WWW Technologies

406.424 Internet Applications

Jonghun Park

jonghun@snu.ac.kr

Dept. of Industrial Eng.

Seoul National University

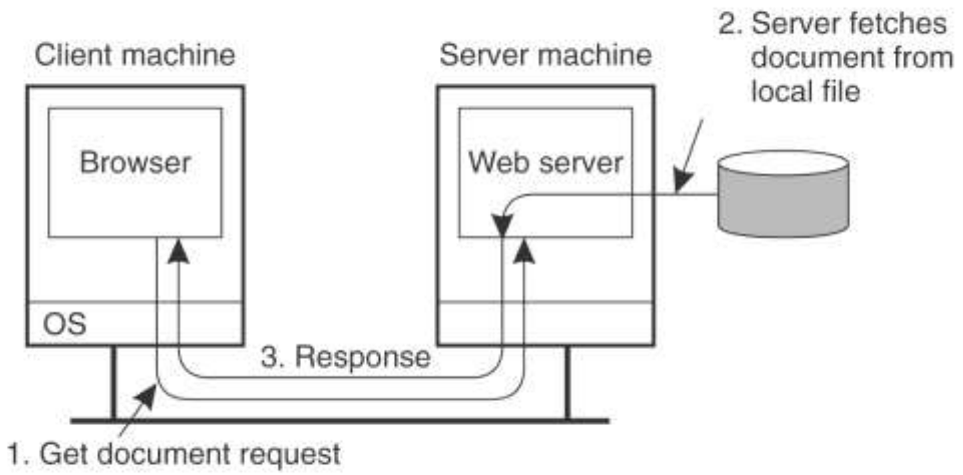
9/1/2010

contents

- **processes in WWW**
  - web clients
  - web proxies
  - web servers
- technology for dynamic Web documents
- technology for active Web documents
- apache Web server

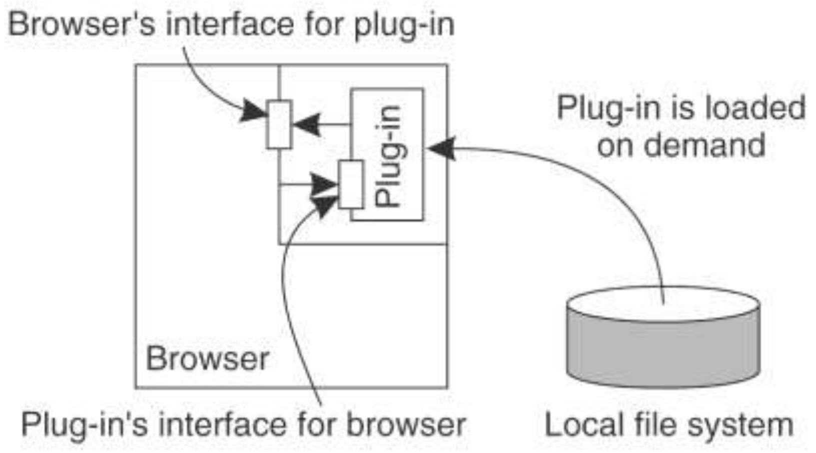
processes in WWW

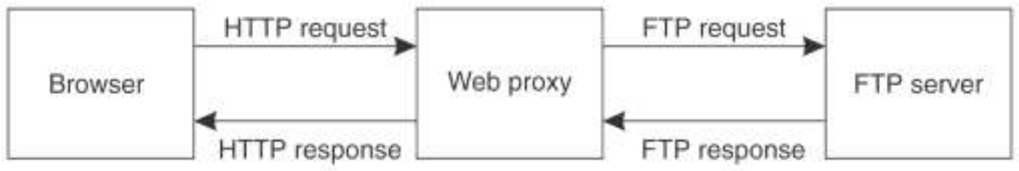
- web server
  - maintains a collection of documents
  - accepts requests for **fetching** a document and transfers it to the client
  - accepts requests for **storing** new documents
- web client
  - browser, spider, agent
  - interacts with web servers
  - should be easily extensible so that it can **support any type of document** returned by a server
  - responsible for **displaying** a document, and accepting input from a user



processes in WWW (cont.)

- plug-in
  - a small program that can be **dynamically downloaded** into a browser for handling a specific document type
  - offers **a standard interface** to the browser, and expect a standard interface from the browser
  - e.g. flash
- web proxy
  - acts as a server to clients and as a client to other proxies or origin servers
  - originally to allow a browser to **handle application-level protocols** other than HTTP

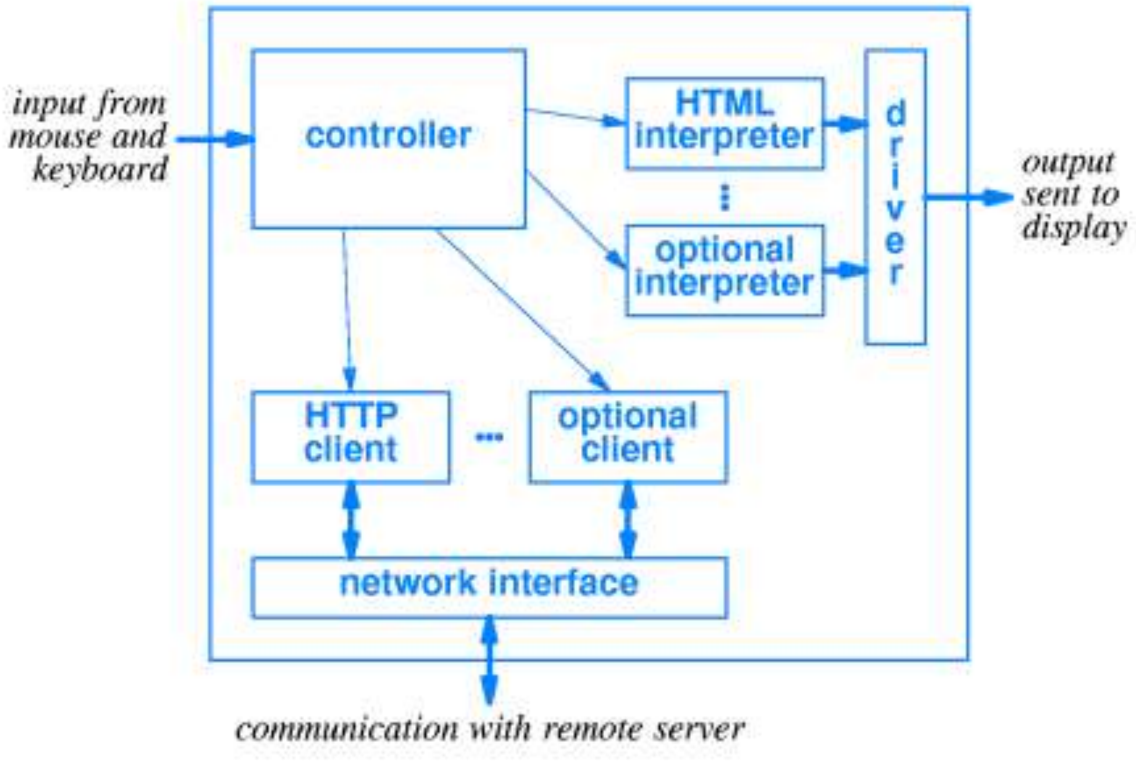

Browser's interface for plug-in
Plug-in
Plug-in is loaded on demand
Browser
Plug-in's interface for browser
Local file system



