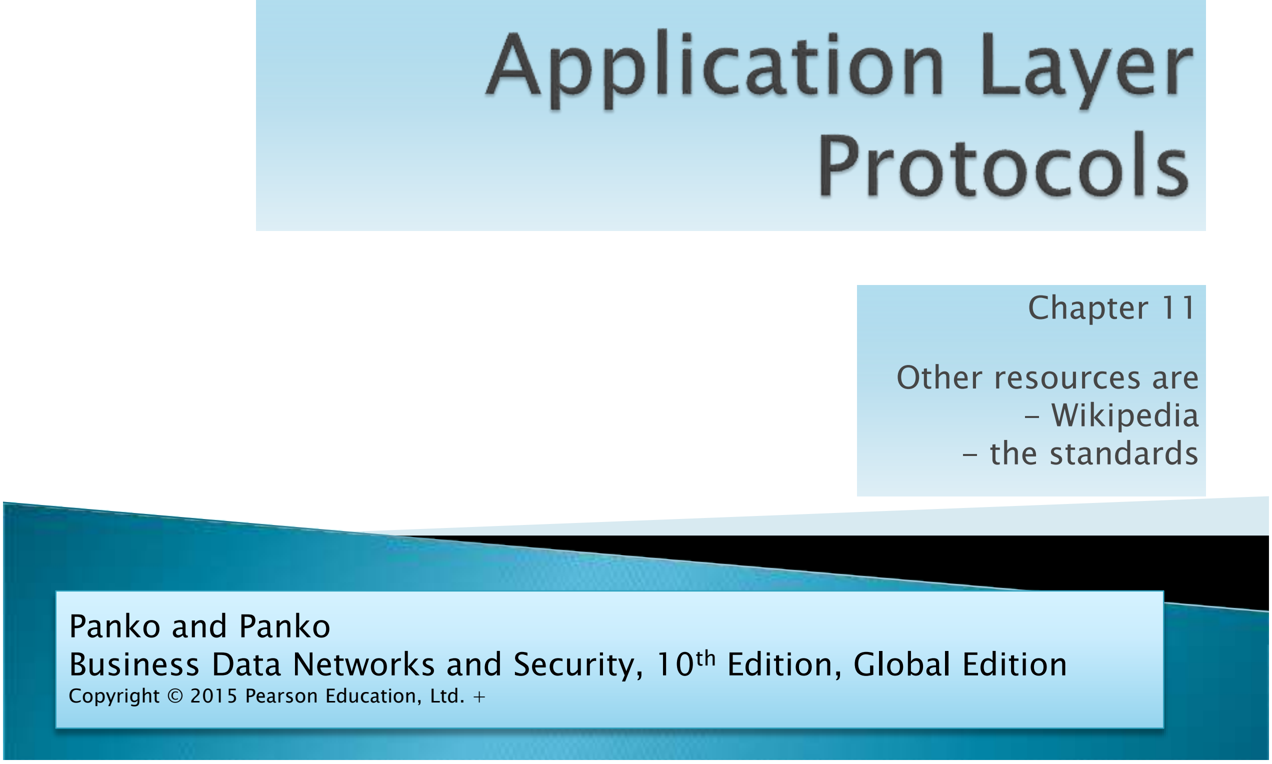


Application Layer Protocols

Chapter 11

Other resources are
– Wikipedia
– the standards

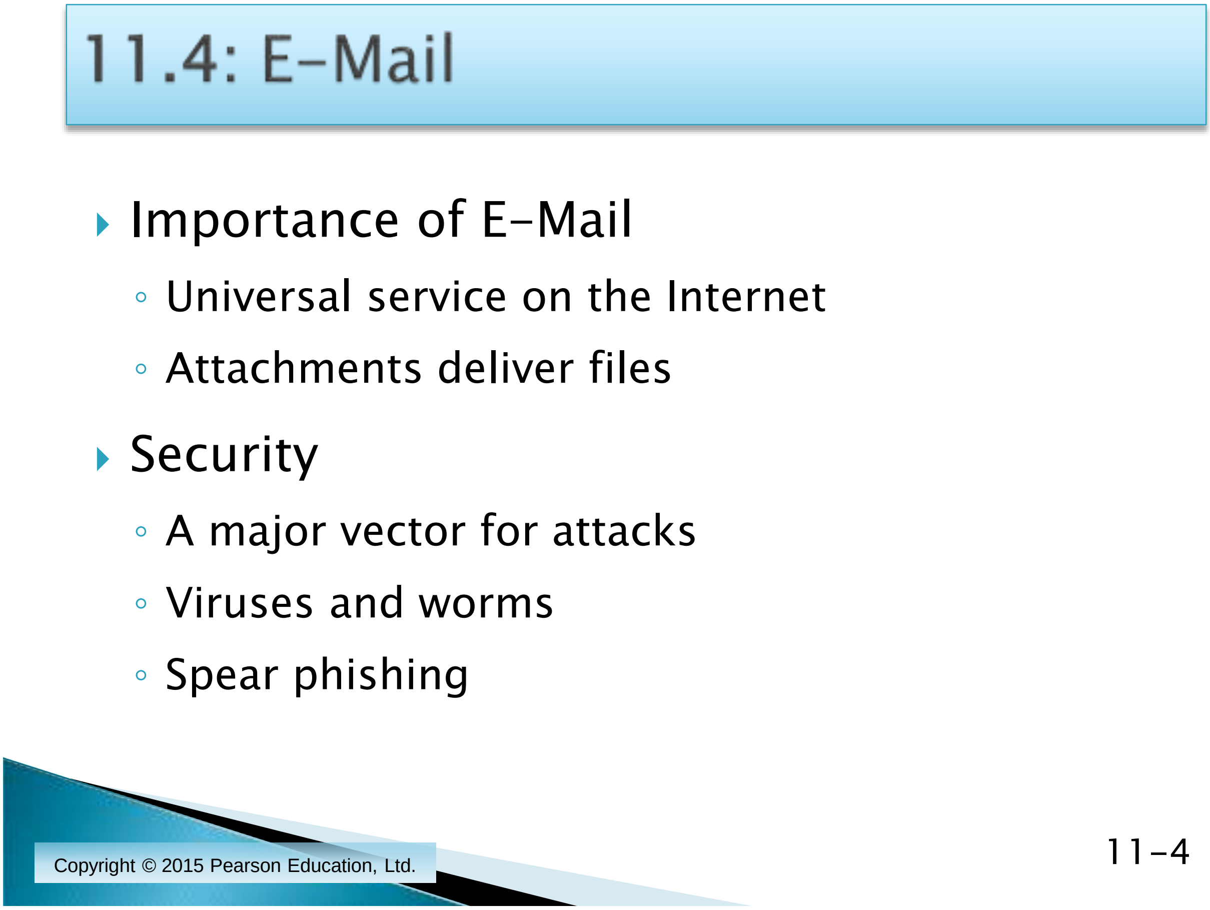
Panko and Panko
Business Data Networks and Security, 10th Edition, Global Edition

Copyright © 2015 Pearson Education, Ltd. +

- Basic Networked Application Concepts
- Terminal-Host v Client/Server Processing
- E-Mail
- Voice over IP
- HTTP and HTML
- Peer-to-Peer (P2P) Applications

Copyright © 2015 Pearson Education, Ltd.+

© 2013 Pearson

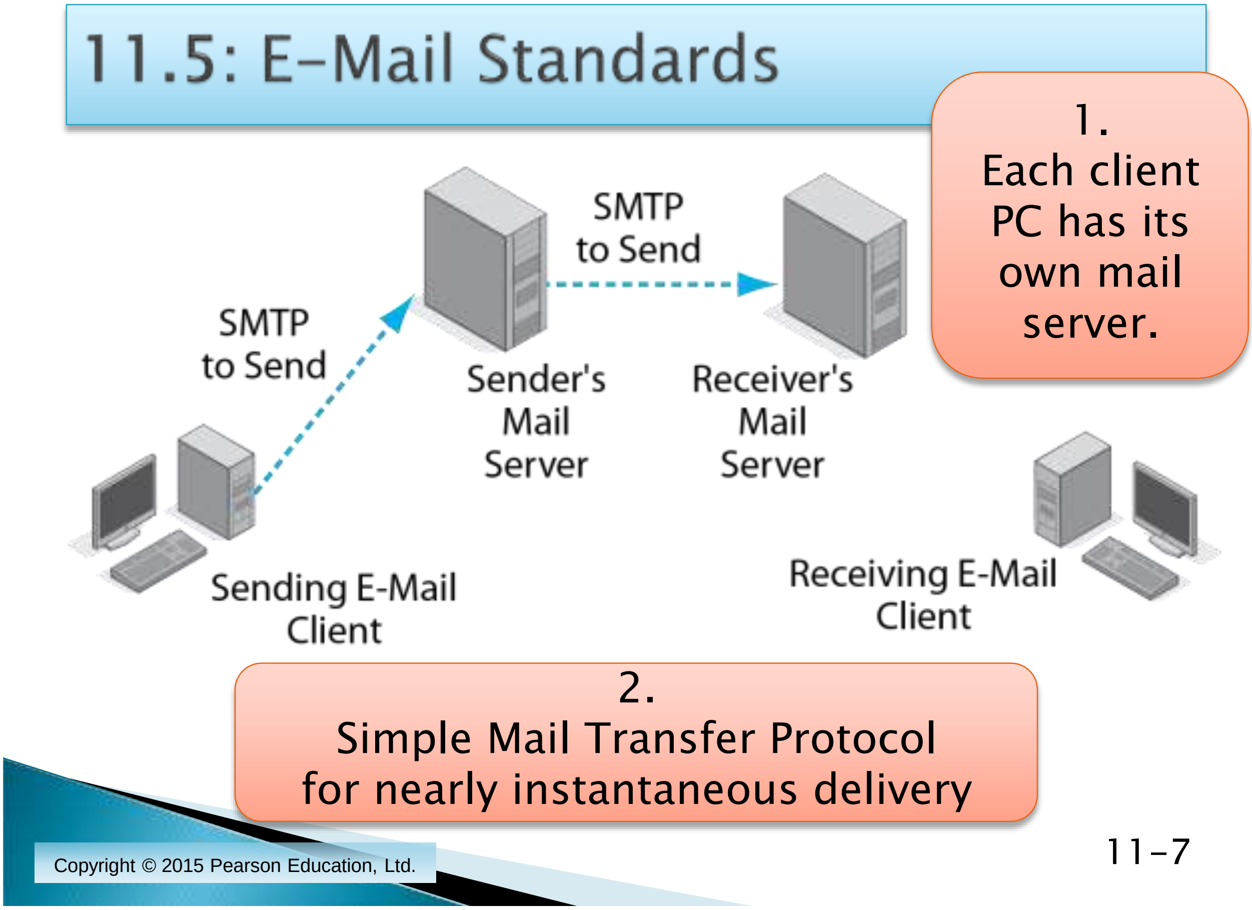


11.4: E-Mail

- Importance of E-Mail
  - Universal service on the Internet
  - Attachments deliver files
- Security
  - A major vector for attacks
  - Viruses and worms
  - Spear phishing

