WWW (World Wide Web) Basics

WWW: all the resources and users on the Internet that are using the HTTP (hypertext Transfer Protocol), a system of interlinked hypertext documents accessed via the Internet

WWW v.s. Internet
Internet — at the bottom, a networking infrastructure and related communication standards
WWW — on the top, an information sharing model

WWW Terminologies
• The Web: a true information superhighway
• URL (Uniform Resource Locator): designate a specific webpage on a specific web-server
• HTTP (HyperText Transfer Protocol): an application-level transfer protocol standard
• HTML (HyperText Markup Language): a document format standard

WWW Components
1. Structural Components
• Clients/browsers
• Servers
• Caches
• Internet
2. Semantic Components
• Hyper Text Transfer Protocol (HTTP)
• Hyper Text Markup Language (HTML)
• Uniform Resource Locators (URLs)

The Web: an information system that links data from many different Internet services under one set of protocols

Web clients (browsers): browsers, interpret HTML delivered from Web servers
Hypertext links: connect different documents and information resources together
HTTP is easily modified to incorporate new data formats and uses

The Web model unites the divers Internet resources under a single system, relying on servers and Web-browsers to "negotiate" or handle data compatibility

WWW Clients
Client is called a "browser", the server is where the data is stored and it is software that runs on well known port (80)
Browser and server talk using a protocol — HTTP

Client properties
All different browsers —> same information but display it differently depend on their capabilities
http:// —> indicate to the browser that it is talking HTTP
A client machine use a browser to download a Web page:
• Entering a URL
• Clicking on a HyperLink
Web browsers = universal clients — can talk other protocols besides HTTP (ftp://)

The Web is capable of accessing data on different internet services (Web pages, FTP, Email service, ...)

WWW Servers
The server is software that is running on a remote location. Its job is to make "pages" available to the client
过程:
Every Web site has a server process listening to TCP port 80 for incoming connections from clients
After a connection has been established, the client sends one request and the server sends one response
Then the connection is released

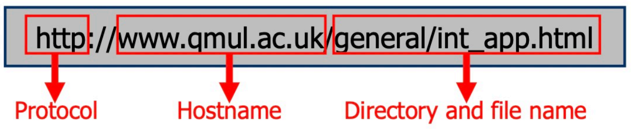
- The protocol that defines the legal request and response — HTTP
- The operation is **Stateless**

URLs (Uniform Resource Locators)
The global address of a Web page is described by its URL
URLs identify (3)
- The protocol you want to talk
- The site (domain name or IP address) you want to go to
- Possible the item you want to see

Form:
protocol://hostname [:port]/directory/item-you-want

Dynamic documents: resources can be dynamically generated on server upon query

Structure of URLs
A URL consists of three parts:
- The protocol — http to ftp...
- The DNS name of the host
- The directory and file name



Defaults
- Protocol: http by default
- Port: 80 by default
- Index.html, default.html if no file-name given ....

HTML
HTML: the agreed upon markup language for the Web
- Depend on the browser and what version you use

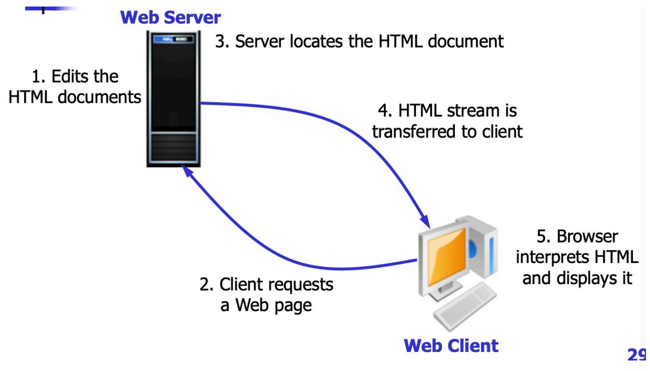
Static v.s. Dynamic
Before: WWW — static documents
Static documents indicates:

1. Each URL corresponded to a single file stored on some hard disk
2. Edit in HTML format .html, .htm

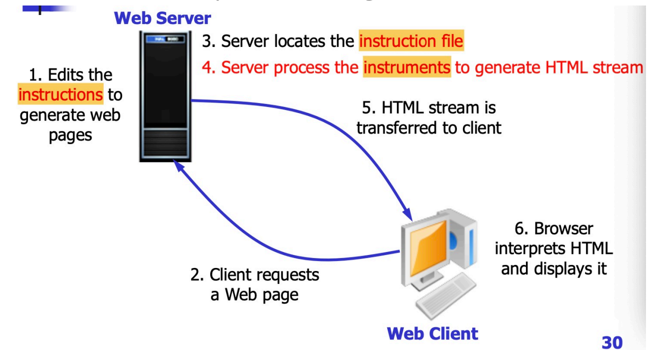
Today: WWW — built at request time
Dynamic indicates:

1. The URL does not correspond to a single file
2. Generated dynamically by some programs such as CGI, PHP

Procedure of Static Pages


Web Server
3. Server locates the HTML document
1. Edits the HTML documents
4. HTML stream is transferred to client
5. Browser interprets HTML and displays it
2. Client requests a Web page
Web Client
29

Procedure of Dynamic Pages


Web Server
3. Server locates the instruction file
4. Server process the instruments to generate HTML stream
1. Edits the instructions to generate web pages
5. HTML stream is transferred to client
6. Browser interprets HTML and displays it
2. Client requests a Web page
Web Client
30

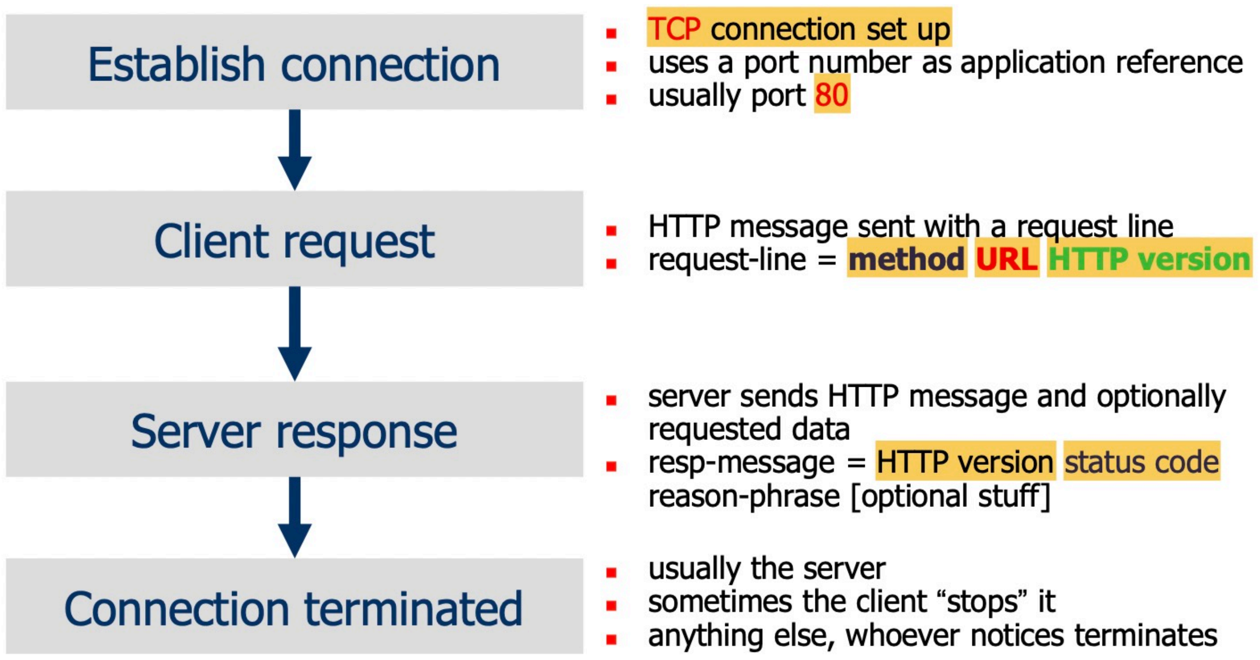
Instrument —> 动态页面的内容一般都是依靠服务器端的程序来生成

CGI (Common Gateway Interface)
CGI: a standard for interfacing external applications with informations servers

- A plain HTML document that the web client retrieves is static: a text file doesn't change
- A CGI program: executed in realtime, can output dynamic information for the server

