
CPE 490: Information Systems Engineering I: Computer Networking

Chap. 7 - The Application Layer

Professor Du
Department of Electrical and Computer Engineering
Stevens Institute of Technology
Email: xdu16@stevens.edu

The Application Layer

- Domain Name System (DNS)
- Email Systems
- Web

DNS – The Domain Name System

➤ DNS

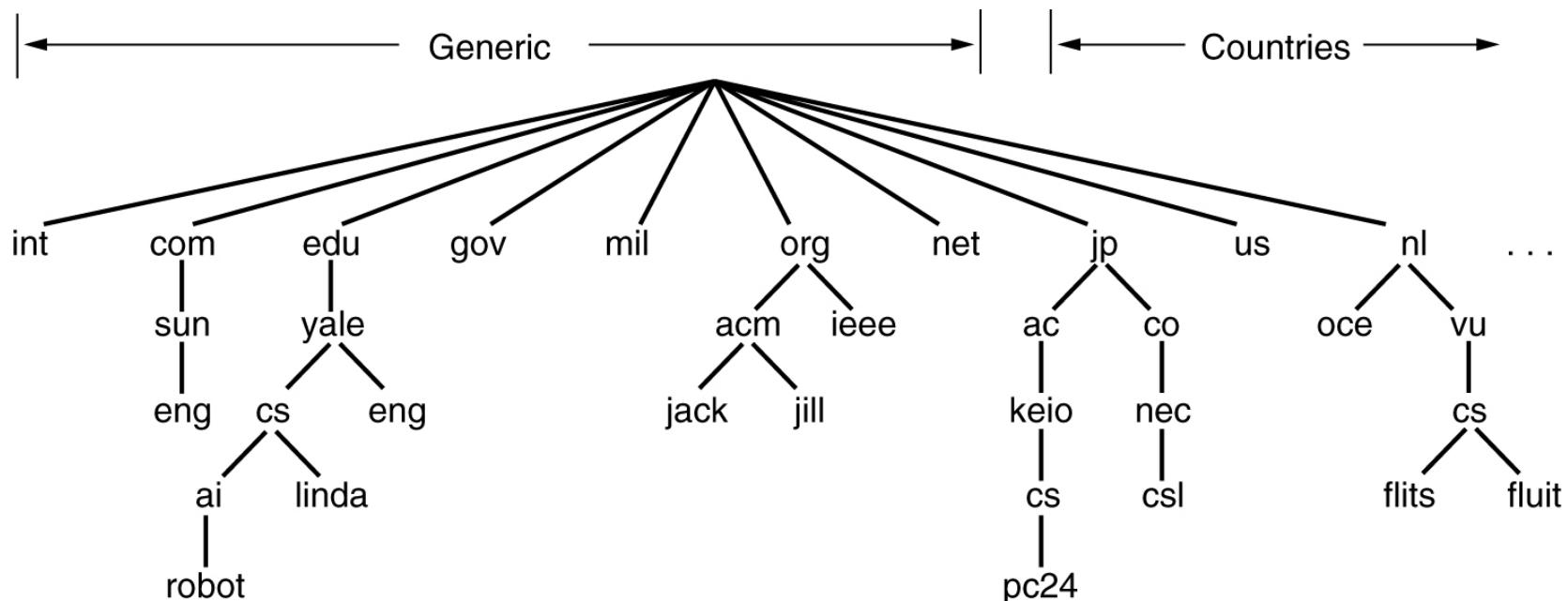
- ✓ To map a computer name onto an IP address.
- ✓ An application passes the computer name to a library procedure – resolver.
- ✓ The resolver calls a local DNS server who returns the IP address.
- ✓ E.g., *gethostbyname*

➤ The DNS Name Space

- ✓ The Internet is divided into over 200 top-level domains.
- ✓ Each domain covers many hosts, and is partitioned into several sub-domains, which can be further divided.
- ✓ All these domains can be represented by a tree.

The DNS Name Space

- The original generic domains are com (commercial), edu (educational institutions), gov (the U.S. Federal Gov.), int (certain international org.), mil (the U.S. military), net (network providers) and org (nonprofit org.)
- In Nov. 2000, four new generic top-level domains were added: biz (businesses), info (information), name (people's names), and pro (professional, such as doctors)



A portion of the Internet domain name space.

Resource Records

- Every domain (host or top-level domain) can have a set of resource records. E.g., the IP address of a single host.
- When a resolver gives a domain name to DNS, the return is the resource records associated with the name. DNS – to map domain names onto resource records.
- A resource record includes: Domain_name, Time_to_live, Class, Type, Value.
 - ✓ Class – IN for Internet info. Value – can be a number, domain name or an ASCII string, and the semantics depend on the record type.
 - ✓ SOA – Info. about the name server's zone.

- ✓ MX – the name of the mail server.
- ✓ CNAME records allow aliases to be created. E.g., cs.mit.edu → lcs.mit.edu
- ✓ HINFO gives CPU and OS types in a domain.

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept e-mail
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

Resource Records (2)

```
; Authoritative data for cs.vu.nl
cs.vu.nl.      86400  IN  SOA   star boss (952771,7200,7200,2419200,86400)
cs.vu.nl.      86400  IN  TXT   "Divisie Wiskunde en Informatica."
cs.vu.nl.      86400  IN  TXT   "Vrije Universiteit Amsterdam."
cs.vu.nl.      86400  IN  MX    1 zephyr.cs.vu.nl.
cs.vu.nl.      86400  IN  MX    2 top.cs.vu.nl.

flits.cs.vu.nl. 86400  IN  HINFO Sun Unix
flits.cs.vu.nl. 86400  IN  A    130.37.16.112
flits.cs.vu.nl. 86400  IN  A    192.31.231.165
flits.cs.vu.nl. 86400  IN  MX   1 flits.cs.vu.nl.
flits.cs.vu.nl. 86400  IN  MX   2 zephyr.cs.vu.nl.
flits.cs.vu.nl. 86400  IN  MX   3 top.cs.vu.nl.
www.cs.vu.nl.   86400  IN  CNAME star.cs.vu.nl
ftp.cs.vu.nl.   86400  IN  CNAME zephyr.cs.vu.nl

rowboat          IN  A    130.37.56.201
                  IN  MX   1 rowboat
                  IN  MX   2 zephyr
                  IN  HINFO Sun Unix

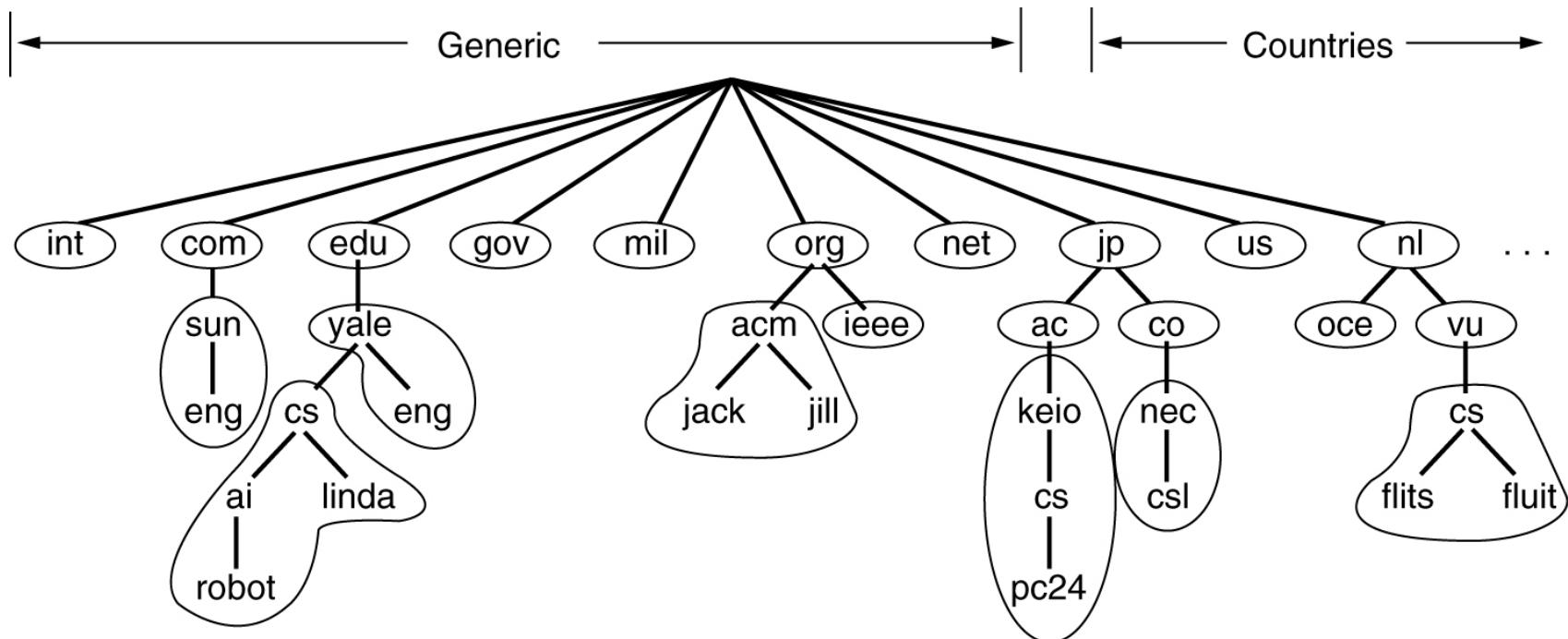
little-sister    IN  A    130.37.62.23
                  IN  HINFO Mac MacOS

laserjet         IN  A    192.31.231.216
                  IN  HINFO "HP Laserjet IISi" Proprietary
```

A portion of a possible DNS database for *cs.vu.nl*.