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



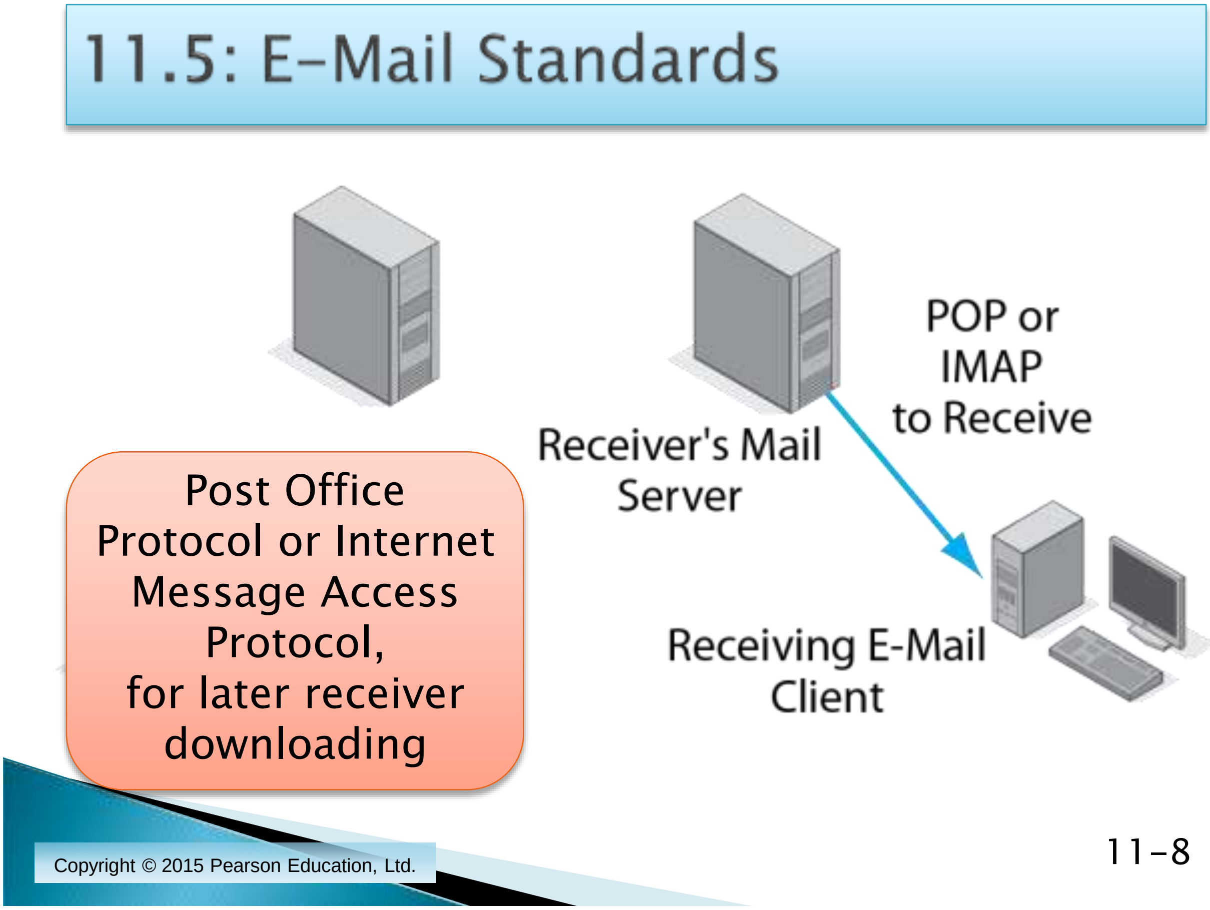
11.5: E–Mail Standards

1.
Each client PC has its own mail server.

2.
Simple Mail Transfer Protocol for nearly instantaneous delivery

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.5: E–Mail Standards

Receiver's Mail Server

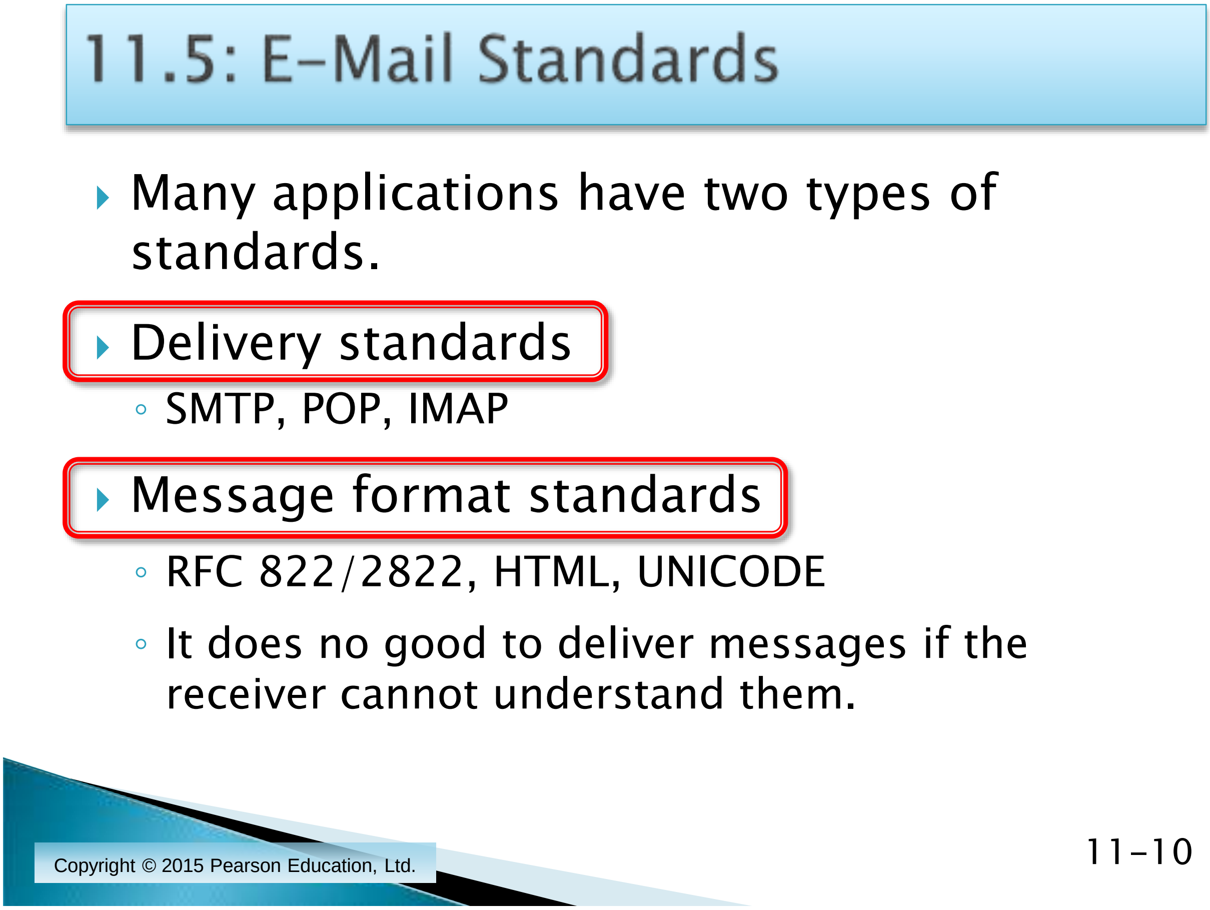
POP or IMAP to Receive

Receiving E-Mail Client

Post Office Protocol or Internet Message Access Protocol, for later receiver downloading

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

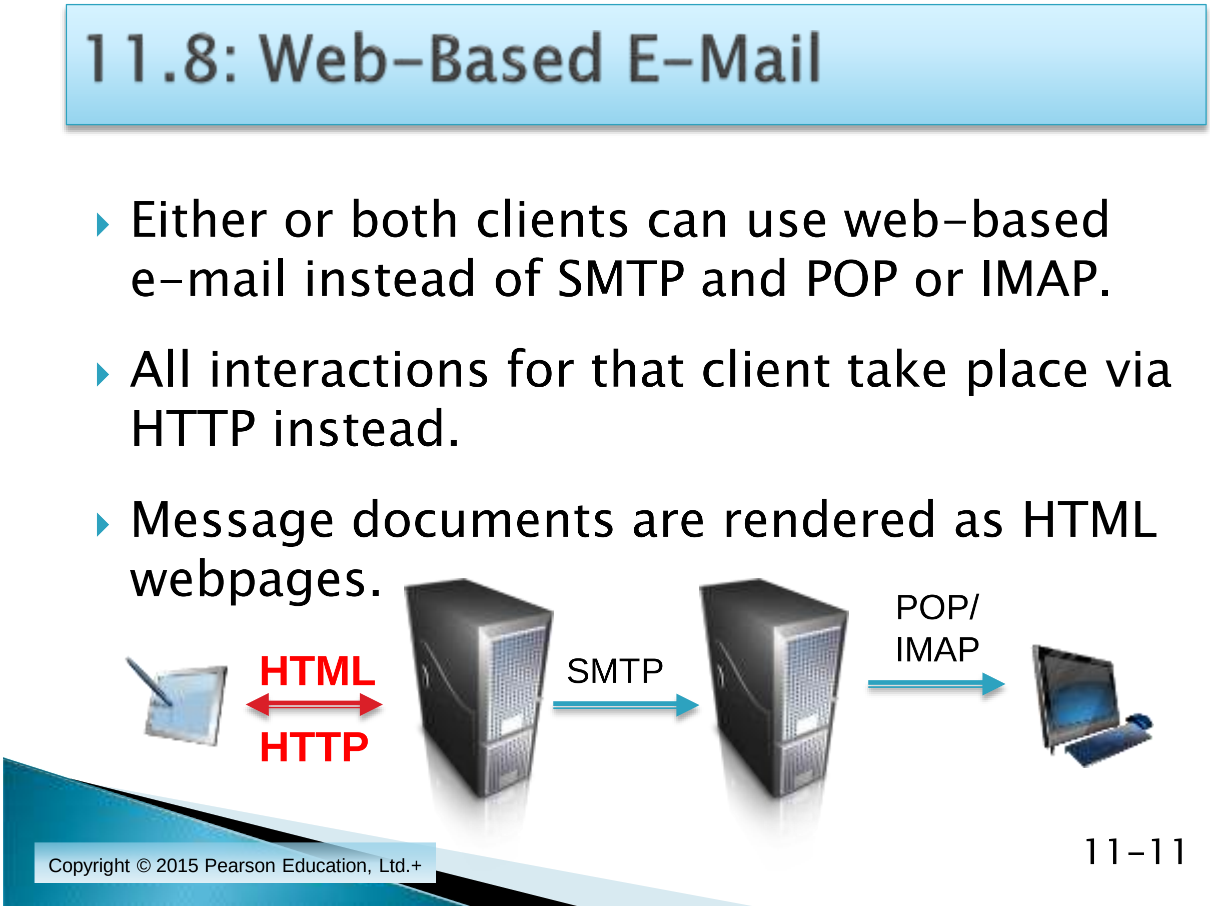


11.5: E-Mail Standards

- Many applications have two types of standards.
- Delivery standards
  - SMTP, POP, IMAP
- Message format standards
  - RFC 822/2822, HTML, UNICODE
  - It does no good to deliver messages if the receiver cannot understand them.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.8: Web-Based E-Mail

- Either or both clients can use web-based e-mail instead of SMTP and POP or IMAP.
- All interactions for that client take place via HTTP instead.
- Message documents are rendered as HTML webpages.

HTML

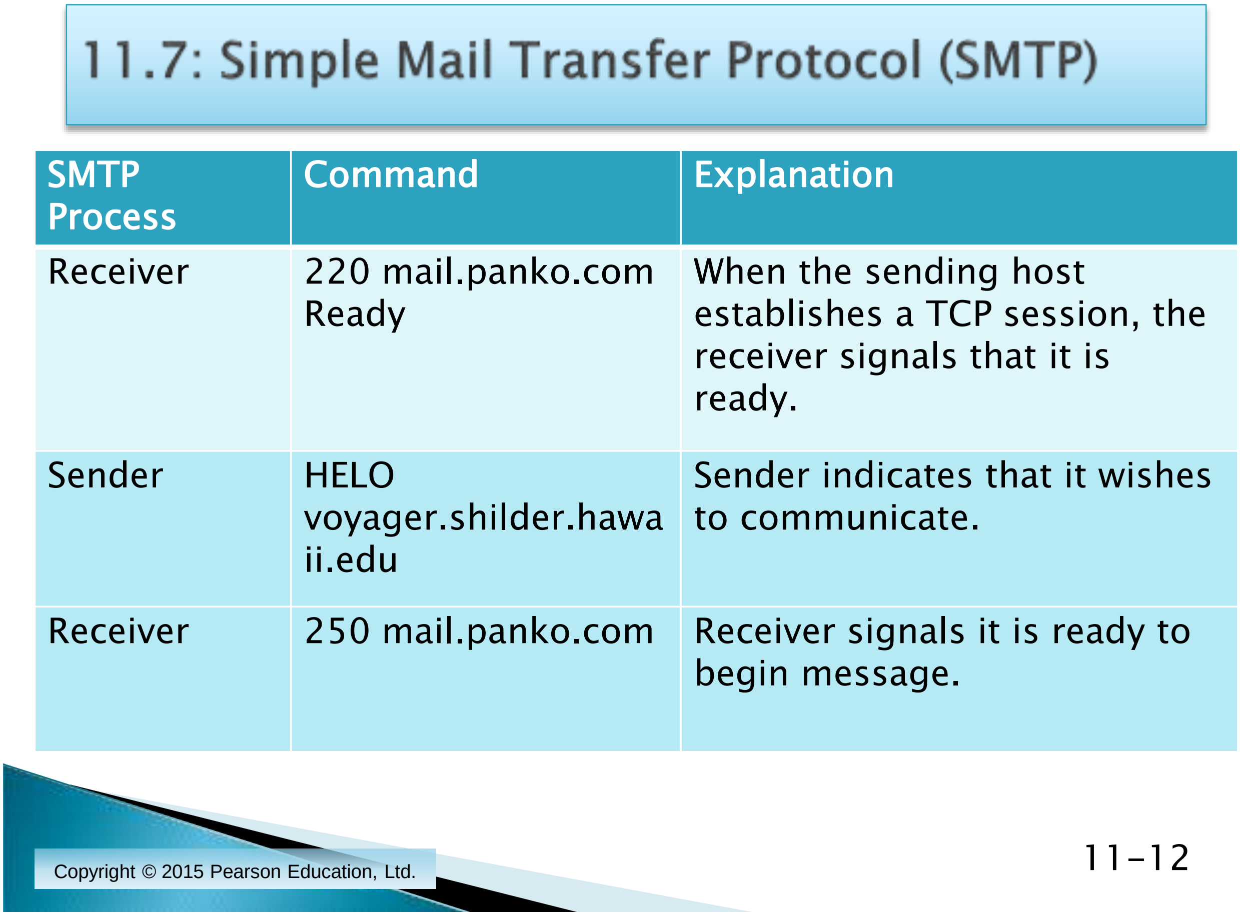
HTTP

SMTP

POP/
IMAP

Copyright © 2015 Pearson Education, Ltd.+

© 2013 Pearson

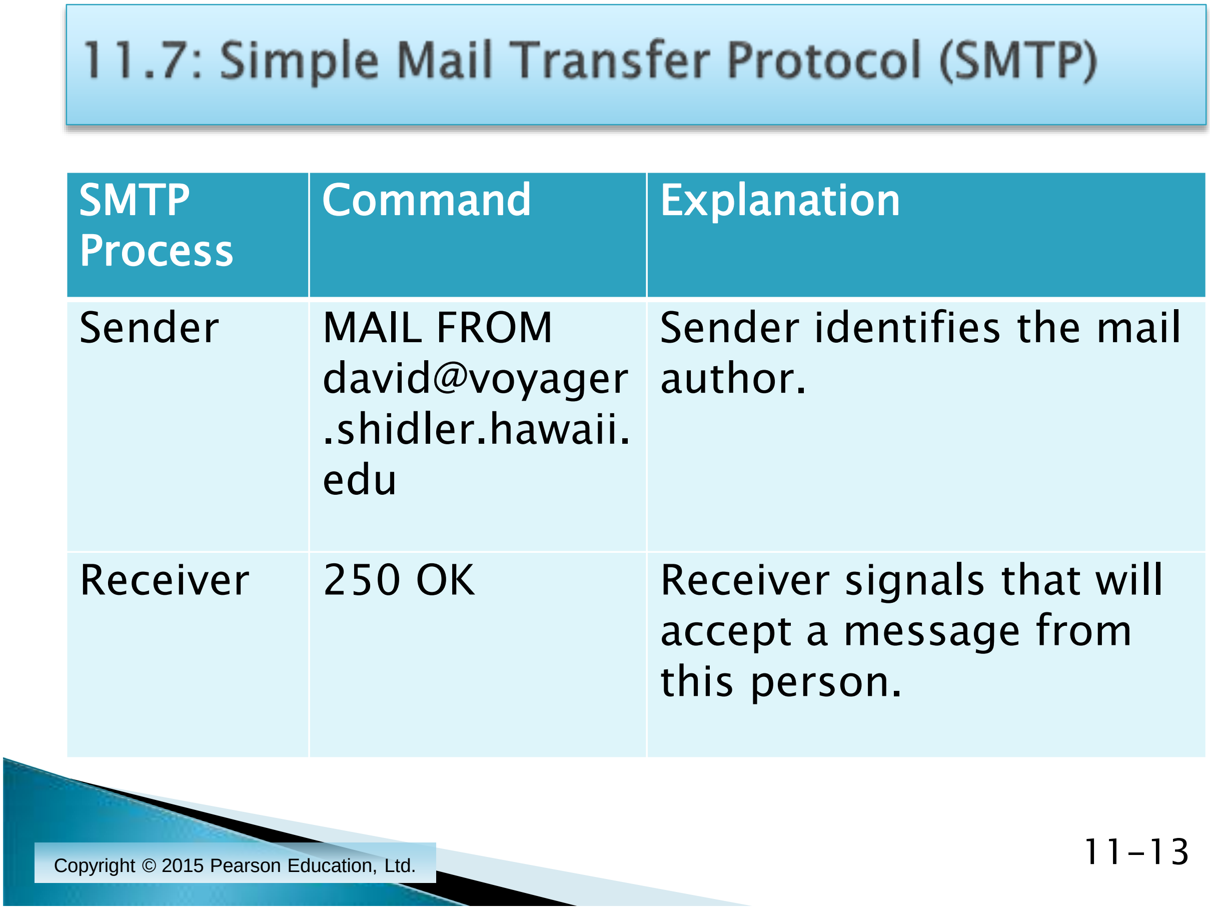


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Receiver	220 mail.panko.com Ready	When the sending host establishes a TCP session, the receiver signals that it is ready.
Sender	HELO voyager.shilder.hawaii.edu	Sender indicates that it wishes to communicate.
Receiver	250 mail.panko.com	Receiver signals it is ready to begin message.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

