

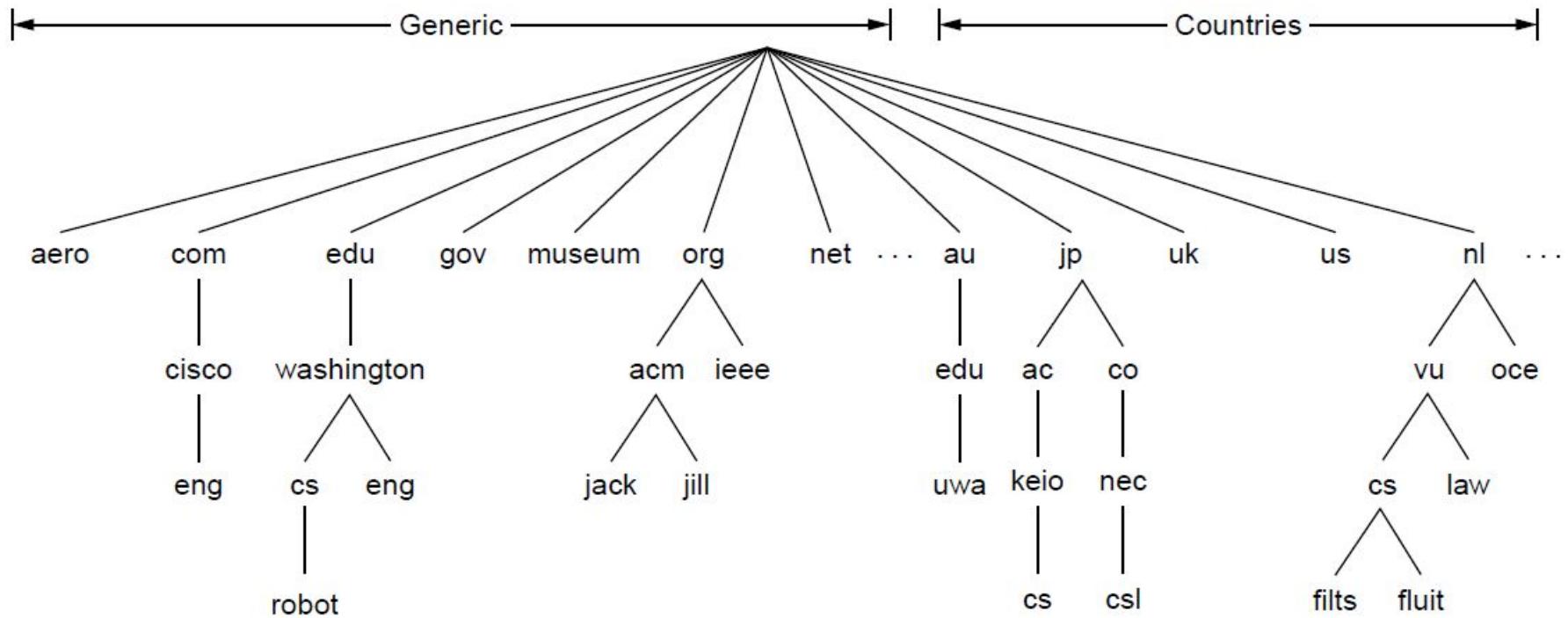
The Application Layer

Chapter 7

DNS – The Domain Name System

- The DNS name space
- Domain Resource records
- Name servers

The DNS Name Space (1)



A portion of the Internet domain name space.

The DNS Name Space (2)

Domain	Intended use	Start date	Restricted?
com	Commercial	1985	No
edu	Educational institutions	1985	Yes
gov	Government	1985	Yes
int	International organizations	1988	Yes
mil	Military	1985	Yes
net	Network providers	1985	No
org	Non-profit organizations	1985	No
aero	Air transport	2001	Yes
biz	Businesses	2001	No
coop	Cooperatives	2001	Yes
info	Informational	2002	No
museum	Museums	2002	Yes
name	People	2002	No
pro	Professionals	2002	Yes
cat	Catalan	2005	Yes
jobs	Employment	2005	Yes
mobi	Mobile devices	2005	Yes
tel	Contact details	2005	Yes
travel	Travel industry	2005	Yes
xxx	Sex industry	2010	No

Generic top-level domains

Domain Resource Records (1)

Type	Meaning	Value
SOA	Start of authority	Parameters for this zone
A	IPv4 address of a host	32-Bit integer
AAAA	IPv6 address of a host	128-Bit integer
MX	Mail exchange	Priority, domain willing to accept email
NS	Name server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
SPF	Sender policy framework	Text encoding of mail sending policy
SRV	Service	Host that provides it
TXT	Text	Descriptive ASCII text

The principal DNS resource record types

Domain Resource Records (2)

```
; Authoritative data for cs.vu.nl
cs.vu.nl.      86400  IN  SOA   star boss (9527,7200,7200,241920,86400)
cs.vu.nl.      86400  IN  MX    1 zephyr
cs.vu.nl.      86400  IN  MX    2 top
cs.vu.nl.      86400  IN  NS    star

star           86400  IN  A     130.37.56.205
zephyr         86400  IN  A     130.37.20.10
top            86400  IN  A     130.37.20.11
www            86400  IN  CNAME star.cs.vu.nl
ftp             86400  IN  CNAME zephyr.cs.vu.nl

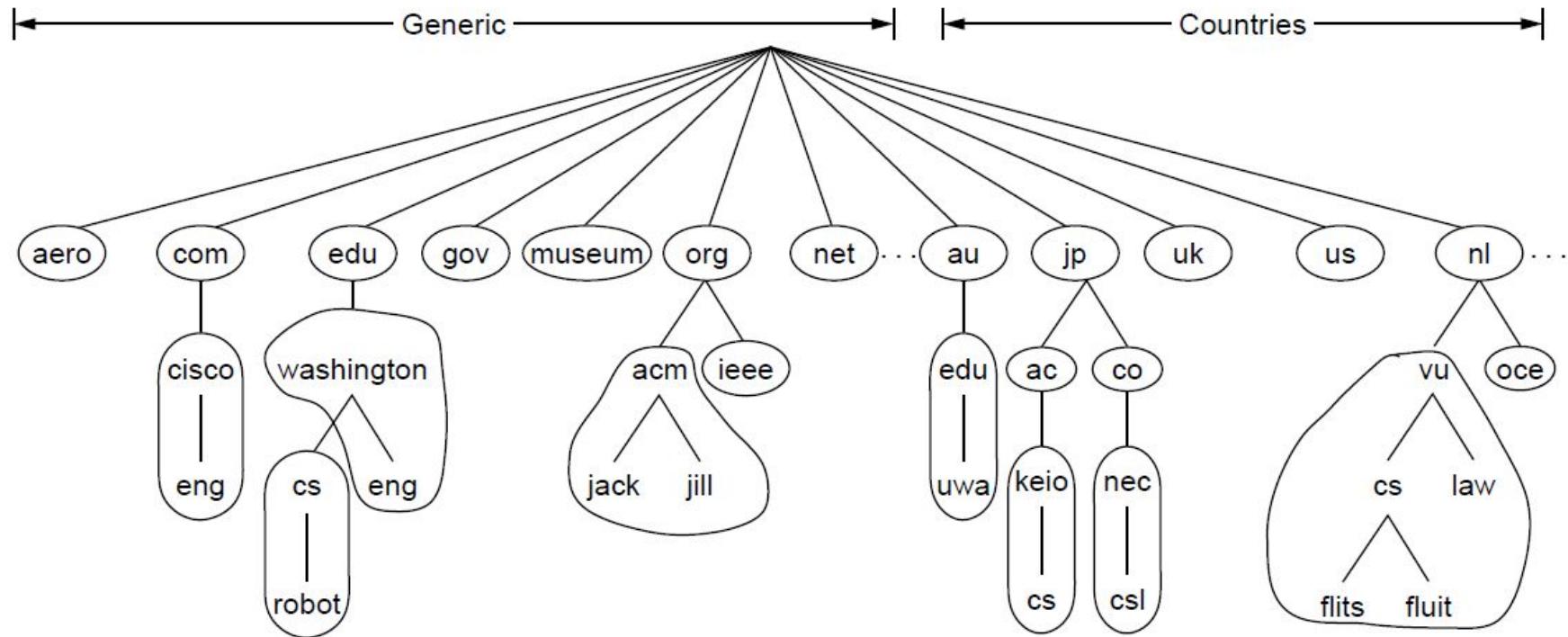
flits          86400  IN  A     130.37.16.112
flits          86400  IN  A     192.31.231.165
flits          86400  IN  MX   1 flits
flits          86400  IN  MX   2 zephyr
flits          86400  IN  MX   3 top

rowboat        IN  A     130.37.56.201
                IN  MX   1 rowboat
                IN  MX   2 zephyr

little-sister  IN  A     130.37.62.23
laserjet       IN  A     192.31.231.216
```

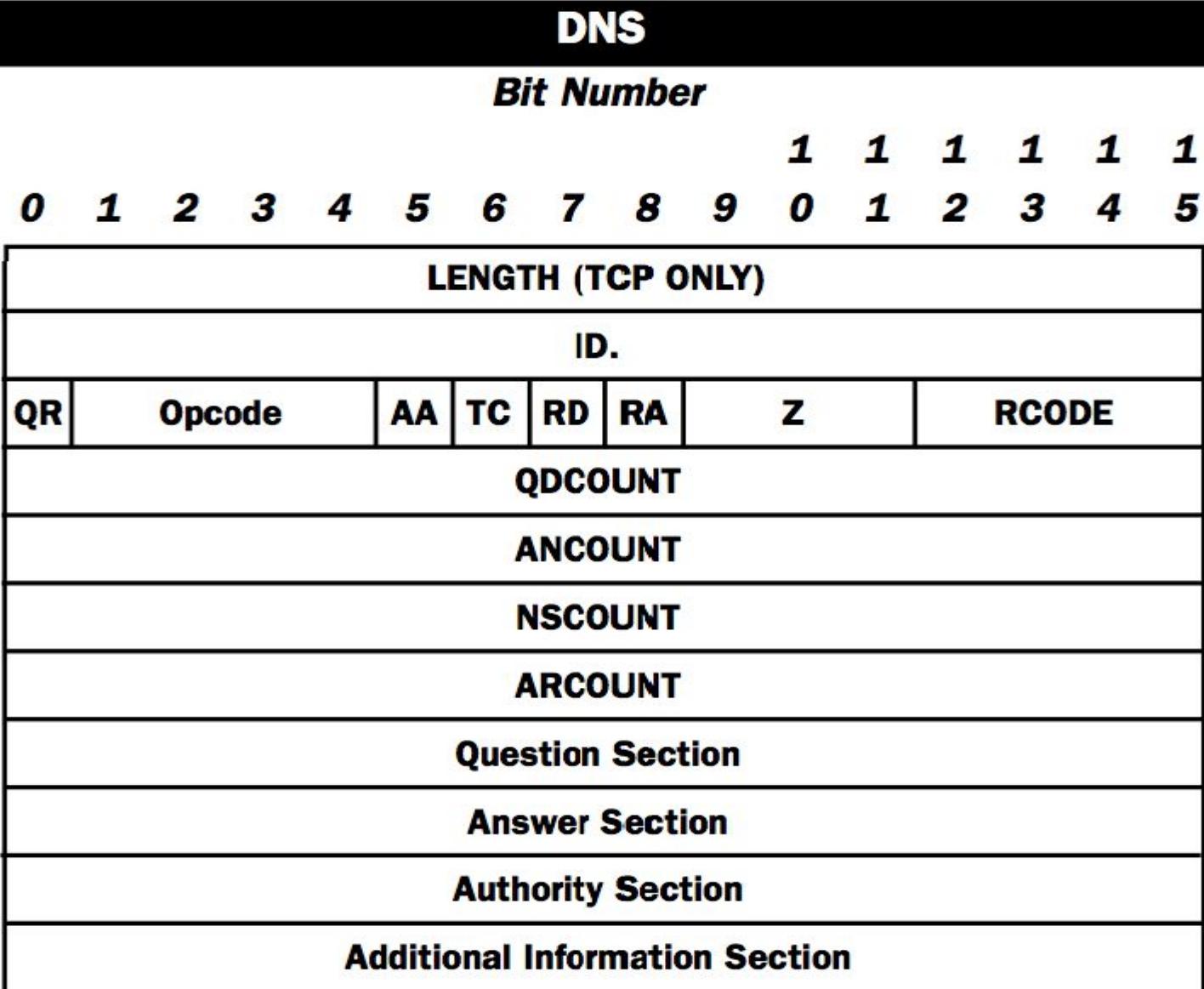
A portion of a possible DNS database for *cs.vu.nl*.

Name Servers (1)



Part of the DNS name space divided into zones
(which are circled).

DNS Header



Identification – This is a 16 bit number through which a query is identified. This number is set by client and when a response is send back to the client, the same identification number is used.

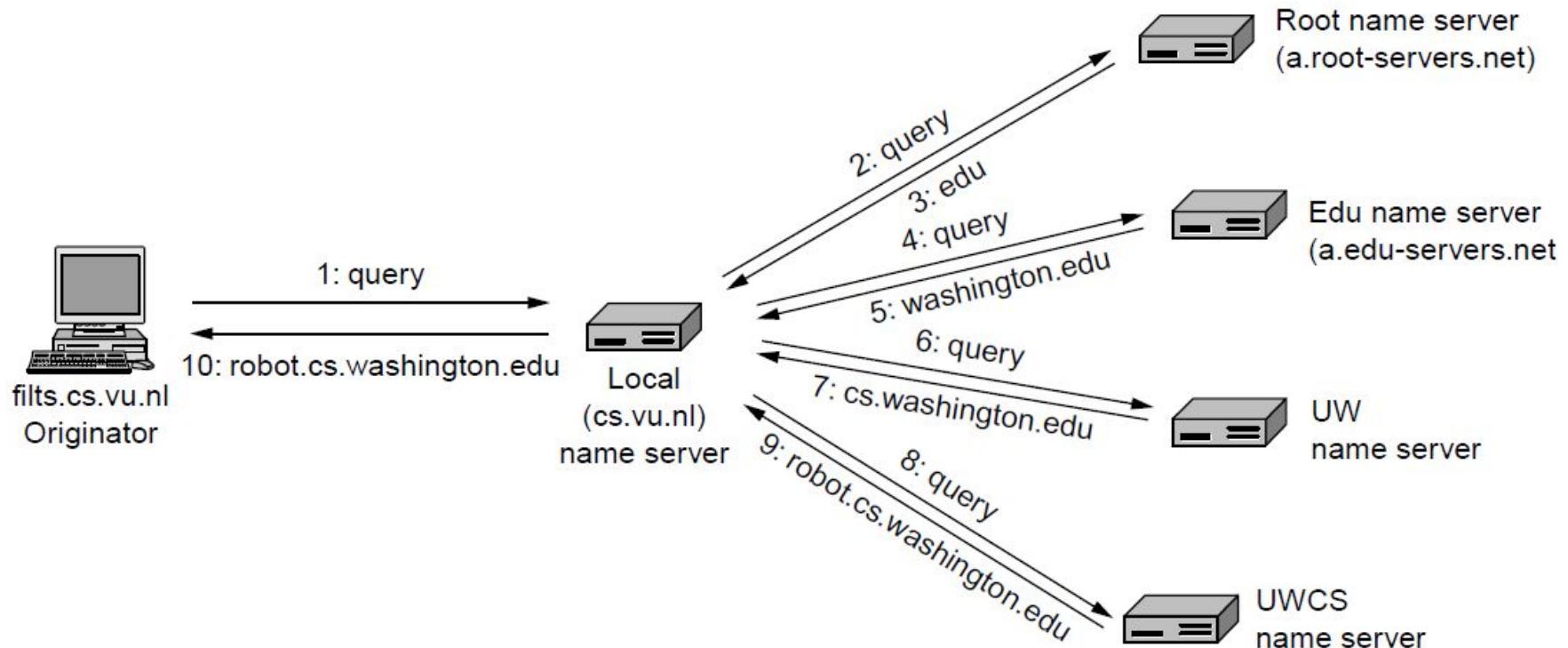
2. The Flag consists of 16-bit parameter:

- a) The first (0th)bit indicates query(0) or response(1)
- b) Next three bits (1-4) indicates ‘Standard Query (0)’, ‘Inverse Query (1)’ and ‘Server Status Request (2)’.
- c) The 5th bit field indicates Authoritative answer. The name server is authoritative for the domain in the question section.
- d) The 6th bit field is set if message was truncated. With UDP this means that the total size of the reply exceeded 512 bytes and only the first 512 bytes of reply were returned.

a)