HTTP
HTTP Features

- Application layer protocol for client/server communication
- Request/response based
- Stateless (不记录每个用户浏览什么)
- Bi-directional transfer (TCP)
- Capability negotiation
- Support for cache
- Support for intermediaries: HTTP proxy



Establish connection
- TCP connection set up
- uses a port number as application reference
- usually port 80

Client request
- HTTP message sent with a request line
- request-line = **method URL HTTP version**

Server response
- server sends HTTP message and optionally requested data
- resp-message = HTTP version status code reason-phrase [optional stuff]

Connection terminated
- usually the server
- sometimes the client "stops" it
- anything else, whoever notices terminates

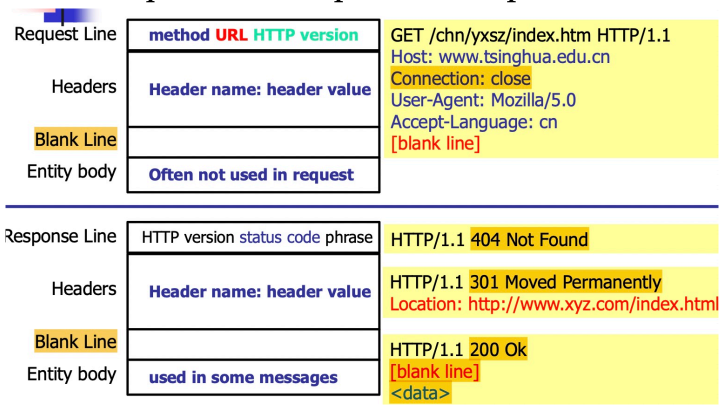
HTTP Transaction (交互过程)

Important points:
- Port 80 —> server port
- Request-line = method URL HTTP version
- Response-message = HTTP version status code

过程:
1. The browser determines the URL
2. Browser asks DNS for the IP address of web-page
3. DNS returns the IP address (ip)
4. The browser makes a TCP connection ~~(IP address (ip), port:60)~~
5. The browser sends a get request (GET /dir/FileName.html HTTP/1.0)
6. The remote server sends the file FileName.html
7. TCP connection is released
8. The browser displays all the text in FileName.html
9. The browser fetches and displays all the images in FileName.html

HTTP methods
- GET: retrieve document specified by URL
- PUT: store specified document under given URL
- POST: give information (e.g. annotation) to the server

HTTP Request and Response Example



Request Line	method URL HTTP version	GET /chn/yxsz/index.htm HTTP/1.1
Headers	Header name: header value	Host: www.tsinghua.edu.cn Connection: close User-Agent: Mozilla/5.0 Accept-Language: cn
Blank Line		[blank line]
Entity body	Often not used in request	
Response Line	HTTP version status code phrase	HTTP/1.1 404 Not Found
Headers	Header name: header value	HTTP/1.1 301 Moved Permanently Location: http://www.xyz.com/index.html
Blank Line		HTTP/1.1 200 Ok [blank line]
Entity body	used in some messages	<data>

Request:
1. Method URL HTTP-version
2. Headers (Host, Connection, User-Agent …)
   - Connection: close —> connection will close later
3. Blank line

Response:
1. HTTP-version status code
2. Blank line before <data>

HTTP Status Codes
1xx — for information only
2xx — action successful
3xx — further action needed (moved permanently, 域名迁移)
4xx — client request error
5xx — server error

HTTP — ASCII/MIME protocol (webpage包含各种type)
It is simple for user at a terminal to communicate directly to a web server
- MIME: multipurpose Internet Mail Extension
- Content types/subtypes defined in MIME

HTTP/1.1 enhancement

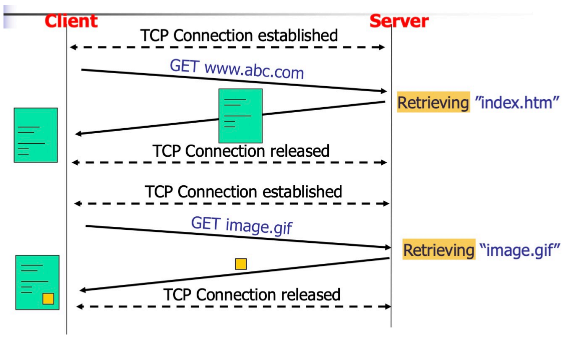
HTTP/1.0 — "stop and wait" protocol
- Separate TCP connection for each file
  - Inefficient use of packet
  - Server must maintain many connections