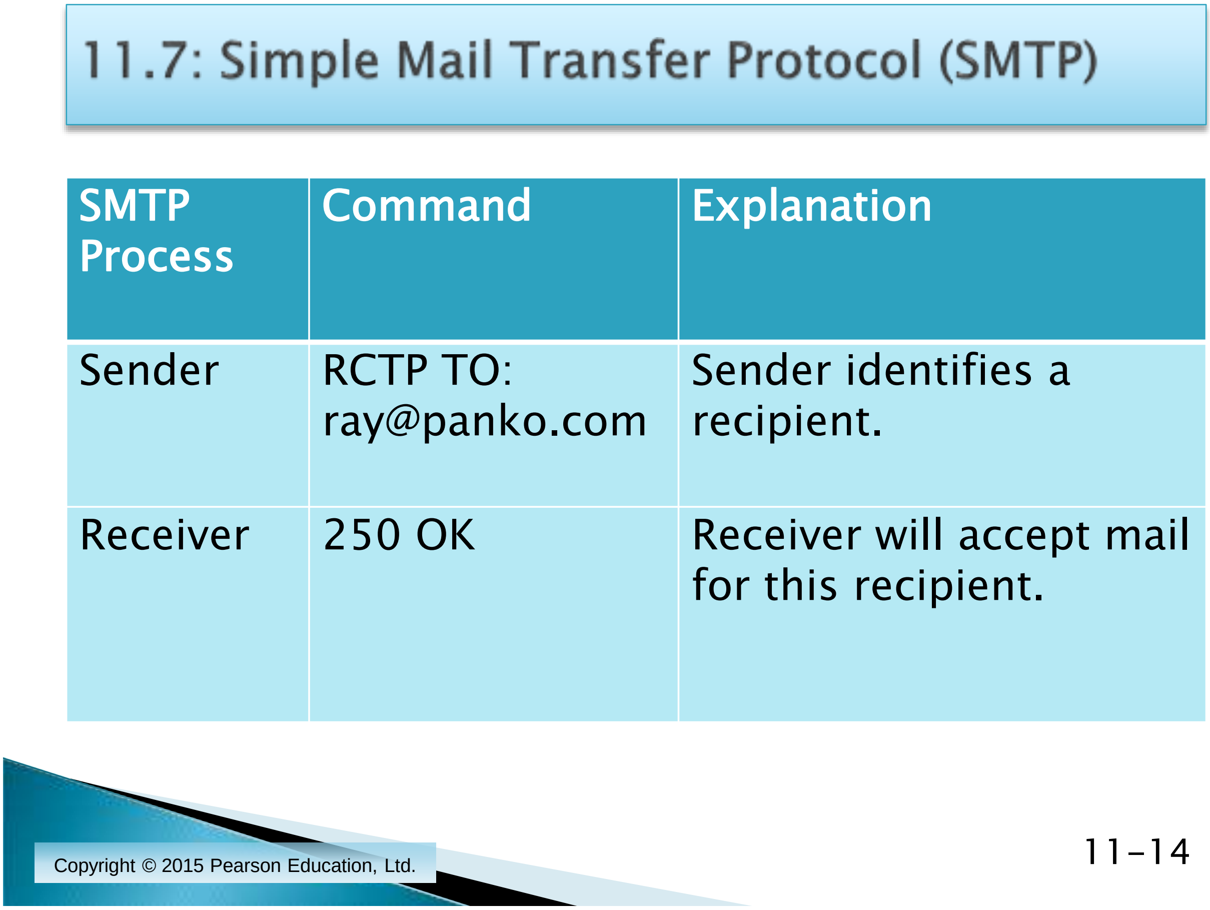


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Sender	MAIL FROM david@voyager.shidler.hawaii.edu	Sender identifies the mail author.
Receiver	250 OK	Receiver signals that will accept a message from this person.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

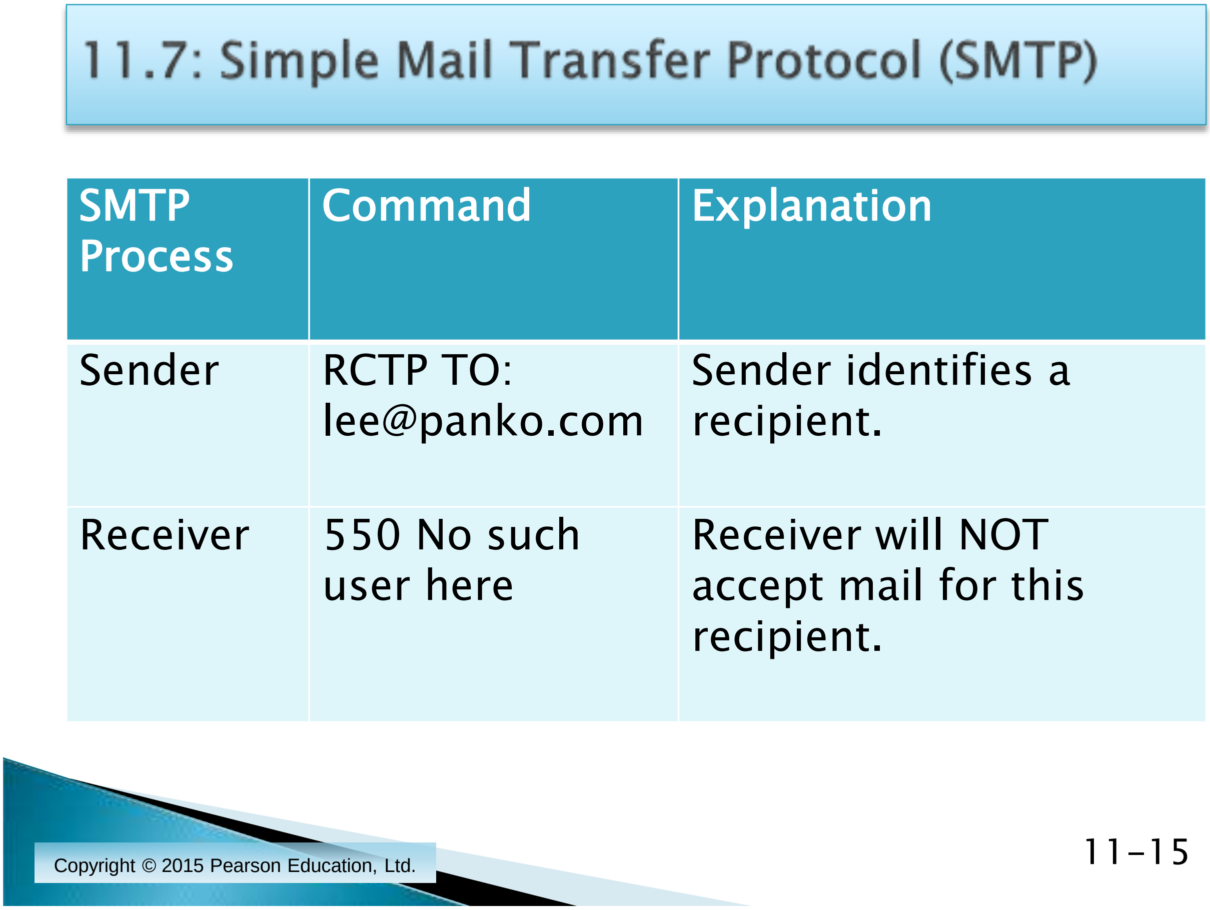


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Sender	RCTP TO: ray@panko.com	Sender identifies a recipient.
Receiver	250 OK	Receiver will accept mail for this recipient.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

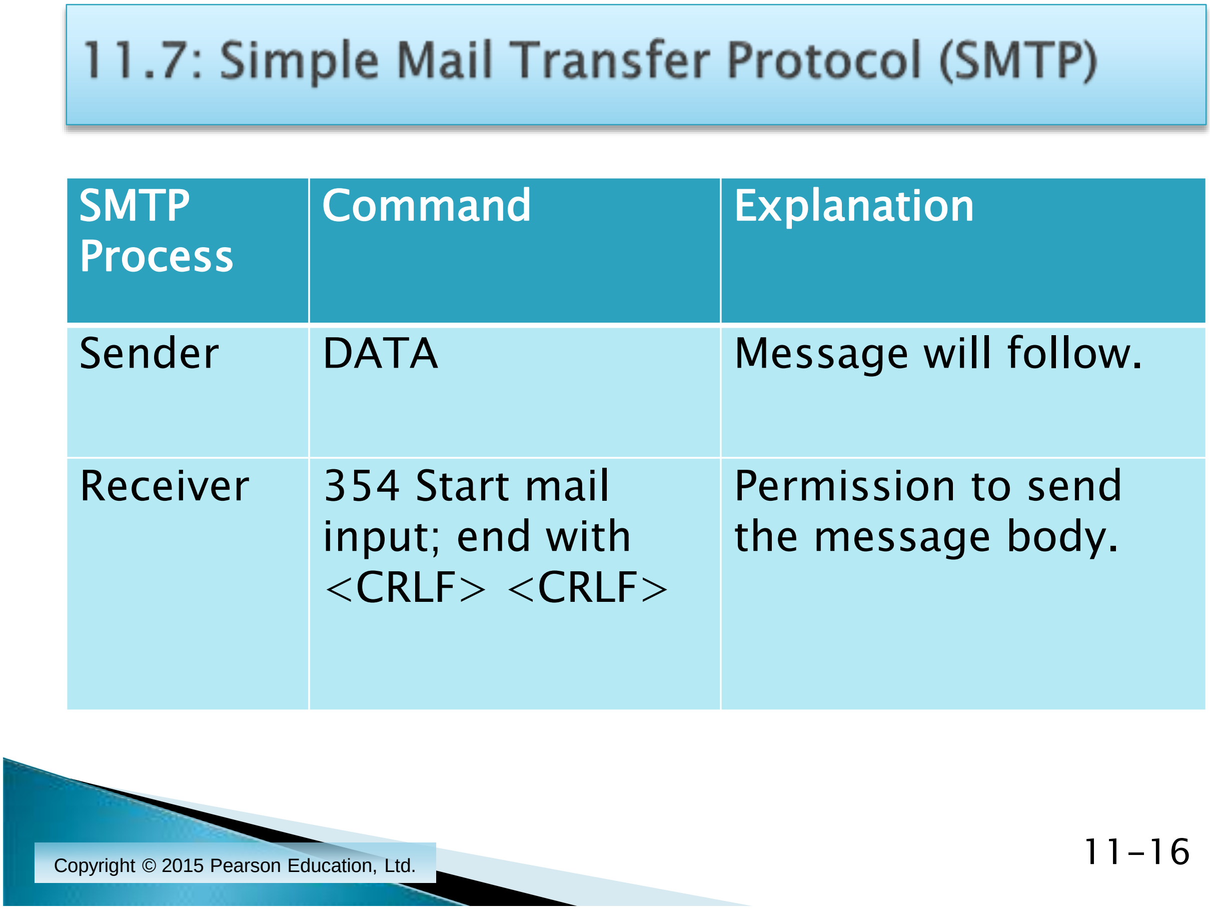


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Sender	RCTP TO: lee@panko.com	Sender identifies a recipient.
Receiver	550 No such user here	Receiver will NOT accept mail for this recipient.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

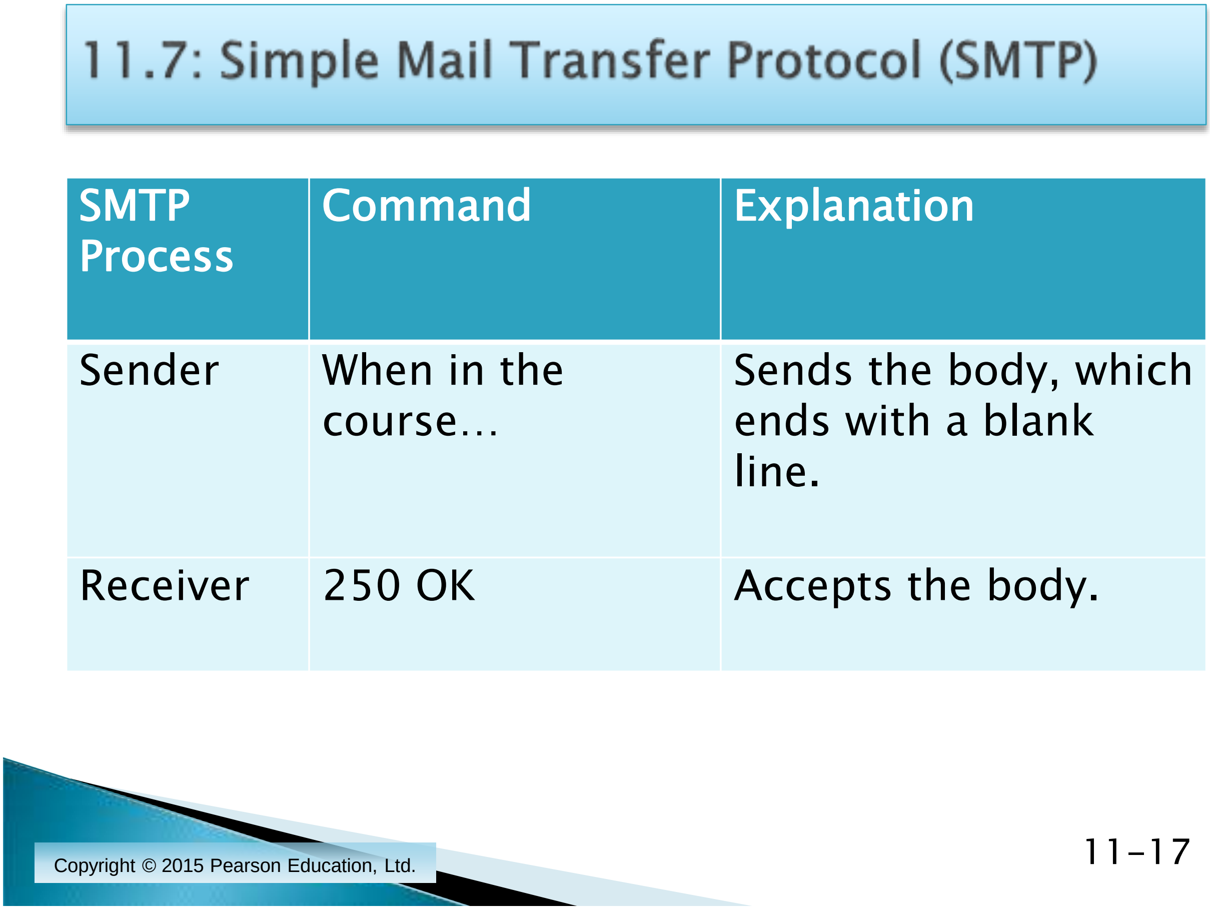


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Sender	DATA	Message will follow.
Receiver	354 Start mail input; end with <CRLF> <CRLF>	Permission to send the message body.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