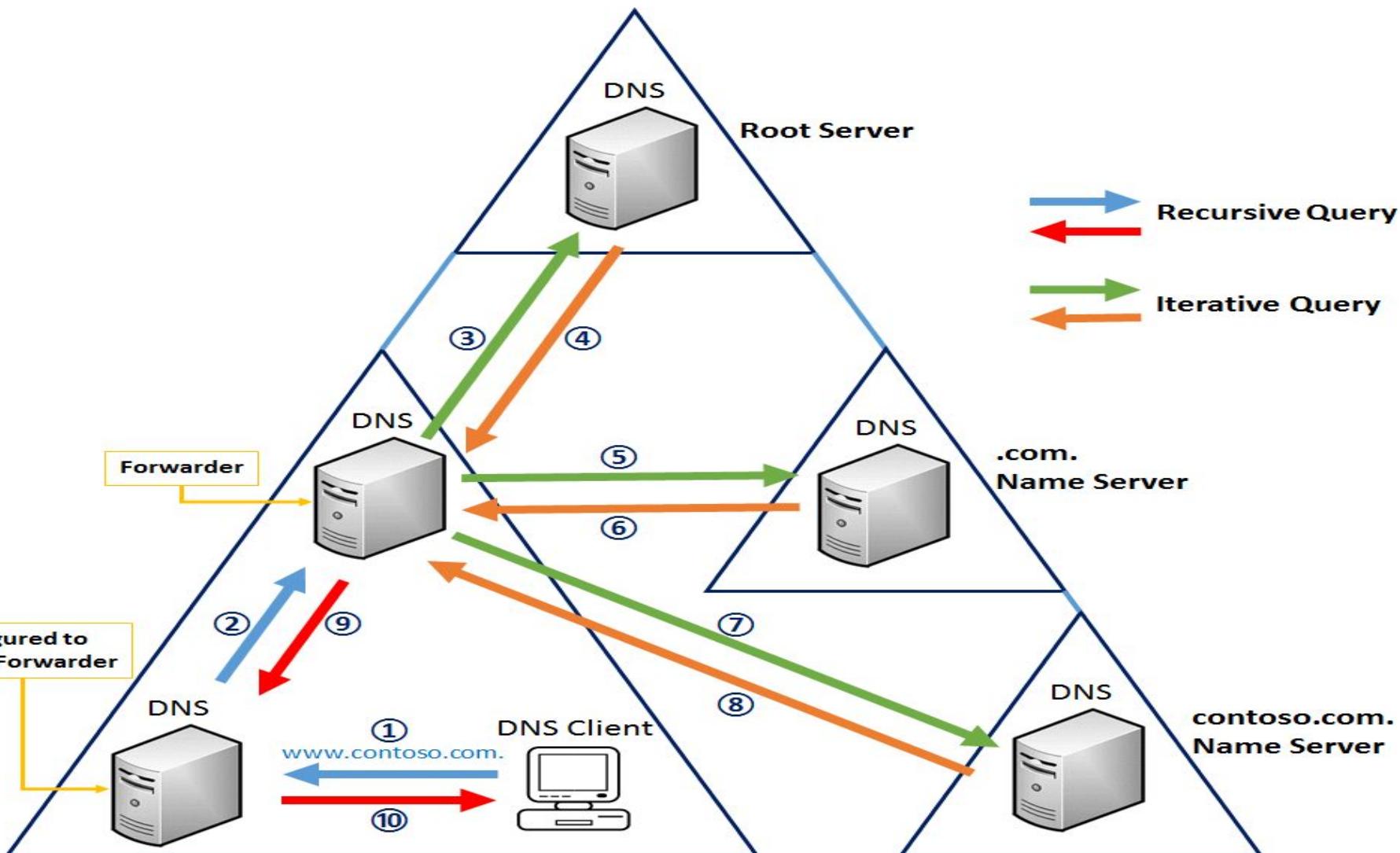
- e) The 7th bit field indicates Recursion Desired .This bit can be set in a query and is returned in the response.
 - f) The 8th bit field indicates Recursion Available or not.
 - g) The next 3 bits (9-11) has to be 0.
 - h) The Next 4 bits (12-15) give a return code where 0 signifies No Error and 3 signifies Name Error.
3. The fields labeled Number of... give each a count of entries in the corresponding sections in the message.
 4. The Question section is filled by the client and contains information about the query that is being made. Each question has a name and type associated with it.
 5. The Answer, Authority, and Additional Information sections consist of a set of resource records that describe the domain names and mappings.

Name Servers (2)



Example of a resolver looking up a remote name in 10 steps.

Name Server: Iterative v/s Recursive

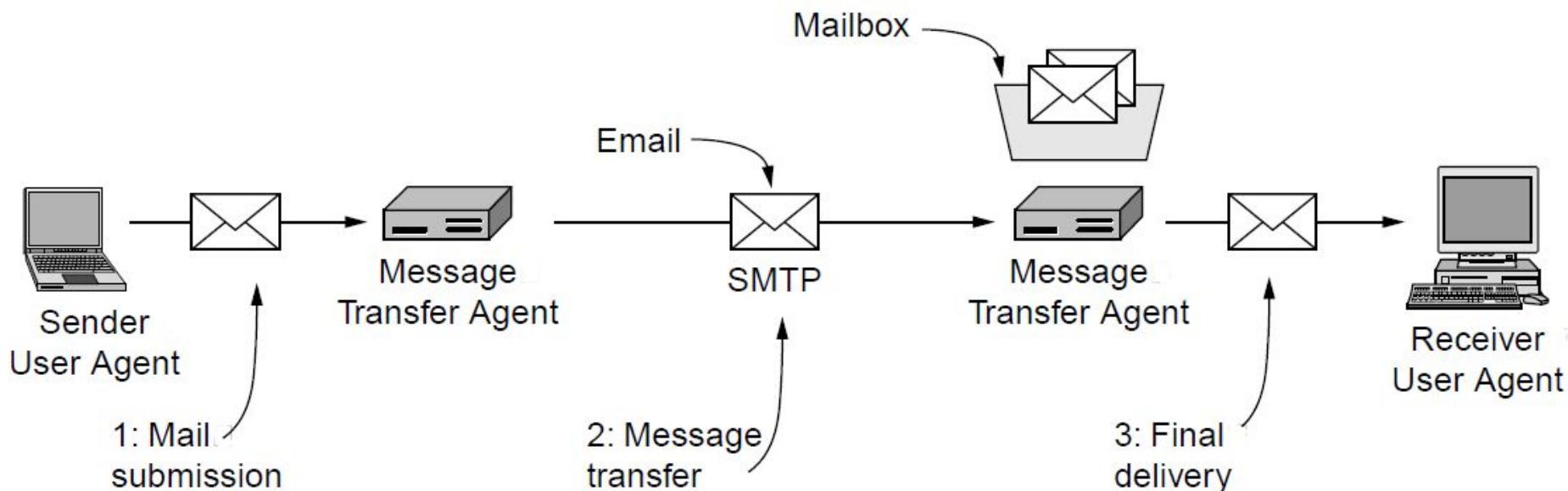


- a) The "Recursion Desired" (RD) flag sent by the client in the DNS request header (see RFC 1035) asks the server "please give me the complete answer to this question".
- b) That server than *iteratively* asks the chain of name servers for the correct answer. Those queries shouldn't themselves have the RD bit set.
- c) Ultimately the recursive server's response will have the "Recursion Available" (RA) flag set, indicating that the answer was indeed fully answered. Conversely an *authoritative* server will not set the RA flag

Electronic Mail

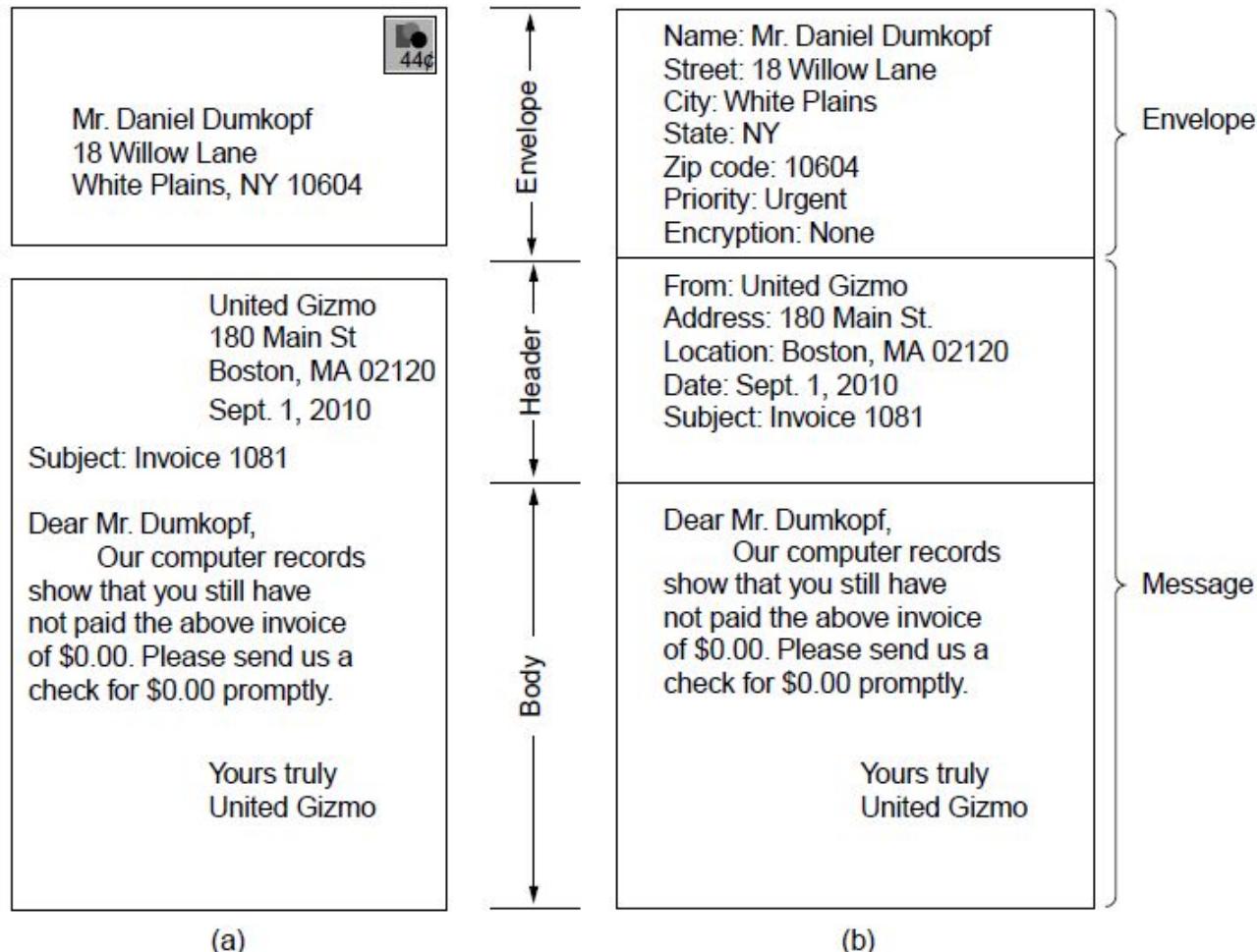
- Architecture and services
- The user agent
- Message formats
- Message transfer
- Final delivery

Architecture and Services (1)



Architecture of the email system

Architecture and Services (2)



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

The User Agent

Message folders

Mail Folders
All items
Inbox
Networks
Travel
Junk Mail

Message summary

From	Subject	Received
trudy	✉ Not all Trudys are nasty	Today
Andy	📎 Material on RFID privacy	Today
djw	❗ Have you seen this?	Mar 4
Amy N. Wong	Request for information	Mar 3
guido	Re: Paper acceptance	Mar 3
lazowska	More on that	Mar 2
lazowska	📎 New report out	Mar 2
...

Search 

Mailbox search

A. Student	Graduate studies?	Mar 1
Dear Professor, I recently completed my undergraduate studies with distinction at an excellent university. I will be visiting your		
... ... Message		

Typical elements of the user agent interface

Message Formats (1)

Header	Meaning
To:	Email address(es) of primary recipient(s)
Cc:	Email address(es) of secondary recipient(s)
Bcc:	Email address(es) for blind carbon copies
From:	Person or people who created the message
Sender:	Email 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 5322 header fields related to message transport.

Message Formats (2)

Header	Meaning
Date:	The date and time the message was sent
Reply-To:	Email 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 5322 message header.

Message Formats (3)

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

Message headers added by MIME.

Message Formats (4)

Type	Example subtypes	Description
text	plain, html, xml, css	Text in various formats
image	gif, jpeg, tiff	Pictures
audio	basic, mpeg, mp4	Sounds
video	mpeg, mp4, quicktime	Movies
model	vrml	3D model
application	octet-stream, pdf, javascript, zip	Data produced by applications
message	http, rfc822	Encapsulated message
multipart	mixed, alternative, parallel, digest	Combination of multiple types

MIME content types and example subtypes.

Message Transfer (1)

From: alice@cs.washington.edu
To: bob@ee.uwa.edu.au
MIME-Version: 1.0
Message-Id: <0704760941.AA00747@cs.washington.edu>
Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
Subject: Earth orbits sun integral number of times

This is the preamble. The user agent ignores it. Have a nice day.

--qwertyuiopasdfghjklzxcvbnm
Content-Type: text/html

<p>Happy birthday to you

Happy birthday to you

Happy birthday dear Bob

Happy birthday to you</p>

• • •

A multipart message containing HTML and audio alternatives.

Message Transfer (2)

```
...  
--qwertyuiopasdfghjklzxcvbnm  
Content-Type: message/external-body;  
access-type="anon-ftp";  
site="bicycle.cs.washington.edu";  
directory="pub";  
name="birthday.snd"
```

```
content-type: audio/basic  
content-transfer-encoding: base64  
--qwertyuiopasdfghjklzxcvbnm--
```

A multipart message containing HTML and audio alternatives.

Message Transfer (3)

```
S: 220 ee.uwa.edu.au SMTP service ready
C: HELO abcd.com
    S: 250 cs.washington.edu says hello to ee.uwa.edu.au
C: MAIL FROM: <alice@cs.washington.edu>
    S: 250 sender ok
C: RCPT TO: <bob@ee.uwa.edu.au>
    S: 250 recipient ok
C: DATA
    S: 354 Send mail; end with "." on a line by itself
C: From: alice@cs.washington.edu
C: To: bob@ee.uwa.edu.au
C: MIME-Version: 1.0
C: Message-Id: <0704760941.AA00747@ee.uwa.edu.au>
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/html
C:
    . . .
```

Sending a message from *alice@cs.washington.edu* to
bob@ee.uwa.edu.au.

Message Transfer (4)

```
...  
C: <p>Happy birthday to you  
C: Happy birthday to you  
C: Happy birthday dear <b> Bob </b>  
C: Happy birthday to you  
C:  
C: --qwertyuiopasdfghjklzxcvbnm  
C: Content-Type: message/external-body;  
C: access-type="anon-ftp";  
C: site="bicycle.cs.washington.edu";  
C: directory="pub";  
C: name="birthday.snd"  
C:  
C: content-type: audio/basic  
C: content-transfer-encoding: base64  
C: --qwertyuiopasdfghjklzxcvbnm  
C: .  
S: 250 message accepted  
C: QUIT  
S: 221 ee.uwa.edu.au closing connection
```

Sending a message from *alice@cs.washington.edu* to
bob@ee.uwa.edu.au.

Message Transfer (5)

Keyword	Description
AUTH	Client authentication
BINARYMIME	Server accepts binary messages
CHUNKING	Server accepts large messages in chunks
SIZE	Check message size before trying to send
STARTTLS	Switch to secure transport (TLS; see Chap. 8)
UTF8SMTP	Internationalized addresses

Some SMTP extensions.

Final Delivery (1)

Command	Description
CAPABILITY	List server capabilities
STARTTLS	Start secure transport (TLS; see Chap. 8)
LOGIN	Log on to server
AUTHENTICATE	Log on with other method
SELECT	Select a folder
EXAMINE	Select a read-only folder
CREATE	Create a folder
DELETE	Delete a folder
RENAME	Rename a folder
SUBSCRIBE	Add folder to active set
UNSUBSCRIBE	Remove folder from active set

...

IMAP (version 4) commands.

Final Delivery (2)

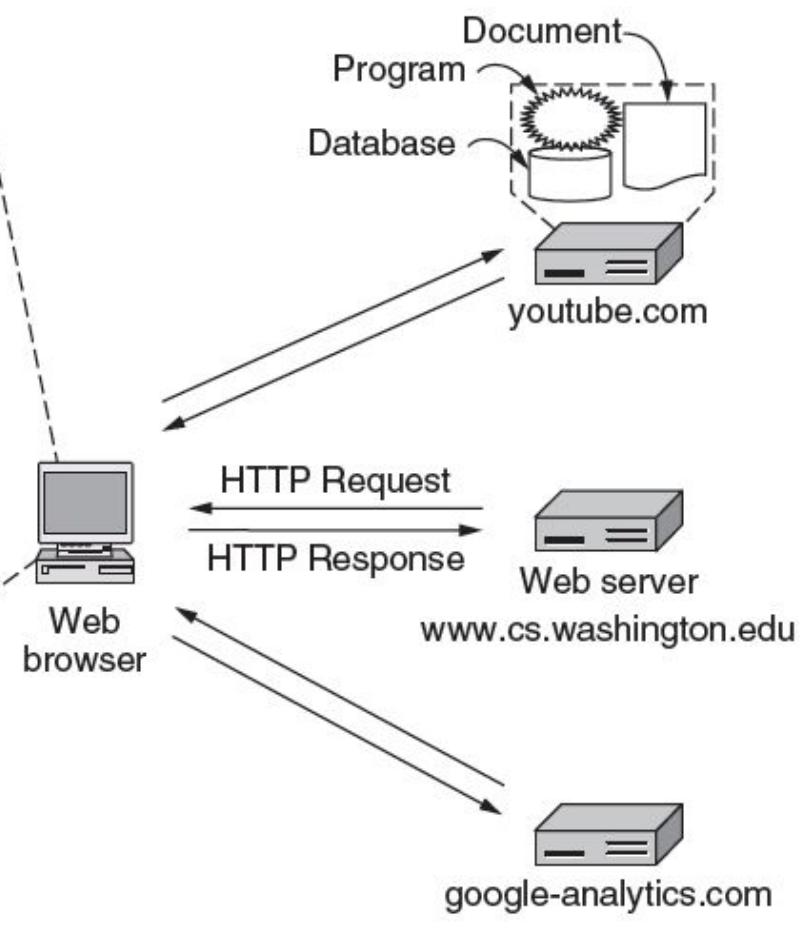
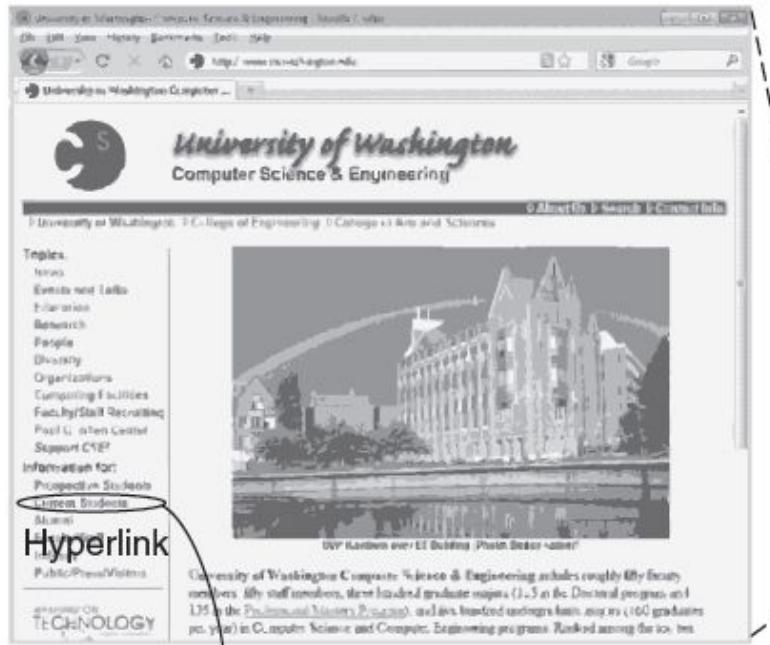
LIST	List the available folders
LSUB	List the active folders
STATUS	Get the status of a folder
APPEND	Add a message to a folder
CHECK	Get a checkpoint of a folder
FETCH	Get messages from a folder
SEARCH	Find messages in a folder
STORE	Alter message flags
COPY	Make a copy of a message in a folder
EXPUNGE	Remove messages flagged for deletion
UID	Issue commands using unique identifiers
NOOP	Do nothing
CLOSE	Remove flagged messages and close folder
LOGOUT	Log out and close connection

IMAP (version 4) commands.