browsers

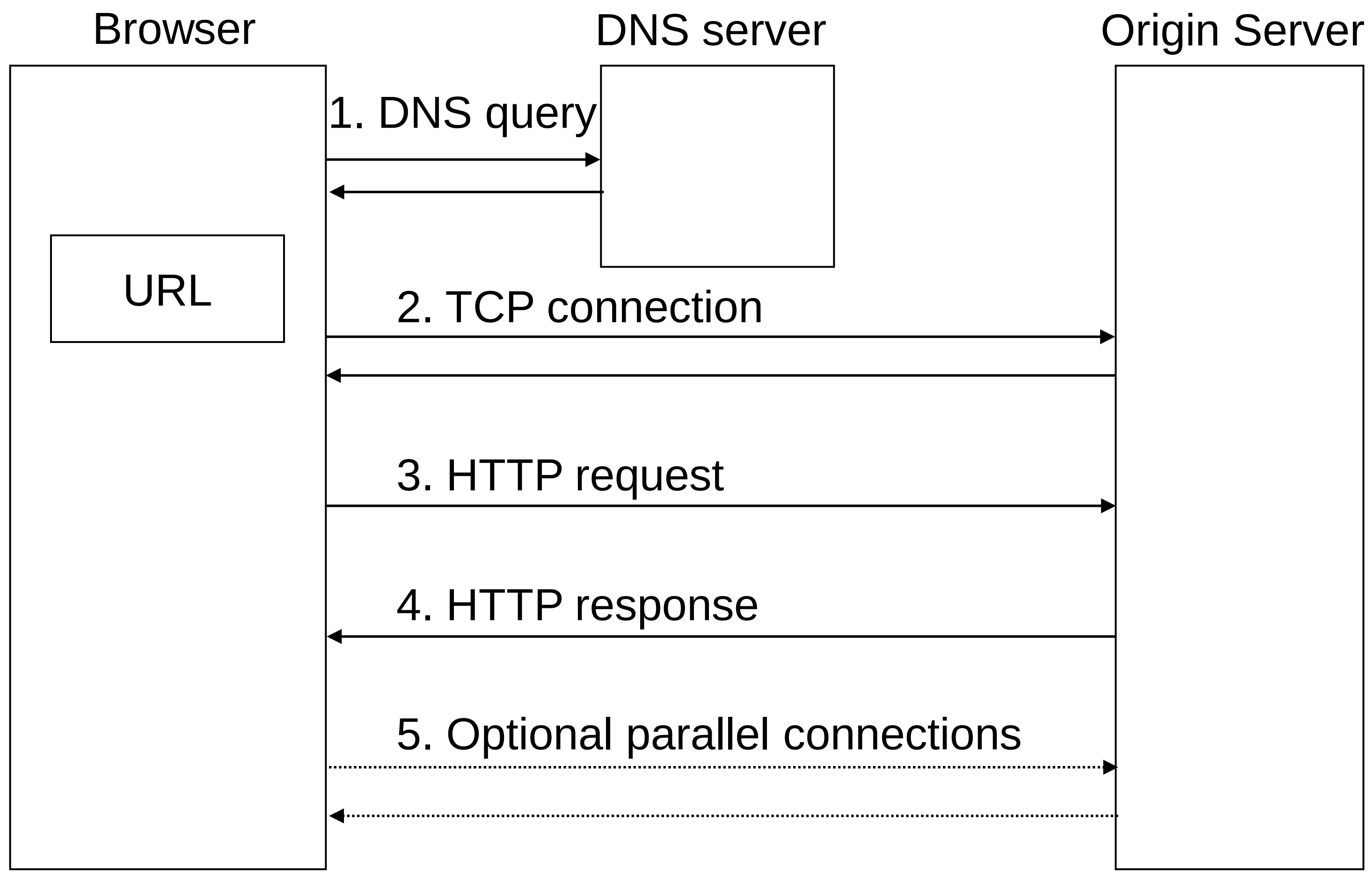
- Mosaic
  - Netscape → Mozilla → Firefox (based on the open source from Netscape Communicator made available in 1998)
    - http://wp.netscape.com/newsref/pr/newsrelease558.html
  - Internet Explorer, Safari, Chrome
- **sends a web request** on behalf of a user and receive the response
- should meet the following requirements
  - notion of current context
  - idea of where else a user could go next and where a user has been recently
  - ability to customize the navigation and the display of the resources accessed
  - ability to search across a collection of resources

browser architecture



- optional clients: FTP client, e-mail client

steps in a browser process



web-related browser functions

- constructs and **sends** an HTTP request, then **receives**, parses, and **displays** the response
- issuing a request
  - clicking on a link, typing the URL, form filling
- browser caching
  - reduces delay perceived by users in obtaining a response from a server
  - cache is said to maintain **consistency** of cached resources if the cache ensures that cached resources are still fresh at the origin server
  - improvements in performance are only helpful if a user decides to view an item again
  - e.g., refresh button: to bypass the browser cache revalidation

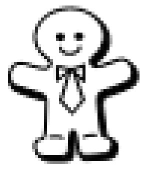
browser configuration

- physical appearance
  - external appearance (window size, presence of buttons and scrollbars), display of embedded images in a page, font, color
  - CSS (Cascading Style Sheets): a standard to describe how documents should be presented
- semantic choices
  - connection related: proxies
  - content related: acceptable language
  - caching
  - handling responses: helper programs
  - cookies

browser security issues

- security violations range from inappropriate access to users' files to even more serious problems such as taking over the computing resources of a user's computer
- denial of service (DoS) attack
  - machine under attack is forced to expend all its CPU, disk, and other resources on the attacking program, which effectively prevents the machine from servicing any other user
  - cf: DDoS
- Java applet and ActiveX controls: subject to a DoS attack
- Javascript and VB script: can execute commands without user's knowledge (e.g. sending an email)

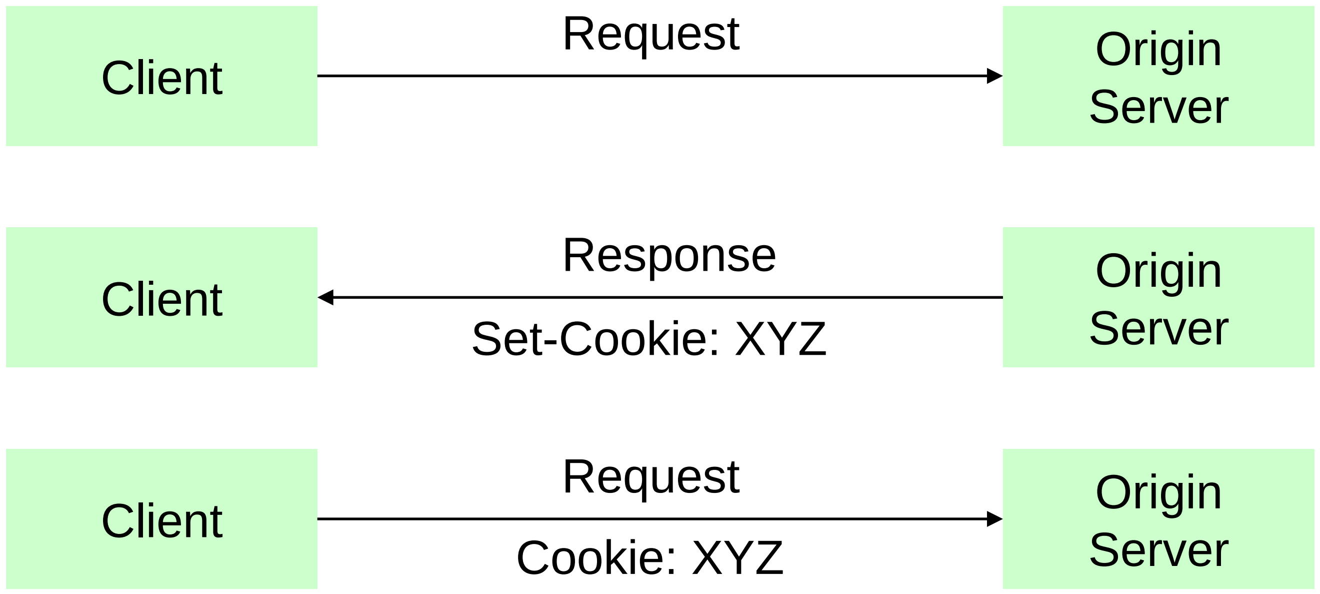
cookies



- HTTP is **stateless**
  - web server does not have to retain any information about past or future requests
- cookies
  - first introduced by Netscape in 1994
  - a small amount of state that is sent by the server to the browser and **stored on the user's machine**
  - can be used to track users and store information about **transactions that span multiple HTTP transfers**
  - consists of a pair of strings which are name/value pair
  - can be up to 4KB long
    - actual state is not fully encoded into cookies
    - cookies are often used as an index into a database maintained by the origin server
  - session cookie: lost as soon as browser exits
- Ref: http://www.cookiecentral.com/

use of cookies

- examples
  - shopping cart
  - redirection: http://ad.doubleclick.net/x22d1/www.foo.com/a.html
  - many web sites require that cookies be used (e.g. nytimes.com)
- a server sends a cookie to a browser with its response, requesting that the browser **include the cookie in subsequent requests to the server**



privacy problems with cookies

- problems
  - cookies are transmitted as **clear text**
  - opt-in model: users explicitly agree to give information about themselves
  - opt-out model: offers ways for users to exclude themselves from sharing information
- user’s control over cookies
  - decide if they accept any cookies at all
  - set a limit on the **size** and **number** of cookies
  - decide whether they accept cookies from all sites or from specific sites only
  - narrow acceptance of cookies only for the duration of a specific session
  - require that cookies must originate from the same server as the current page being viewed

spiders (crawlers, robots, bots)