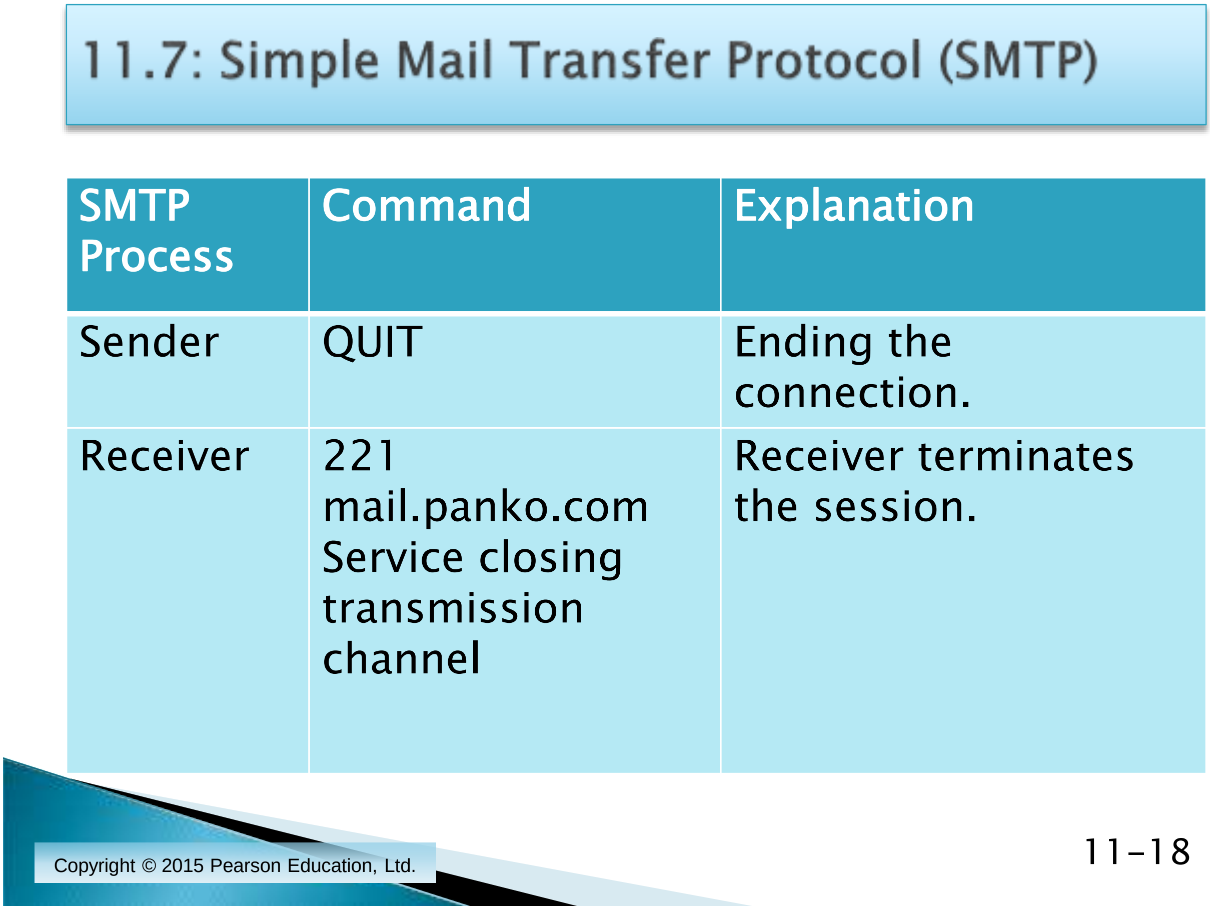


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Sender	When in the course…	Sends the body, which ends with a blank line.
Receiver	250 OK	Accepts the body.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

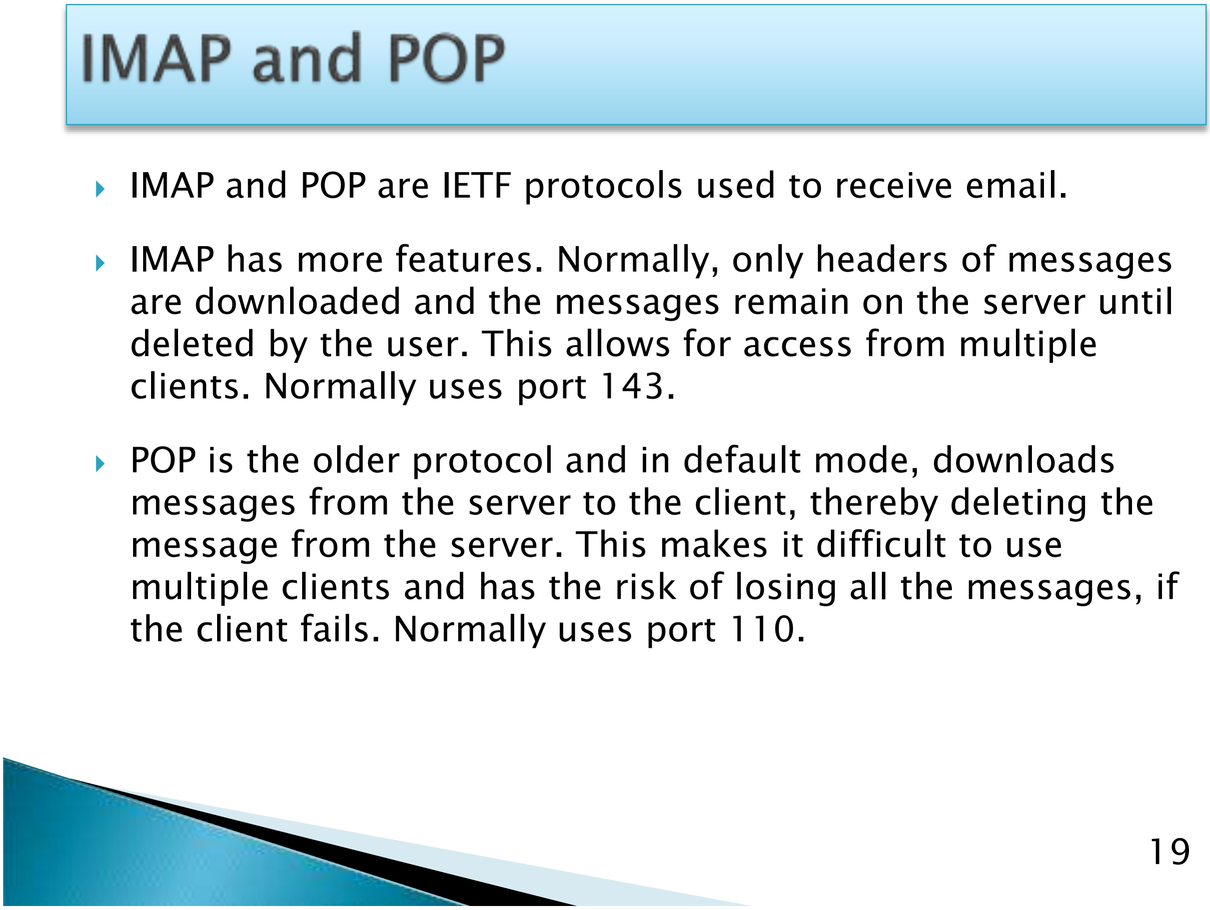


11.7: Simple Mail Transfer Protocol (SMTP)

SMTP Process	Command	Explanation
Sender	QUIT	Ending the connection.
Receiver	221 mail.panko.com Service closing transmission channel	Receiver terminates the session.

Copyright © 2015 Pearson Education, Ltd.

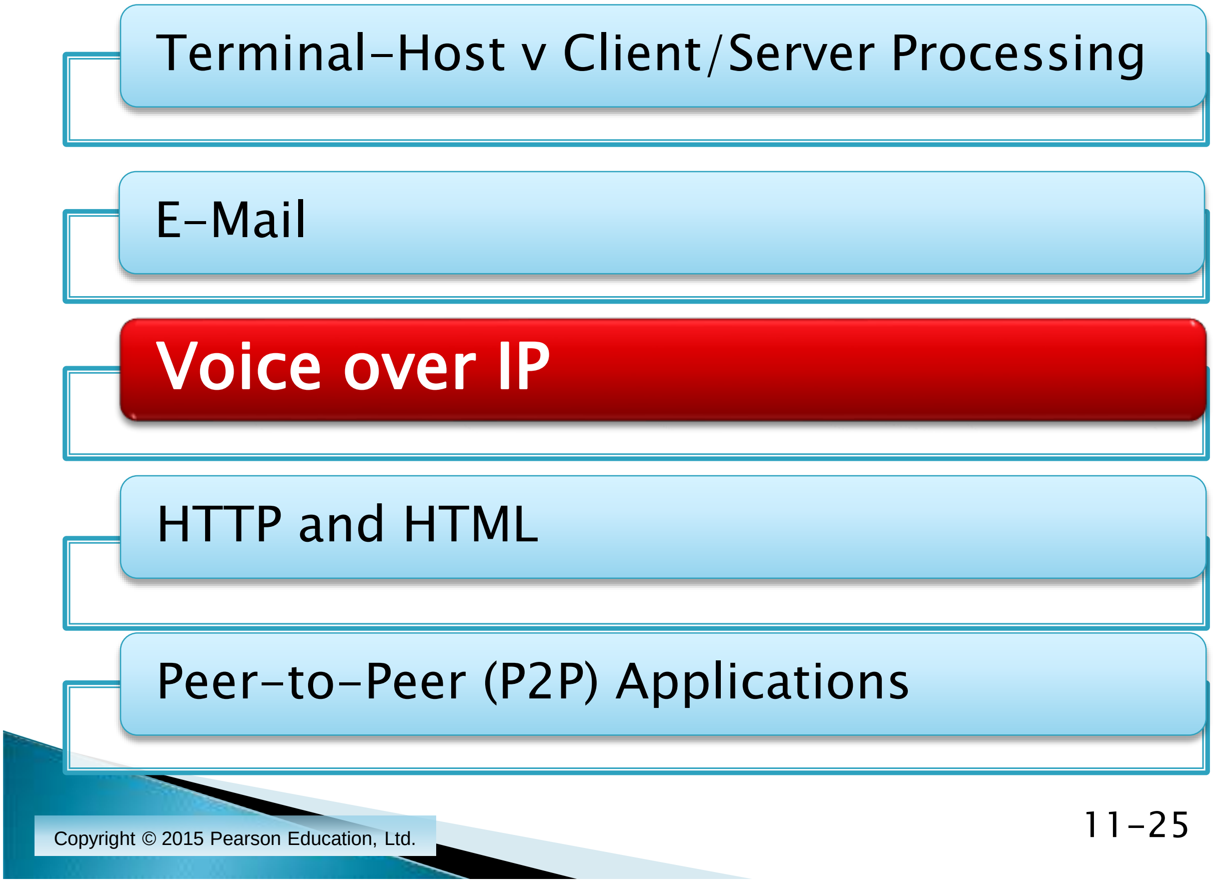
© 2013 Pearson



IMAP and POP

- IMAP and POP are IETF protocols used to receive email.
- IMAP has more features. Normally, only headers of messages are downloaded and the messages remain on the server until deleted by the user. This allows for access from multiple clients. Normally uses port 143.
- POP is the older protocol and in default mode, downloads messages from the server to the client, thereby deleting the message from the server. This makes it difficult to use multiple clients and has the risk of losing all the messages, if the client fails. Normally uses port 110.

