- **Content-length: 50** （属于实体标题域）

- **song_title=Hello&song_artist=Jones&limit_number=20**

HTTP 的报文结构（响应报文）



响应报文的开始行是状态行。

状态行包括三项内容，即 HTTP 的版本，状态码，以及解释状态码的简单短语。

状态码都是三位数字

- 1xx : 通知信息，如请求收到了或正在进行处理
- 2xx : 成功
- 3xx : 重定向，表示要完成请求还必须采取进一步的行动
- 4xx : 客户的差错，如请求中有错误的语法
- 5xx : 服务器的差错，如服务器失效

举例：响应报文

- **HTTP/1.1 200 ok**
- **Content-Type: text/html**
- **Set-Cookie: sessionid=5H2HXGYAAAAAEWQAAAAZJCI**
- **Expires: Thu, 01 Dec 1994 16:00:00 GMT**
- **Set-Cookie: USERID=spot;Expires=Fri, 08-Jun-2001 21:54:37 GMT**
- **Content-Language: en**

- **<HTML>**
- **<BODY>**
- **<H1>Very simple dynamic document created on 01-Jun-2001</H1>**
- **</BODY>**
- **</HTML>**

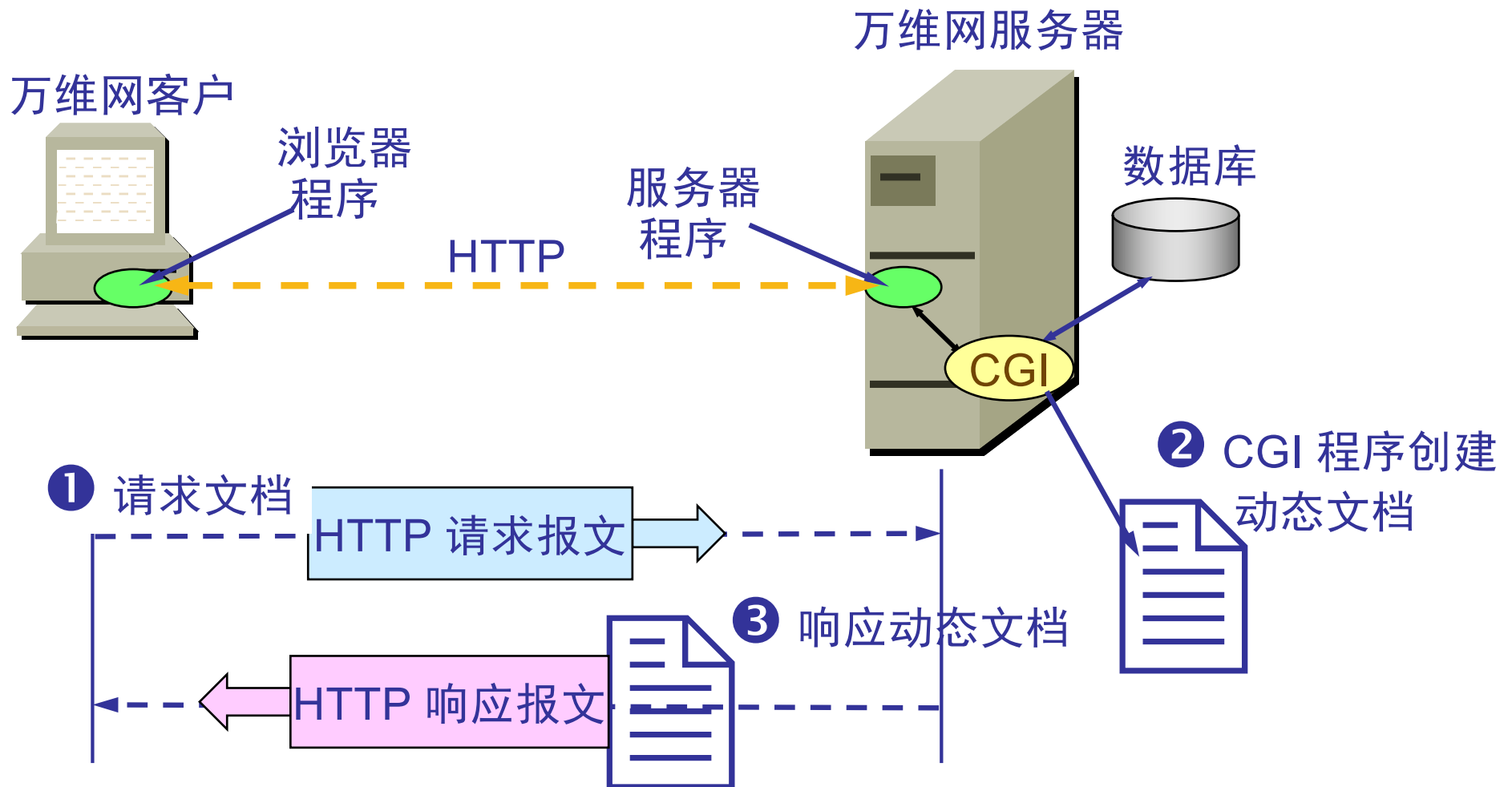
4. 在服务器上存放用户的信息

- HTTP是无状态的
- 万维网站点可使用 Cookie 来跟踪用户
 - ▣ Cookie表示在HTTP服务器和客户之间传递的状态信息
 - ▣ 使用Cookie的网站服务器为用户产生一个唯一的识别码。利用此识别码，网站就能够跟踪该用户在该网站的活动。