The World Wide Web

- Architectural overview
- Static web pages
- Dynamic web pages, web applications
- The hypertext transfer protocol
- The mobile web
- Web search

Architectural Overview (1)



Architecture of the Web.

Architectural Overview (2)

Three questions had to be answered before a selected page could be displayed:

1. What is the page called?
2. Where is the page located?
3. How can the page be accessed?

Architectural Overview (3)

Steps that occur when link is selected:

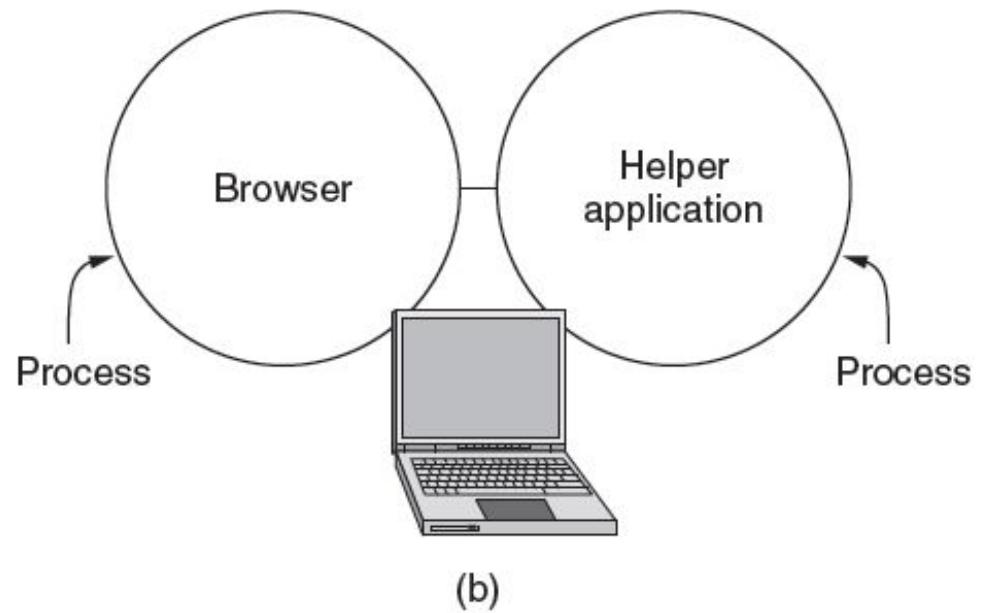
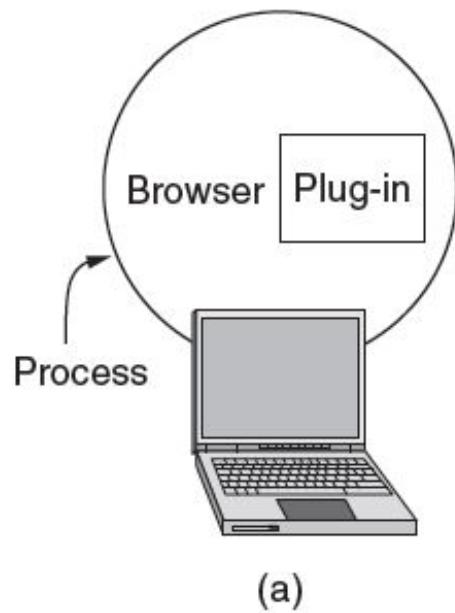
1. Browser determines the URL
2. Browser asks DNS for the IP address of the server
3. DNS replies
4. The browser makes a TCP connection
5. Sends HTTP request for the page
6. Server sends the page as HTTP response
7. Browser fetches other URLs as needed
8. The browser displays the page
9. The TCP connections are released

Architectural Overview (4)

Name	Used for	Example
http	Hypertext (HTML)	http://www.ee.uwa.edu/~rob/
https	Hypertext with security	https://www.bank.com/accounts/
ftp	FTP	ftp://ftp.cs.vu.nl/pub/minix/README
file	Local file	file:///usr/suzanne/prog.c
mailto	Sending email	mailto:JohnUser@acm.org
rtsp	Streaming media	rtsp://youtube.com/montypython.mpg
sip	Multimedia calls	sip:eve@adversary.com
about	Browser information	about:plugins

Some common URL schemes.

Architectural Overview (5)



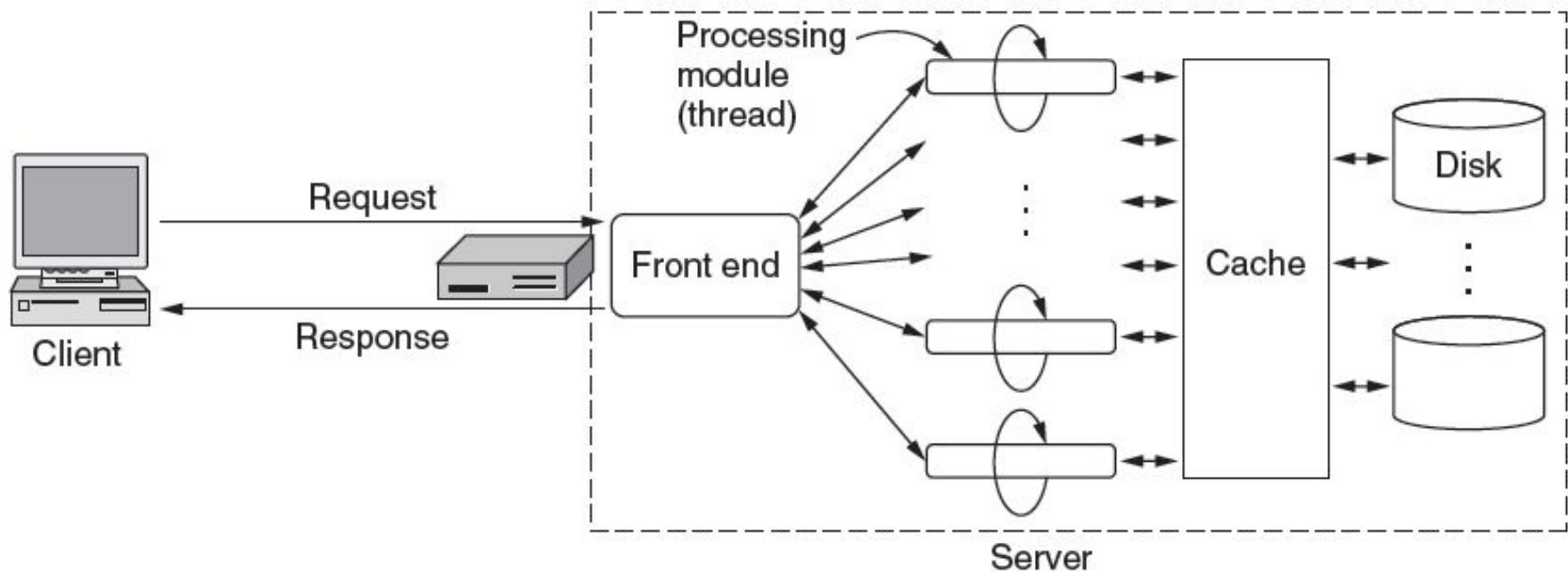
(a) A browser plug-in. (b) A helper application.

Architectural Overview (6)

Steps server performs in main loop

1. Accept a TCP connection from client
2. Get path to page, name of file requested.
3. Get the file (from disk).
4. Send contents of the file to the client.
5. Release the TCP connection.

Architectural Overview (7)



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

Architectural Overview (8)

A processing module performs a series of steps:

1. Resolve name of Web page requested.
2. Perform access control on the Web page.
3. Check the cache.
4. Fetch requested page from disk or run program
5. Determine the rest of the response
6. Return the response to the client.
7. Make an entry in the server log.

Architectural Overview (9)

Domain	Path	Content	Expires	Secure
toms-casino.com	/	CustomerID=297793521	15-10-10 17:00	Yes
jills-store.com	/	Cart=1-00501;1-07031;2-13721	11-1-11 14:22	No
aportal.com	/	Prefs=Stk:CSCO+ORCL;Spt:Jets	31-12-20 23:59	No
sneaky.com	/	UserID=4627239101	31-12-19 23:59	No

Some examples of cookies

Static Web Pages (1)

```
<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 email. </p>
<hr>
<h2> Product information </h2>
<ul>
  <li> <a href="http://widget.com/products/big"> Big widgets </a> </li>
  <li> <a href="http://widget.com/products/little"> Little widgets </a> </li>
</ul>
<h2> Contact information </h2>
<ul>
  <li> By telephone: 1-800-WIDGETS </li>
  <li> By email: info@amalgamated-widget.com </li>
</ul>
</body>
</html>
```

The HTML for a sample Web page.

Static Web Pages (2)

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 email.

Product Information

- [Big widgets](#)
- [Little widgets](#)

Contact information

- By telephone: 1-800-WIDGETS
- By email: info@amalgamated-widget.com

The formatted page.

Static Web Pages (3)

Item	HTML 1.0	HTML 2.0	HTML 3.0	HTML 4.0	HTML 5.0
Hyperlinks	x	x	x	x	x
Images	x	x	x	x	x
Lists	x	x	x	x	x
Active maps & images		x	x	x	x
Forms		x	x	x	x
Equations			x	x	x
Toolbars			x	x	x
Tables			x	x	x
Accessibility features				x	x
Object embedding				x	x
Style sheets				x	x
Scripting				x	x
Video and audio					x
Inline vector graphics					x
XML representation					x
Background threads					x
Browser storage					x
Drawing canvas					x

Some differences between HTML versions.

Static Web Pages (4)

```
<html>
<head> <title> AWI CUSTOMER ORDERING FORM </title> </head>
<body>
<h1> Widget Order Form </h1>
<form ACTION="http://widget.com/cgi-bin/order.cgi" method=POST>
<p> Name <input name="customer" size=46> </p>
<p> Street address <input name="address" size=40> </p>
<p> City <input name="city" size=20> State <input name="state" size =4>
Country <input name="country" size=10> </p>
<p> Credit card # <input name="cardno" size=10>
Expires <input name="expires" size=4>
M/C <input name="cc" type=radio value="mastercard">
VISA <input name="cc" type=radio value="visacard"> </p>
<p> Widget size Big <input name="product" type=radio value="expensive">
Little <input name="product" type=radio value="cheap">
Ship by express courier <input name="express" type=checkbox> </p>
<p><input type=submit value="Submit order"> </p>
Thank you for ordering an AWI widget, the best widget money can buy!
</form>
</body>
</html>
```

The HTML for an order form.

Static Web Pages (5)

Widget Order Form

Name

Street address

City State Country

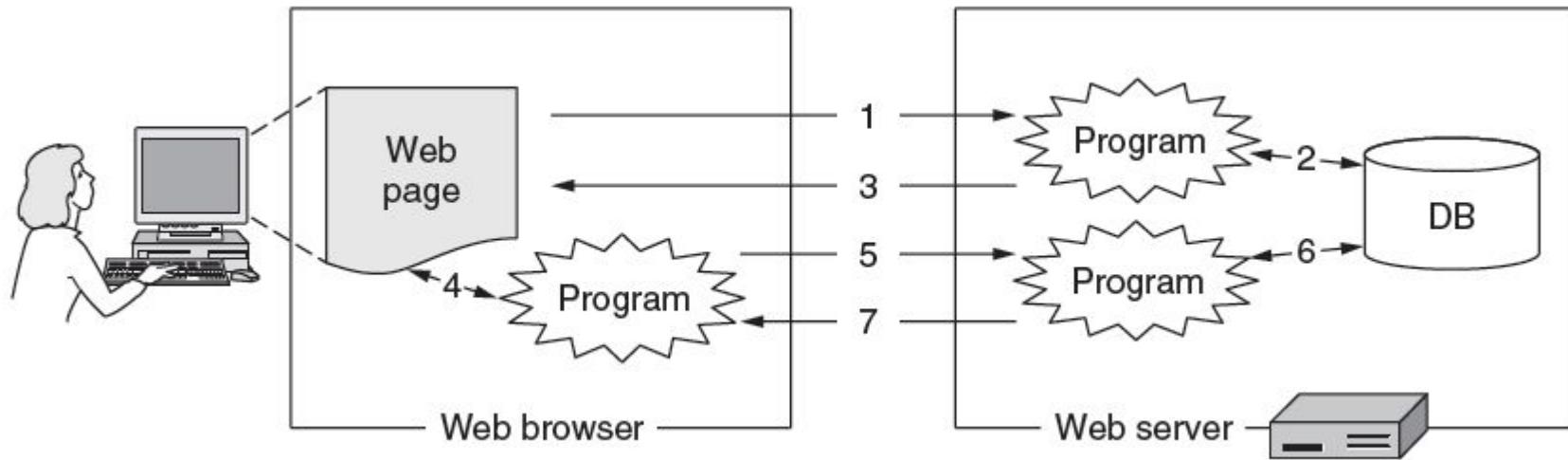
Credit card # Expires M/C Visa

Widget size Big Little Ship by express courier

Thank you for ordering an AWI widget, the best widget money can buy!

The formatted page.

Dynamic Web Pages, Web Applications (1)



Dynamic pages

Dynamic Web Pages, Web Applications (2)

```
<html>
<body>
<form action="action.php" method="post">
<p> Please enter your name: <input type="text" name="name"> </p>
<p> Please enter your age: <input type="text" name="age"> </p>
<input type="submit">
</form>
</body>      (a)
</html>
```

```
<html>
<body>
<h1> Reply: </h1>
Hello <?php echo $name; ?>.
Prediction: next year you will be <?php echo $age + 1; ?>
</body>
</html>      (b)
```

```
<html>
<body>
<h1> Reply: </h1>
Hello Barbara.
Prediction: next year you will be 33
</body>
</html>      (c)
```

- (a) A Web page containing a form. (b) A PHP script for handling the output of the form. (c) Output from the PHP script when the inputs are “Barbara” and “32”, respectively.

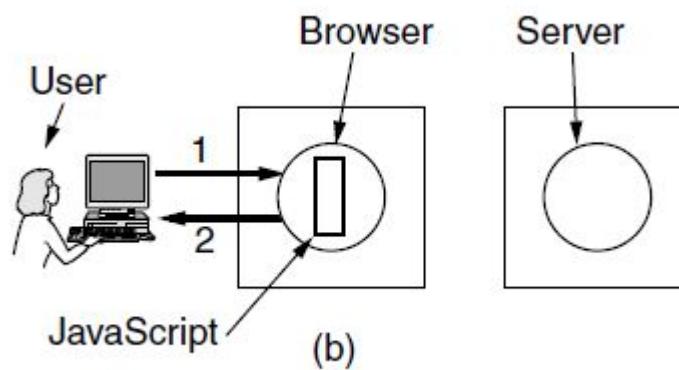
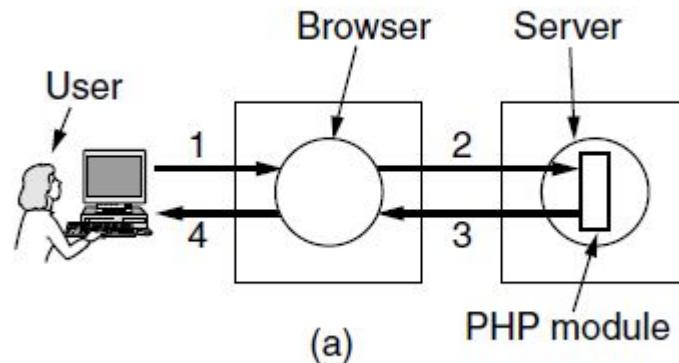
Dynamic Web Pages, Web Applications (3)

```
<html>
<head>
<script language="javascript" type="text/javascript">
function response(test_form) {
    var person = test_form.name.value;
    var years = eval(test_form.age.value) + 1;
    document.open();
    document.writeln("<html> <body>");
    document.writeln("Hello " + person + ".<br>");
    document.writeln("Prediction: next year you will be " + years + ".");
    document.writeln("</body> </html>");
    document.close();
}
</script>
</head>

<body>
<form>
Please enter your name: <input type="text" name="name">
<p>
Please enter your age: <input type="text" name="age">
<p>
<input type="button" value="submit" onclick="response(this.form)">
</form>
</body>
</html>
```

Use of JavaScript for processing a form.