© 2013 Pearson



Terminal-Host v Client/Server Processing

E-Mail

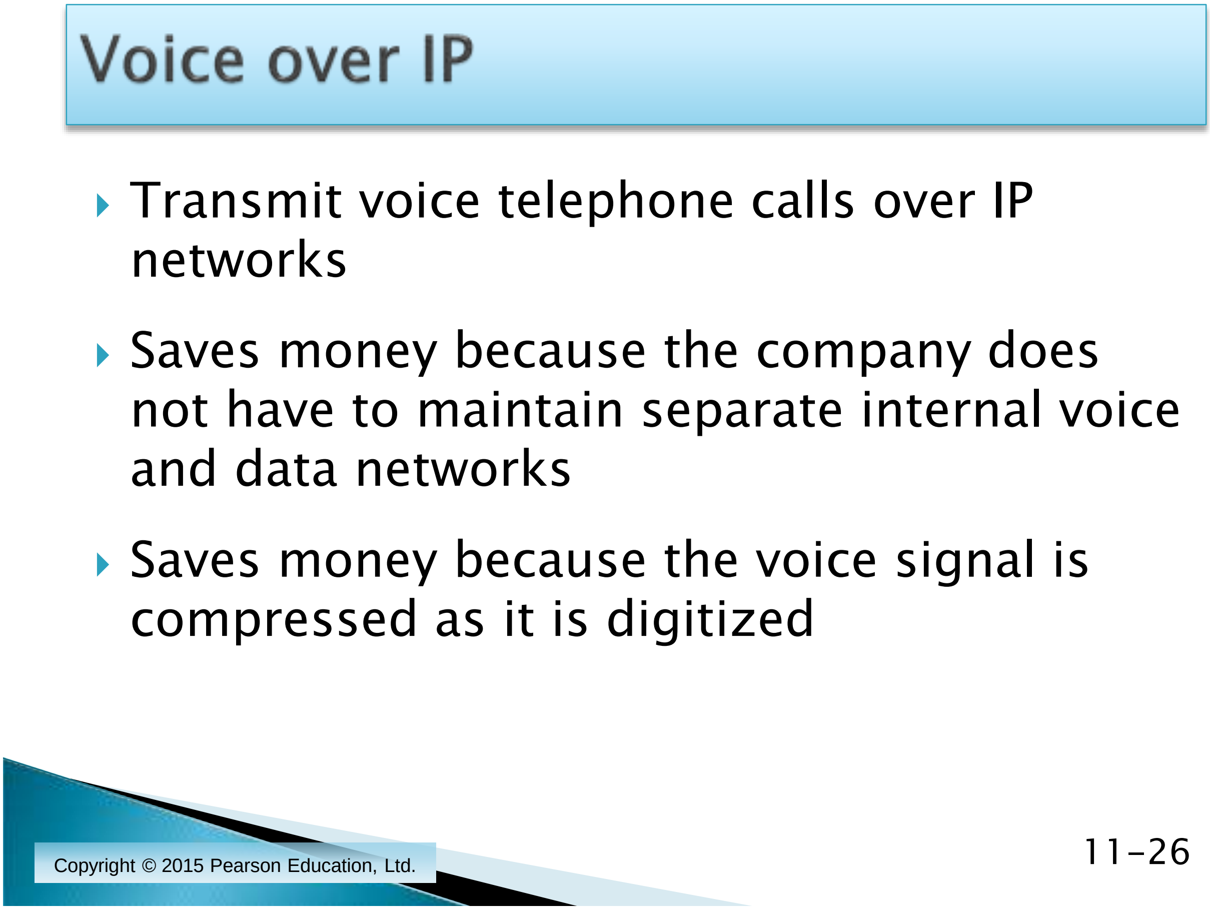
Voice over IP

HTTP and HTML

Peer-to-Peer (P2P) Applications

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

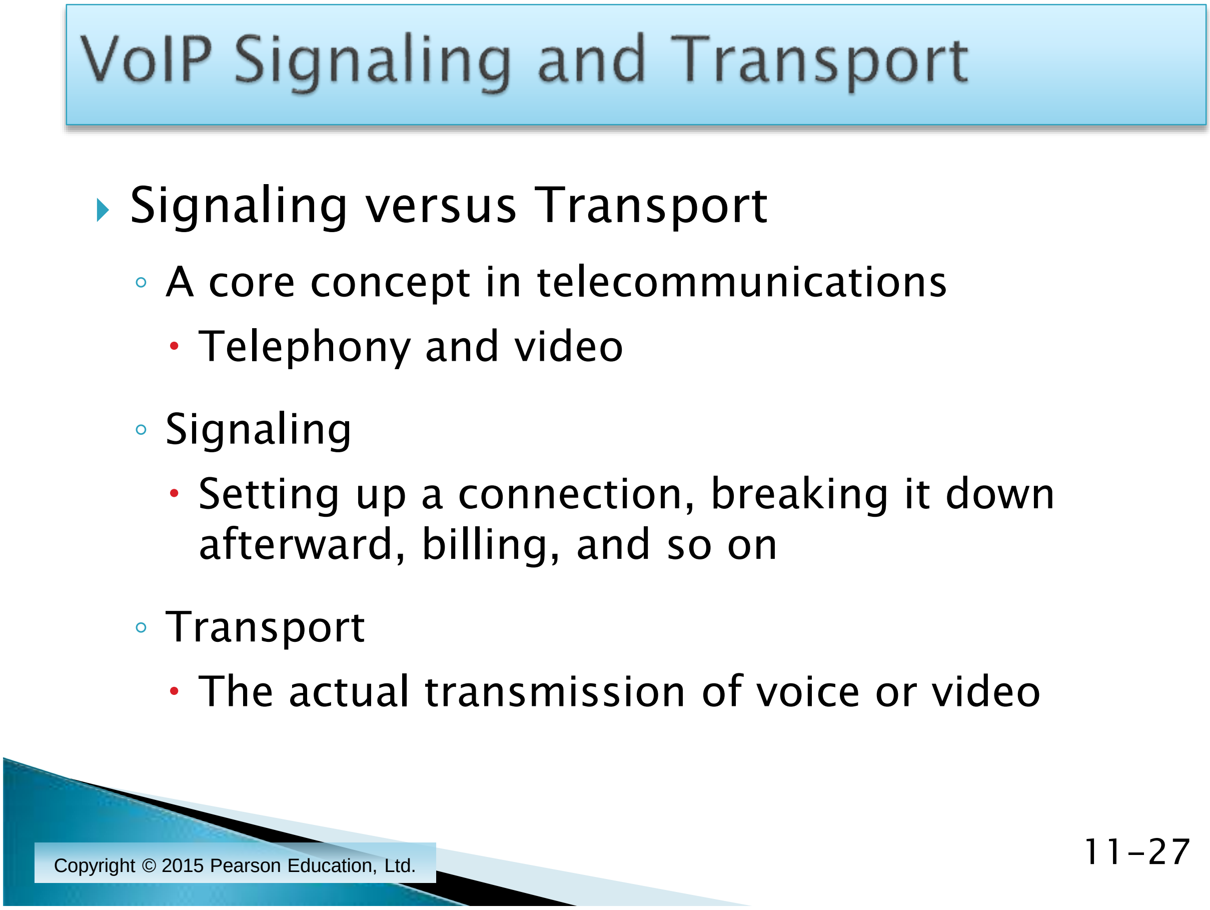


Voice over IP

- Transmit voice telephone calls over IP networks
- Saves money because the company does not have to maintain separate internal voice and data networks
- Saves money because the voice signal is compressed as it is digitized

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



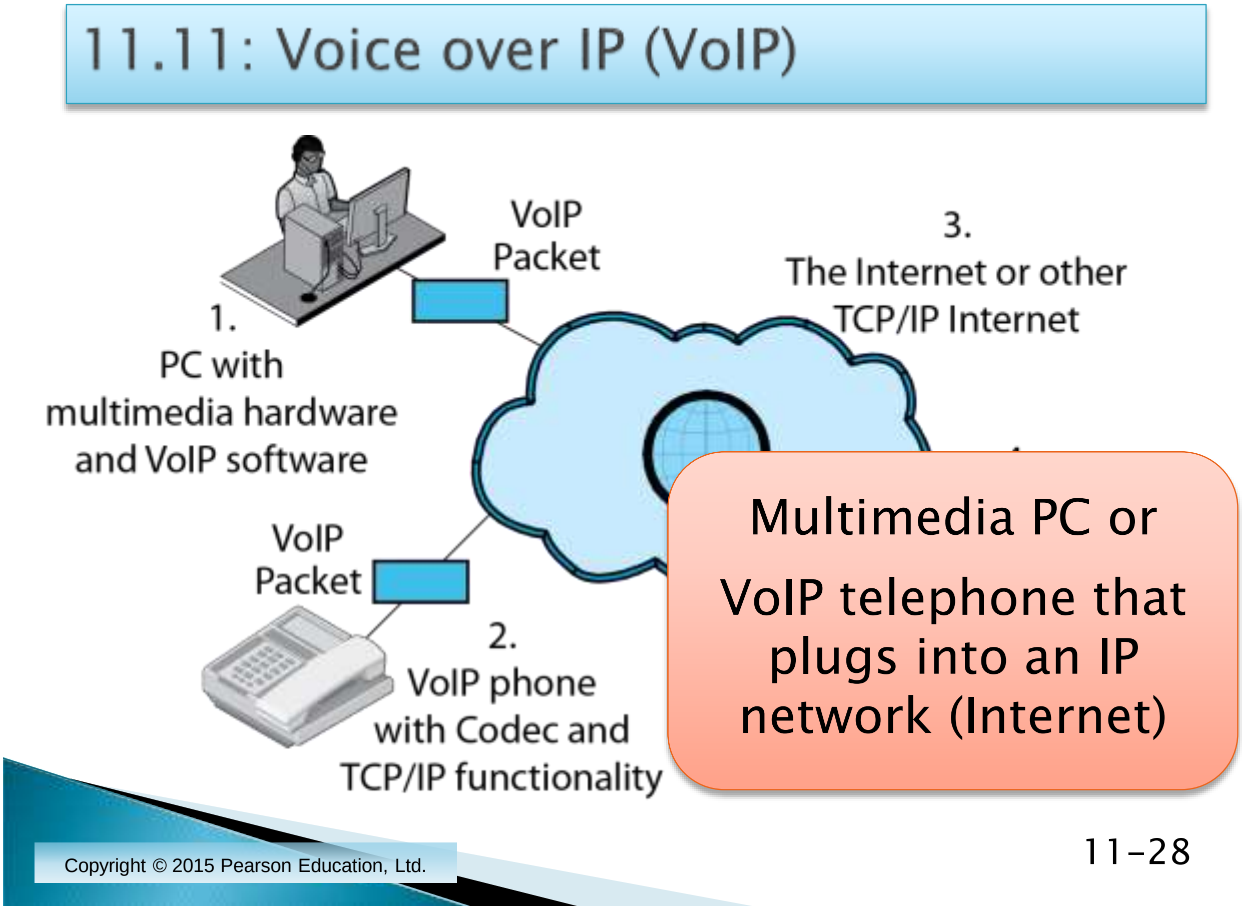
VoIP Signaling and Transport

- Signaling versus Transport
  - A core concept in telecommunications
    - Telephony and video
  - Signaling
    - Setting up a connection, breaking it down afterward, billing, and so on
  - Transport
    - The actual transmission of voice or video

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

11.11: Voice over IP (VoIP)

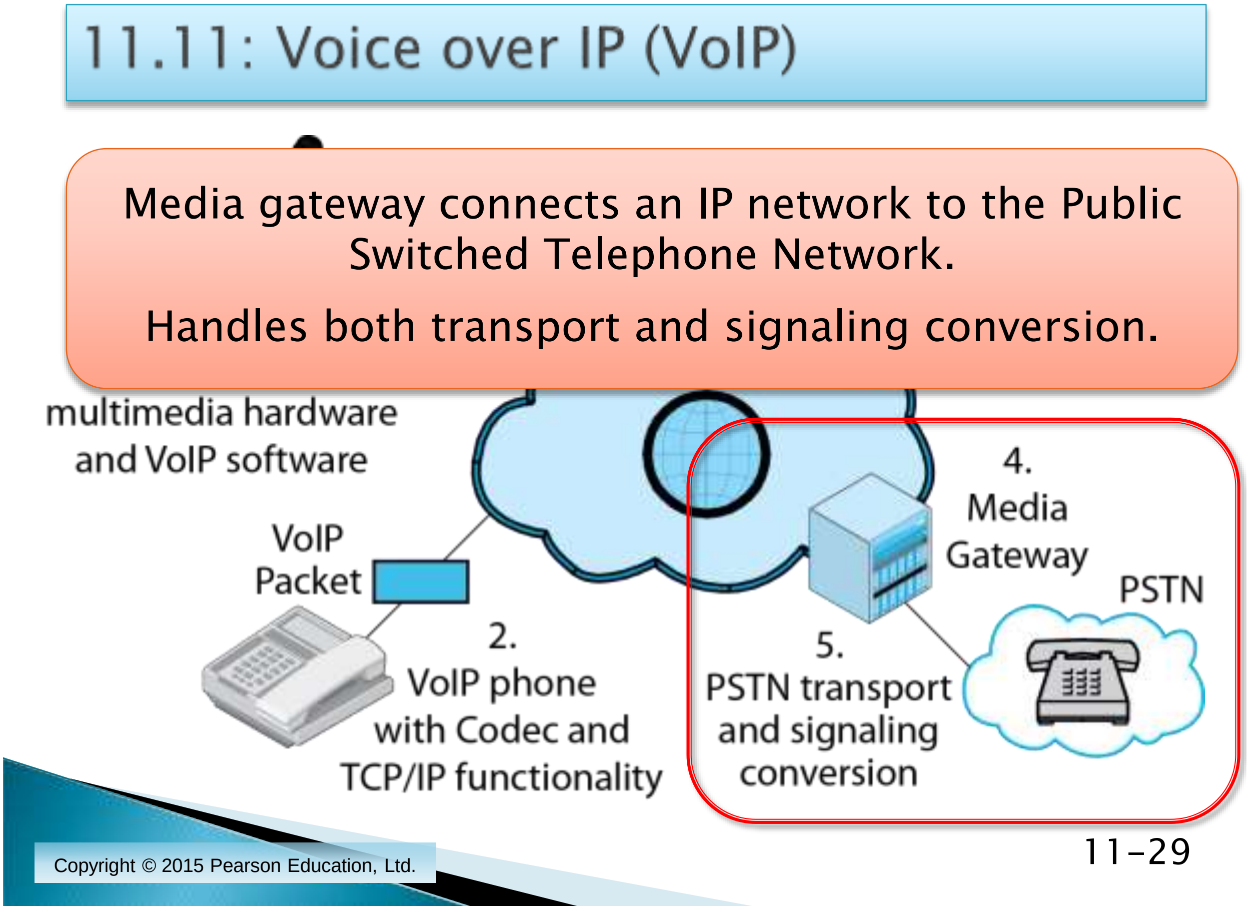


Multimedia PC or

VoIP telephone that plugs into an IP network (Internet)

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.11: Voice over IP (VoIP)

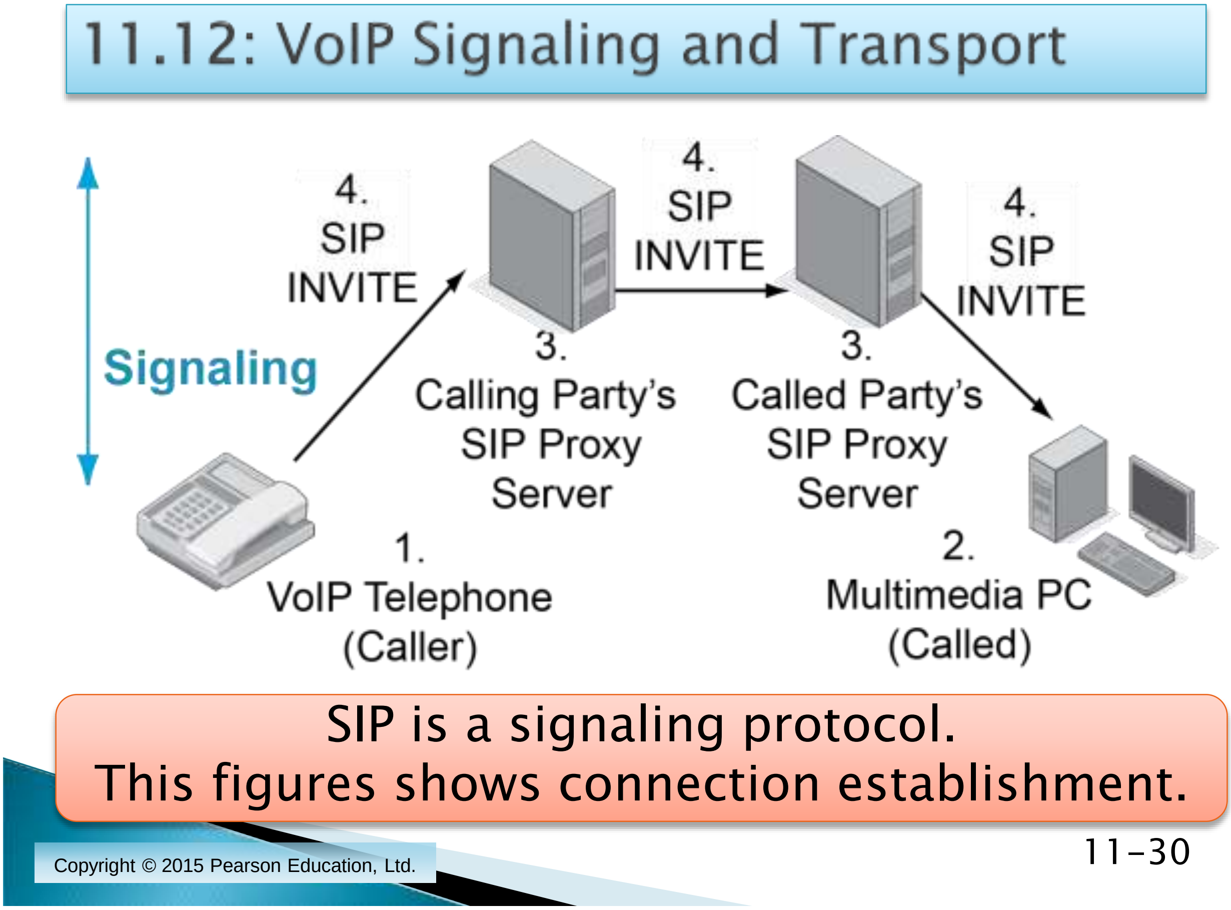
Media gateway connects an IP network to the Public Switched Telephone Network.