Name Servers

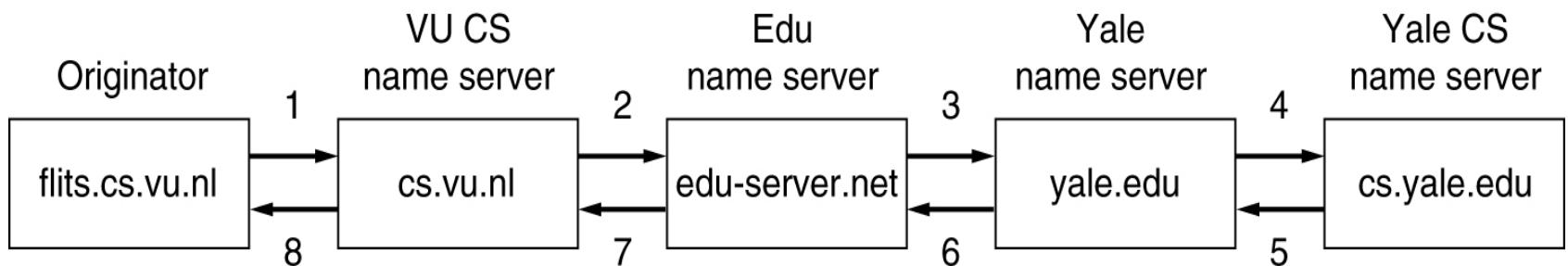
- The DNS name space is divided into several non-overlapping zones.
- Each zone has one primary name server, and one or more secondary name servers.



Part of the DNS name space showing the division into zones.

Name Servers (2)

- An example: a resolver on flits.cs.vu.nl wants to know the IP address of the host linda.cs.yale.edu



How a resolver looks up a remote name in eight steps.

Electronic Mail

- Architecture and Services
- The User Agent
- Message Formats
- Message Transfer
- Final Delivery

Electronic Mail (2)

Smiley	Meaning	Smiley	Meaning	Smiley	Meaning
:-)	I'm happy	= :-)	Abe Lincoln	:+)	Big nose
:-('	I'm sad/angry	=):-)	Uncle Sam	:-))	Double chin
:-	I'm apathetic	*<:-)	Santa Claus	:-{})	Mustache
;:-)	I'm winking	<:-('	Dunce	#:-)	Matted hair
:-)(O)	I'm yelling	(-:	Australian	8-)	Wears glasses
:-(*)	I'm vomiting	:-)X	Man with bowtie	C:-)	Large brain

Some smileys. They will not be on the final exam :-).

Architecture and Services

- Email systems consist of two subsystems:
 - ✓ User agents – allows people to read and send email.
 - ✓ User agents are local programs that provide a command-based, menu-based, or graphical method for interacting with the email sys.
 - ✓ Message transfer agents move the messages from the source to the destination
 - ✓ Message transfer agents are system daemons - processes that run in the background.
- Basic functions
 - ✓ Composition – the process of creating messages and replying.
 - ✓ Transfer – moving messages from the originator to the recipient.
 - Establishing a connection, sending the message, and releasing the conn.
 - ✓ Reporting – telling the originator what happened to the message.
 - Delivered? Rejected? Lost?
 - ✓ Displaying incoming message.
 - ✓ Disposition – What the recipient does with the message after receiving it.
 - Throwing it away before/after reading it, saving it, etc.

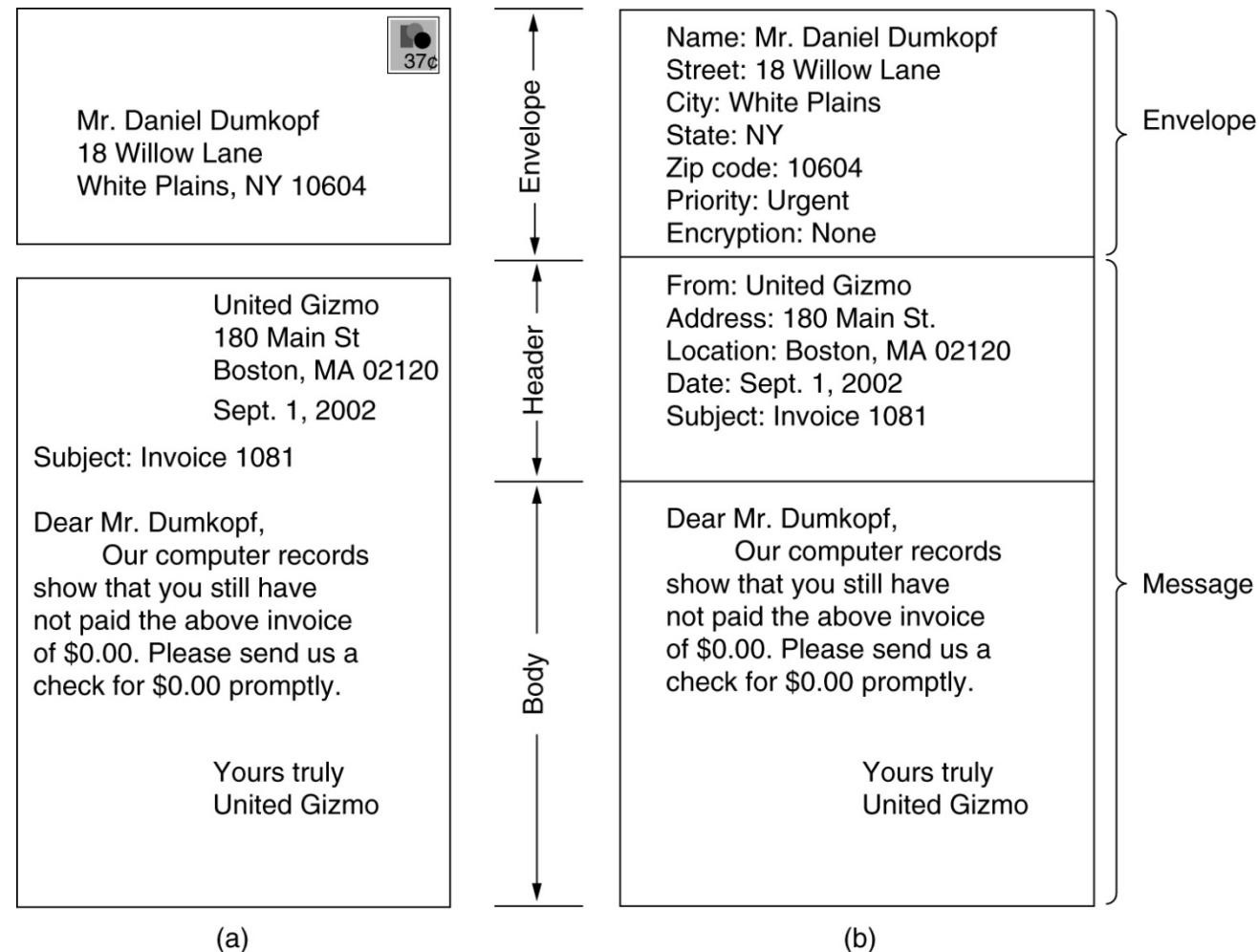
The User Agent

An email consists of:

- An envelop – contains all the info.

Needed for transporting the message, such as the destination address, priority, and security level, which can be used by the message transport agents for routing.

- The message includes two part:
 - header – control info. for the user agents
 - Body –message text.



Envelopes and messages. (a) Paper mail. (b) Electronic mail.

Reading E-mail

- ✓ When a user agent is started up, it looks at the user's mailbox for incoming email before displaying anything on the screen.
- ✓ Then it may announce the number of messages in the mailbox.
- ✓ An example – A summary of the user's email.
- ✓ Each line contains seven fields extracted from the envelop or header of the message.
 - K – kept email; A – answered; F – forwarded.

#	Flags	Bytes	Sender	Subject
1	K	1030	asw	Changes to MINIX
2	KA	6348	trudy	Not all Trudys are nasty
3	K F	4519	Amy N. Wong	Request for information
4		1236	bal	Bioinformatics
5		104110	kaashoek	Material on peer-to-peer
6		1223	Frank	Re: Will you review a grant proposal
7		3110	guido	Our paper has been accepted
8		1204	dmr	Re: My student's visit

An example display of the contents of a mailbox.

Message Formats – RFC 822

- ✓ Email message formats were defined in RFC (Request For Comment) 822.
- ✓ A line containing Received is added by each message transfer agent along the way.
 - The line contains the agent's identity, the date and time the message was received, etc.
- ✓ The Return-Path field is added by the final message transfer agent to tell how to get back to the sender.

Header	Meaning
To:	E-mail address(es) of primary recipient(s)
Cc:	E-mail address(es) of secondary recipient(s)
Bcc:	E-mail address(es) for blind carbon copies
From:	Person or people who created the message
Sender:	E-mail address of the actual sender
Received:	Line added by each transfer agent along the route
Return-Path:	Can be used to identify a path back to the sender

RFC 822 header fields related to message transport.

Message Formats – RFC 822 (2)

- ✓ Header fields used by user agents or human recipients.