- a program used to obtain some or all of the resources on a large number of web sites primarily for **generating an inverted index** to be used later in a search application
- behavior of spiders
  - constructs **HTTP requests** to access resources at a web site and **parses the responses**
  - typically **starts with a base list of popular sites**, and follow all of the URLs within the site (e.g. yahoo.com)
    - breadth-first, depth-first traversals
  - once a site has been indexed, it has to **revisit the site** occasionally because the contents of the site may change
- dynamic resources typically are not indexed

control over spiders

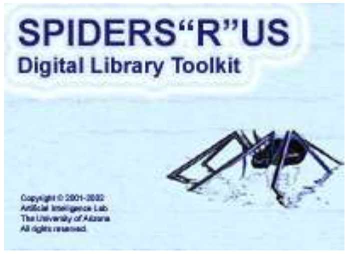
- robot exclusion standard: robots.txt

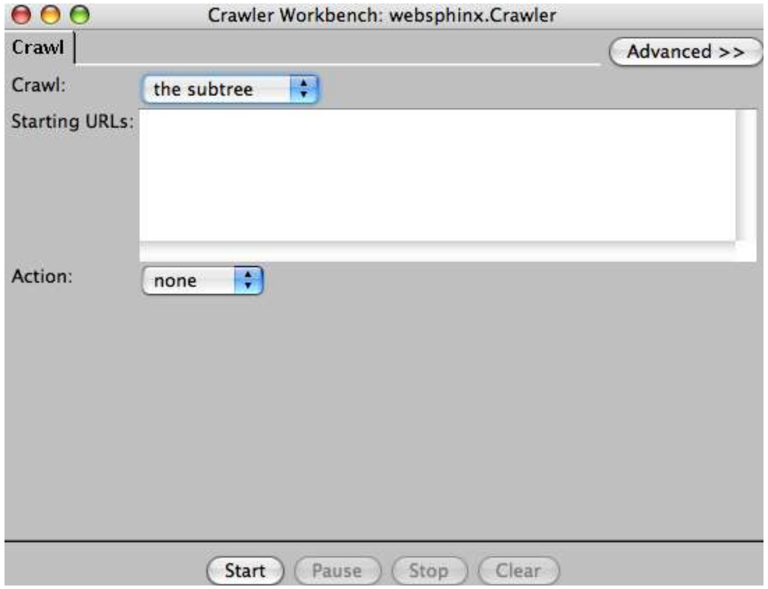
  -
    ```
    User-agent: *
    Disallow: /stats/
    Disallow: /cgi-bin/
    ```
  - A convention for a well-behaved robot to follow
- robots META tag in HTML

  -
    ```
    <META NAME="ROBOTS" CONTENT="NOINDEX,
    NOFOLLOW">
    ```

spider examples

- WebSPHINX (Website-Specific Processors for HTML INformation eXtraction)
  - a java class library and interactive development environment for web crawlers that browse and process web pages automatically.
  - http://www-2.cs.cmu.edu/~rcm/websphinx/index.html
  - java -jar websphinx.jar
- Spiders are Us
  - http://ai.bpa.arizona.edu/go/spider/
- MySpiders
  - http://myspiders.biz.uiowa.edu/

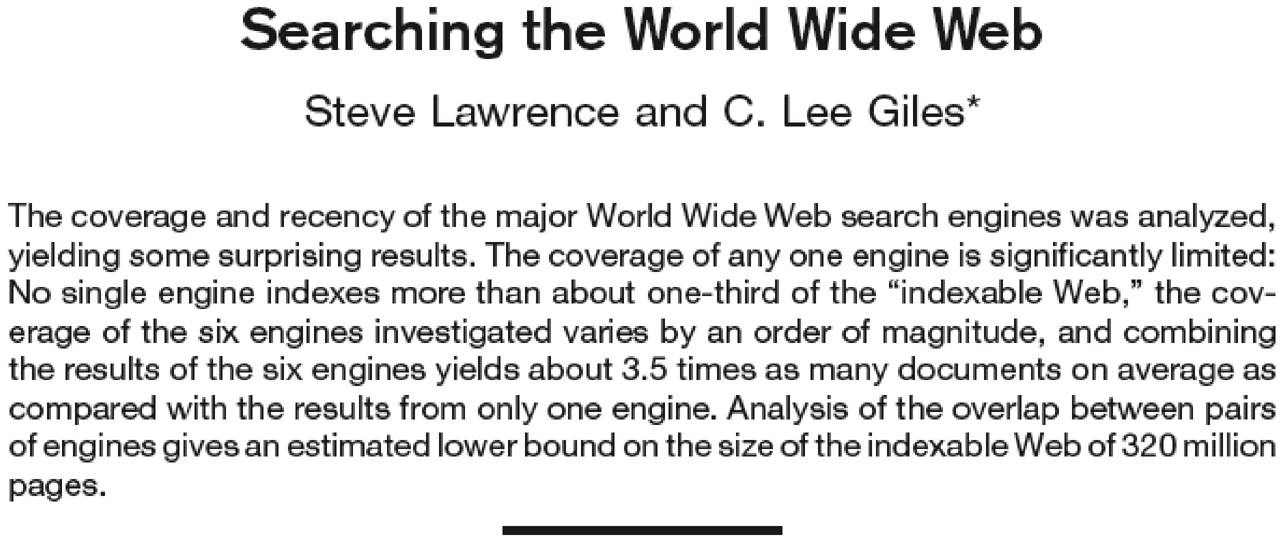




Crawl: the subtree
Starting URLs:
Action: none

use of spiders in search engines

- search engine provides a simple interface to the user to input one or more keywords
  - around 90% of queries consists of a single term
- keywords are looked up in the index, and the pointers to the documents containing them are returned
- some document creators deliberately fill their documents with terms known as "**query traps**": making terms invisible in the rendered HTML document by setting their font size to be zero or the foreground and background colors of the terms to be black



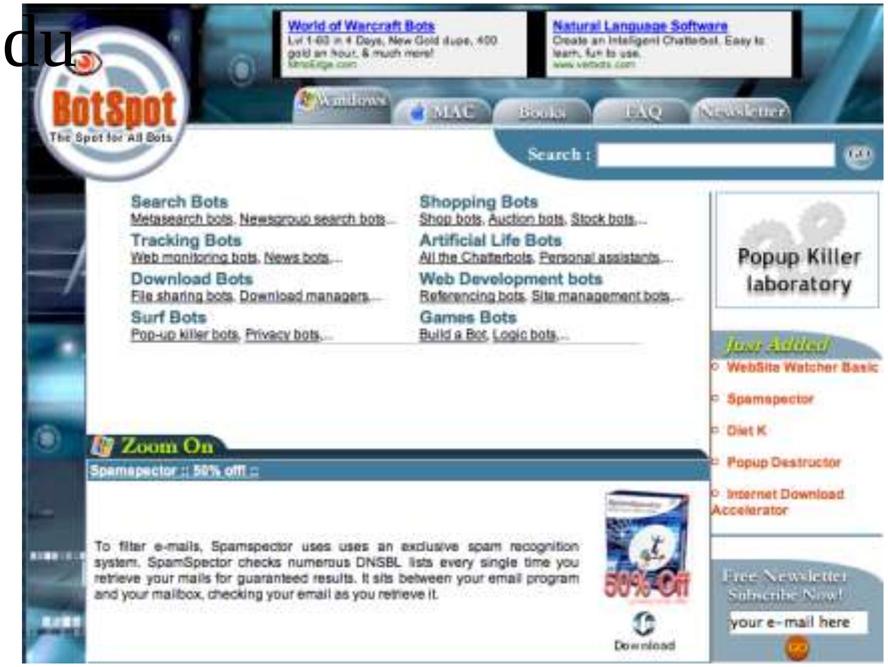
Searching the World Wide Web

Steve Lawrence and C. Lee Giles*

The coverage and recency of the major World Wide Web search engines was analyzed, yielding some surprising results. The coverage of any one engine is significantly limited: No single engine indexes more than about one-third of the "indexable Web," the coverage of the six engines investigated varies by an order of magnitude, and combining the results of the six engines yields about 3.5 times as many documents on average as compared with the results from only one engine. Analysis of the overlap between pairs of engines gives an estimated lower bound on the size of the indexable Web of 320 million pages.