Handles both transport and signaling conversion.

multimedia hardware and VoIP software
VoIP Packet
2. VoIP phone with Codec and TCP/IP functionality
4. Media Gateway
PSTN
5. PSTN transport and signaling conversion

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

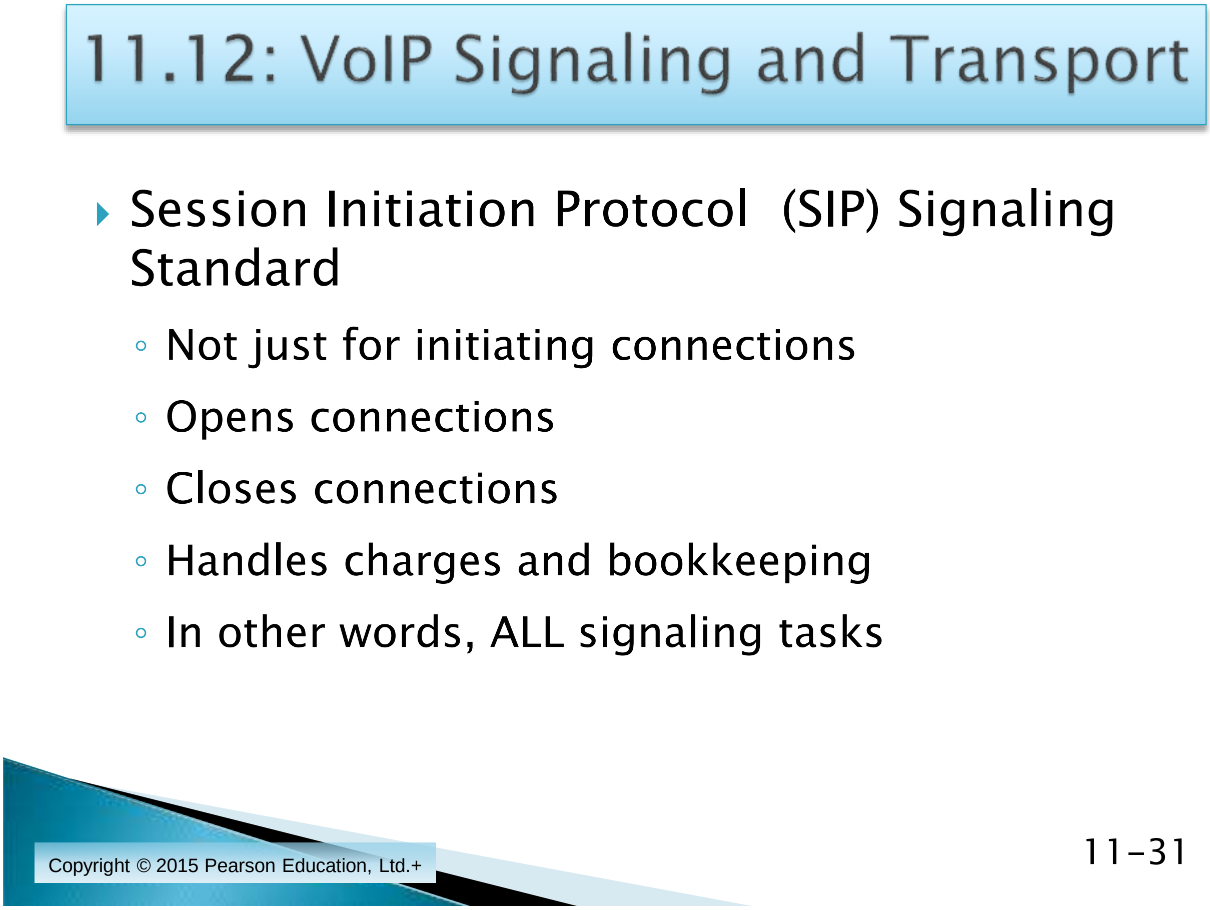


11.12: VoIP Signaling and Transport

SIP is a signaling protocol.
This figures shows connection establishment.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

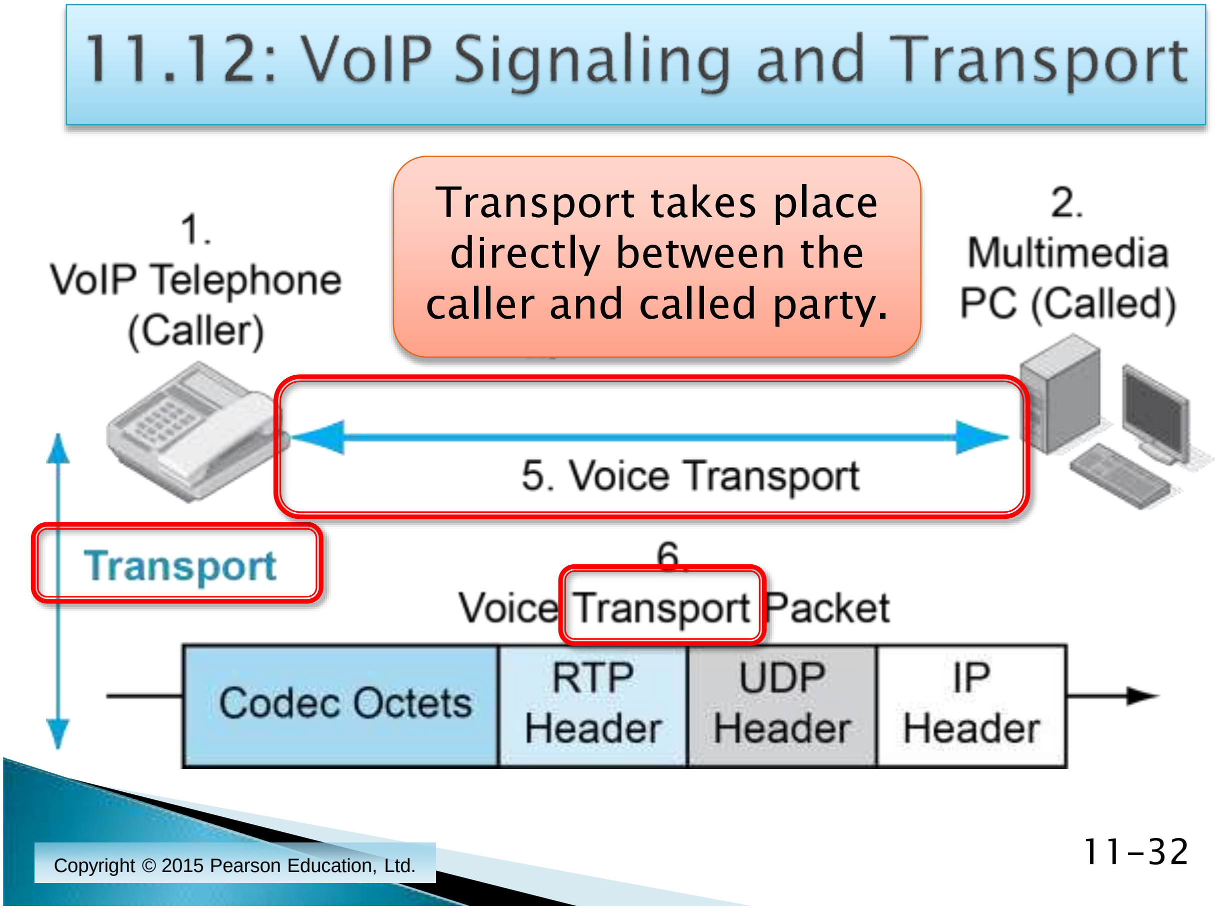


11.12: VoIP Signaling and Transport

- Session Initiation Protocol (SIP) Signaling Standard
  - Not just for initiating connections
  - Opens connections
  - Closes connections
  - Handles charges and bookkeeping
  - In other words, ALL signaling tasks

Copyright © 2015 Pearson Education, Ltd.+

© 2013 Pearson



11.12: VoIP Signaling and Transport

Transport takes place directly between the caller and called party.

1. VoIP Telephone (Caller)

2. Multimedia PC (Called)

5. Voice Transport

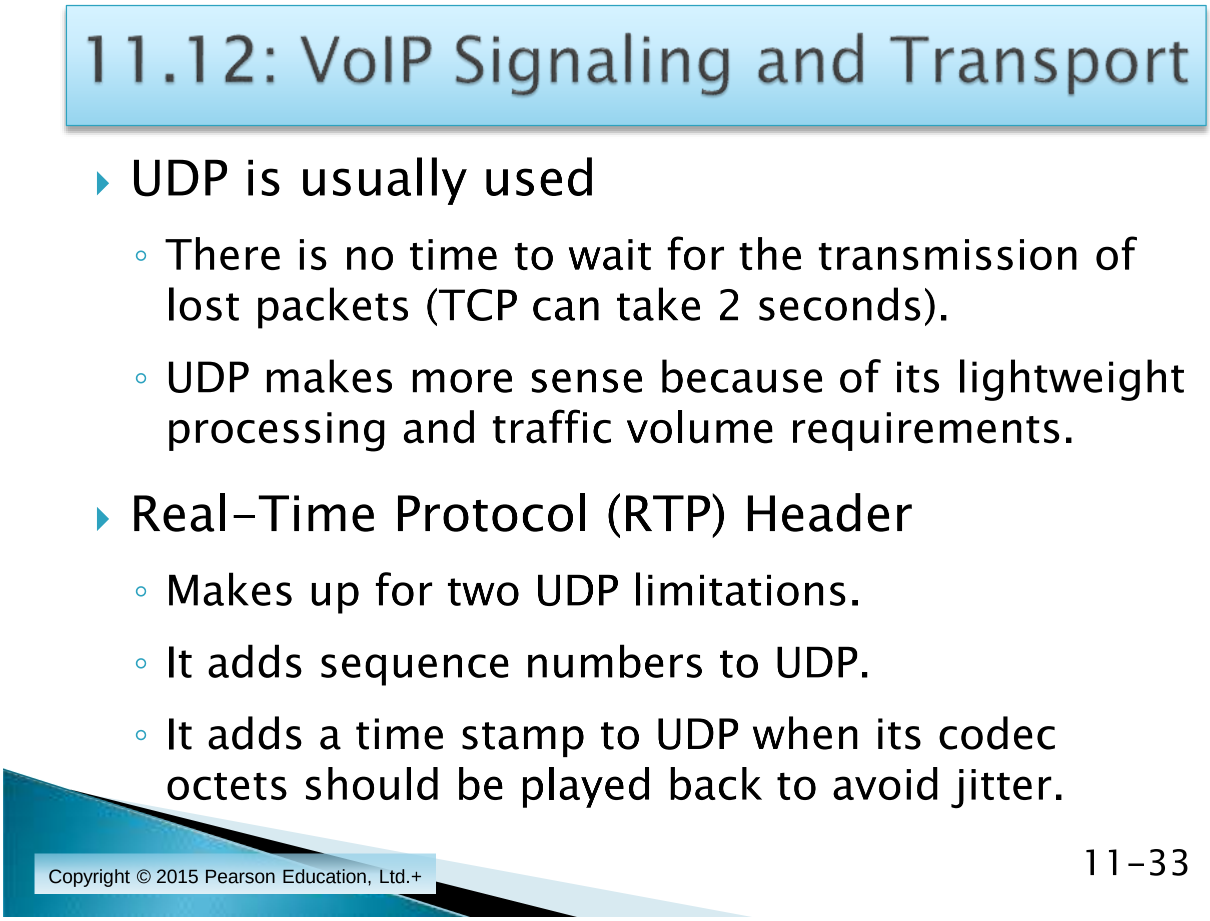
Transport

6. Voice Transport Packet

Codec Octets	RTP Header	UDP Header	IP Header

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.12: VoIP Signaling and Transport

- UDP is usually used
  - There is no time to wait for the transmission of lost packets (TCP can take 2 seconds).
  - UDP makes more sense because of its lightweight processing and traffic volume requirements.
- Real-Time Protocol (RTP) Header
  - Makes up for two UDP limitations.
  - It adds sequence numbers to UDP.
  - It adds a time stamp to UDP when its codec octets should be played back to avoid jitter.

Copyright © 2015 Pearson Education, Ltd.+

© 2013 Pearson

Terminal-Host v Client/Server Processing

E-Mail

Voice over IP

HTTP and HTML

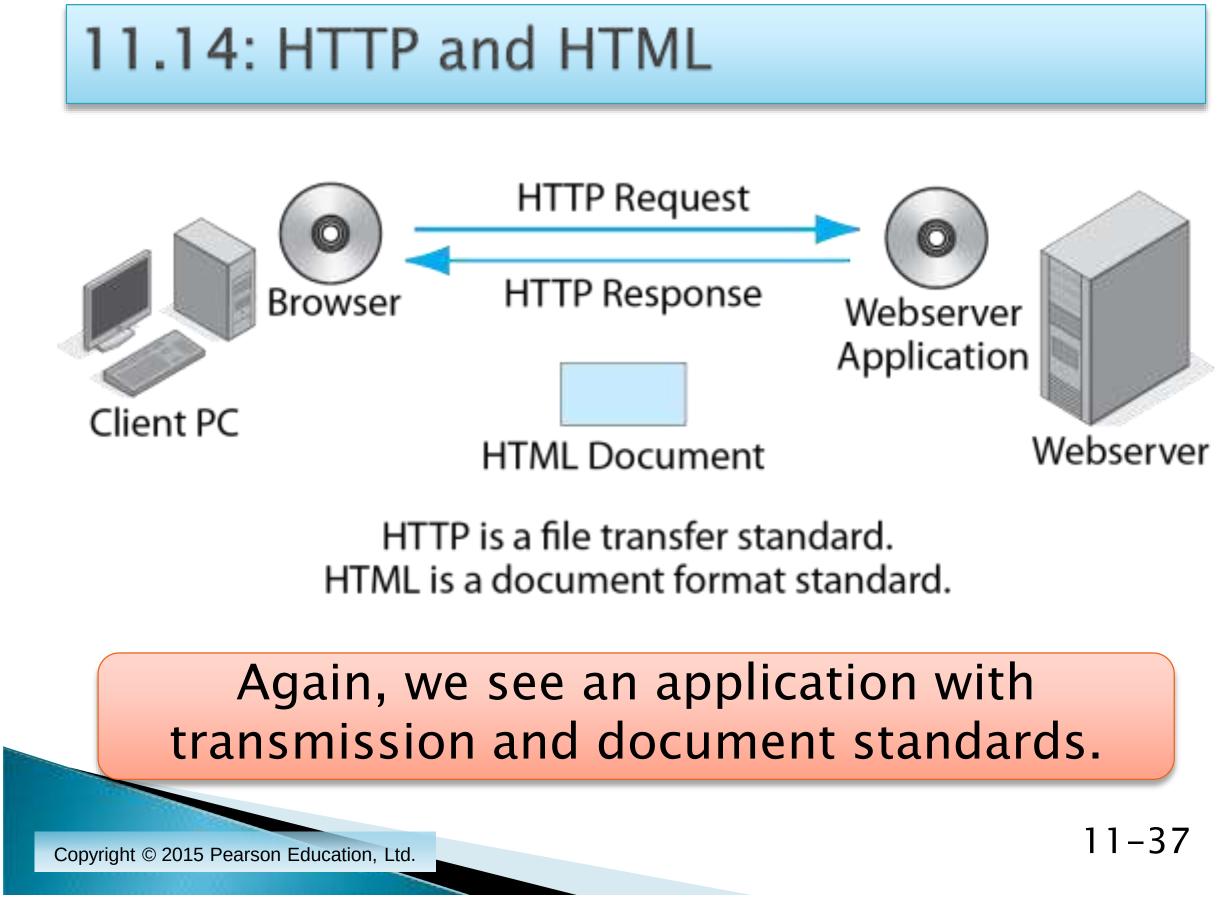
Peer-to-Peer (P2P) Applications

Copyright © 2015 Pearson Education, Ltd.