Header	Meaning
Date:	The date and time the message was sent
Reply-To:	E-mail address to which replies should be sent
Message-Id:	Unique number for referencing this message later
In-Reply-To:	Message-Id of the message to which this is a reply
References:	Other relevant Message-Ids
Keywords:	User-chosen keywords
Subject:	Short summary of the message for the one-line display

Some fields used in the RFC 822 message header.

MIME – Multipurpose Internet Mail Extensions

- The early email consisted exclusively of text message written in English and expressed in ASCII.
- Problems with international languages:
 - ✓ Languages with accents
(French, German).
 - ✓ Languages in non-Latin alphabets
(Hebrew, Russian).
 - ✓ Languages without alphabets
(Chinese, Japanese).
 - ✓ Messages not containing text at all
(audio or images).
- MIME was defined to solve the problem.

MIME (2)

- ✓ MIME continues to use RFC 822, but to add structure to the message body and define encoding rules for non-ASCII messages.
- ✓ MIME defines five new message headers.
 - **MIME-Version** – tells the user agent that this is a MIME message.
 - Any message not containing **MIME-Version** header is assumed to be an English plaintext message.
 - **Content-Transfer-Encoding** – ASCII, binary encoding (exe. files), base64 encoding, etc.
 - **Content-Type** – Type/subtype, e.g., video/mpeg.

Header	Meaning
MIME-Version:	Identifies the MIME version
Content-Description:	Human-readable string telling what is in the message
Content-Id:	Unique identifier
Content-Transfer-Encoding:	How the body is wrapped for transmission
Content-Type:	Type and format of the content

RFC 822 headers added by MIME.

MIME (3)

- Rfc822 – A message is fully encapsulated inside another, e.g., mail forwarding.
- The Partial subtype makes it possible to break an encapsulated message into pieces.
- External-body – e.g. the FTP address of a MPEG file.
- Mixed – e.g., several attachments.
- Alternative – e.g., in plain ASCII, enriched text, PostScript.
- Parallel – e.g., audio and video parts of a movie.

Type	Subtype	Description
Text	Plain	Unformatted text
	Enriched	Text including simple formatting commands
Image	Gif	Still picture in GIF format
	Jpeg	Still picture in JPEG format
Audio	Basic	Audible sound
Video	Mpeg	Movie in MPEG format
Application	Octet-stream	An uninterpreted byte sequence
	Postscript	A printable document in PostScript
Message	Rfc822	A MIME RFC 822 message
	Partial	Message has been split for transmission
	External-body	Message itself must be fetched over the net
Multipart	Mixed	Independent parts in the specified order
	Alternative	Same message in different formats
	Parallel	Parts must be viewed simultaneously
	Digest	Each part is a complete RFC 822 message

The MIME types and subtypes defined in RFC 2045.

Message Transfer

- ✓ The message transfer system is concerned with relaying messages from the sender to the recipient.
- ✓ The simplest way is to establish a transport connection.
- ✓ Within the Internet, email is delivered by having the source establish a TCP connection to port 25 of the destination.
- ✓ The daemon uses SMTP (Simple Mail Transfer Protocol), accepts incoming connections and copies messages into mailboxes.
- ✓ After setting up a TCP connection, the sender acts as the Client.
- ✓ The receiver – Server starts by sending a line of text giving its identity and telling whether it is prepared to receive mail.
- ✓ If it is not, the client releases the connection and tries again later.
- ✓ If the server is willing to accept email, the client announces the sender and the receiver.
- ✓ If the recipient (process) exists at the destination machine, the server tells the client to go-ahead and send the message.
- ✓ Then the client sends the message and the server Ack. it.

Message Transfer

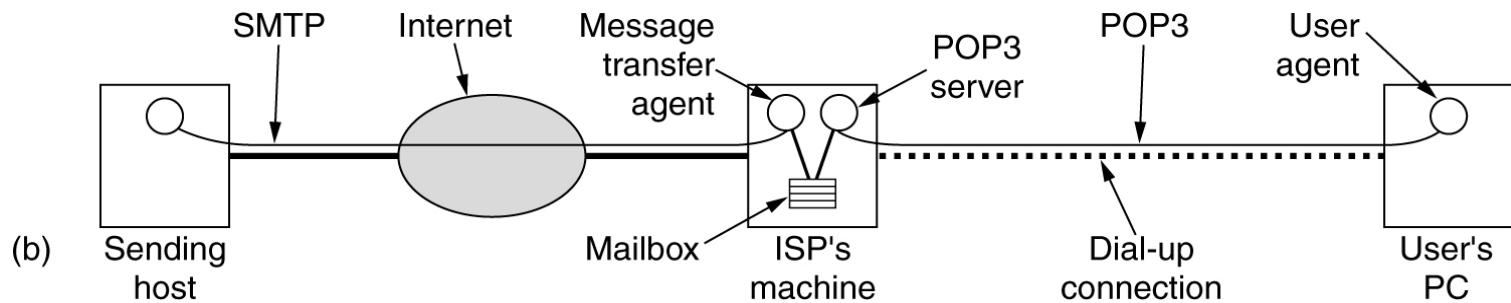
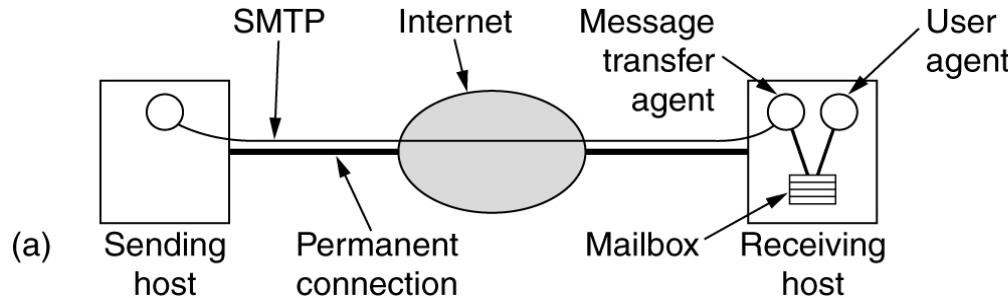
- ✓ C – Client
- ✓ S – Server.
- ✓ E.g., transferring a message from elinore@abc.com to carolyn@xyz.com.

S: 220 xyz.com SMTP service ready
C: HELO abcd.com
S: 250 xyz.com says hello to abcd.com
C: MAIL FROM: <elinor@abcd.com>
S: 250 sender ok
C: RCPT TO: <carolyn@xyz.com>
S: 250 recipient ok
C: DATA
S: 354 Send mail; end with "." on a line by itself
C: From: elinor@abcd.com
C: To: carolyn@xyz.com
C: MIME-Version: 1.0
C: Message-Id: <0704760941.AA00747@abcd.com>
C: Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
C: Subject: Earth orbits sun integral number of times
C:
C: This is the preamble. The user agent ignores it. Have a nice day.
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: text/enriched
C:
C: Happy birthday to you
C: Happy birthday to you
C: Happy birthday dear <bold> Carolyn </bold>
C: Happy birthday to you
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: message/external-body;
C: access-type="anon-ftp";
C: site="bicycle.abcd.com";
C: directory="pub";
C: name="birthday.snd"
C:
C: content-type: audio/basic
C: content-transfer-encoding: base64
C: --qwertyuiopasdfghjklzxcvbnm
C: .

CPE 490: Chap. 7, D S: 250 message accepted
C: QUIT S: 221 xyz.com closing connection

Final Delivery

- ✓ What if the receiver is not online? - An email server in the ISP machine.
- ✓ POP (Post Office Protocol) – allows user agent to contact the message transfer agent and copy the message.



- (a) Sending and reading mail when the receiver has a permanent Internet connection and the user agent runs on the same machine as the message transfer agent.
- (b) Reading e-mail when the receiver has a dial-up connection to an ISP.

POP3

- ✓ POP3
 - Authorization
 - Transactions
 - Update (e.g., deletion of messages)
 - ✓ PASS - Password
 - ✓ RETR – Retrieve
 - ✓ DELE – Marked the message for deletion.
Later the server deletes the message before breaks the TCP conn.
- | | |
|--------------------|----------------------------------|
| C: USER carolyn | S: +OK POP3 server ready |
| C: PASS vegetables | S: +OK |
| C: LIST | S: +OK login successful |
| | S: 1 2505 |
| | S: 2 14302 |
| | S: 3 8122 |
| | S: . |
| C: RETR 1 | S: (sends message 1) |
| C: DELE 1 | S: (sends message 2) |
| C: RETR 2 | S: (sends message 3) |
| C: DELE 2 | |
| C: RETR 3 | |
| C: DELE 3 | |
| C: QUIT | S: +OK POP3 server disconnecting |

Using POP3 to fetch three messages.

IMAP

- ✓ IMAP (Internet Message Access Protocol) – assumes that all the emails will remain on the server indefinitely in multiple mailboxes.
- ✓ IMAP supports email access from multiple locations, e.g., work, home, traveling.
- ✓ While POP3 assumes that the user will clear out the mailbox on every contact.