spider applications

- general shopbots
  - mysimon.com, pricegrabber.com, dealtime.com
- specialized shopbots
  - allbookstores.com (books), autobytel.com (cars), zdnet.com/computershopper (computers and SW), offce.com
- bot directory: agents.umbc.edu
- bots on sale: botspot.com



World of Warcraft Bots
Lvl 1-60 in 4 Days, New Gold dupe, 400 gold an hour, & much more!
MmoEdge.com

Natural Language Software
Create an Intelligent Chatterbot. Easy to learn, fun to use.
www.verbots.com

Search Bots
Metasearch bots, Newsgroup search bots...

Shopping Bots
Shop bots, Auction bots, Stock bots...

Tracking Bots
Web monitoring bots, News bots...

Artificial Life Bots
All the Chatterbots, Personal assistants...

Download Bots
File sharing bots, Download managers...

Web Development bots
Referencing bots, Site management bots...

Surf Bots
Pop-up killer bots, Privacy bots...

Games Bots
Build a Bot, Logic bots...

- WebSite Watcher Basic
- Spamspector
- Diet K
- Popup Destructor
- Internet Download Accelerator

Spamspector :: 50% off! ::

To filter e-mails, Spamspector uses uses an exclusive spam recognition system. SpamSpector checks numerous DNSBL lists every single time you retrieve your mails for guaranteed results. It sits between your email program and your mailbox, checking your email as you retrieve it.

intelligent agents and special-purpose browsers

- intelligent agents
  - created for specific client tasks
  - meta-search engine: searches on multiple search engines
  - auction agent: performs bidding on behalf of a user
- special-purpose browsers
  - co-browser: helps a user by proactively requesting relevant resources in advance (e.g. Letizia)
  - collaborative browser: lets a group of users browse together
  - offline browser: enables a user to browse offline a collection of pages that were downloaded earlier

web proxies

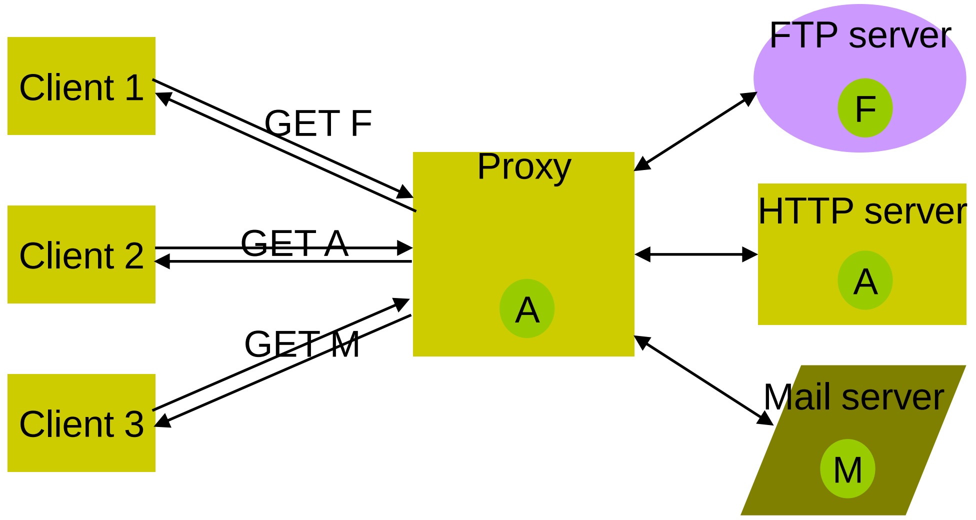
- proxy
  - an intermediary program which acts as **both a server and a client** for the purpose of forwarding requests
  - proxies are often used to act as a portal through a network firewall
  - proxy server accepts requests from other clients and services them either internally or by passing them (with possible translation) on to other servers
- benefits of proxy (caching proxy)
  - latency perceived by a client is lowered if the proxy is closer to the client
  - load on the work is reduced
  - load on the server is lowered

proxy applications

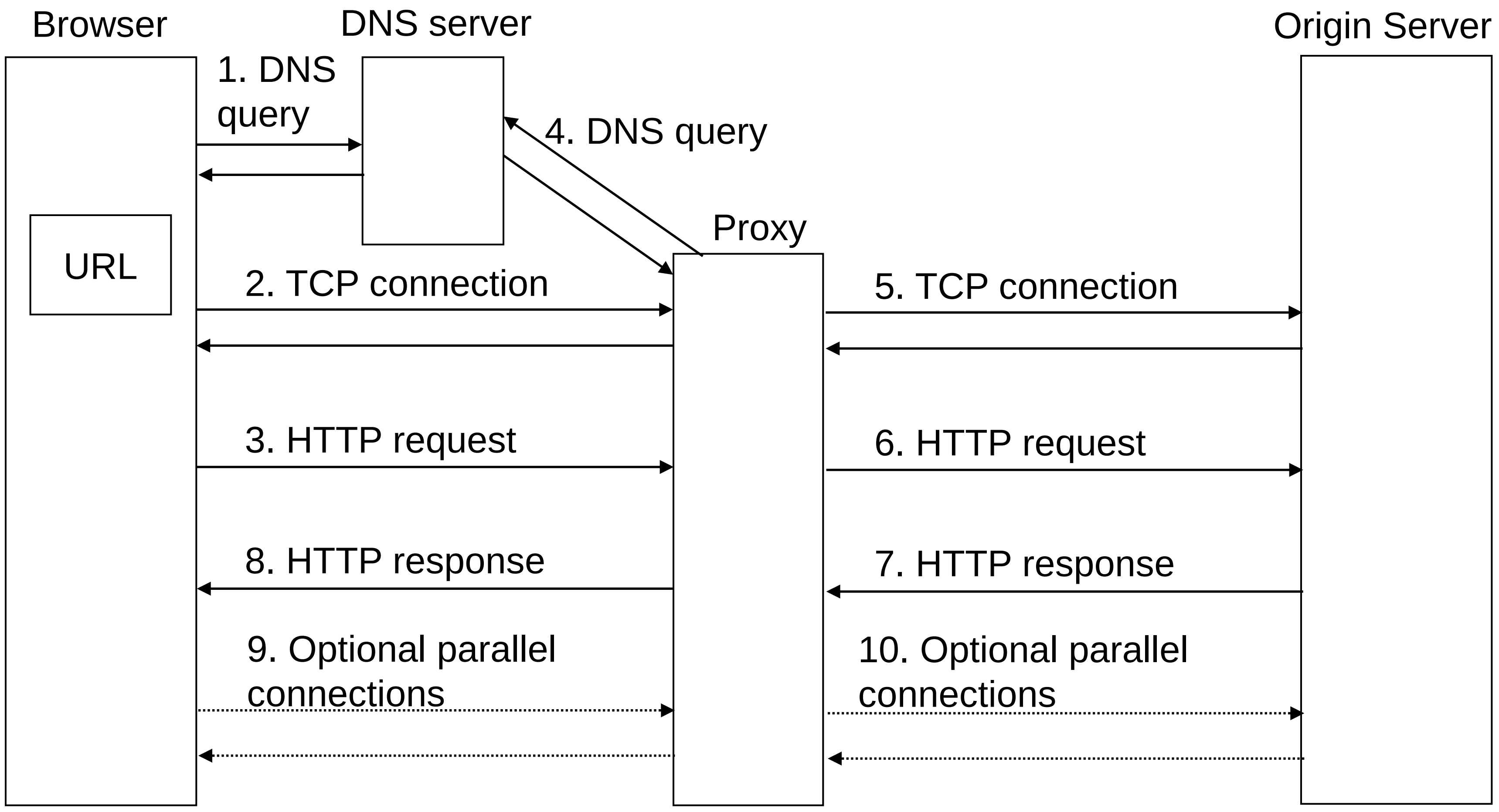
- sharing access to the web
  - many clients request the same resource → a single connection between the proxy and the server
  - many clients request the different resources → serialization
  - some ISPs mandate users to go through proxies to access the web (e.g. AOL)
- caching responses
  - caching: storage of a response obtained earlier for later use, when clients request the same resource
  - not effective for spiders
- anonymizing clients
- transforming requests and responses
  - **compression** for speeding up a transfer to a user behind a low-bandwidth connection
  - HTTP version compatibility