Dynamic Web Pages, Web Applications (4)



(a) Server-side scripting with PHP.

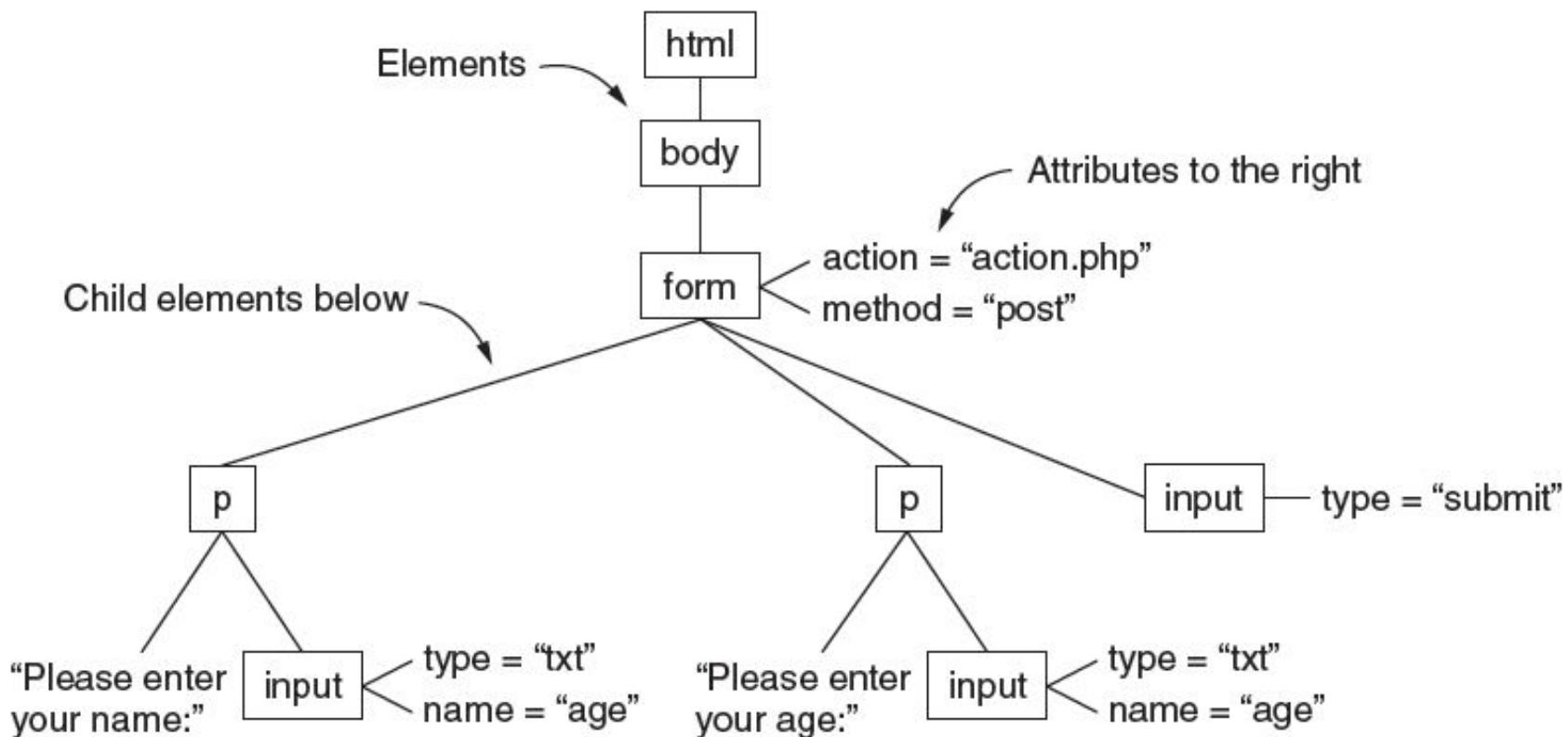
(b) Client-side scripting with JavaScript.

Dynamic Web Pages, Web Applications (5)

AJAX Technologies

1. HTML and CSS: present information as pages.
2. DOM: change parts of pages while they are viewed.
3. XML: let programs exchange data with the server.
4. An asynchronous way to send and retrieve XML data.
5. JavaScript as a language to bind all this together.

Dynamic Web Pages, Web Applications (6)



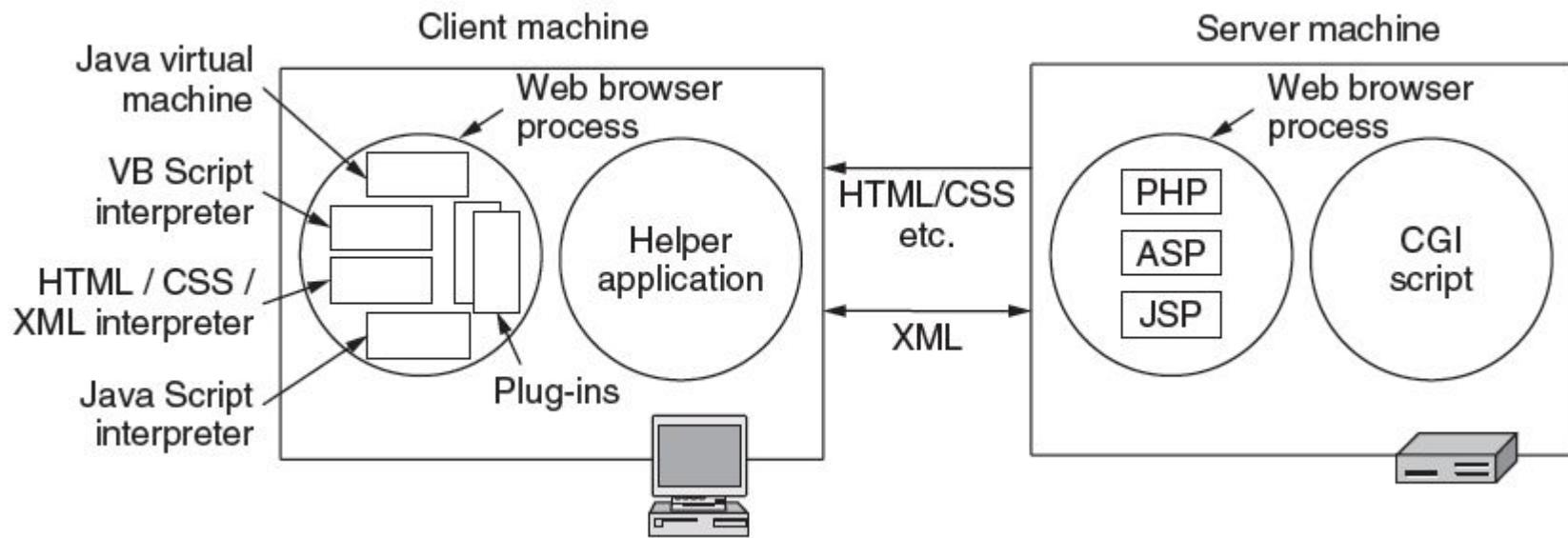
The DOM tree for the HTML in Fig. 7-30(a).

Dynamic Web Pages, Web Applications (7)

```
<?xml version="1.0" ?>
<book_list>
  <book>
    <title> Human Behavior and the Principle of Least Effort </title>
    <author> George Zipf </author>
    <year> 1949 </year>
  </book>
  <book>
    <title> The Mathematical Theory of Communication </title>
    <author> Claude E. Shannon </author>
    <author> Warren Weaver </author>
    <year> 1949 </year>
  </book>
  <book>
    <title> Nineteen Eighty-Four </title>
    <author> George Orwell </author>
    <year> 1949 </year>
  </book>
</book_list>
```

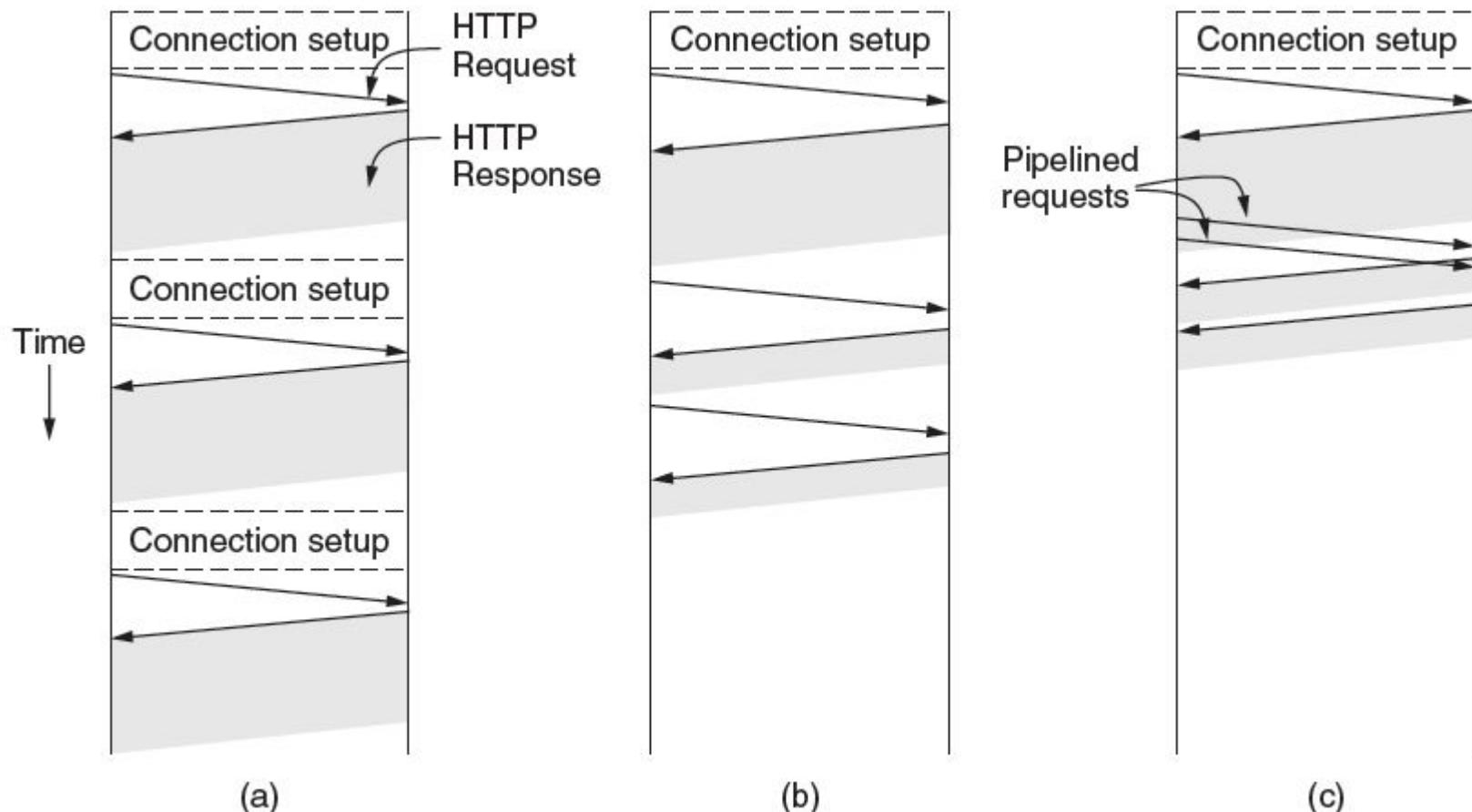
A simple XML document.

Dynamic Web Pages, Web Applications (8)



Various technologies used to generate dynamic pages.

The HyperText Transfer Protocol (1)



- HTTP with (a) multiple connections and sequential requests.
(b) A persistent connection and sequential requests.
(c) A persistent connection and pipelined requests.

The HyperText Transfer Protocol (2)

Method	Description
GET	Read a Web page
HEAD	Read a Web page's header
POST	Append to a Web page
PUT	Store a Web page
DELETE	Remove the Web page
TRACE	Echo the incoming request
CONNECT	Connect through a proxy
OPTIONS	Query options for a page

The built-in HTTP request methods.

The HyperText Transfer Protocol (3)

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

The HyperText Transfer Protocol (4)

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
If-Modified-Since	Request	Time and date to check freshness
If-None-Match	Request	Previously sent tags to check freshness
Host	Request	The server's DNS name
Authorization	Request	A list of the client's credentials
Referer	Request	The previous URL from which the request came
Cookie	Request	Previously set cookie sent back to the server
Set-Cookie	Response	Cookie for the client to store
Server	Response	Information about the server

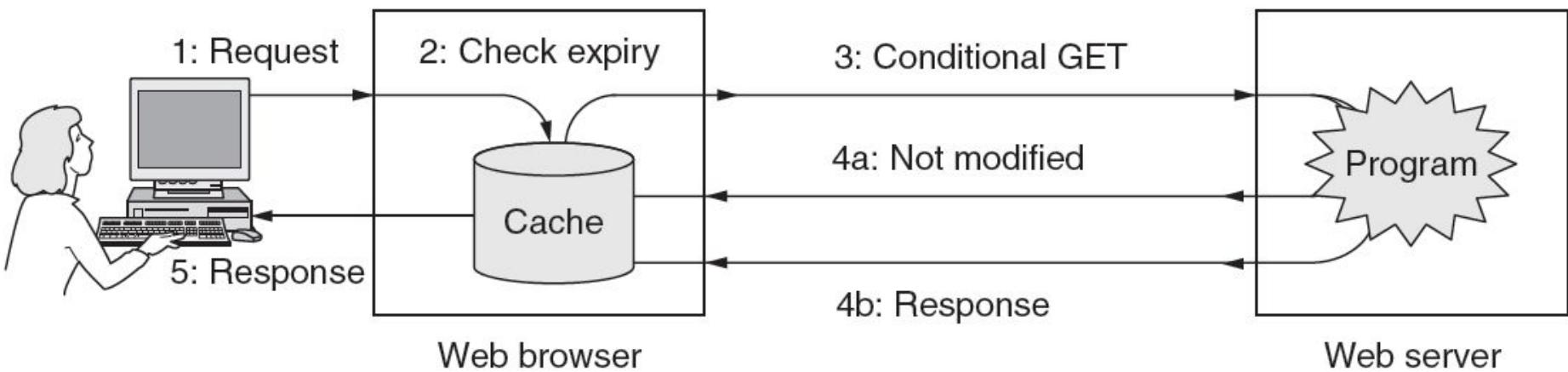
Some HTTP message headers.

The HyperText Transfer Protocol (5)

Content-Encoding	Response	How the content is encoded (e.g., <i>gzip</i>)
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
Content-Range	Response	Identifies a portion of the page's content
Last-Modified	Response	Time and date the page was last changed
Expires	Response	Time and date when the page stops being valid
Location	Response	Tells the client where to send its request
Accept-Ranges	Response	Indicates the server will accept byte range requests
Date	Both	Date and time the message was sent
Range	Both	Identifies a portion of a page
Cache-Control	Both	Directives for how to treat caches
ETag	Both	Tag for the contents of the page
Upgrade	Both	The protocol the sender wants to switch to

Some HTTP message headers.

The HyperText Transfer Protocol (6)



HTTP caching.

The Mobile Web (1)

Difficulties for mobile phones browsing the web

1. Relatively small screens
2. Limited input capabilities, lengthy input.
3. Network bandwidth is limited
4. Connectivity may be intermittent.
5. Computing power is limited

The Mobile Web (2)

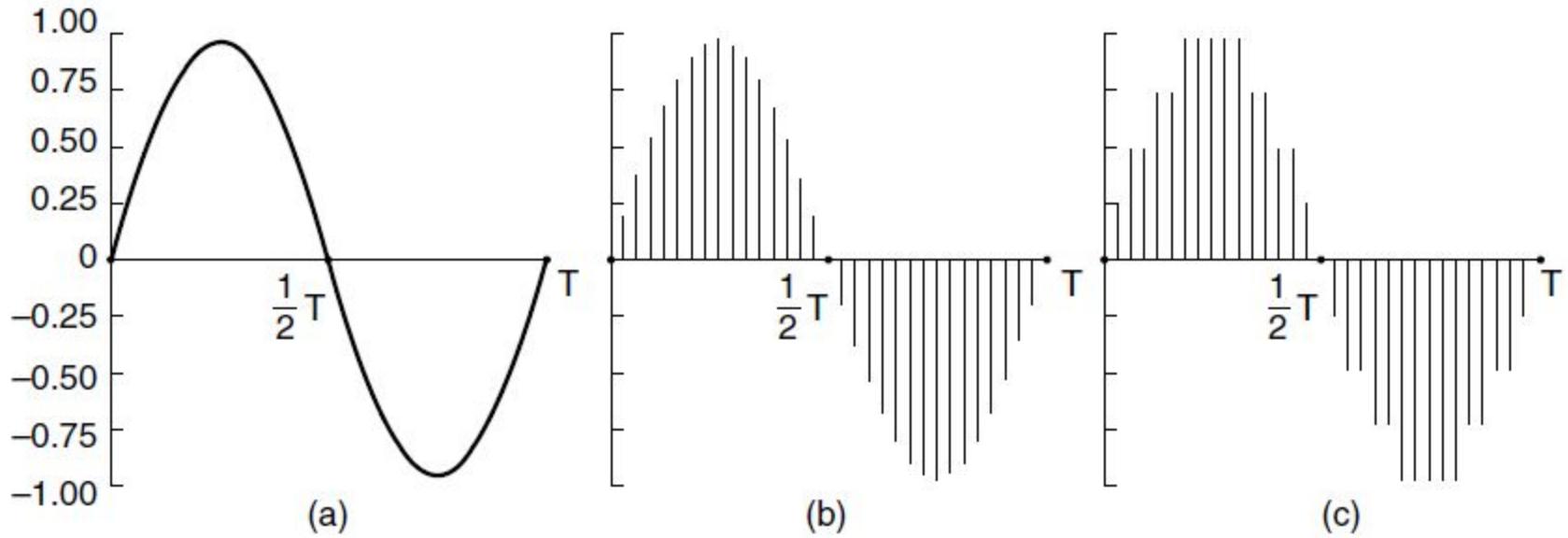
Module	Req.?	Function	Example tags
Structure	Yes	Doc. structure	body, head, html, title
Text	Yes	Information	br, code, dfn, em, hn, kbd, p, strong
Hypertext	Yes	Hyperlinks	a
List	Yes	Itemized lists	dl, dt, dd, ol, ul, li
Forms	No	Fill-in forms	form, input, label, option, textarea
Tables	No	Rectangular tables	caption, table, td, th, tr
Image	No	Pictures	img
Object	No	Applets, maps, etc.	object, param
Meta-information	No	Extra info	meta
Link	No	Similar to <a>	link
Base	No	URL starting point	base

The XHTML Basic modules and tags.