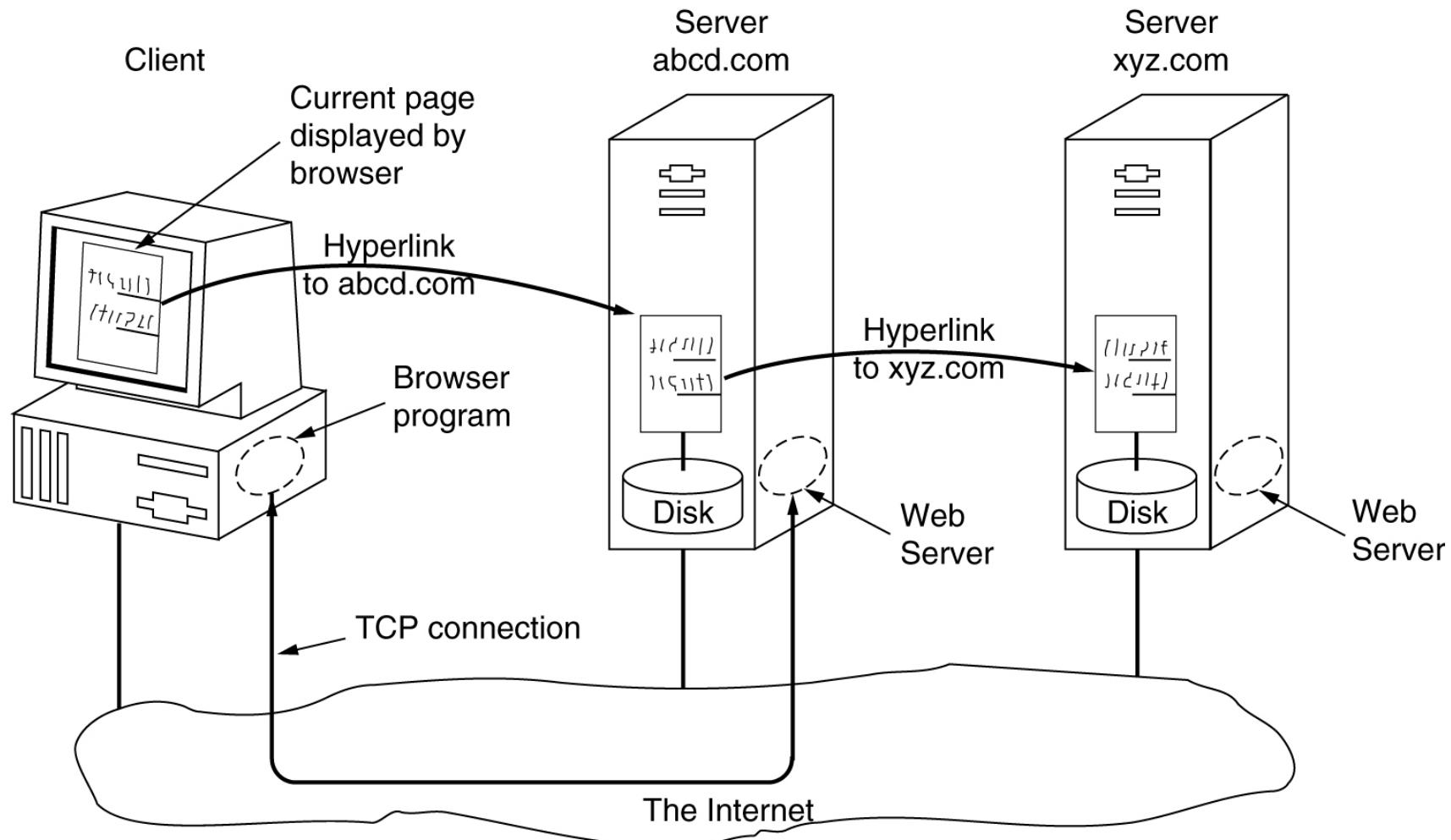
Feature	POP3	IMAP
Where is protocol defined?	RFC 1939	RFC 2060
Which TCP port is used?	110	143
Where is e-mail stored?	User's PC	Server
Where is e-mail read?	Off-line	On-line
Connect time required?	Little	Much
Use of server resources?	Minimal	Extensive
Multiple mailboxes?	No	Yes
Who backs up mailboxes?	User	ISP
Good for mobile users?	No	Yes
User control over downloading?	Little	Great
Partial message downloads?	No	Yes
Are disk quotas a problem?	No	Could be in time
Simple to implement?	Yes	No
Widespread support?	Yes	Growing

A comparison of POP3 and IMAP.

The World Wide Web

- Architectural Overview
- Static Web Documents
- Dynamic Web Documents
- HTTP – The HyperText Transfer Protocol
- Performance Enhancements
- The Wireless Web

Architectural Overview



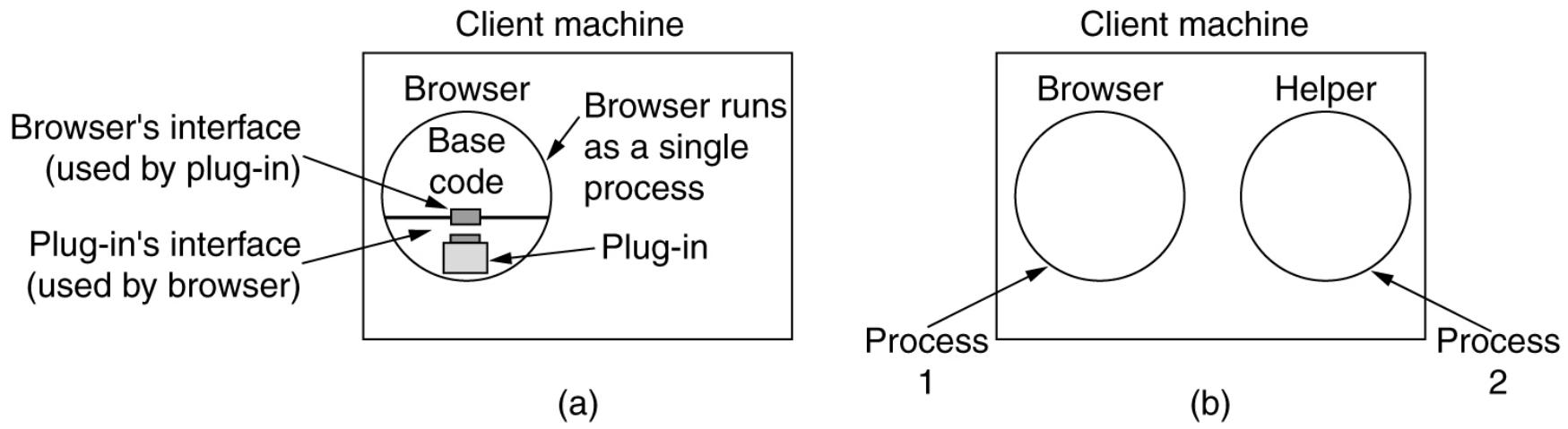
The parts of the Web model.

The Client Side

- Web pages are named using URL (Uniform Resource Locators).
- When a user clicks on a hyperlink, the browser carries out the following steps:
 - ✓ The browser determines the URL.
 - ✓ The browser asks DNS for the IP address of www.itu.org.
 - ✓ DNS replies with 156.106.192.32.
 - ✓ The browser makes a TCP connection to port 80 on 156.106.192.32.
 - ✓ It then sends over a request asking for file /home/index.html.
 - ✓ The www.itu.org server sends the file /home/index.html.
 - ✓ The TCP connection is released.
 - ✓ The browser displays all the text in /home/index.html
 - ✓ The browser fetches all displays all images in this file.

The Client Side

- ✓ If a MIME type is not one of the built-in ones, the browser consults its table of MIME types.
- ✓ This table associates a MIME type with a viewer.
- ✓ Plug-ins and helper applications
 - A plug-in is a code module that the browser fetches from a special directory on the disk and installs as an extension to itself.
 - A helper application is a complete program running as a separate process.

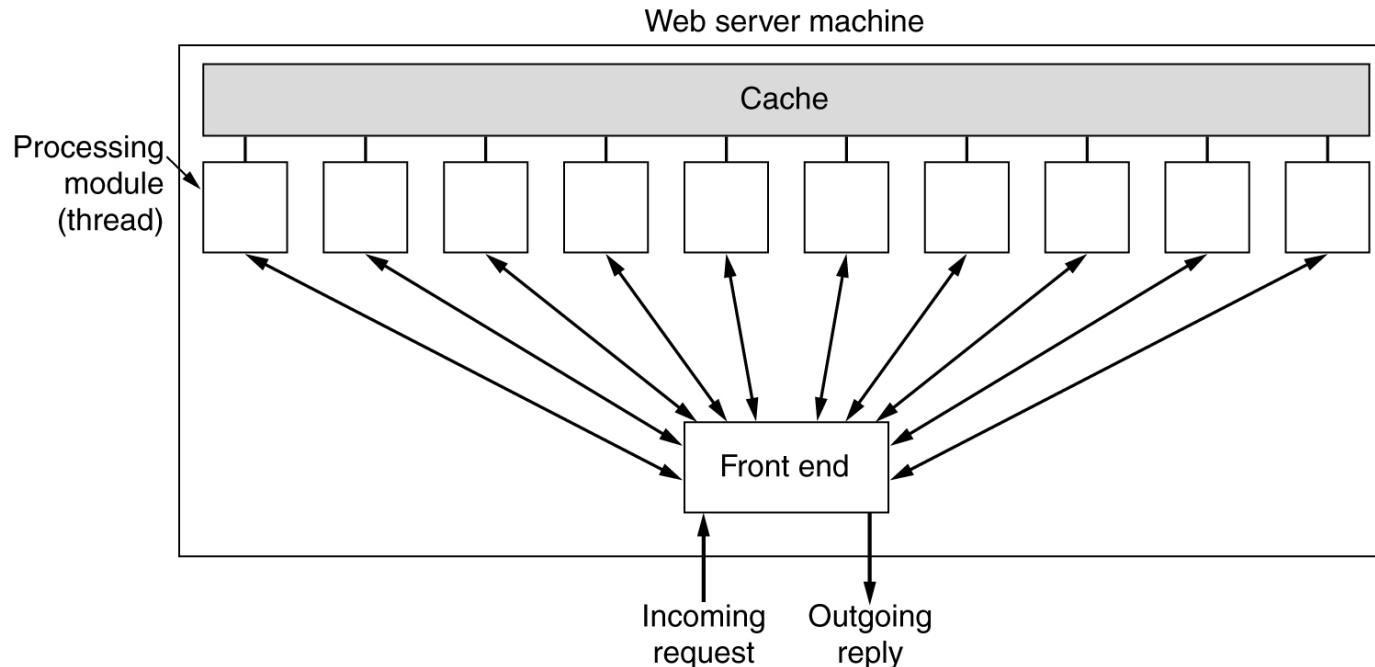


(a) A browser plug-in. (b) A helper application.