proxy applications (cont.)

- gateway to non-HTTP systems
  - plays a role of an intermediary to other systems that do not communicate using HTTP
- filtering requests and responses
  - filtering certain requests based on destination sites and filtering responses based on characteristics such as response size
  - e.g. http://muffin.doit.org/
  - e.g. SurfWatch, Netnanny



steps in request-response with a proxy



other kinds of proxies

- reverse proxies
  - proxies positioned **closer to the origin server**
  - prevent the origin server from being vulnerable to direct attacks from the outside world
  - help in balancing the load
- interception proxies
  - proxy that either directly examines network traffic and intercepts web traffic or receives redirected traffic flow from network elements performing traffic interception

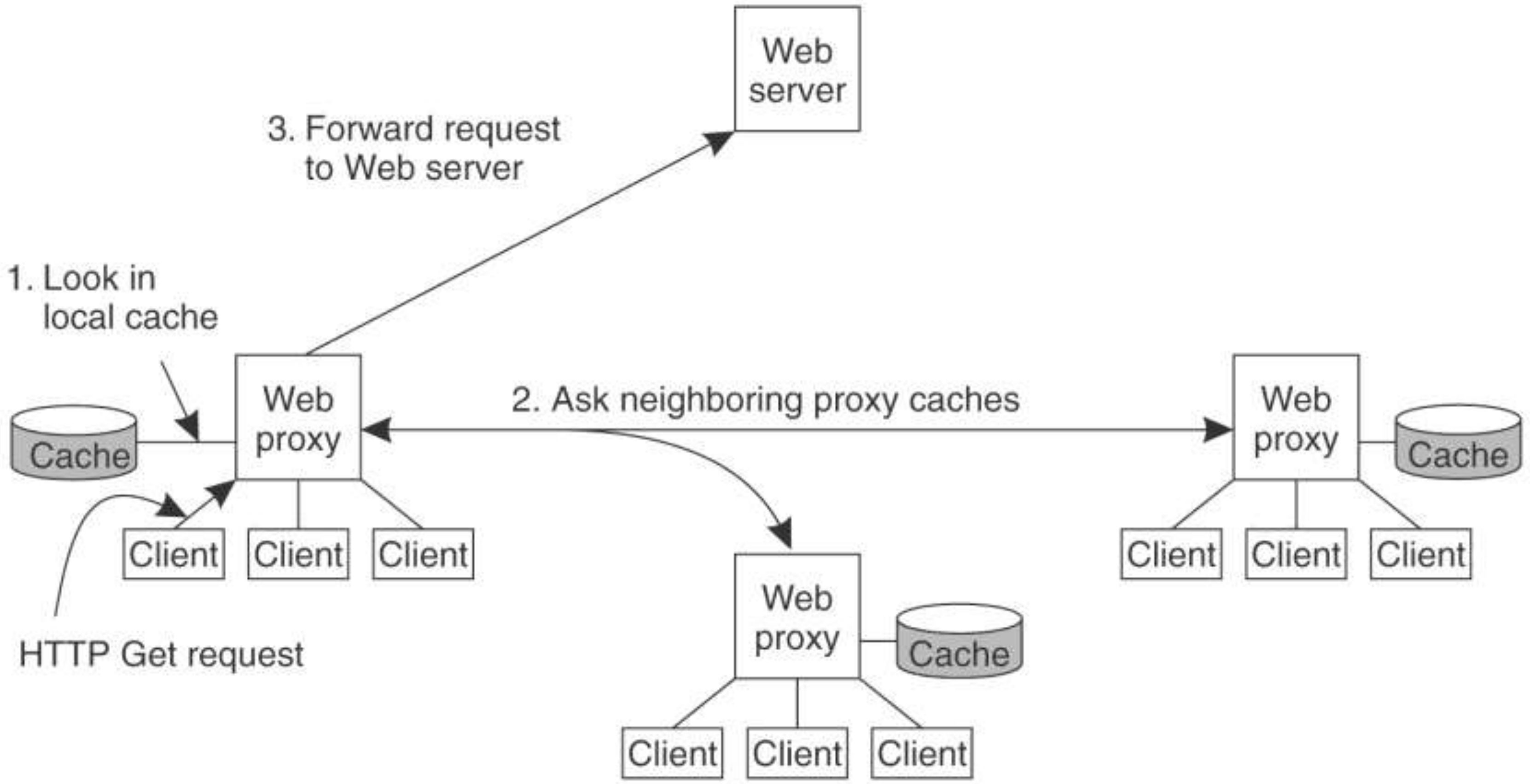
cache-consistency protocols

- pull-based protocol
  - proxies first send a conditional HTTP GET request to the server with an additional **If-Modified-Since** request header
  - proxy contacts a server **for each request**
- squid web proxy
  - $T_{expire} = \alpha * (T_{cached} - T_{last\_modified}) + T_{cached}$
  - after the expiration time, proxy requests the server to send a fresh copy, unless it had not been modified
  - $\alpha = 0 \rightarrow$ pull-based control

cache-consistency protocols (cont.)

- push-based protocol
  - server notifies proxies that a document has been modified by sending an invalidation
  - server may need to keep track of a large number of proxies, leading to a **scalability problem**
- cooperative caching
  - whenever a cache miss occurs at a web proxy, proxy first checks a number of **neighboring proxies** to see if one of them contains the requested document
  - effective for relatively small groups of clients

Cooperative caching



web server

- program that generates and transmits responses to client requests for web resources, ranging from static files to scripts that generate customized responses
- web site: a collection of web pages associated with a particular hostname
- steps in handling a client request
  - read and parse the HTTP request message
  - translate the URL to a file name
  - determine whether the request is authenticated and authorized (e.g., by use of access control list)
  - generate and transmit the response

sharing information across requests

- sharing HTTP responses
  - **server-side cache**: stores frequently requested resources or the results of common queries in main memory
- sharing metadata
  - server stores the information generated in the process of handling an HTTP request instead of re-generating every time
  - e.g., translation of URL to file name, control information about resource, HTTP response headers, current date/time, client’s host name (from IP address)

server architecture

- event-driven architecture
  - has a **single process** that alternates between servicing different requests (i.e., one at a time)
  - typically performs **nonblocking system calls** that allow the calling process to continue executing while awaiting a response from OS
  - can **serialize operations** that modify the same data
  - works well when each event introduces small, bounded delay
- process-driven architecture
  - allocates each request to a **separate process**
  - each process works as an event-driven server that performs all of the steps involved in handling a single request
  - overhead of process forking, terminating, and context switching can be significant
  - coordination of resource sharing can be complex
  - preforking: a set of processes is created when the server starts

server architecture: enhancement

- multithreaded web server
  - assigns a thread to each request
  - Each thread is a sequential flow of execution in the shared address space of a containing process
  - The overhead of switching between threads is lower than switching between processes because threads in the same process share a common address space
- hybrid architecture
  - Server has a main event-driven process that handles the initial stages of every request
  - If a request requires significant computation, then the main process instructs a separate process to perform the computation