Streaming Audio and Video

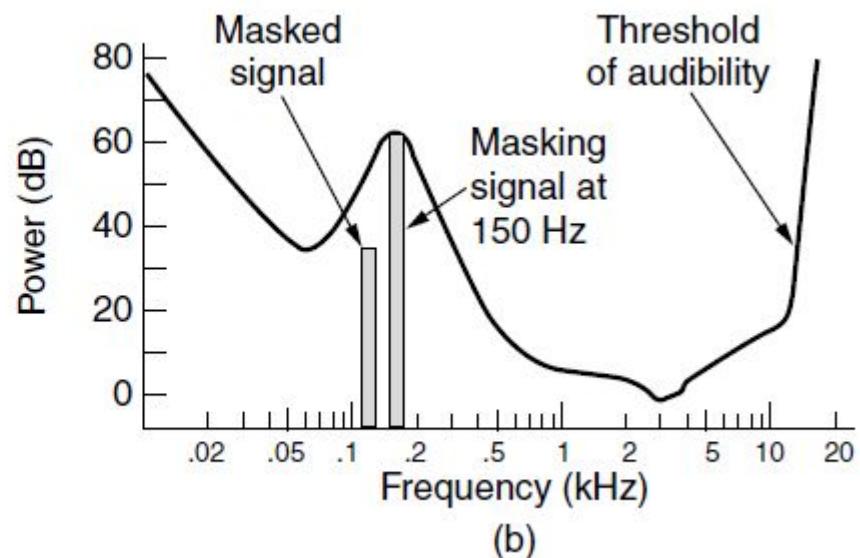
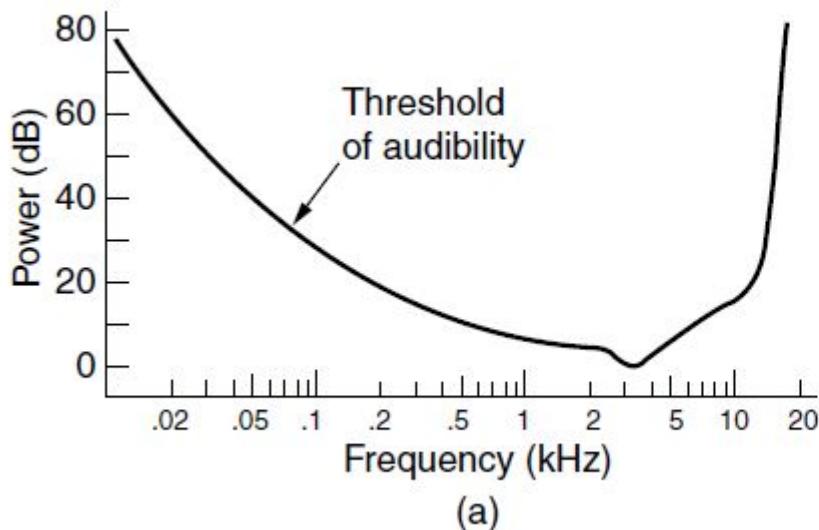
- Digital audio
- Digital video
- Streaming stored media
- Streaming live media
- Real-time conferencing

Digital Audio (1)



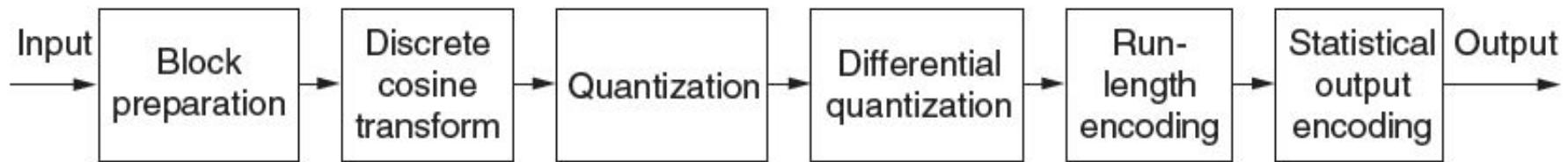
- (a) A sine wave. (b) Sampling the sine wave.
(c) Quantizing the samples to 4 bits.

Digital Audio (2)



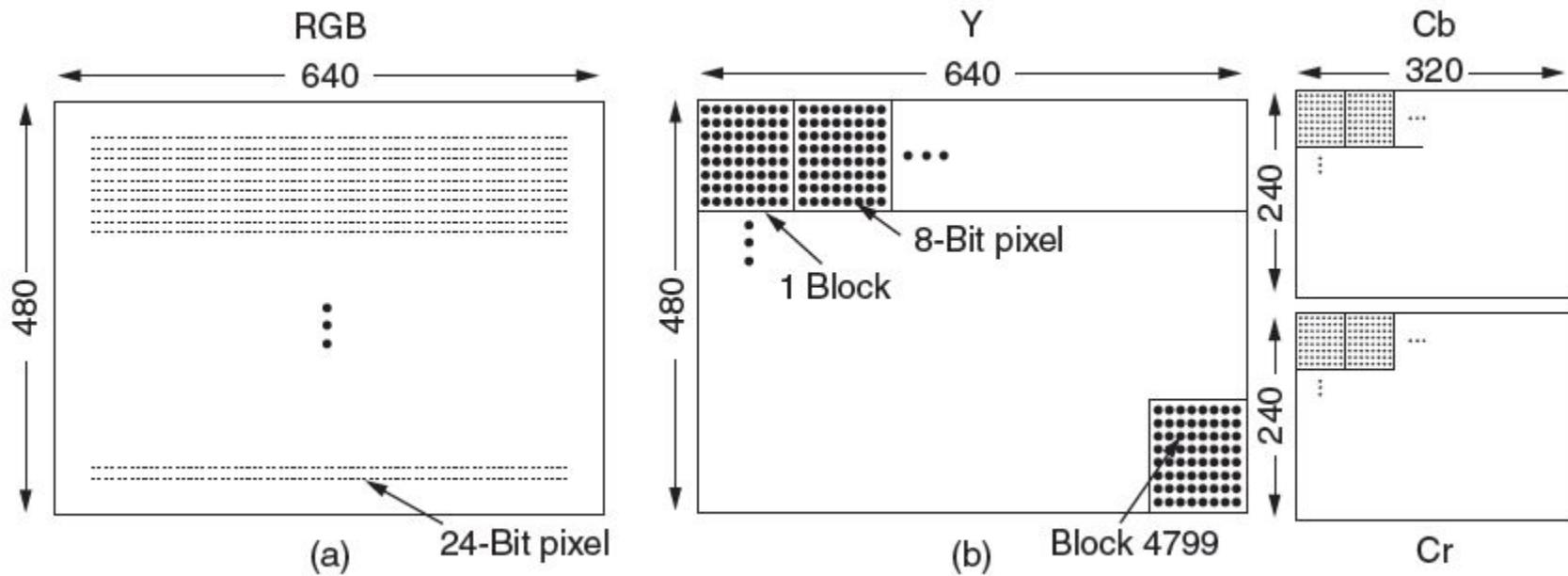
- (a) The threshold of audibility as a function of frequency.
(b) The masking effect.

Digital Video (1)



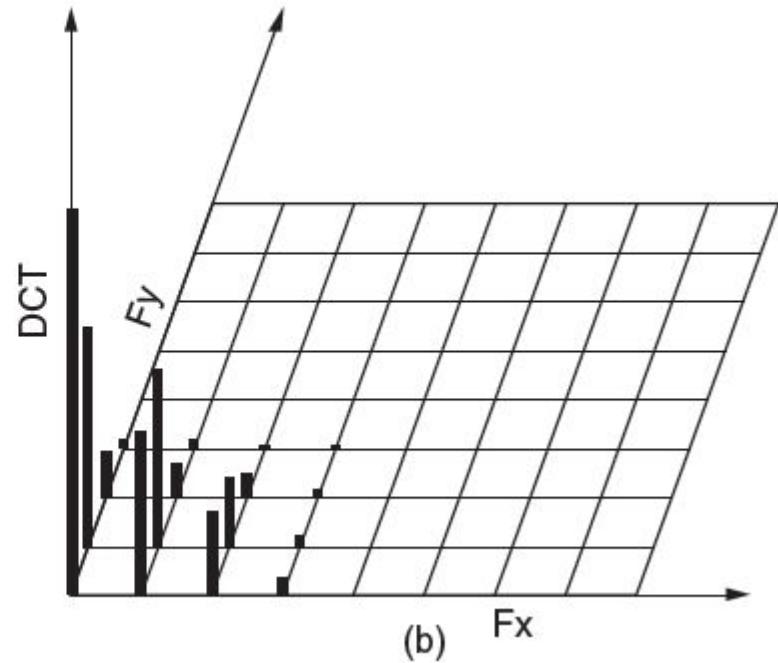
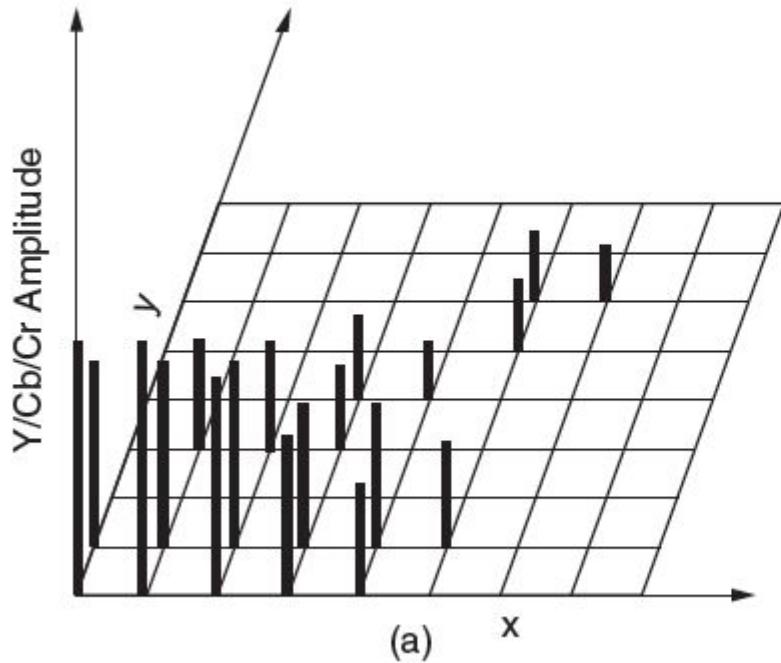
Steps in JPEG lossy sequential encoding.

Digital Video (2)



(a) RGB input data. (b) After block preparation.

Digital Video (3)



(a) One block of the Y matrix. (b) The DCT coefficients.

Digital Video (4)

DCT coefficients

150	80	40	14	4	2	1	0
92	75	36	10	6	1	0	0
52	38	26	8	7	4	0	0
12	8	6	4	2	1	0	0
4	3	2	0	0	0	0	0
2	2	1	1	0	0	0	0
1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Quantization table

1	1	2	4	8	16	32	64
1	1	2	4	8	16	32	64
2	2	2	4	8	16	32	64
4	4	4	4	8	16	32	64
8	8	8	8	8	16	32	64
16	16	16	16	16	16	32	64
32	32	32	32	32	32	32	64
64	64	64	64	64	64	64	64

Quantized coefficients

150	80	20	4	1	0	0	0
92	75	18	3	1	0	0	0
26	19	13	2	1	0	0	0
3	2	2	1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Computation of the quantized DCT coefficients.

Digital Video (5)

150	80	20	4	1	0	0	0
92	75	18	3	1	0	0	0
26	19	13	2	1	0	0	0
3	2	2	1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

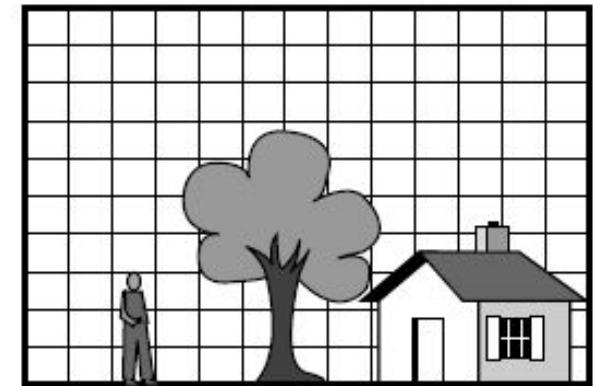
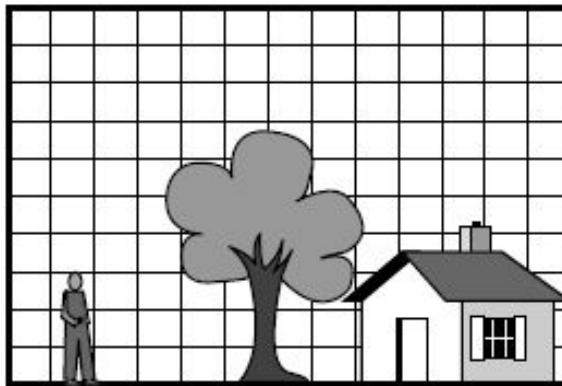
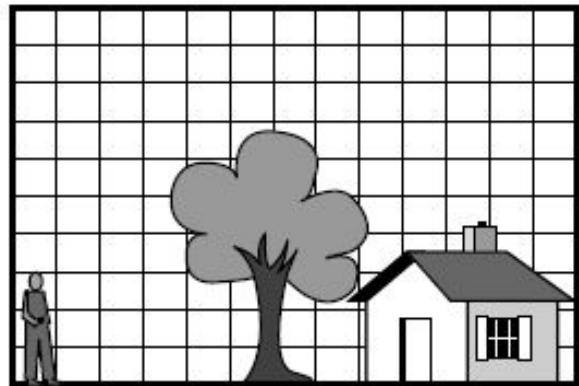
The order in which the quantized values are transmitted.

Digital Video (6)

MPEG output consists of three kinds of frames:

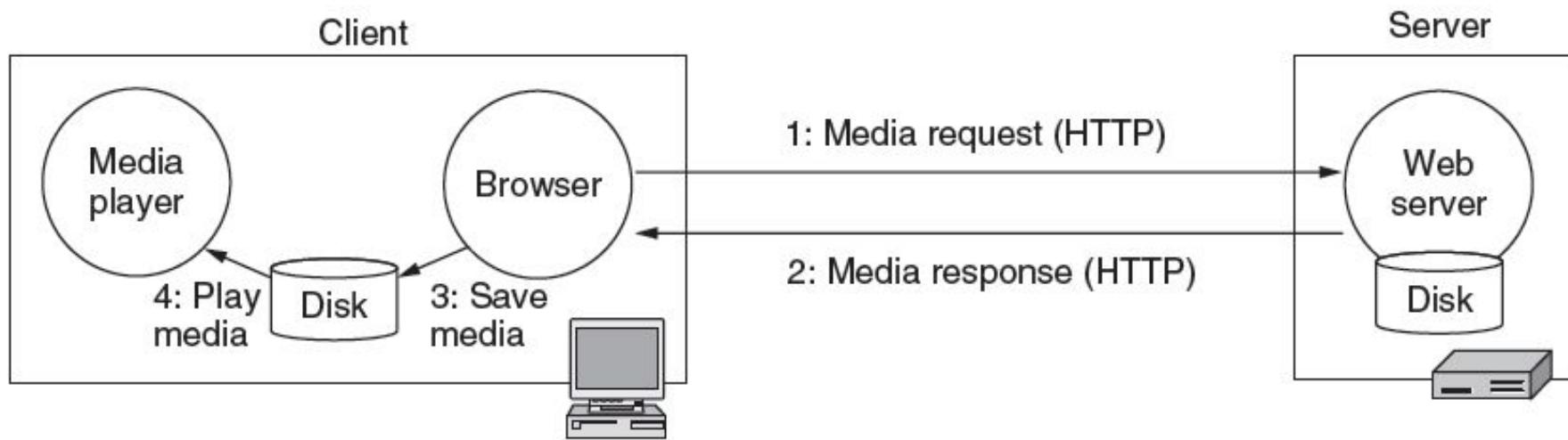
- a) I- (Intracoded) : Self-contained compressed still pictures.
- b) P- (Predictive) : Block-by-block difference with previous frames.
- c) B- (Bidirectional) : block-by-block differences between previous and future frames.