The Server Side

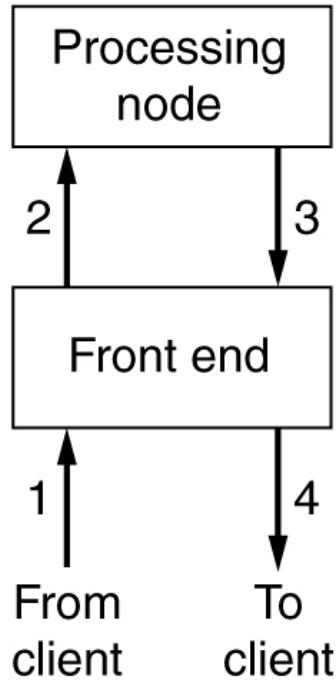
✓ The steps at the server side.

- Accept a TCP connection from a client (a browser).
- Get the name of the file requested.
- Get the file (from disk).
- Return the file to the client.
- Release the TCP connection.

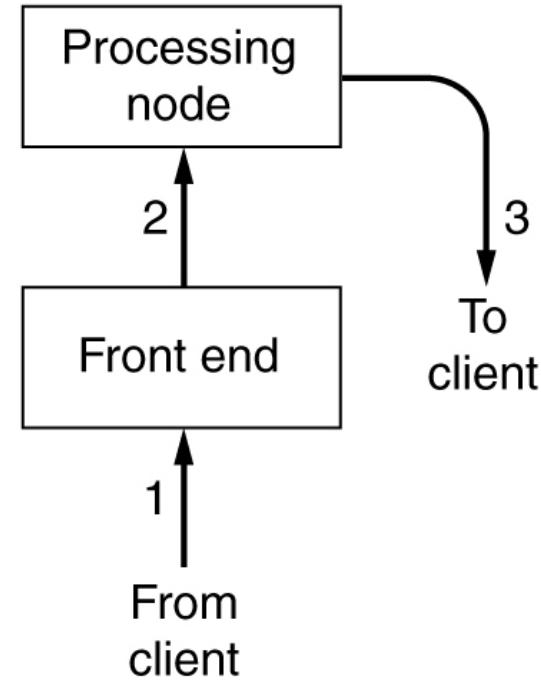


A multithreaded Web server with a front end and processing modules.

The Server Side



(a)



(b)

(a) Normal request-reply message sequence.

(b) Sequence when TCP handoff is used.

Front end – A software process.

TCP handoff – the TCP end point is passed to the processing node so it can reply directly to the client.

URLs – Uniform Resource Locators

Name	Used for	Example
http	Hypertext (HTML)	http://www.cs.vu.nl/~ast/
ftp	FTP	ftp://ftp.cs.vu.nl/pub/minix/README
file	Local file	file:///usr/suzanne/prog.c
news	Newsgroup	news:comp.os.minix
news	News article	news:AA0134223112@cs.utah.edu
gopher	Gopher	gopher://gopher.tc.umn.edu/11/Libraries
mailto	Sending e-mail	mailto:JohnUser@acm.org
telnet	Remote login	telnet://www.w3.org:80

Some common URLs.

Statelessness and Cookies

- ✓ The Web is basically stateless.
- ✓ Cookies can be used to record certain connection state info.
- ✓ When a client requests a Web page, the server can supply additional info. along with the requested page, such as cookies.
- ✓ A cookie may contain up to five fields:
 - Domain – where the cookie came from.
 - Path – a path in the server's directory structure that identifies which parts of the server's file tree may use the cookie. E.g., / means the whole tree.
 - Content – takes the form: name = value, which could be anything the server wants.

Domain	Path	Content	Expires	Secure
toms-casino.com	/	CustomerID=497793521	15-10-02 17:00	Yes
joes-store.com	/	Cart=1-00501;1-07031;2-13721	11-10-02 14:22	No
aportal.com	/	Prefs=Stk:SUNW+ORCL;Spt:Jets	31-12-10 23:59	No
sneaky.com	/	UserID=3627239101	31-12-12 23:59	No

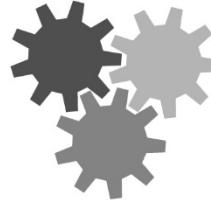
Some examples of cookies.

HTML – HyperText Markup Language

```
<html>
<head><title> AMALGAMATED WIDGET, INC. </title> </head>
<body><h1>Welcome to AWI's Home Page</h1>
 <br>
We are so happy that you have chosen to visit <b>Amalgamated Widget's </b>
home page. We hope <i>you </i> will find all the information you need here.
<p>Below we have links to information about our many fine products.
You can order electronically (by WWW), by telephone, or by fax.</p>
<hr>
<h2>Product information </h2>
<ul>
  <li> <a href="http://widget.com/products/big"> Big widgets </a>
  <li> <a href="http://widget.com/products/little"> Little widgets </a>
</ul>
<h2>Telephone numbers</h2>
<ul>
  <li> By telephone: 1-800-WIDGETS
  <li> By fax: 1-415-765-4321
</ul>
</body>
</html>
```

(a)

Welcome to AWI's Home Page



We are so happy that you have chosen to visit **Amalgamated Widget's** home page. We hope you will find all the information you need here.

Below we have links to information about our many fine products. You can order electronically (by WWW), by telephone, or by FAX.

Product Information

- [Big widgets](http://widget.com/products/big)
- [Little widgets](http://widget.com/products/little)

Telephone numbers

- 1-800-WIDGETS
- 1-415-765-4321

(b)

(a) The HTML for a sample Web page. (b) The formatted page.

HTML (2)

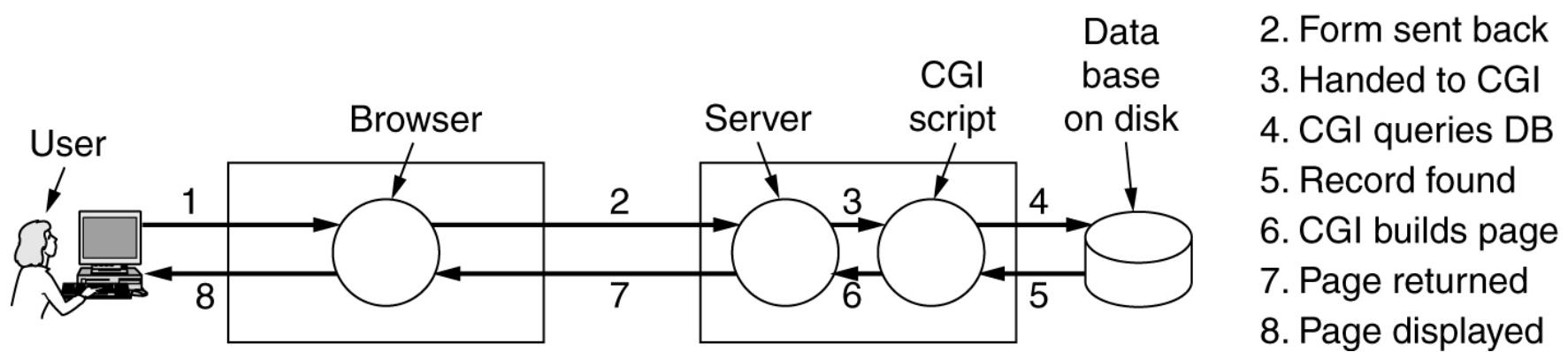
Tag	Description
<html> ... </html>	Declares the Web page to be written in HTML
<head> ... </head>	Delimits the page's head
<title> ... </title>	Defines the title (not displayed on the page)
<body> ... </body>	Delimits the page's body
<h <i>n</i> > ... </h <i>n</i> >	Delimits a level <i>n</i> heading
 ... 	Set ... in boldface
<i> ... </i>	Set ... in italics
<center> ... </center>	Center ... on the page horizontally
 ... 	Brackets an unordered (bulleted) list
 ... 	Brackets a numbered list
	Starts a list item (there is no)
 	Forces a line break here
<p>	Starts a paragraph
<hr>	Inserts a Horizontal rule
	Displays an image here
 ... 	Defines a hyperlink

A selection of common HTML tags. some can have additional parameters.

Static Web Documents

- ✓ Tables
- ✓ Forms
- ✓ XML (eXtensible Markup Language) describes Web content in a structured way.
- ✓ XSL (eXtensible Style Language) describes the formatting independently of the content.