Java-based multi-threaded web server

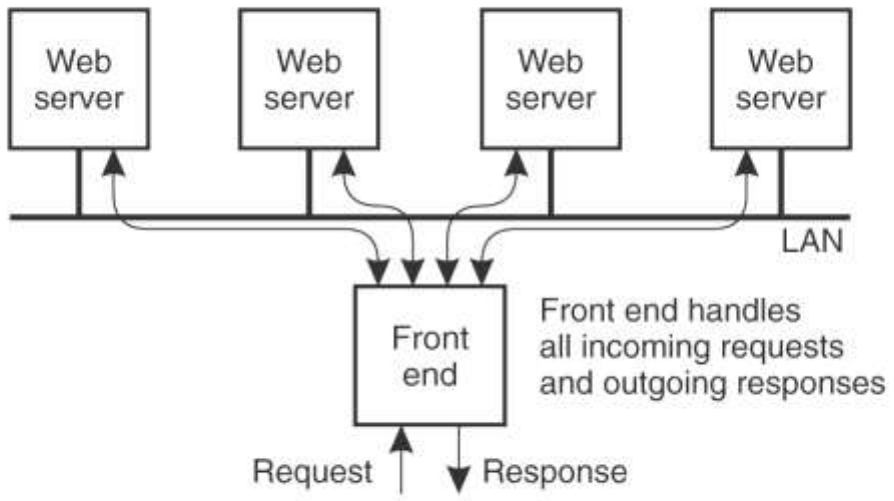
- http://developer.java.sun.com/developer/technicalArticles/Networking/Webserver/
- web server
  - WServer.java at local computer
  - e.g., /Users/drburix02/Dropbox/IA2010Spring/Demo/WebServerInJava
- web client
  - any browser with http://localhost:18081/

server hosting

- multiple web sites on a single machine
  - virtual servers: used by a web hosting company
  - assigning multiple IP addresses to a single machine in HTTP
  - use of host header field in HTTP
- multiple machines for a single web site
  - translation of a server name into a particular IP address is performed **by DNS**, or
  - single IP address that corresponds to a surrogate that resides in front of the web hosting complex can be used
  - introduces the **replication management** problem

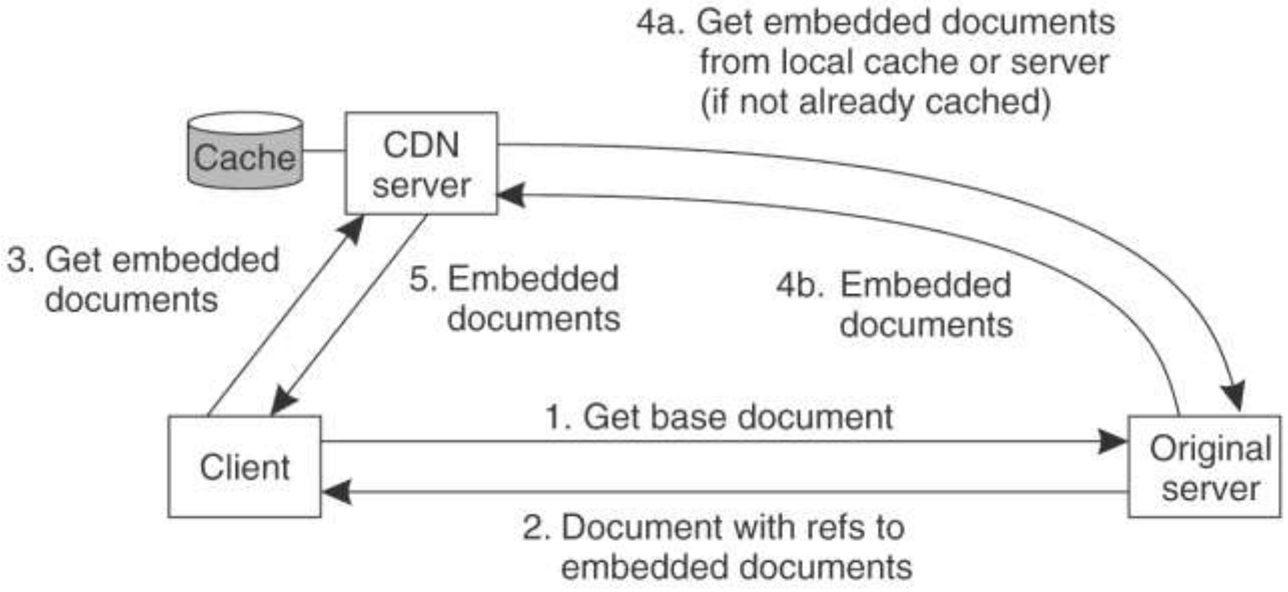
server clusters

- front end as a transport-layer switch: L4
  - simply passes the data sent along the TCP connection to one of the servers
- front end as a application-layer switch: L7
  - content-aware request distribution
  - it is possible to actually distribute the collection of documents among the servers instead of having to replicate each document for each server



CDN (Content Delivery Network)

- collection of servers is distributed across the Internet, offering the facilities for hosting web documents by **replicating documents** across the servers, taking client access characteristics into account (e.g. geographical location)
- based on the idea that each web document consists of a main HTML page in which several other documents such as images, video, and audio have been embedded
- akamai.com (based on MIT Students' contest)
- popular for game downloads

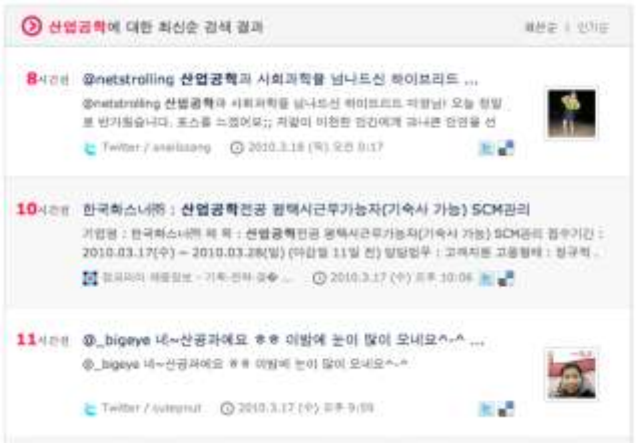

4a. Get embedded documents from local cache or server (if not already cached)
Cache
CDN server
3. Get embedded documents
5. Embedded documents
4b. Embedded documents
1. Get base document
Client
Original server
2. Document with refs to embedded documents

contents

- introduction
- processes in WWW
  - Web clients
  - Web proxies
  - Web servers
- **technology for dynamic Web documents**
- technology for active Web documents
- apache Web server

3 basic types of web documents

- **static** document
  - contents do not change
  - simple, reliable, but inflexible
- **dynamic** document
  - **created by a web server** whenever a client requests the document
  - when a request arrives, the web server runs an application program that creates the dynamic document (i.e., HTML)
  - report current information, but not suitable for changing information (e.g. stock prices)
- **active** document
  - consists of a **computer program** that understands how to compute and display values
  - when a browser requests an active document, the server **returns a copy of the program** that the browser must run locally
  - can update information continuously and autonomously, but may have security problems


산업공학에 대한 최신순 검색 결과 최신순 | 인기순

8시간 전 @netstrolling 산업공학과 사회과학을 넘나드신 하이브리드 ...
@netstrolling 산업공학과 사회과학을 넘나드신 하이브리드 [illegible] 오늘 정말 보 반가웠습니다. 포스를 느꼈어요;; 치앙이 미천한 인간에게 [illegible] 안경을 선
Twitter / [illegible] 2010.3.18 (목) 오전 0:17

10시간 전 한국하스너(주) : 산업공학전공 평택시근무가능자(기숙사 가능) SCM관리
기업명 : 한국하스너 제 목 : 산업공학전공 평택시근무가능자(기숙사 가능) SCM관리 접수기간 : 2010.03.17(수) ~ 2010.03.28(일) (마감일 11일 전) 담당업무 : 고객지원 고용형태 : 정규직 ..
[illegible] 2010.3.17 (수) 오후 10:06