Digital Video (7)



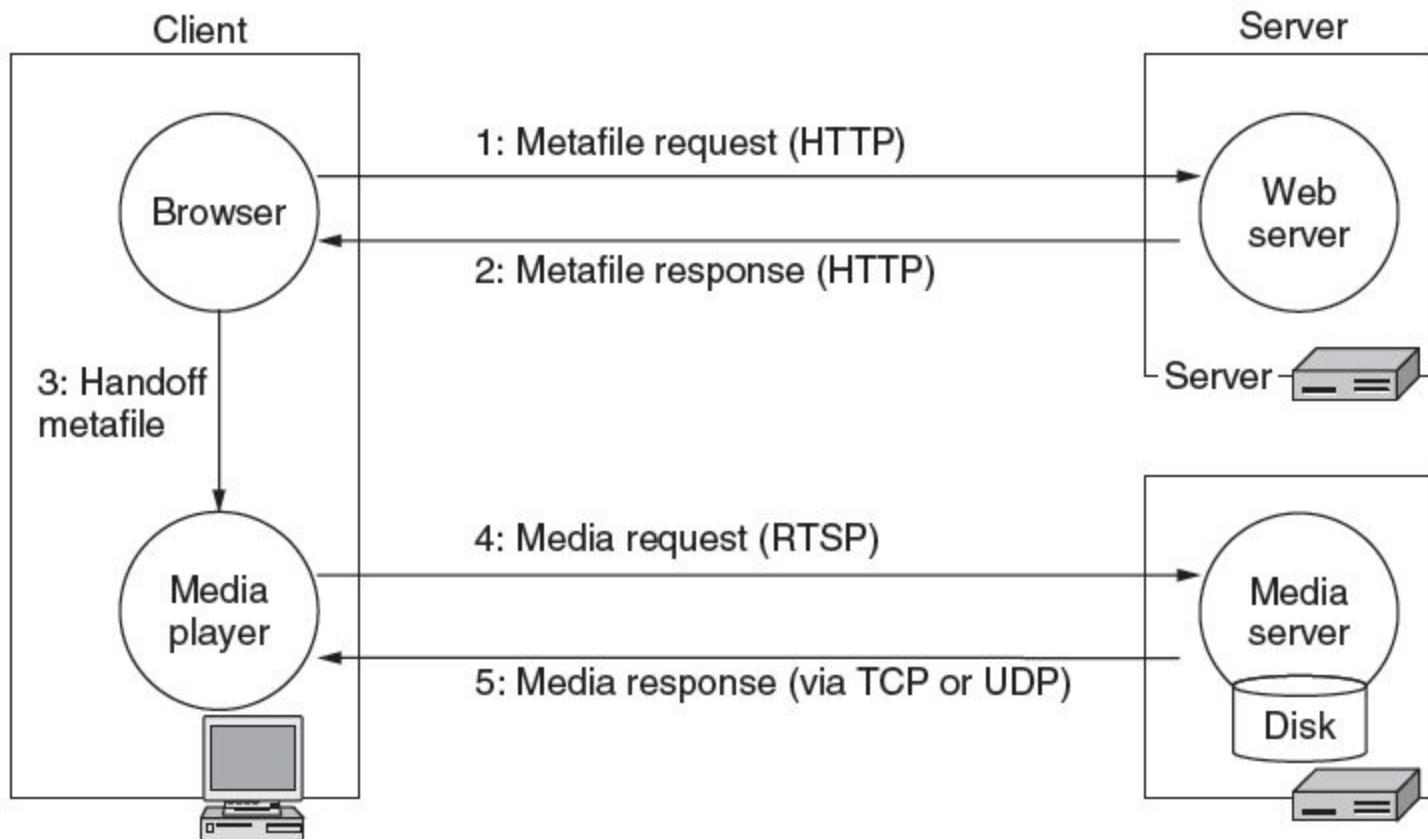
Three consecutive frames

Streaming Stored Media (1)



Playing media over the Web via simple downloads.

Streaming Stored Media (2)



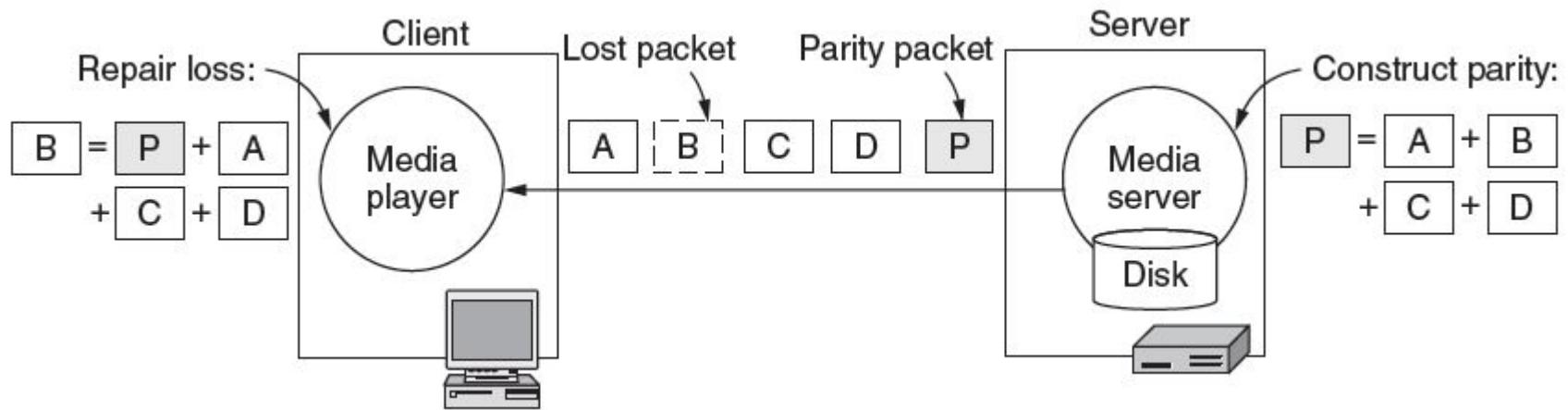
Streaming media using the Web and a media server.

Streaming Stored Media (3)

Major tasks of the media player:

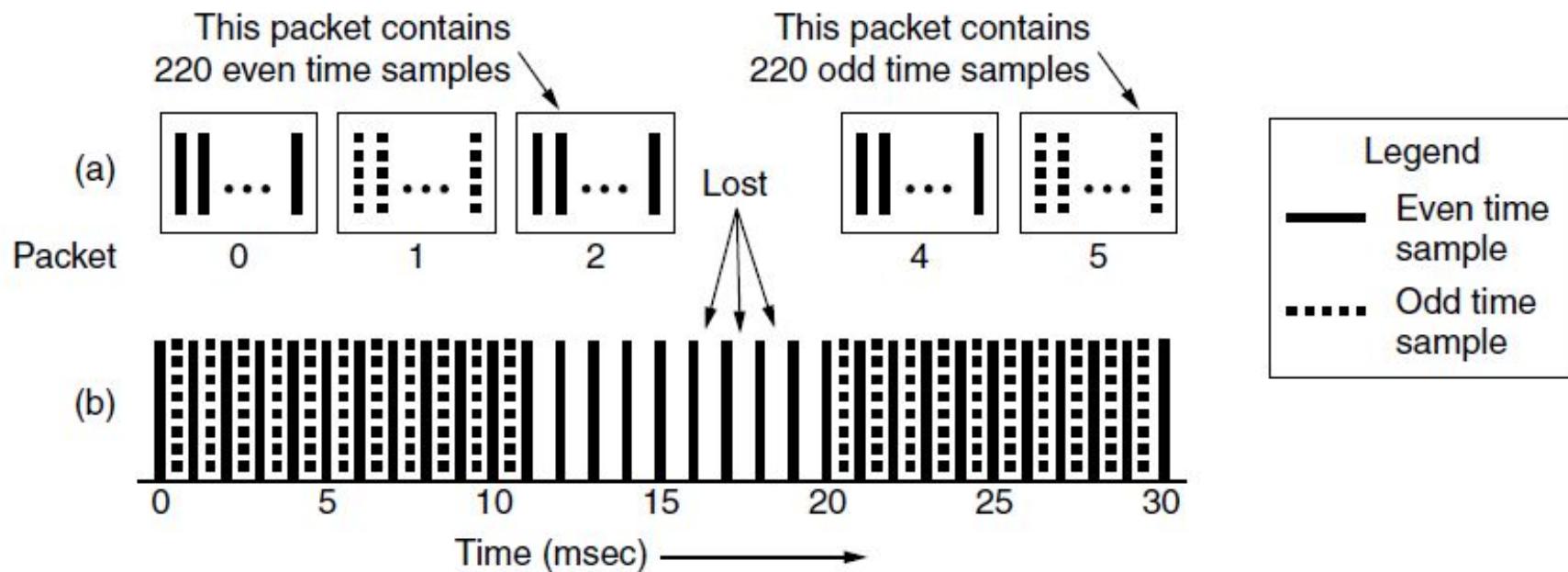
1. Manage the user interface.
2. Handle transmission errors.
3. Decompress the content.
4. Eliminate jitter.

Streaming Stored Media (4)



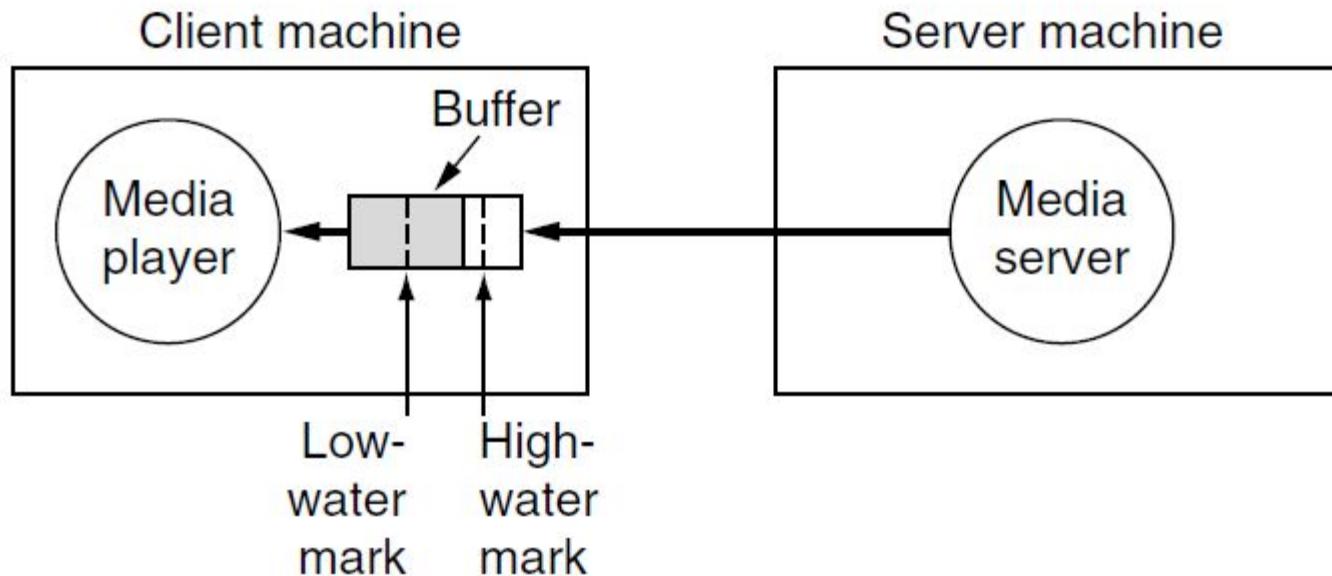
Using a parity packet to repair loss.

Streaming Stored Media (5)



When packets carry alternate samples, the loss of a packet reduces the temporal resolution rather than creating a gap in time.

Streaming Stored Media (6)



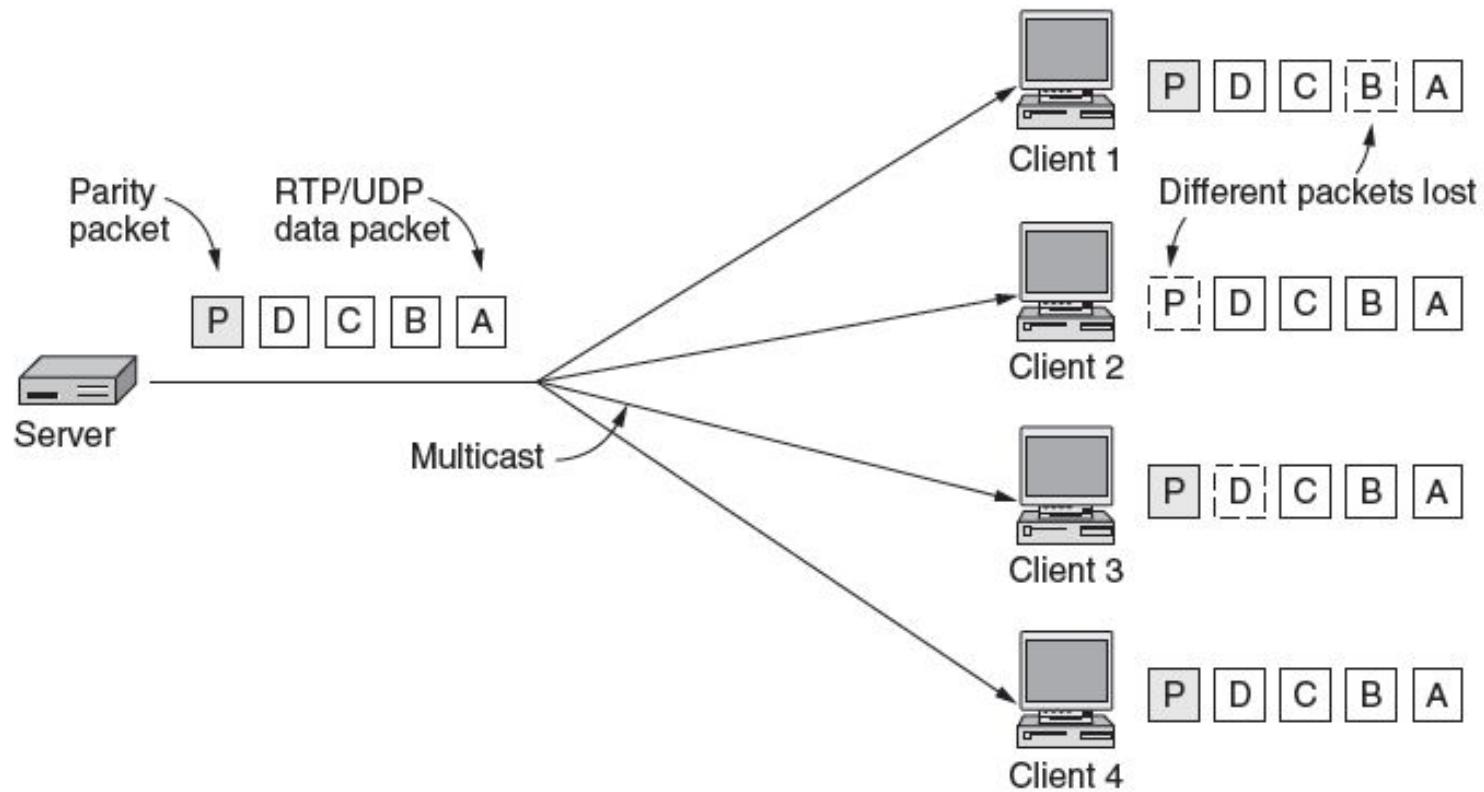
The media player buffers input from the media server and plays from the buffer rather than directly from the network.

Streaming Stored Media (7)

Command	Server action
DESCRIBE	List media parameters
SETUP	Establish a logical channel between the player and the server
PLAY	Start sending data to the client
RECORD	Start accepting data from the client
PAUSE	Temporarily stop sending data
TEARDOWN	Release the logical channel

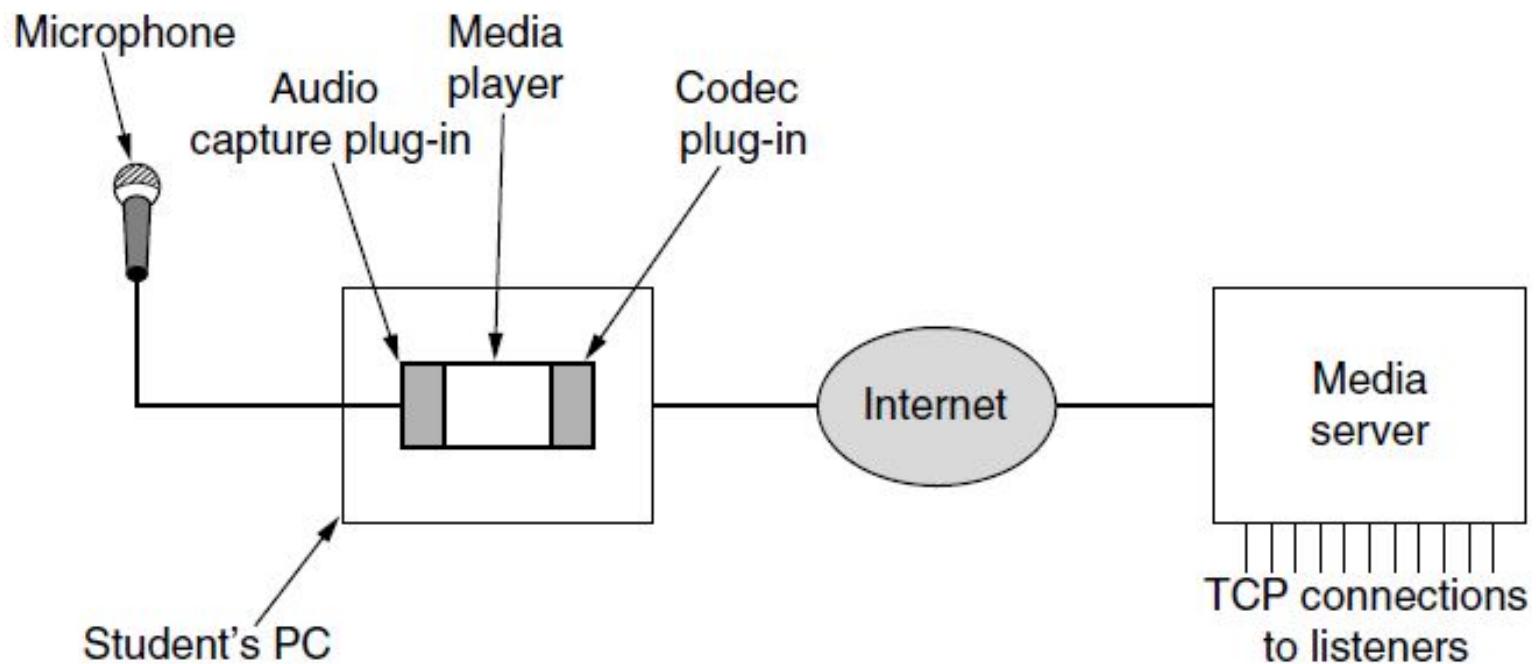
RTSP commands from the player to the server.