Dynamic Web Documents

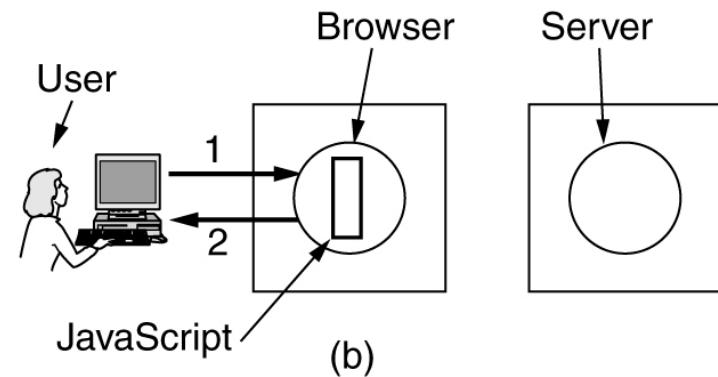
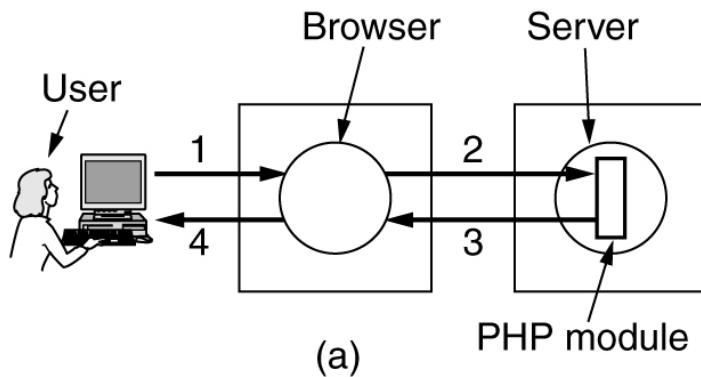


Steps in processing the information from an HTML form.

Scripts for generating Dynamic Web Documents:

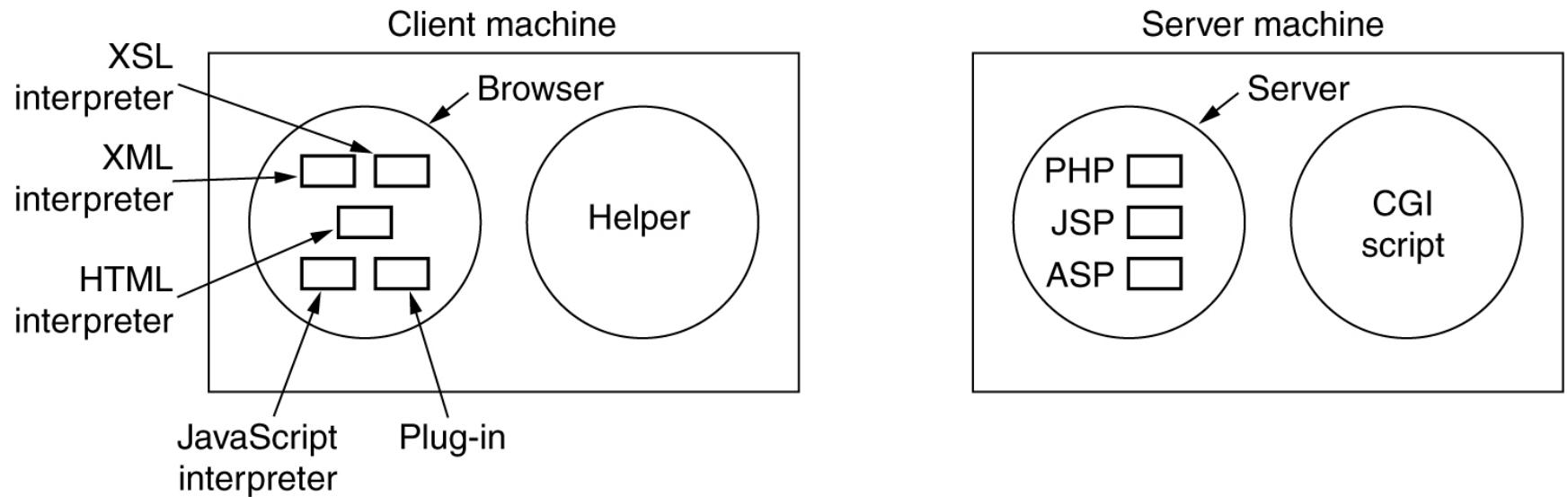
- CGI – Common Gateway Interface
- PHP – PHP Hypertext Preprocessor
- Java Script
- VB Script

Dynamic Web Pages



- (a) Server-side scripting (interaction with a remote database) with PHP.
- (b) Client-side scripting (interaction with the user at the client computer) with JavaScript.

Dynamic Web Pages



The various ways to generate and display content.

HTTP Methods

HTTP (HyperText Transfer Protocol) – WWW

- Each interaction consists of one ASCII request, followed by one RFC 822 MIME-like response

Method	Description
GET	Request to read a Web page
HEAD	Request to read a Web page's header
PUT	Request to store a Web page
POST	Append to a named resource (e.g., a Web page)
DELETE	Remove the Web page
TRACE	Echo the incoming request
CONNECT	Reserved for future use
OPTIONS	Query certain options

The built-in HTTP request methods.

HTTP Methods (2)

Code	Meaning	Examples
1xx	Information	100 = server agrees to handle client's request
2xx	Success	200 = request succeeded; 204 = no content present
3xx	Redirection	301 = page moved; 304 = cached page still valid
4xx	Client error	403 = forbidden page; 404 = page not found
5xx	Server error	500 = internal server error; 503 = try again later

The status code response groups.

HTTP Message Headers

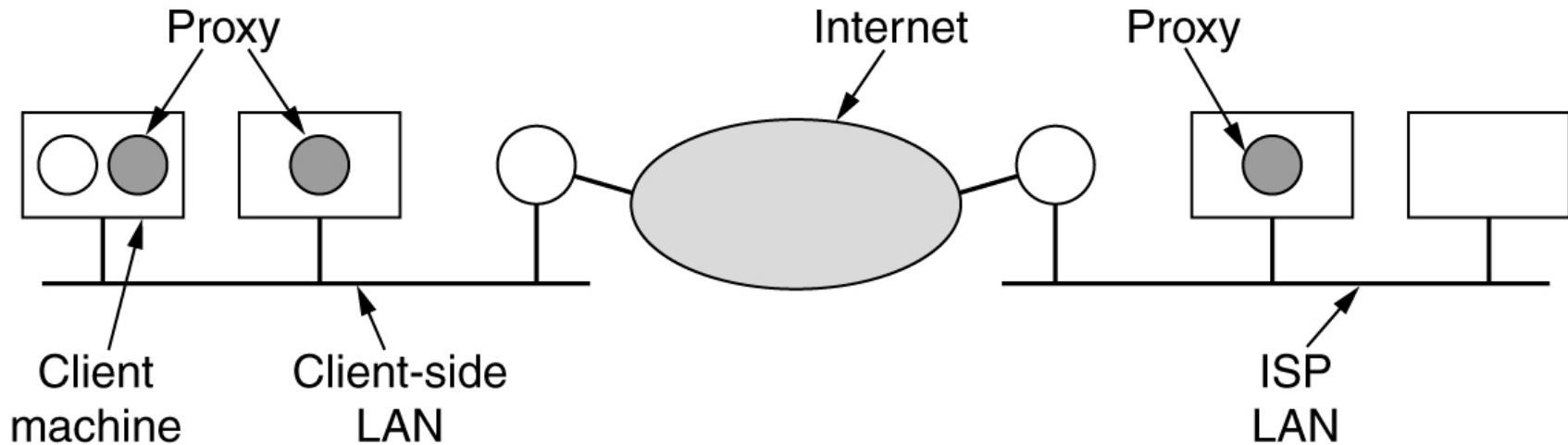
- ✓ The request line may be followed by additional lines with more info. – request headers (Response headers are similar).

Header	Type	Contents
User-Agent	Request	Information about the browser and its platform
Accept	Request	The type of pages the client can handle
Accept-Charset	Request	The character sets that are acceptable to the client
Accept-Encoding	Request	The page encodings the client can handle
Accept-Language	Request	The natural languages the client can handle
Host	Request	The server's DNS name
Authorization	Request	A list of the client's credentials
Cookie	Request	Sends a previously set cookie back to the server
Date	Both	Date and time the message was sent
Upgrade	Both	The protocol the sender wants to switch to
Server	Response	Information about the server
Content-Encoding	Response	How the content is encoded (e.g., gzip)
Content-Language	Response	The natural language used in the page
Content-Length	Response	The page's length in bytes
Content-Type	Response	The page's MIME type
Last-Modified	Response	Time and date the page was last changed
Location	Response	A command to the client to send its request elsewhere
Accept-Ranges	Response	The server will accept byte range requests
Set-Cookie	Response	The server wants the client to save a cookie

Some HTTP message
headers.