11시간 전 @_bigeye 네~산공과에요 ㅎㅎ 이밤에 눈이 많이 오네요^-^ ...
@_bigeye 네~산공과에요 ㅎㅎ 이밤에 눈이 많이 오네요^-^
Twitter / [illegible] 2010.3.17 (수) 오후 9:59

핫이슈 ›

1	서희원	–
2	구준엽 서희원	–
3	메시	–
4	한양대학교	↑
5	캐나다 유학생	↑
6	박진영	↑
7	샤이니민호	↑
8	간동희	↑
9	이승훈	↑
10	구준엽	↑

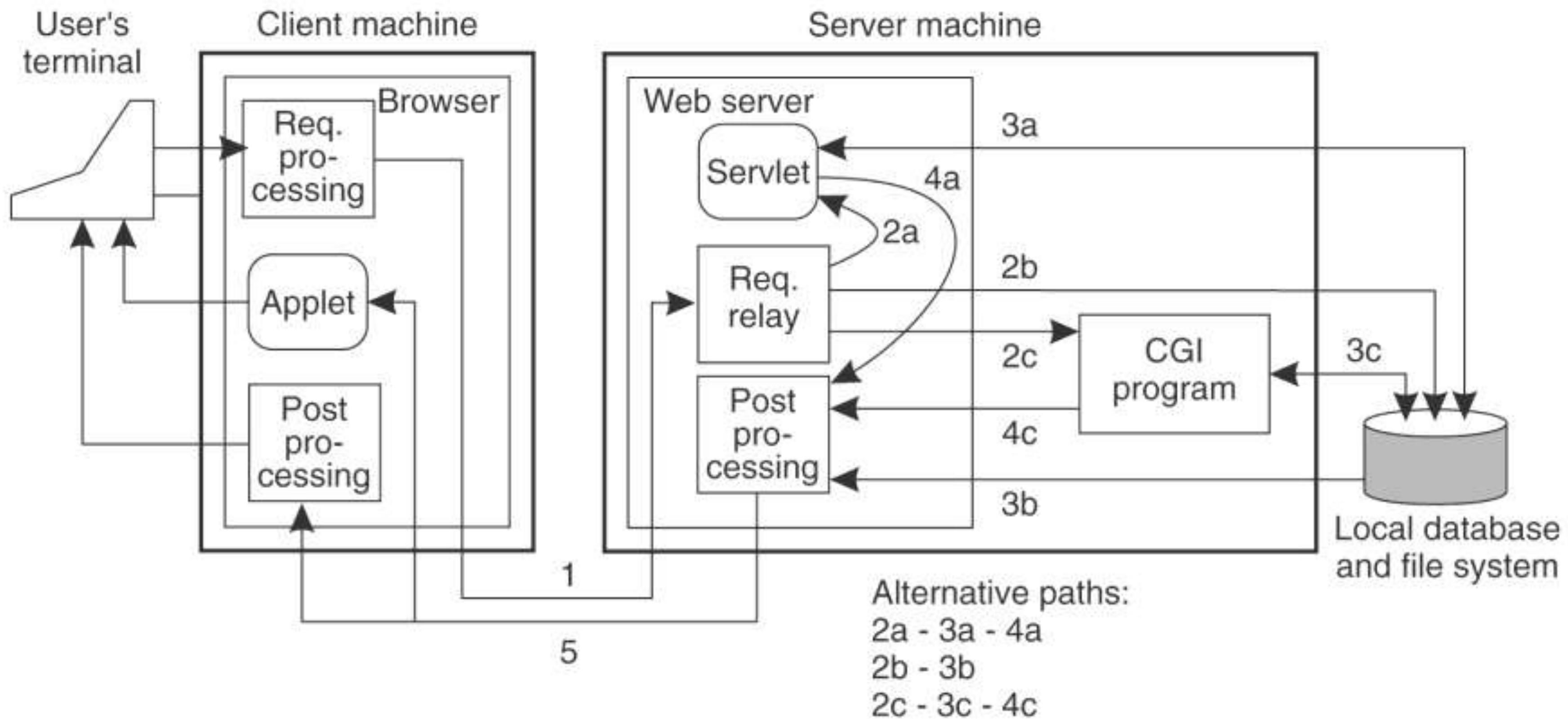
Implementation of dynamic documents

- some additions are required
  - server program must be extended so that it is capable of executing a separate **application program that creates a document each time a request arrives**
  - separate application program must be written for each dynamic document
  - server must be configured so it knows which URLs correspond to dynamic documents and which correspond to static documents
  - for each dynamic document, the configuration must specify the application program that generates the document
- executing the program as a script at the server ensures that the software and raw data remain private

Server-side scripting technologies

- separate program that generates the requested resource
  - as a separate process invoked by the server (e.g. CGI)
  - as a separate software module that executes as part of the server (e.g., Servlets, Apache’s mod_perl module, MS’s Internet Server Application Programming Interface (ISAPI))
- the stored form of the page, known as a **template** or skeleton, that contains a mixture of conventional HTML and scripting information
  - interpreter replaces the scripting information with the results of interpreting the script
  - SSIs: commands embedded in XHTML documents to allow the creation of simple dynamic content
  - JSP, ASP, and PHP

Architecture of server-side program



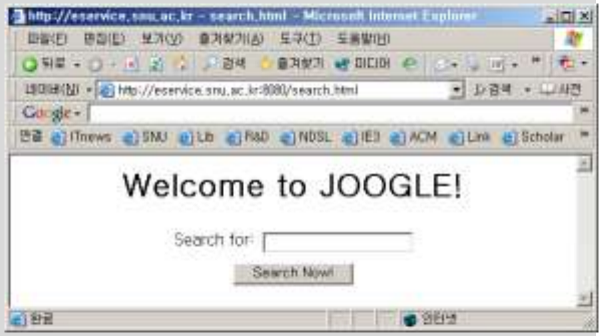
Major server-side scripting technologies

- Servlet
  - a precompiled program that is executed at the server
  - is executed when a server receives an HTTP request addressed to a servlet
- JSP (Java Server Pages)
  - a dynamic page technology that is intended to be platform-independent
  - script is written in Java
  - Apache Tomcat
- ASP.NET (Active Server Pages)
  - a dynamic page technology from Microsoft
  - script is written in Visual Basic
  - interpreter is integrated with IIS
- PHP (PHP: Hypertext Preprocessor)
  - www.php.net
  - faster than JSP or ASP

an example JSP code

```
…
<BODY>
<% if (error != null) { %>
<TABLE width="100%" border="0">
<TR><TD><B>Exception encountered:</B></TD><TD><FONT color="#ff0000">
  <%= error %></FONT></TD></TR>
<TR><TD><B>Exception details:</B></TD><TD><FONT color="#ff0000">
  <%= errorDetails %></FONT></TD></TR>
</TABLE>
<% } else { %>
<P><B>Google Search Engine</B> reported it took
   <B><%= searchTime %></B> milliseconds
   to complete searching for your query <B>[<FONT color="#0000ff"><U>
  <%= request.getAttribute("searchStr") %></U></FONT>]</B>.</P>
<P>Now returning the best results of about
   <B><%= numTotalResults %></B> items found.</P>
<HR>
<% for (int i=0; i<elements.length; i++) { %>
<TABLE width="100%" border="0" cellspacing="3" cellpadding="3">

<TR>
<TD width="15%" bgcolor="#000099"><FONT color="#ffffff"
   size="2" face="Courier New"><B>Title:</B></FONT></TD>
…
</BODY>
</HTML>
```


http://eservice.snu.ac.kr - search.html - Microsoft Internet Explorer
http://eservice.snu.ac.kr:8080/search.html
Welcome to JOOGLE!
Search for:
Search Now!