6.4.4 万维网的文档

- 1. 超文本标记语言 HTML
- 2. 动态万维网文档
 - ▣ 文档的内容是在浏览器访问万维网服务器时才由应用程序动态创建

扩充了功能的万维网服务器



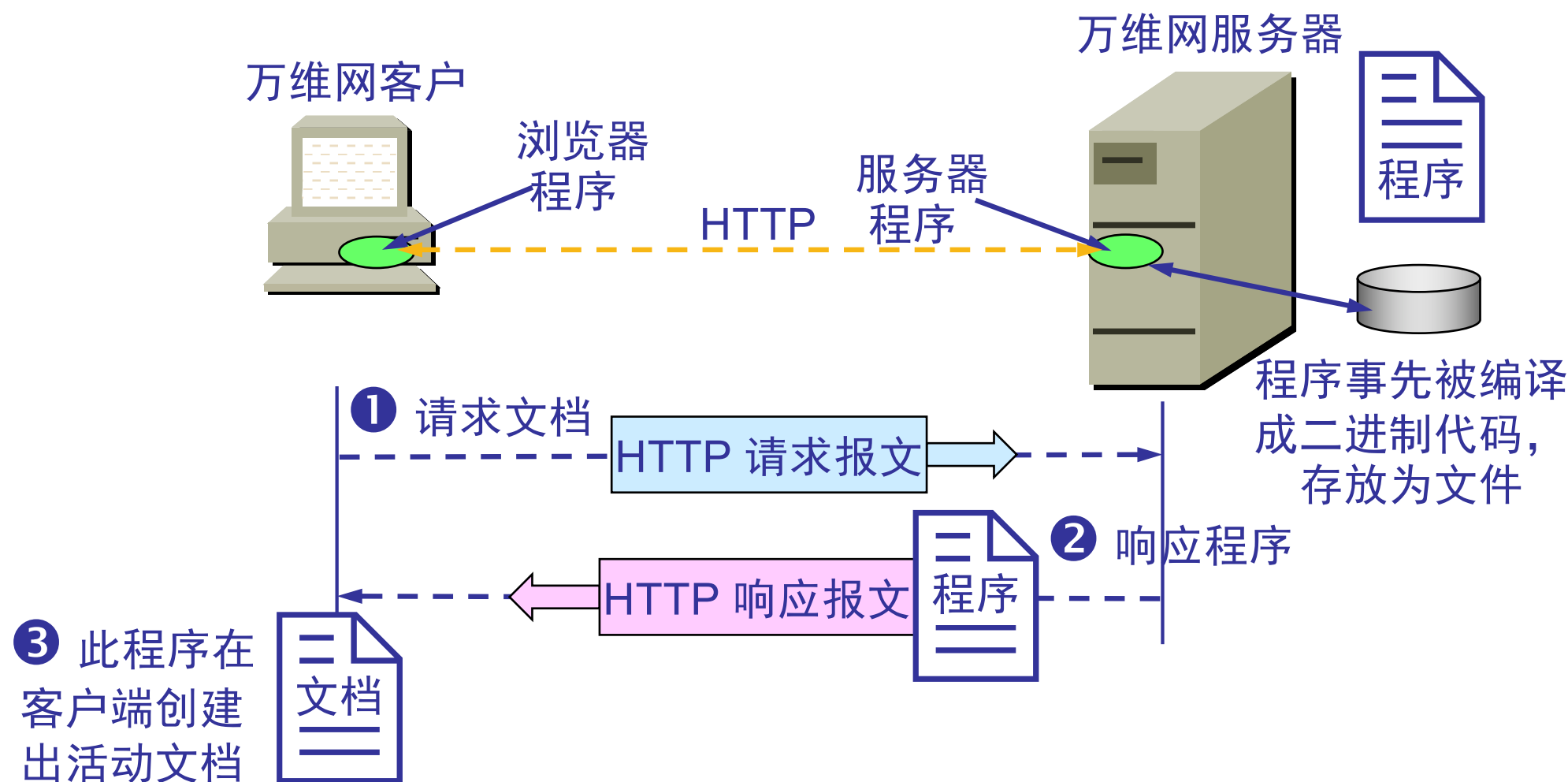
通用网关接口 CGI (Common Gateway Interface)

- CGI是一种标准
 - ▣ 定义了动态文档应如何创建
 - ▣ 输入数据应如何提供给应用程序
 - ▣ 输出结果应如何使用
- 万维网服务器与CGI 的通信遵循CGI 标准