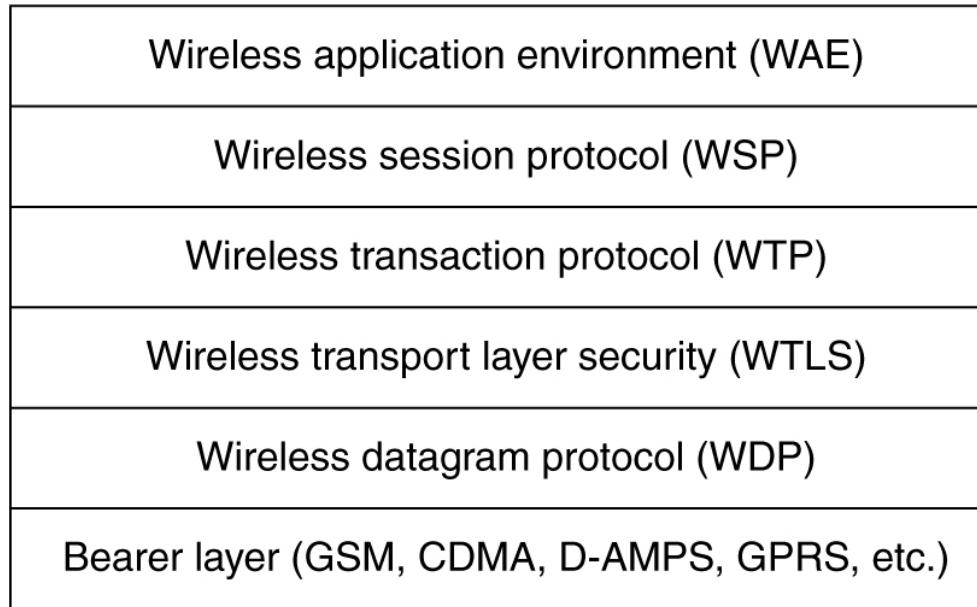
Caching

- Caching - to save pages that have been requested in case they are used again.
- Some process (called a proxy) maintains the cache.



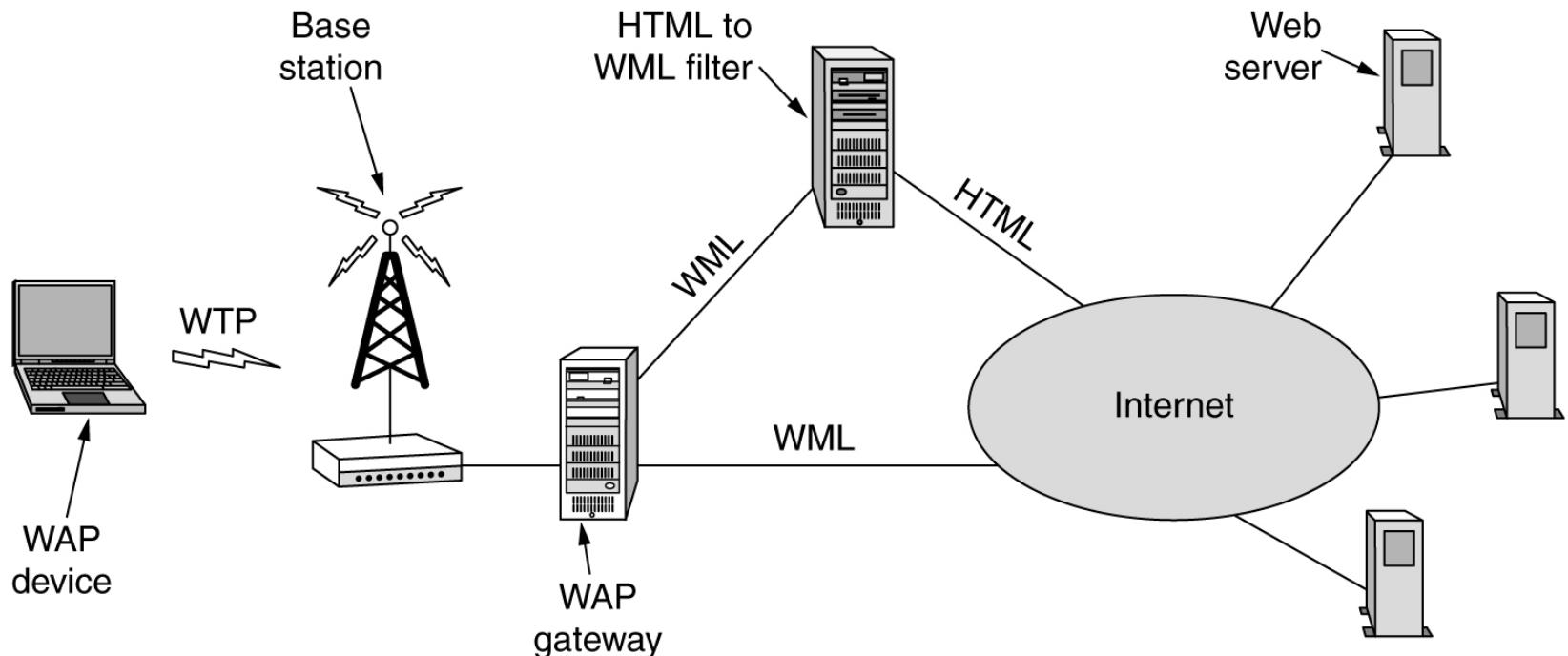
Hierarchical caching with three proxies.

WAP – The Wireless Application Protocol



The WAP (Wireless Application Protocol) protocol stack.

WAP (2)



The WAP architecture.

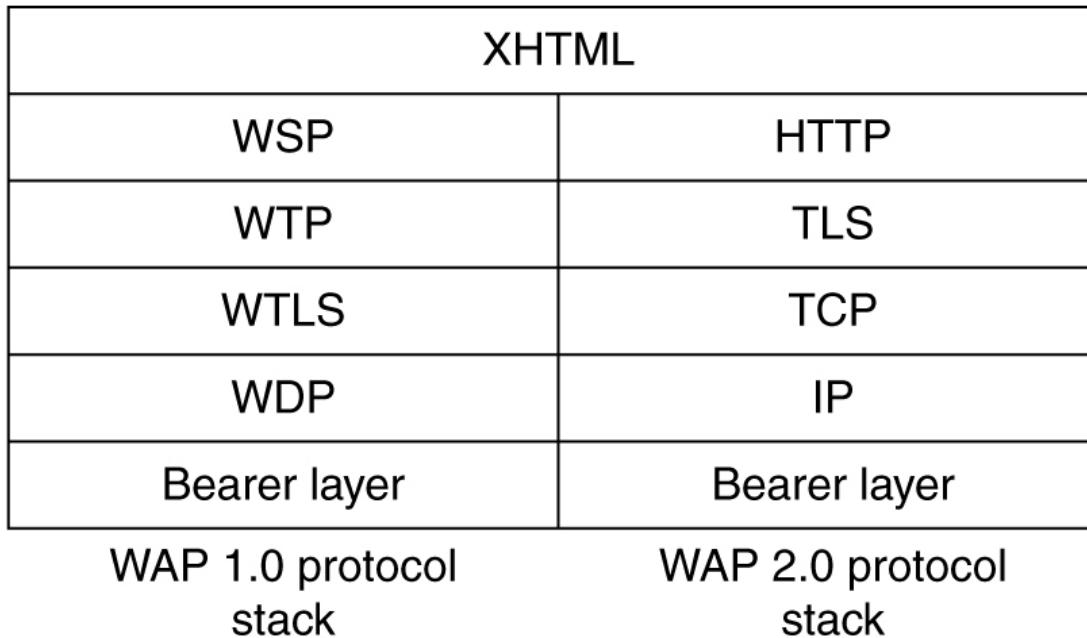
WML – Wireless Markup Language

Second-Generation Wireless Web

New features of WAP 2.0.

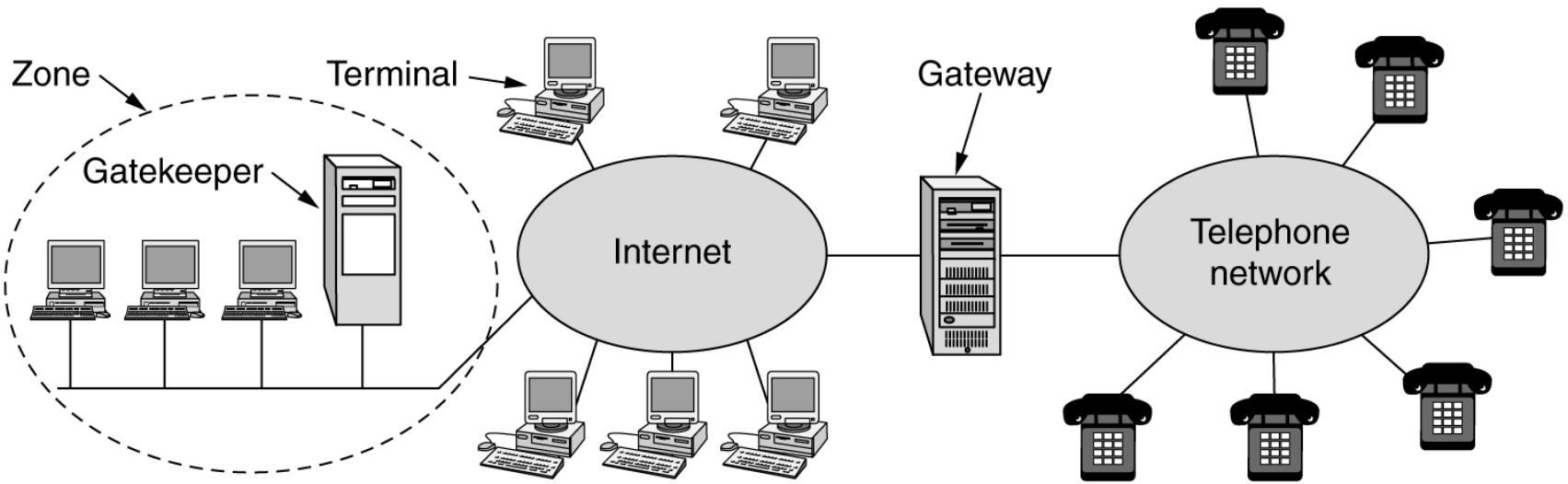
- Push model as well as pull model.
- Support for integrating telephony into apps.
- Multimedia messaging.
- Inclusion of 264 pictograms.
 - ✓ A pictogram (pictograph) is a symbol which represents an object or a concept by illustration.
- Interface to a storage device.
- Support for plug-ins in the browser.