HTTP/1.1 — better performance
- Persistent connections (TCP 一直connect)
- Pipelining
- Enhanced caching options
- Support for compression

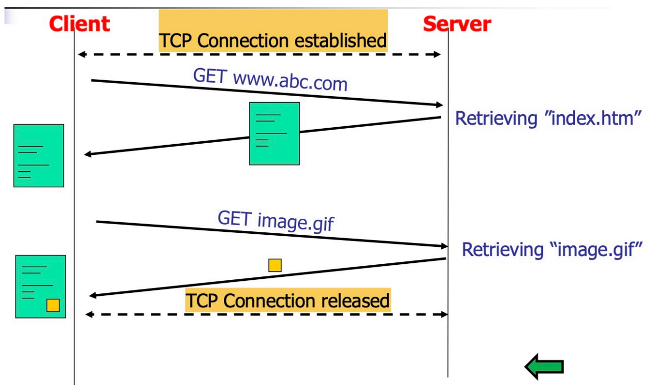
Persistent connections
Use the same TCP connection for transfer of multiple files

Pipelining
Pack several HTTP requests into one TCP/IP packet

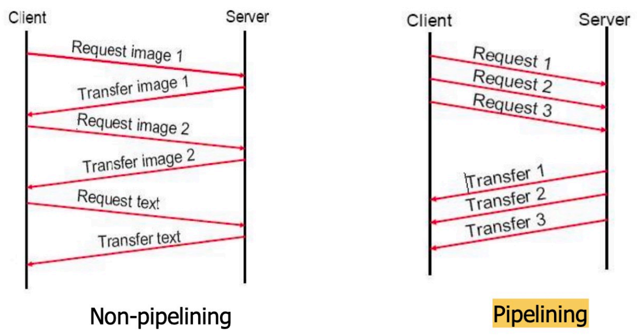
Non-persistent connection



Persistent Connections

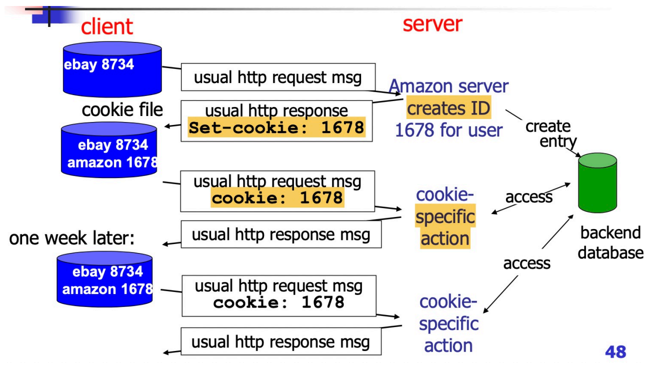


Pipelining (同时发送多个请求)



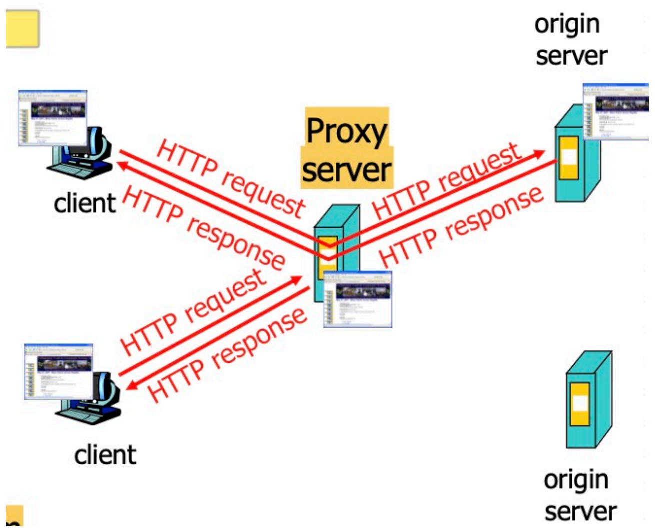
User-server state: cookies
Four component:

1. Cookie header line of HTTP response message
2. Cookie header line of HTTP request message
3. Cookie file kept on user's host managed by user's browser
4. Back-end database at website