Streaming Live Media (1)



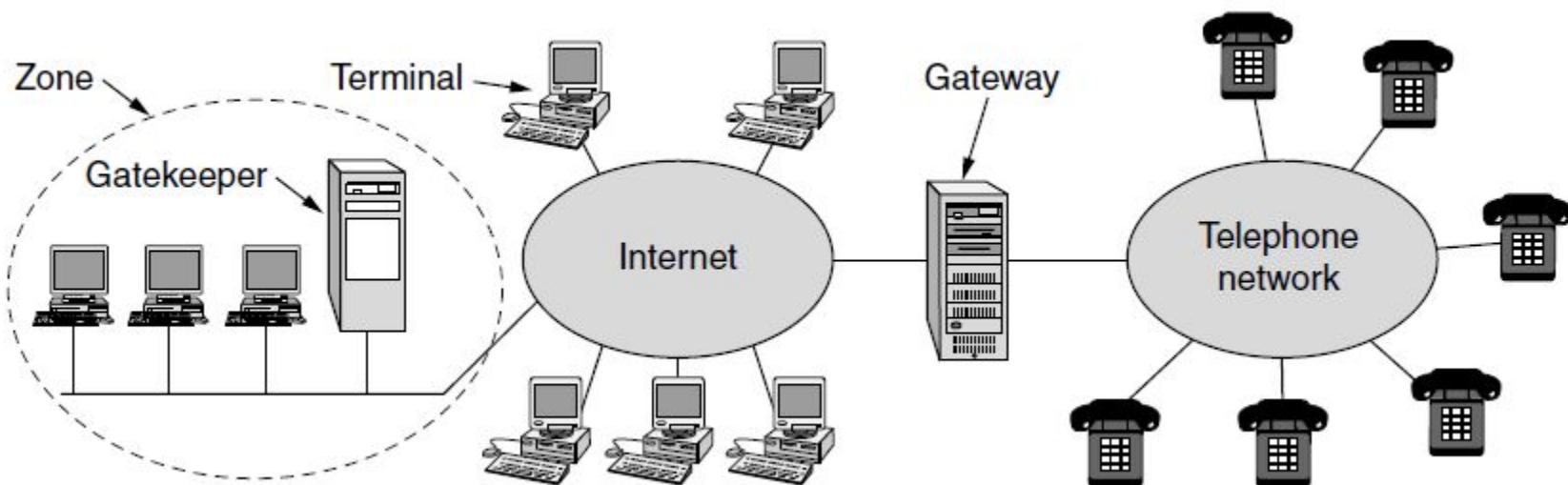
Multicast streaming media with a parity packet.

Streaming Live Media (2)



A student radio station.

Real-Time Conferencing (1)



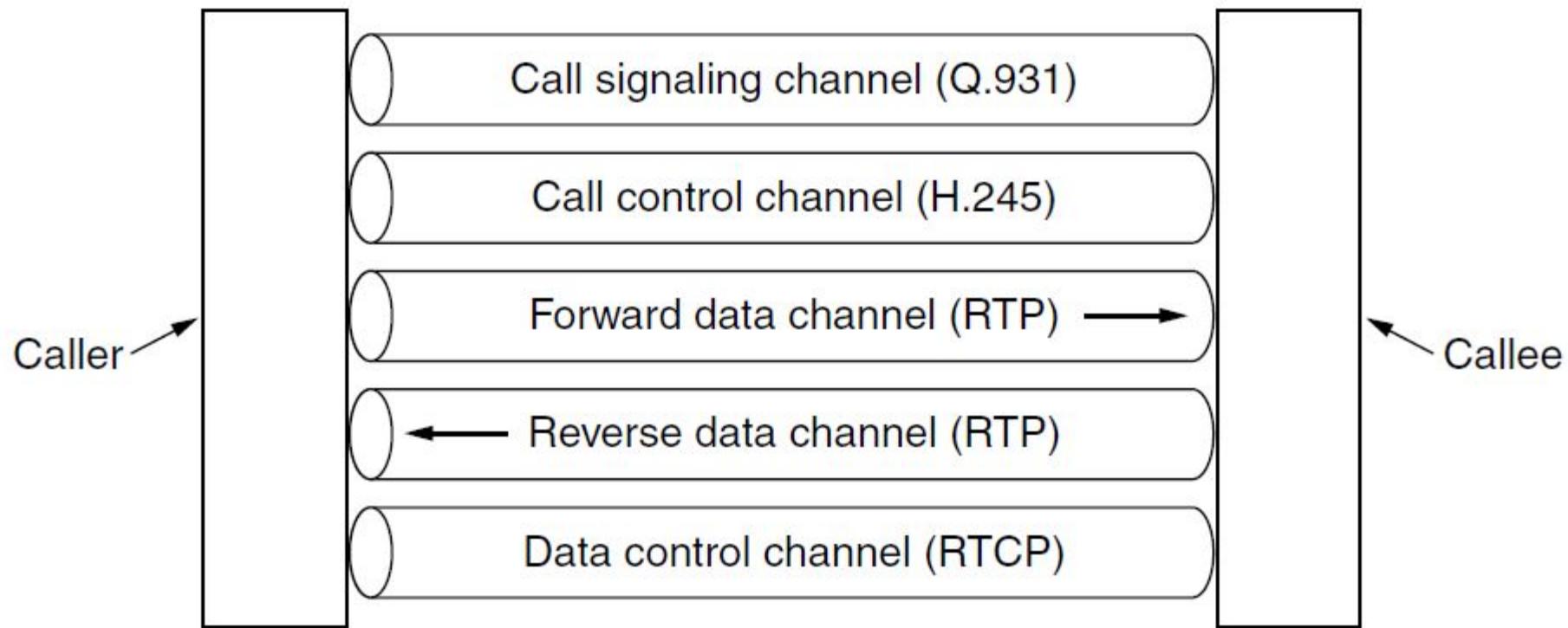
The H.323 architectural model for Internet telephony.

Real-Time Conferencing (2)

Audio	Video	Control					
G.7xx	H.26x	RTCP	H.225 (RAS)	Q.931 (Signaling)	H.245 (Call Control)		
RTP		UDP					
TCP							
IP							
Link layer protocol							
Physical layer protocol							

The H.323 protocol stack.

Real-Time Conferencing (3)



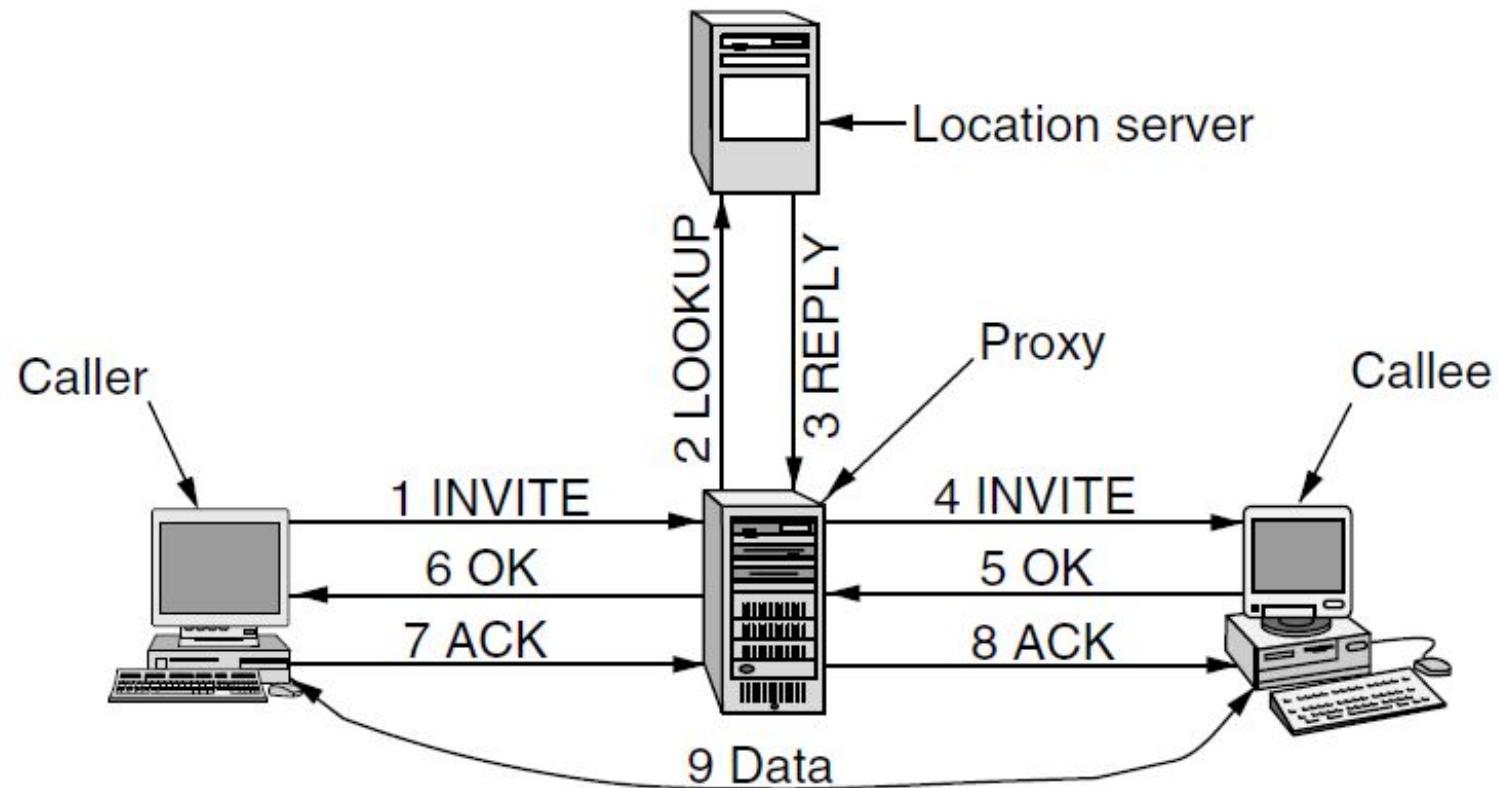
Logical channels between the caller and callee during a call.

Real-Time Conferencing (4)

Method	Description
INVITE	Request initiation of a session
ACK	Confirm that a session has been initiated
BYE	Request termination of a session
OPTIONS	Query a host about its capabilities
CANCEL	Cancel a pending request
REGISTER	Inform a redirection server about the user's current location

SIP methods.

Real-Time Conferencing (5)



Use of a proxy server and redirection with SIP.

Real-Time Conferencing (6)

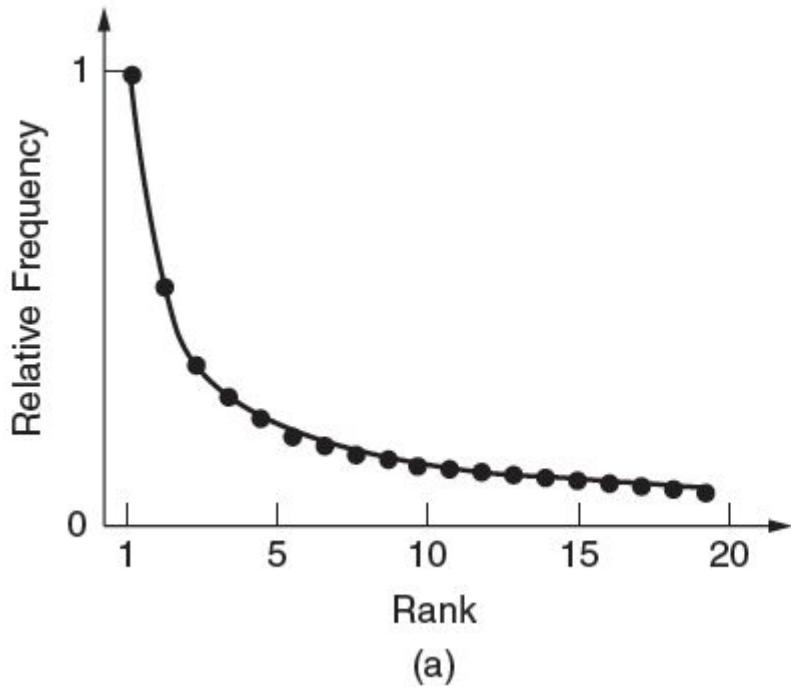
Item	H.323	SIP
Designed by	ITU	IETF
Compatibility with PSTN	Yes	Largely
Compatibility with Internet	Yes, over time	Yes
Architecture	Monolithic	Modular
Completeness	Full protocol stack	SIP just handles setup
Parameter negotiation	Yes	Yes
Call signaling	Q.931 over TCP	SIP over TCP or UDP
Message format	Binary	ASCII
Media transport	RTP/RTCP	RTP/RTCP
Multiparty calls	Yes	Yes
Multimedia conferences	Yes	No
Addressing	URL or phone number	URL
Call termination	Explicit or TCP release	Explicit or timeout
Instant messaging	No	Yes
Encryption	Yes	Yes
Size of standards	1400 pages	250 pages
Implementation	Large and complex	Moderate, but issues
Status	Widespread, esp. video	Alternative, esp. voice

Comparison of H.323 and SIP.

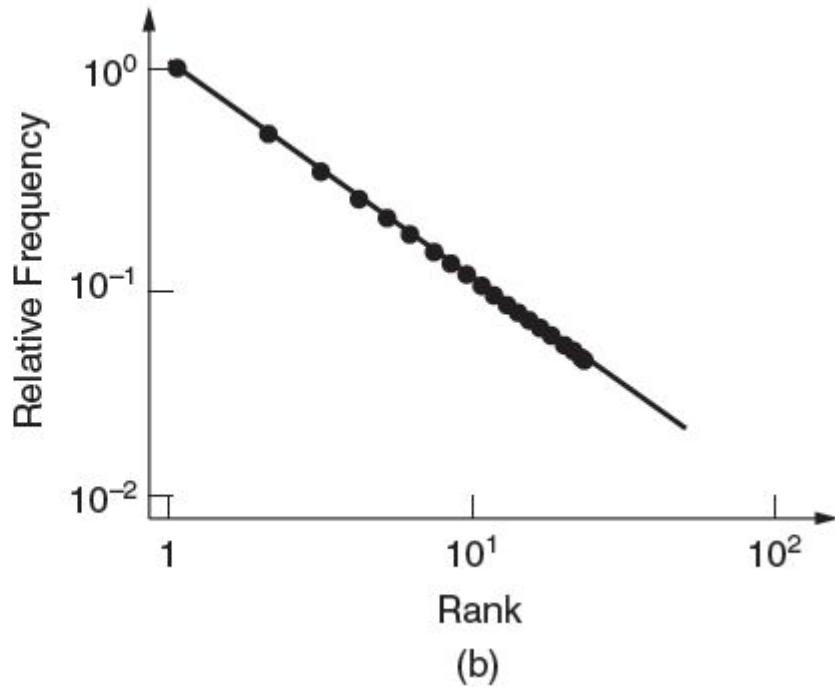
Content Delivery

- Content and internet traffic
- Server farms and web proxies
- Content delivery networks
- Peer-to-peer networks

Content and Internet Traffic



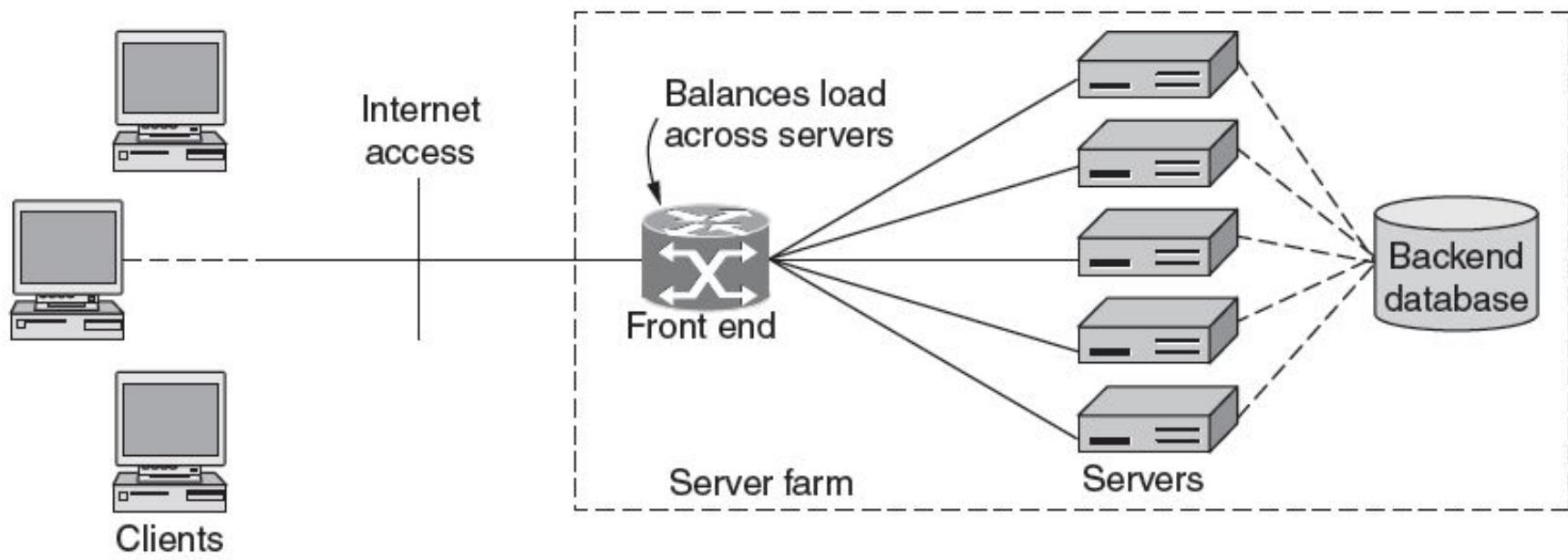
(a)