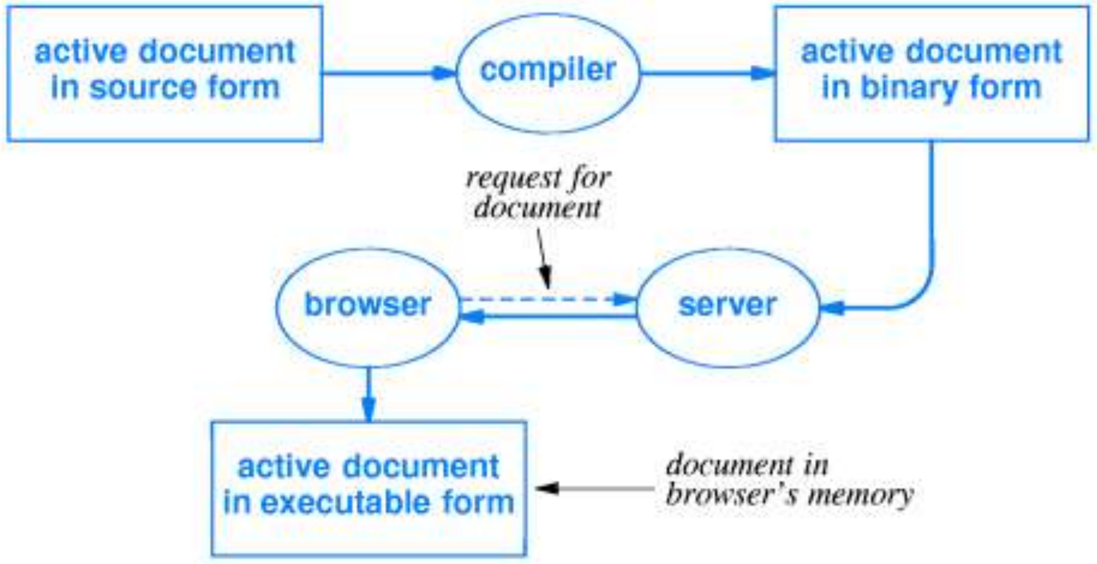
contents

- introduction
- processes in WWW
  - Web clients
  - Web proxies
  - Web servers
- technology for dynamic Web documents
- **technology for active Web documents**
- apache Web server

server push and active documents

- a dynamic document can’t update information on the screen as information changes
- server push
  - requires a server to periodically generate and send new copies of a document
  - causes excessive server overhead and introduces delay
- active documents
  - moves the computation to the browser
  - when a browser requests an active document, the **server returns a program** that the browser runs locally
  - the browser that runs active documents should contain support software for the program execution

Active document representations


active document in source form
compiler
active document in binary form
request for document
browser
server
active document in executable form
document in browser's memory

- some active document technologies arrange for a browser to accept and interpret a source document

Java applet

- small stand-alone application that can be sent to the client and executed in the browser’s address space
- browser must support a Java interpreter and run-time environment (i.e., Java plug-in)
  - Java interpreter must be able to communicate with the browser’s HTTP client and HTML interpreter
- invoking an applet
  - URL: http://www.my.com/dir/MyApp.class
  - Embedding in HTML using an applet tag: <applet codebase=www.my.com/dir code=“MyApp.class”>

Applet example

```
import java.applet.*;
import java.net.*;
import java.awt.*;

public class Buttons extends Applet {

  public void init() {
    setBackground(Color.white);
    add(new Button("Harvard"));
    add(new Button("SNU"));
  }

  public boolean action(Event e, Object arg) {
    if (((Button) e.target).getLabel() == "Harvard") {
      try {
                getAppletContext().showDocument(
                  new URL("http://www.harvard.edu"));
      }
      catch (Exception ex) {
                // note: code to handle the exception goes here //
                                    }
    }
    else if (((Button) e.target).getLabel() == "SNU") {
      try {
                getAppletContext().showDocument(
                  new URL("http://www.snu.ac.kr"));
      }
      catch (Exception ex) {
                // note: code to handle the exception goes here //
                                    }
    }
    return true;
  }
}
```

Running the applet

- Buttons.html
  -
```
<applet code="Buttons.class" width="300"
height="300">
</applet>
```

JavaScript

- provides a **scripting language**, and arranges for a browser to read and interpret a script in source form
- used for active pages that do not contain large or complex code
- can be integrated with HTML
  - an HTML page can contain a JavaScript function that provides simple interaction with a user
- advantages: simplicity and ease of use
- disadvantages: speed and scalability (due to the source-based code shipping)

JavaScript example

```
<html>
<head>
  <title>First JavaScript</title>
</head>
<body>
<script language="JavaScript"
type="text/javascript">
  document.write("Hello, World");
</script>
</body>
</html>
```

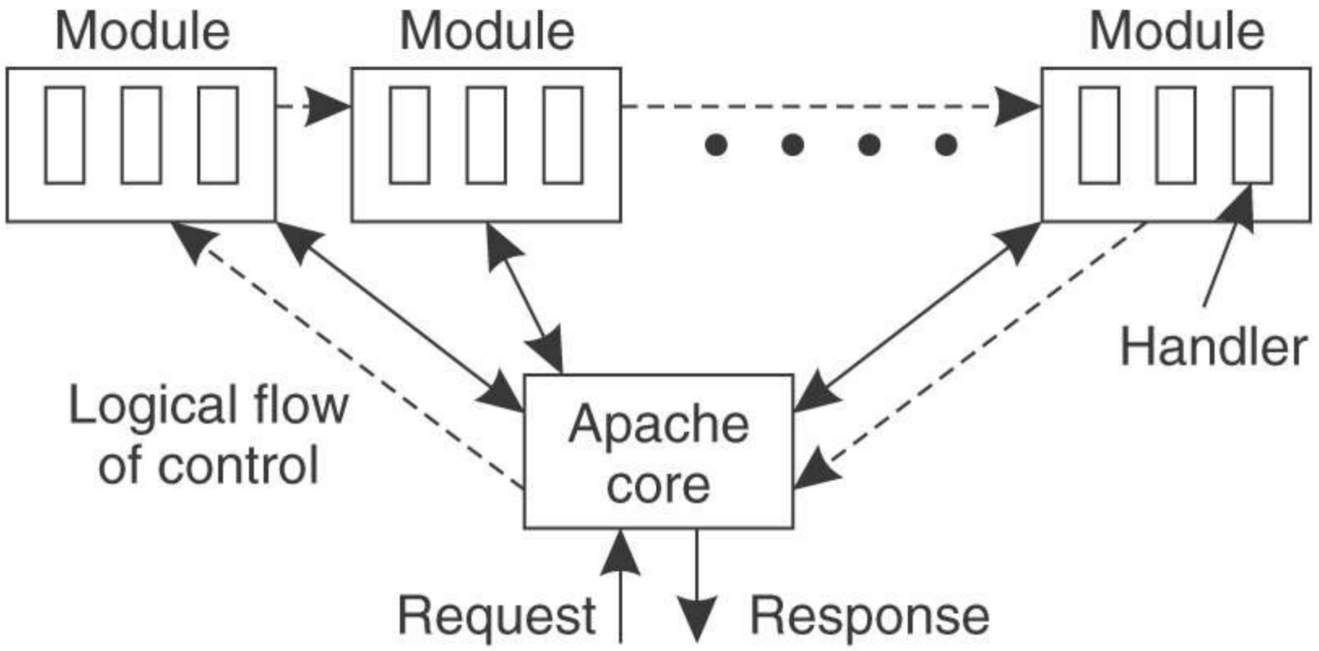
contents

- introduction
- processes in WWW
  - Web clients
  - Web proxies
  - Web servers
- technology for dynamic Web documents
- technology for active Web documents
- **apache Web server**

Apache web server

- the first version 0.6.2 released in 1995
  - based on the NCSA software
- “Apache” refers to the fact that the software consisted of existing code along with some “patches”
- the server
  - consists of a number of modules that are controlled by a single core module
  - the core module determines the flow of control
  - for each request, the core module allocates a request record
  - each module is required to provide one or more handlers that can be invoked by the core module

Apache Web server architecture



Resource management in Apache

- Apache follows a process-driven model, with a parent process that assigns a process to each new connection
  - the parent preforks several child processes when the server starts
  - imposes a limit on the min and max # of idle processes
  - when the # of idle processes is low, creating new processes prepares the server for future requests
- Apache 2.0 has a **multithreaded** implementation
  - consists of software modules performing various tasks, such as listening for new connections, parsing requests, generating responses, and transmitting response messages
  - uses a pool abstraction which is a data structure that tracks a group of OS resources that are created and destroyed together