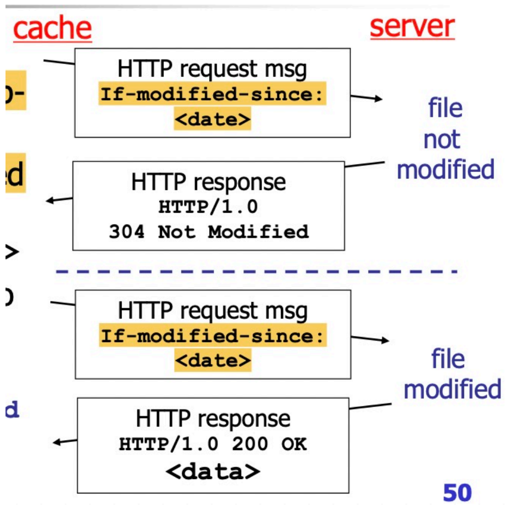
Web Caches (Proxy Server)
Motivation: client与origin server不通，但Proxy server 与origin server通
User sets browser: web accesses via a proxy server (用户设置代理)
Browser sends all HTTP requests to proxy server

- If requested file in cache: proxy server returns file
- Else proxy requests file from origin server, then forwards to client



How to get file between proxy and server?
Conditional get — used between proxy and server
Server does not send required files if cache has up-to-date cached version
(如果已经有最新版)

具体交互行为
Cache: specified date of cached copy in HTTP request
(cache) request msg: If-modified-since: <data>
Server: response contains no object if cached copy is up-to-date
(server) HTTP/1.0 304 Not Modified



HTTPS
A communication protocol for secure communication over a computer network

WWW (Summary)

WWW Components
Structural Components
- Clients/browsers
- Servers
- Caches
- Internet

Semantic Components
- HTTP
- HTML (XML)
- URLs

WWW Clients (browser)
Clients properties:
- Different browsers, same information, different display
- Web browsers (universal clients): most cam talk other protocols (http, ftp …)
- Accessing data on many different Internet services

WWW servers: software that makes "pages" available
- Every Web site has a server process listening to TCP port 80 for incoming connections
- After connection, client sends one request and the server sends one response, then the connection is released.
- The operation is stateless

URL (Uniform resource locators)
1. Protocol
2. Site (domain name or IP address)
3. Item (directory or file name)

HTML: agreed upon markup language for the web
Static v.s. Dynamic
Static: each URL corresponded to a single file
Dynamic: URL doesn't correspond to a single file, generated dynamically by other programs, built at request time

CGI
CGI: interfacing external applications with information servers
It is executed in realtime, so that it can output dynamic information for the server

HTTP (features)
- Application layer protocol for c/s communication
- Request/response based
- Stateless
- Bi-directional transfer
- Capability negotiation
- Support for cache
- Support for intermediaries: HTTP proxy

HTTP Transaction
Request line: method URL HTTP-version
Response message: HTTP-version status code

Status code
1xx, 2xx, 3xx, 4xx, 5xx

HTTP methods
1. GET — retrieve document specified by URL
2. PUT — store specified document under given URL
3. POST — give information to the server

HTTP — Getting Remote Web Pages (全过程)
1. Browser determines the URL
2. Browser asks DNS for IP address
3. DNS returns IP address
4. Browser makes TCP connection to port 80 at this IP address
5. The browser sends a get request
6. Th remote server sends the file
7. TCP connection released
8. The browser displays the content

MIME —> allow multiple content types and subtypes

HTTP/1.1 performance
HTTP/1.0 "stop and wait" — each file a separate TCP connection

HTTP/1.1: 1. Persistent connections 2. Pipelining 3. Enhanced caching options 3. Support for compression

Persistent connections: use the same TCP connection for transfer of multiple files

Pipelining:pack several HTTP requests into one TCP/IP packet

User-server state: cookies
Four components:
1. Cookie header: a. Request message, b. Response message
2. Cookie file kept on user's host managed by browser
3. Back-end database at website

Web Caches (Proxy server)
Browser sends all HTTP requests to proxy server
If requested file in cache, proxy server returns file
Else, proxy requests file from origin server, then forward to client

Conditional get
Server does not send files if cache has up-to-date cached version
Cache: specify date of cached copy (If-modified-since: <date>)
Server: response contains no object if cached copy is up-to-date (HTTP/1.0 304 Not Modified)

HTTPS — a communication protocol for secure communication