Second-Generation Wireless Web (2)



WAP 2.0 supports two protocol stacks.

Multimedia - Voice over IP

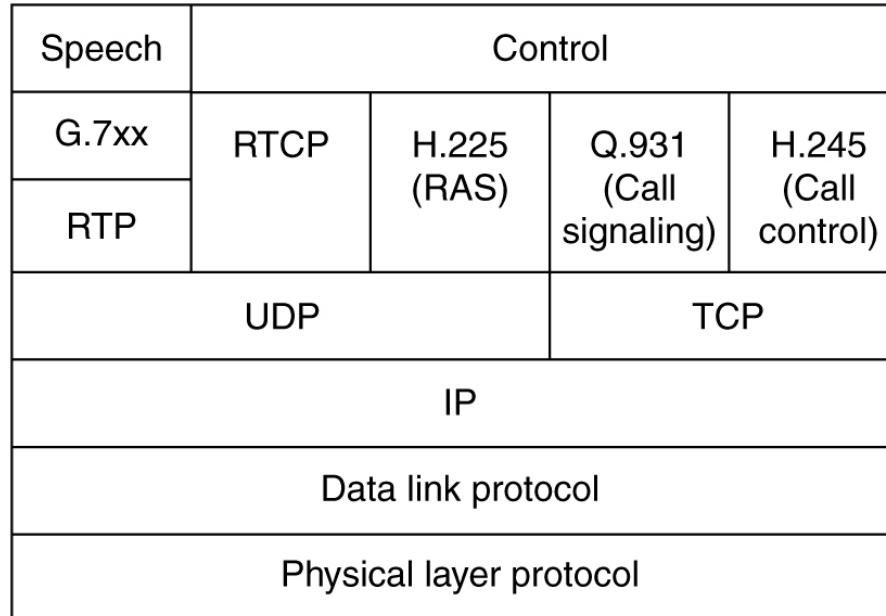


The H323 architectural model for Internet telephony.

Voice over IP (2)

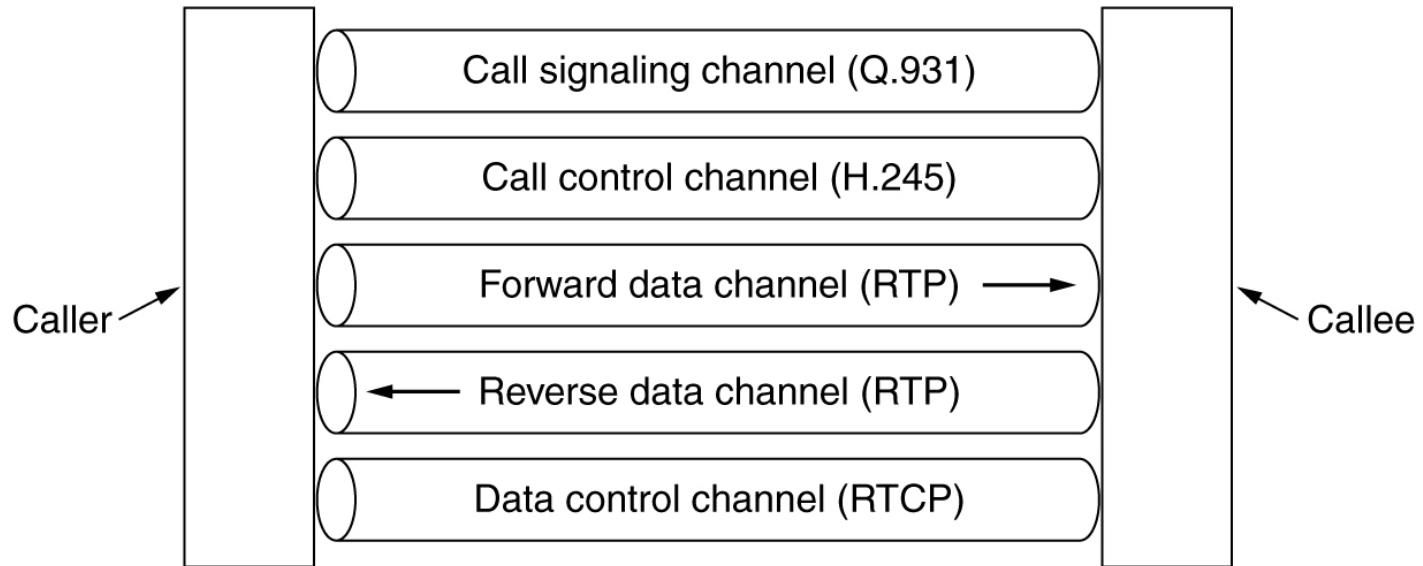
Call signaling: Establish and release connections, providing dial tones, making ringing sounds.

Call control: Selecting compression algorithms for voice, negotiating bit rate, etc.



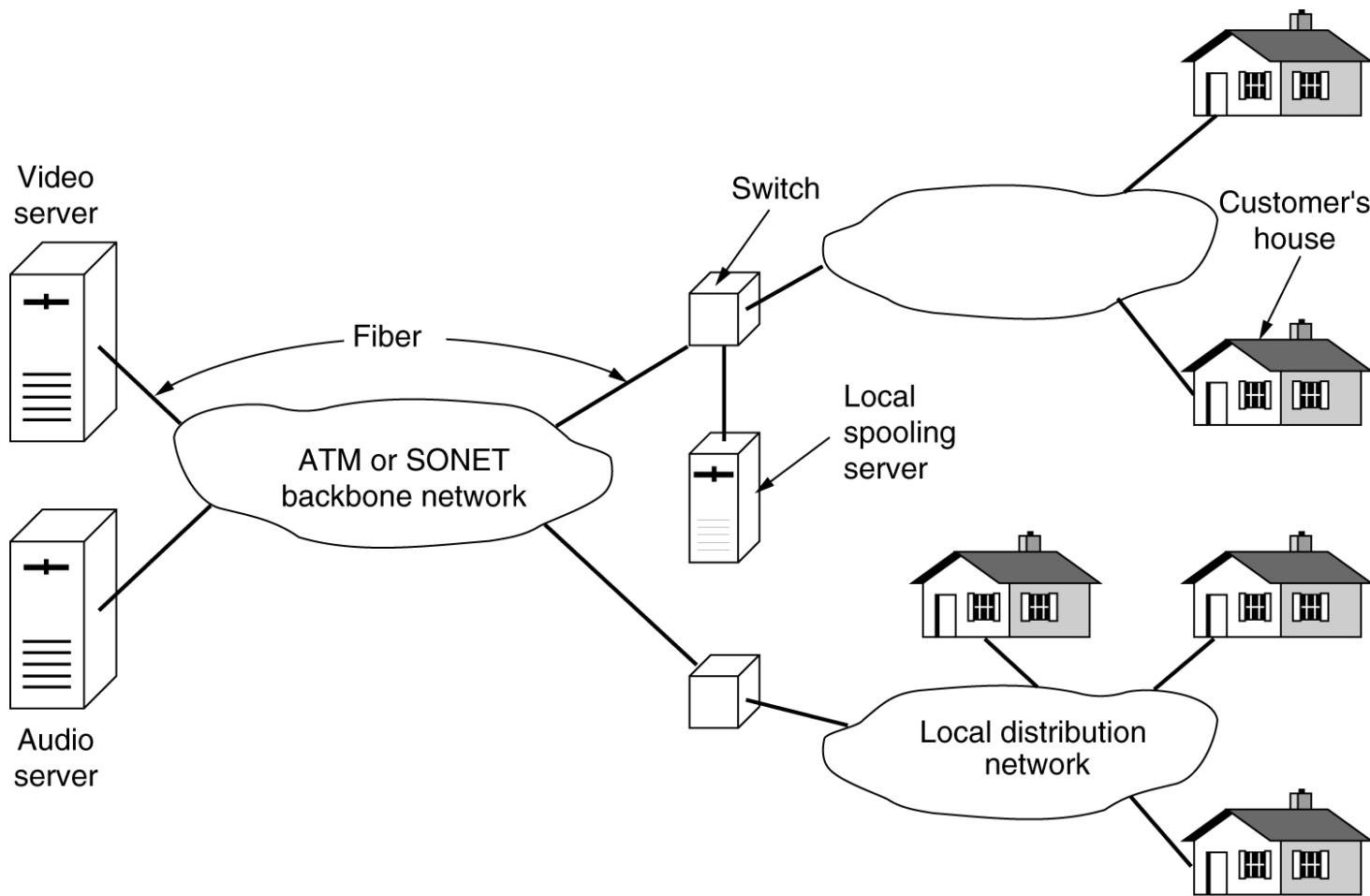
The H323 protocol stack.

Voice over IP (3)



Logical channels between the caller and callee during a call.

Video on Demand



Overview of a video-on-demand system.