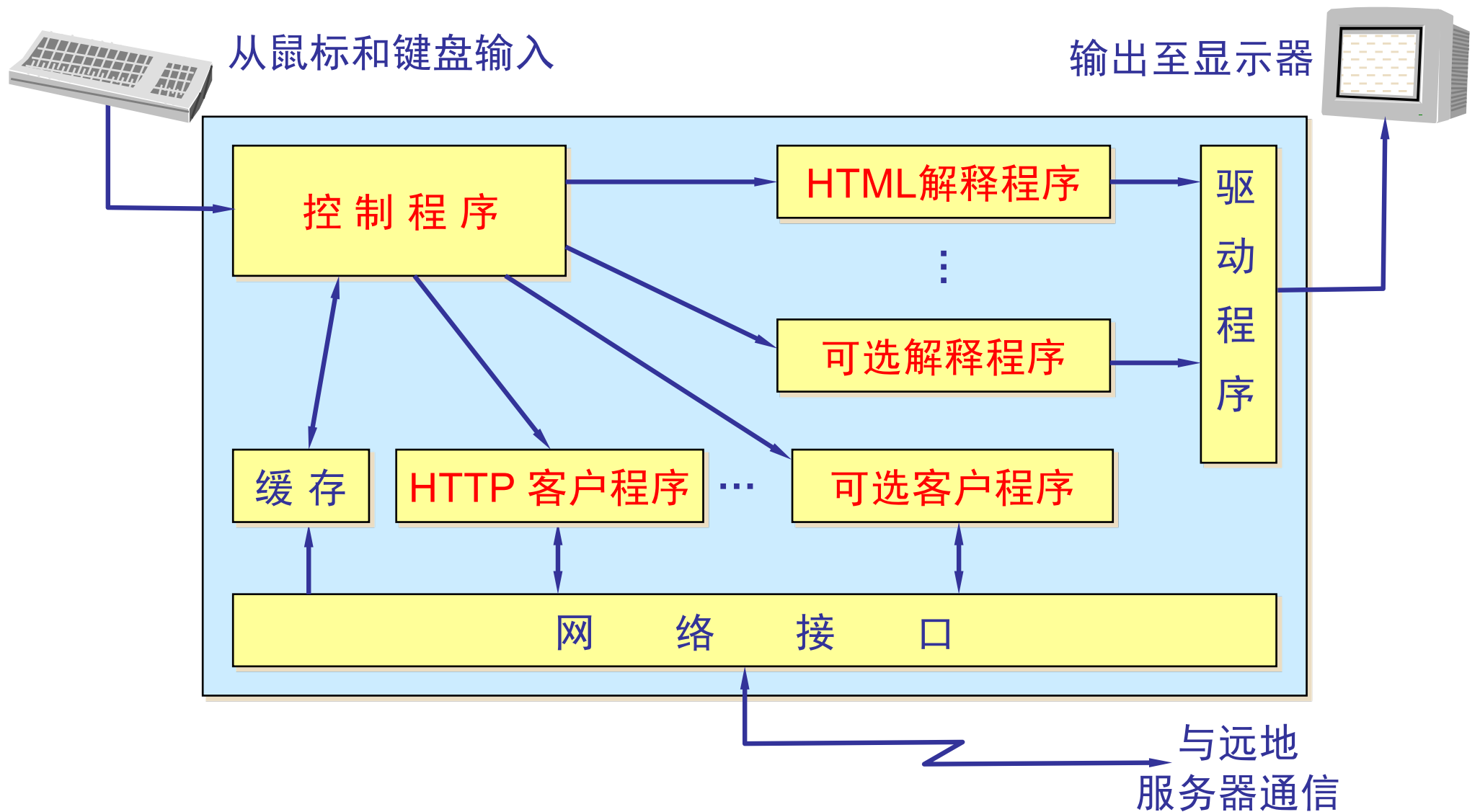
3. 活动万维网文档

- 活动文档(active document)
 - ▣ 浏览器请求一个活动文档时，服务器就返回一段程序副本在浏览器端运行
 - ▣ 活动文档程序可与用户直接交互
- Java applet

活动文档在客户端创建



4. 浏览器的结构



浏览器的主要组成部分

- 浏览器有一组客户、一组解释程序，以及管理这些客户和解释程序的控制程序
- 控制程序
 - ▣ 是其中的核心部件，解释鼠标的点击和键盘的输入，并调用有关的组件来执行用户指定的操作。
 - ▣ 例如，当用户用鼠标点击一个超链时，控制程序就调用一个客户从远地服务器上取回该文档，并调用解释程序向用户显示该文档。

浏览器中的缓存



- 浏览器将取回的页面副本放入本地磁盘的缓存中
- 访问页面时，若缓存中保存了该项，浏览器就直接从缓存中得到该项副本

Thanks, Any Question?

The End.

chenwh@nwnu.edu.cn