(b)

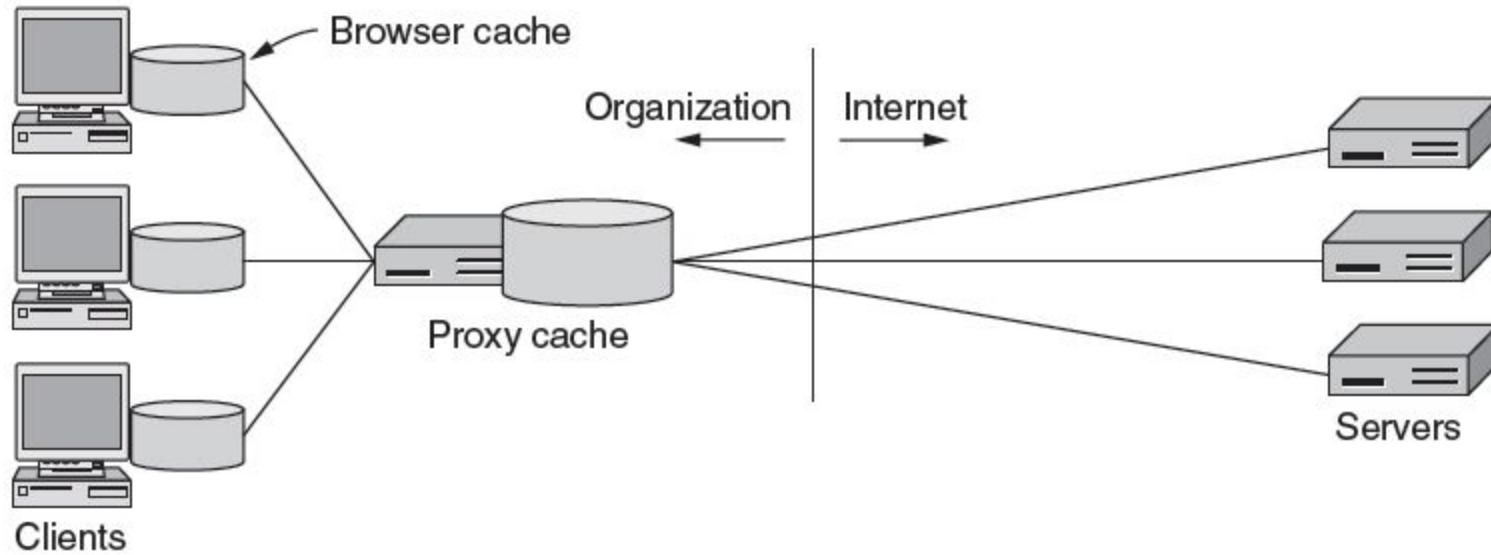
Zipf distribution (a) On a linear scale. (b) On a log-log scale.

Server Farms and Web Proxies (1)



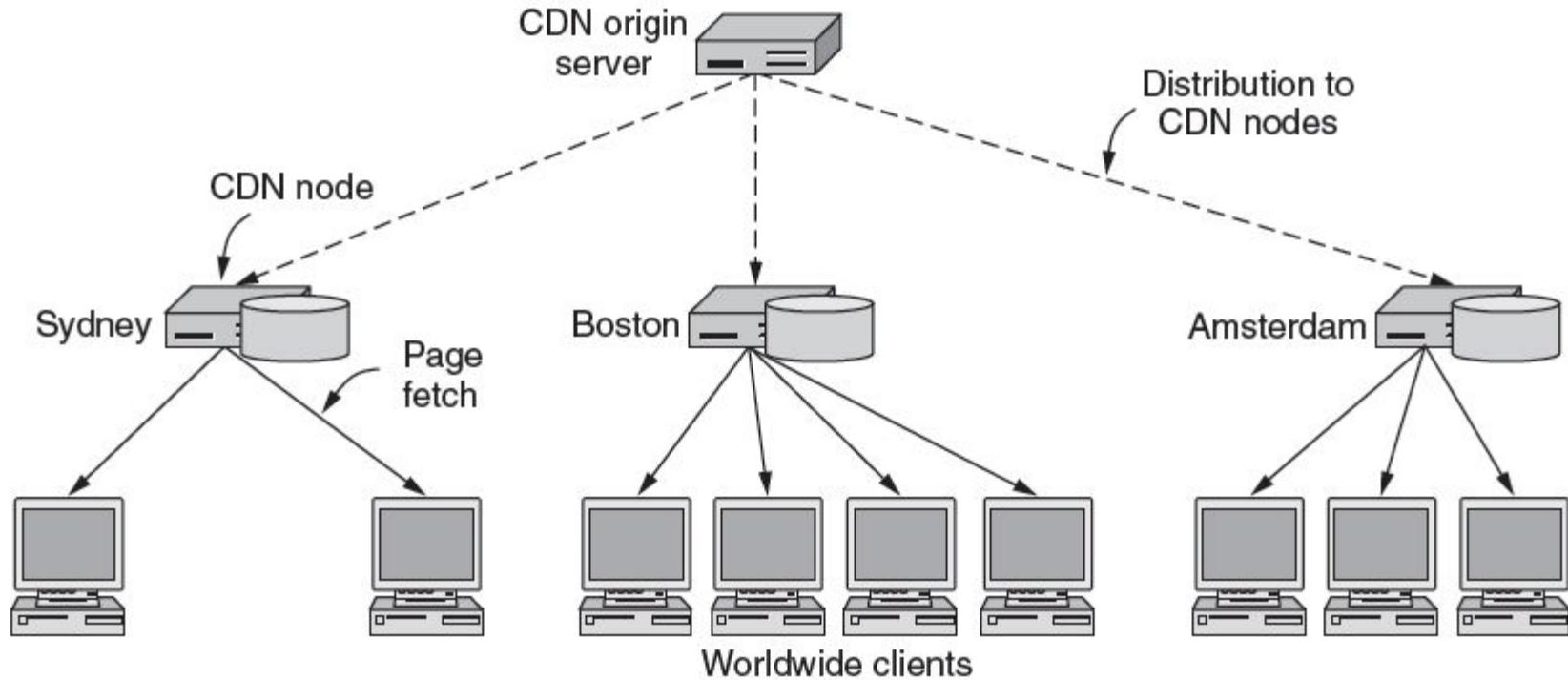
A server farm.

Server Farms and Web Proxies (2)



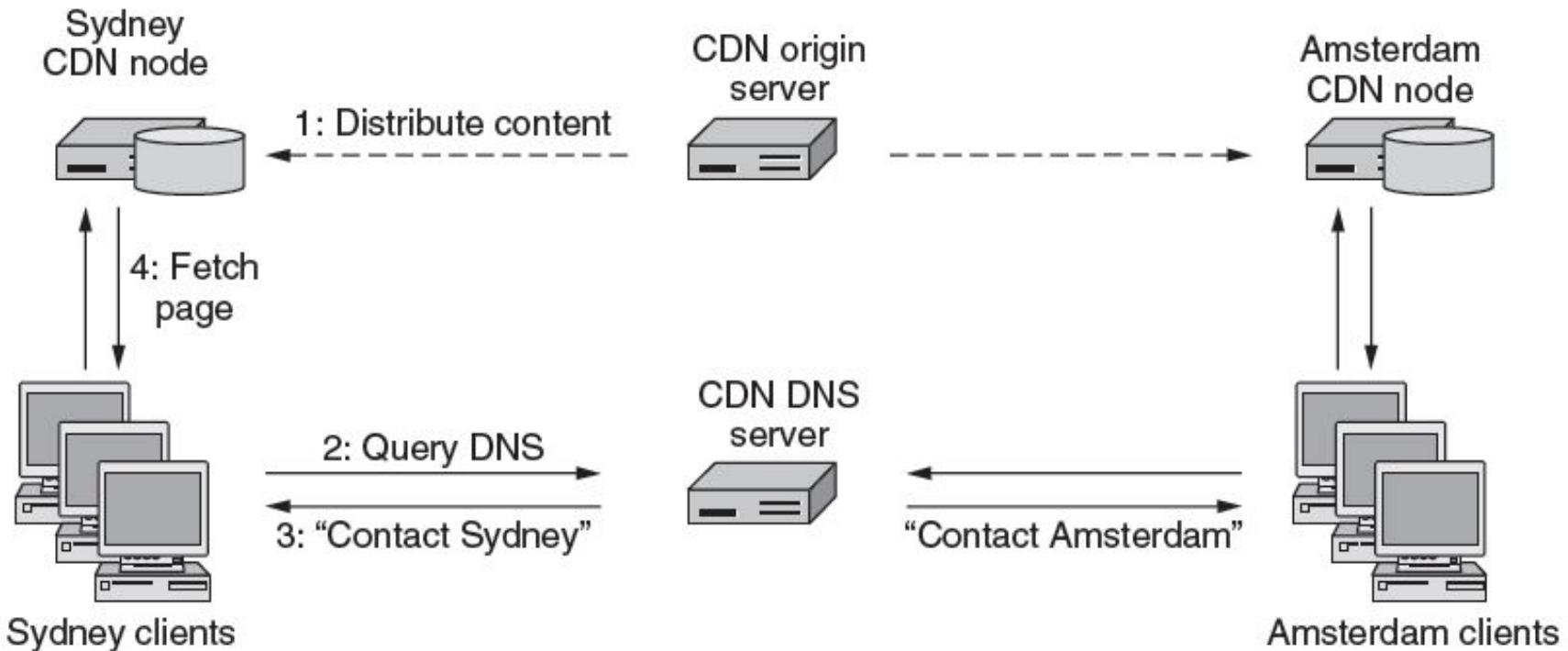
A proxy cache between Web browsers and Web servers.

Content Delivery Networks (1)



CDN distribution tree.

Content Delivery Networks (2)



Directing clients to nearby CDN nodes using DNS.

Content Delivery Networks (3)

```
<html>
<head> <title> Fluffy Video </title> </head>
<body>
<h1> Fluffy Video's Product List </h1>
<p> Click below for free samples. </p>
<a href="koalas.mpg"> Koalas Today </a> <br>
<a href="kangaroos.mpg"> Funny Kangaroos </a> <br>
<a href="wombats.mpg"> Nice Wombats </a> <br>
</body>
</html>
```

(a)

```
<html>
<head> <title> Fluffy Video </title> </head>
<body>
<h1> Fluffy Video's Product List </h1>
<p> Click below for free samples. </p>
<a href="http://www.cdn.com/fluffyvideo/koalas.mpg"> Koalas Today </a> <br>
<a href="http://www.cdn.com/fluffyvideo/kangaroos.mpg"> Funny Kangaroos </a> <br>
<a href="http://www.cdn.com/fluffyvideo/wombats.mpg"> Nice Wombats </a> <br>
</body>
</html>
```

(b)

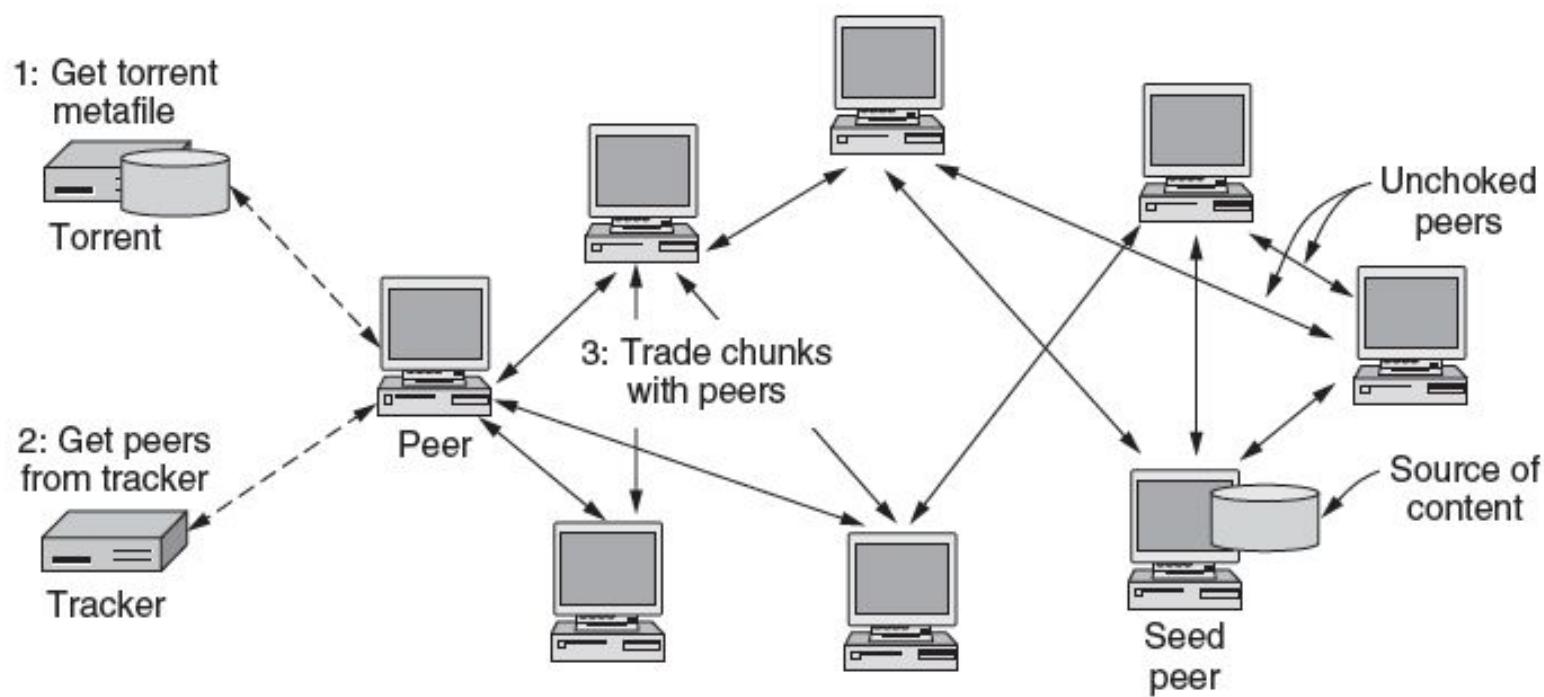
(a) Original Web page. (b) Same page after linking to the CDN

Peer-to-Peer Networks (1)

Problems to be solved with BitTorrent sharing

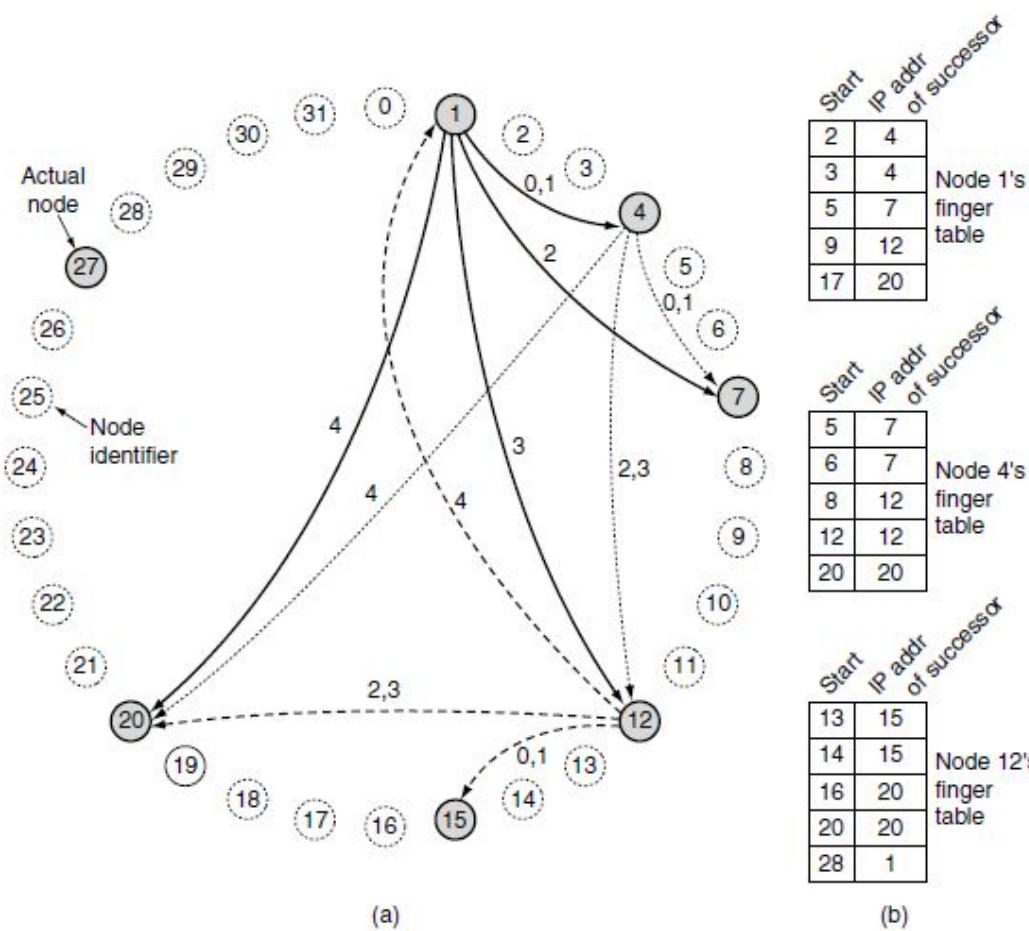
1. How does a peer find other peers
2. How is content replicated by peers to provide high-speed downloads
3. How do peers encourage each other to upload content to others

Peer-to-Peer Networks (2)



BitTorrent.

Peer-to-Peer Networks (3)



(a) A set of 32 node identifiers arranged in a circle. The shaded ones correspond to actual machines. The arcs show the fingers from nodes 1, 4 and 12. The labels on the arcs are the table indices. (b) Examples of the finger tables.

End

Chapter 7