WWW is an application of the Internet

- The World Wide Web (WWW) was invented by Sir Tim Berners-Lee in 1989.
- The protocol of the Web is HTTP (Hypertext Transfer Protocol).
- The message syntax of the Web is HTML (Hypertext Markup Language).

© 2013 Pearson



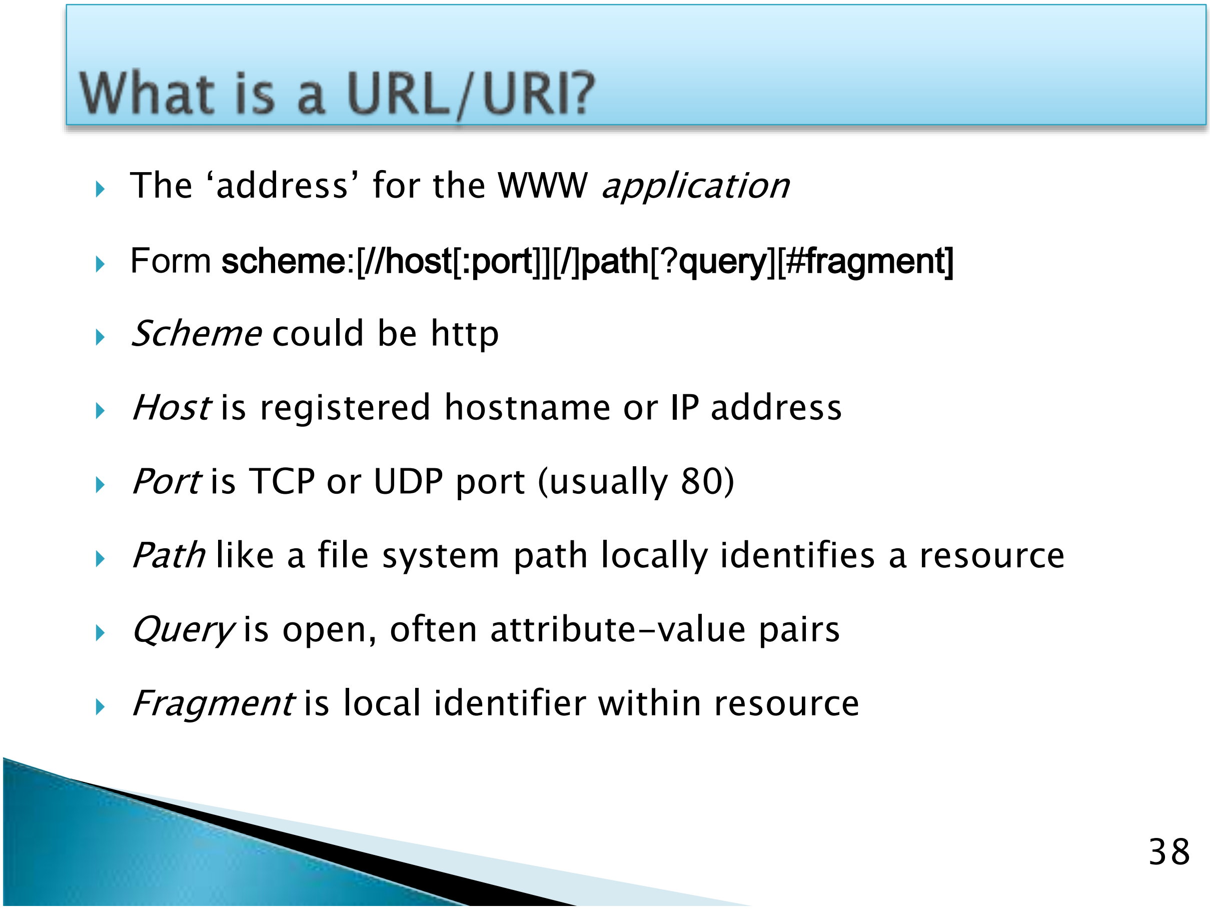
11.14: HTTP and HTML

HTTP is a file transfer standard.
HTML is a document format standard.

Again, we see an application with transmission and document standards.

Copyright © 2015 Pearson Education, Ltd.

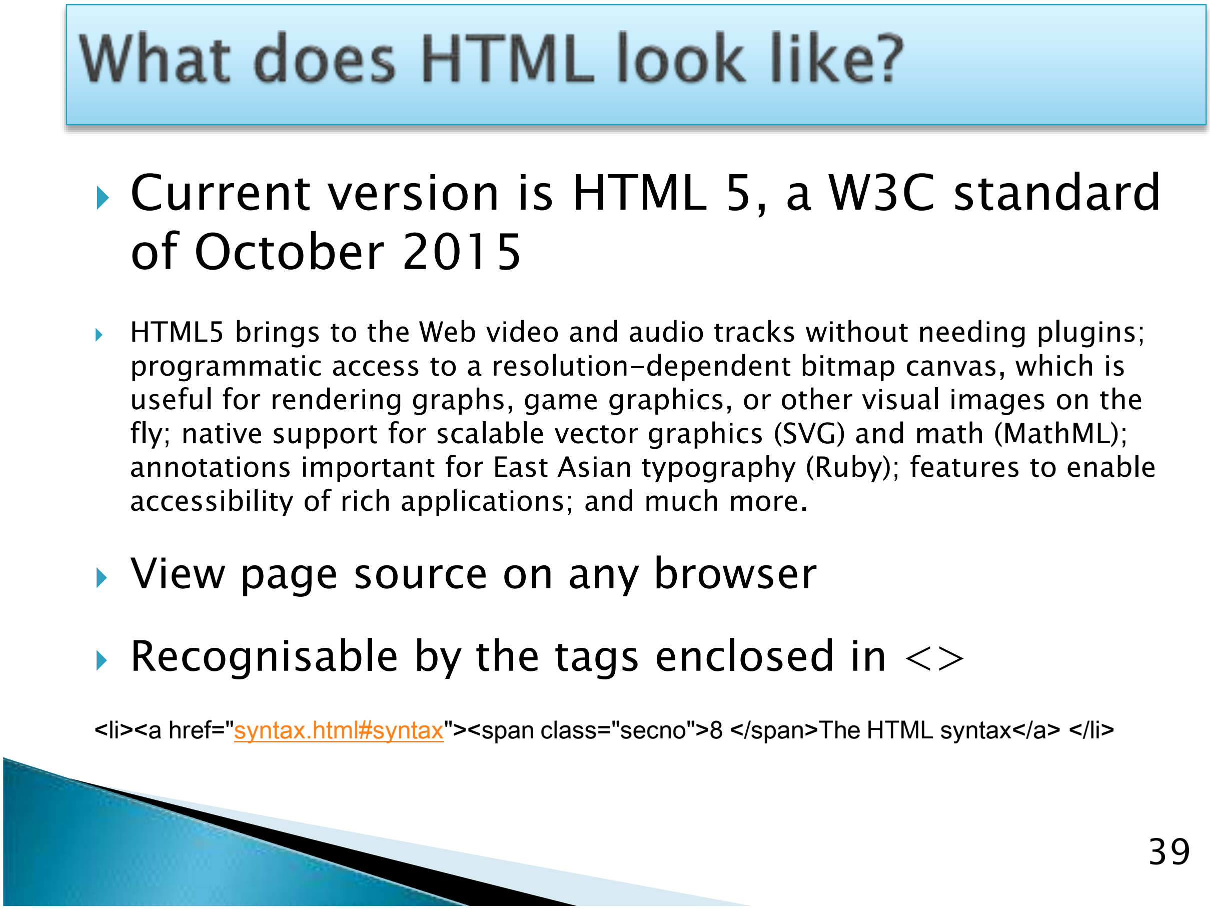
© 2013 Pearson



What is a URL/URI?

- The ‘address’ for the WWW *application*
- Form **scheme**:[**//host[:port]**][/]**path**[?**query**][**#fragment]**
- *Scheme* could be http
- *Host* is registered hostname or IP address
- *Port* is TCP or UDP port (usually 80)
- *Path* like a file system path locally identifies a resource
- *Query* is open, often attribute-value pairs
- *Fragment* is local identifier within resource

© 2013 Pearson

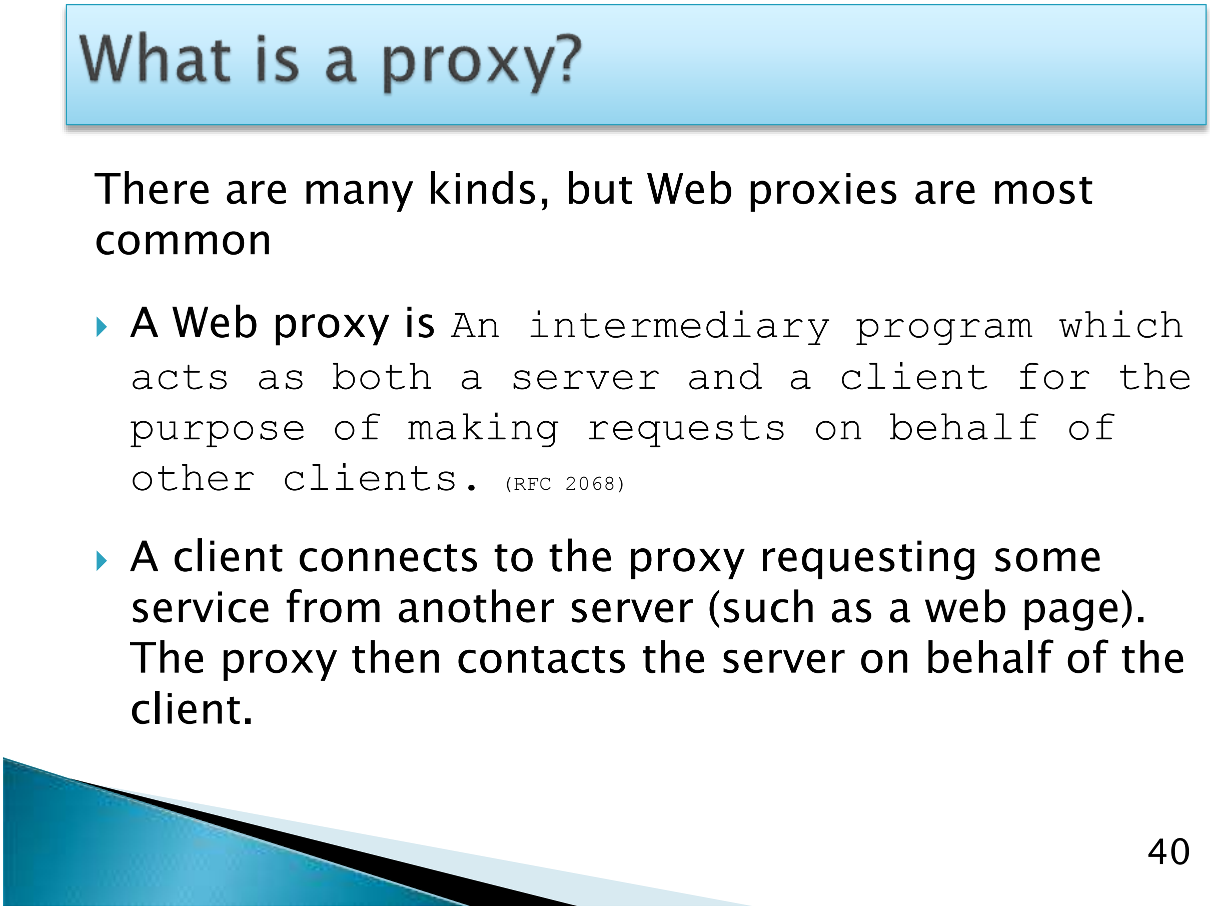


What does HTML look like?

- Current version is HTML 5, a W3C standard of October 2015
- HTML5 brings to the Web video and audio tracks without needing plugins; programmatic access to a resolution-dependent bitmap canvas, which is useful for rendering graphs, game graphics, or other visual images on the fly; native support for scalable vector graphics (SVG) and math (MathML); annotations important for East Asian typography (Ruby); features to enable accessibility of rich applications; and much more.
- View page source on any browser
- Recognisable by the tags enclosed in <>

```
<li><a href="syntax.html#syntax"><span class="secno">8 </span>The HTML syntax</a> </li>
```

© 2013 Pearson

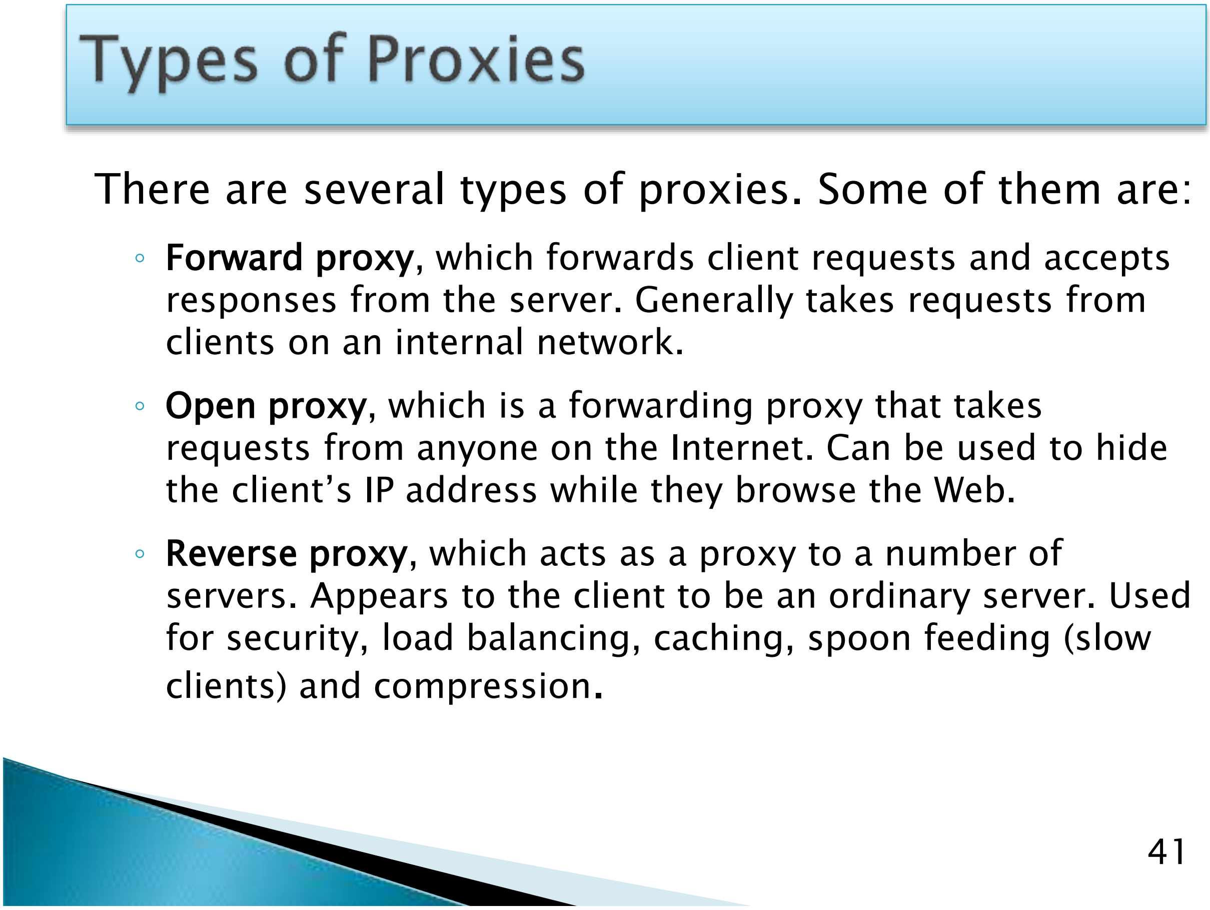


What is a proxy?

There are many kinds, but Web proxies are most common

- A Web proxy is An intermediary program which acts as both a server and a client for the purpose of making requests on behalf of other clients. (RFC 2068)
- A client connects to the proxy requesting some service from another server (such as a web page). The proxy then contacts the server on behalf of the client.

© 2013 Pearson

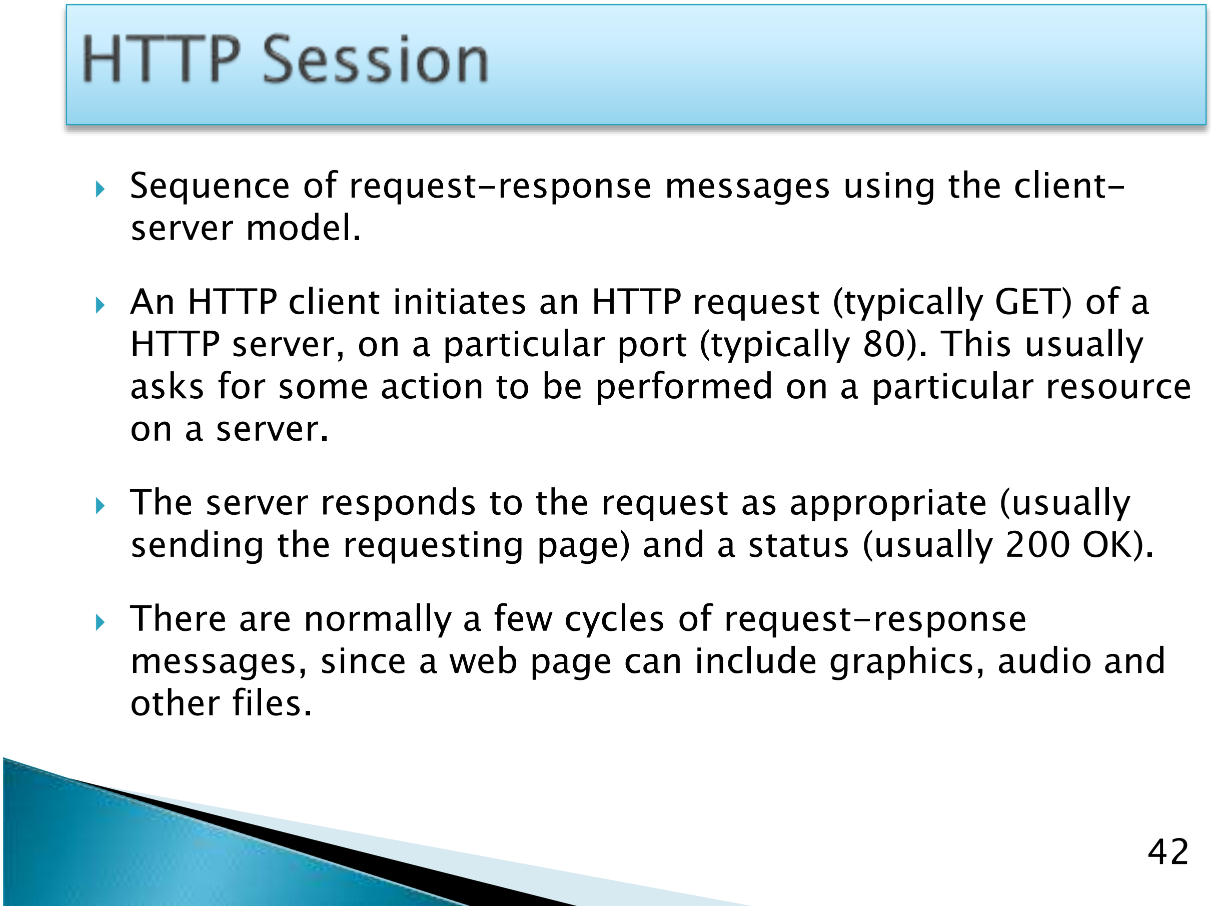


Types of Proxies

There are several types of proxies. Some of them are:

- **Forward proxy**, which forwards client requests and accepts responses from the server. Generally takes requests from clients on an internal network.
- **Open proxy**, which is a forwarding proxy that takes requests from anyone on the Internet. Can be used to hide the client’s IP address while they browse the Web.
- **Reverse proxy**, which acts as a proxy to a number of servers. Appears to the client to be an ordinary server. Used for security, load balancing, caching, spoon feeding (slow clients) and compression**.**

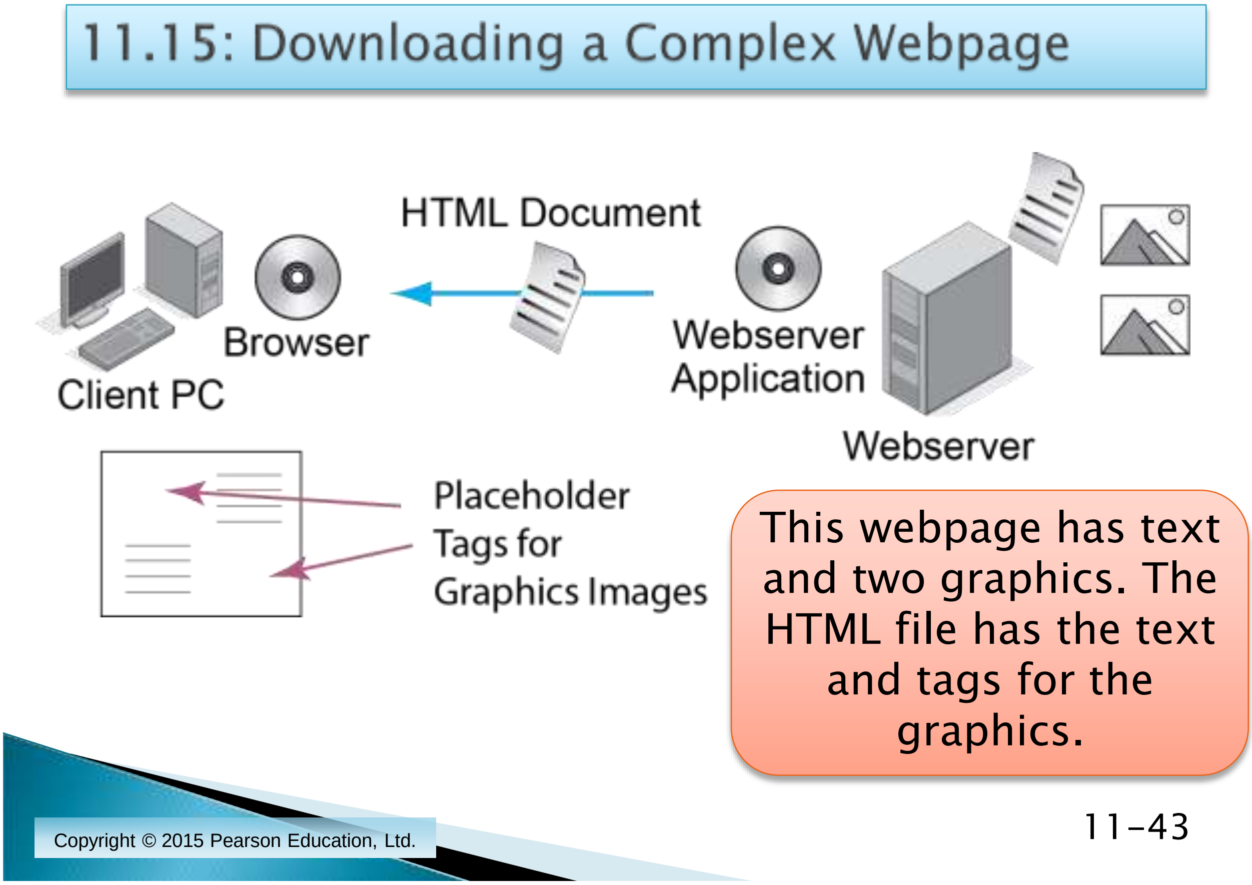
© 2013 Pearson



HTTP Session

- Sequence of request-response messages using the client-server model.
- An HTTP client initiates an HTTP request (typically GET) of a HTTP server, on a particular port (typically 80). This usually asks for some action to be performed on a particular resource on a server.
- The server responds to the request as appropriate (usually sending the requesting page) and a status (usually 200 OK).
- There are normally a few cycles of request-response messages, since a web page can include graphics, audio and other files.

© 2013 Pearson



11.15: Downloading a Complex Webpage

HTML Document

Client PC

Browser

Webserver Application

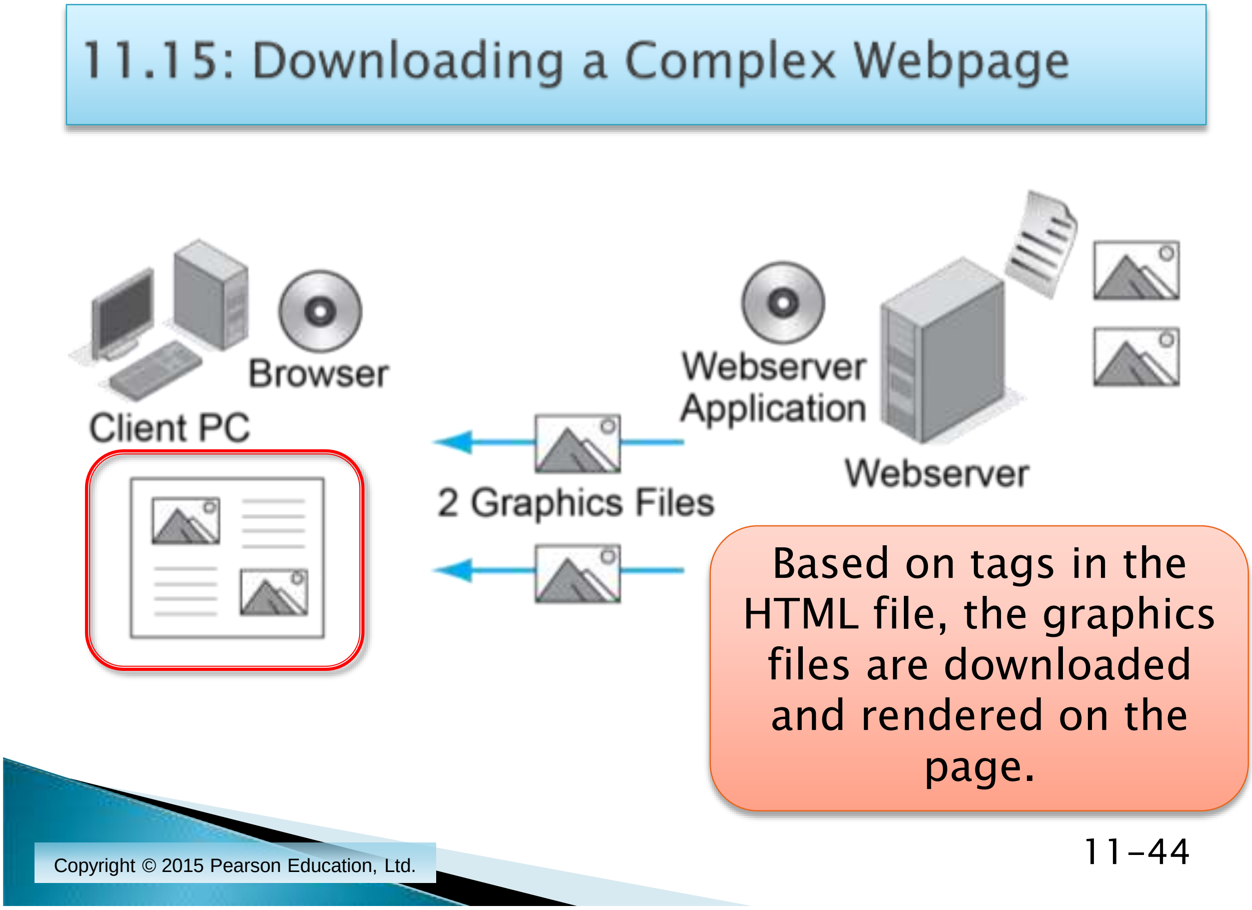
Webserver

Placeholder Tags for Graphics Images

This webpage has text and two graphics. The HTML file has the text and tags for the graphics.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

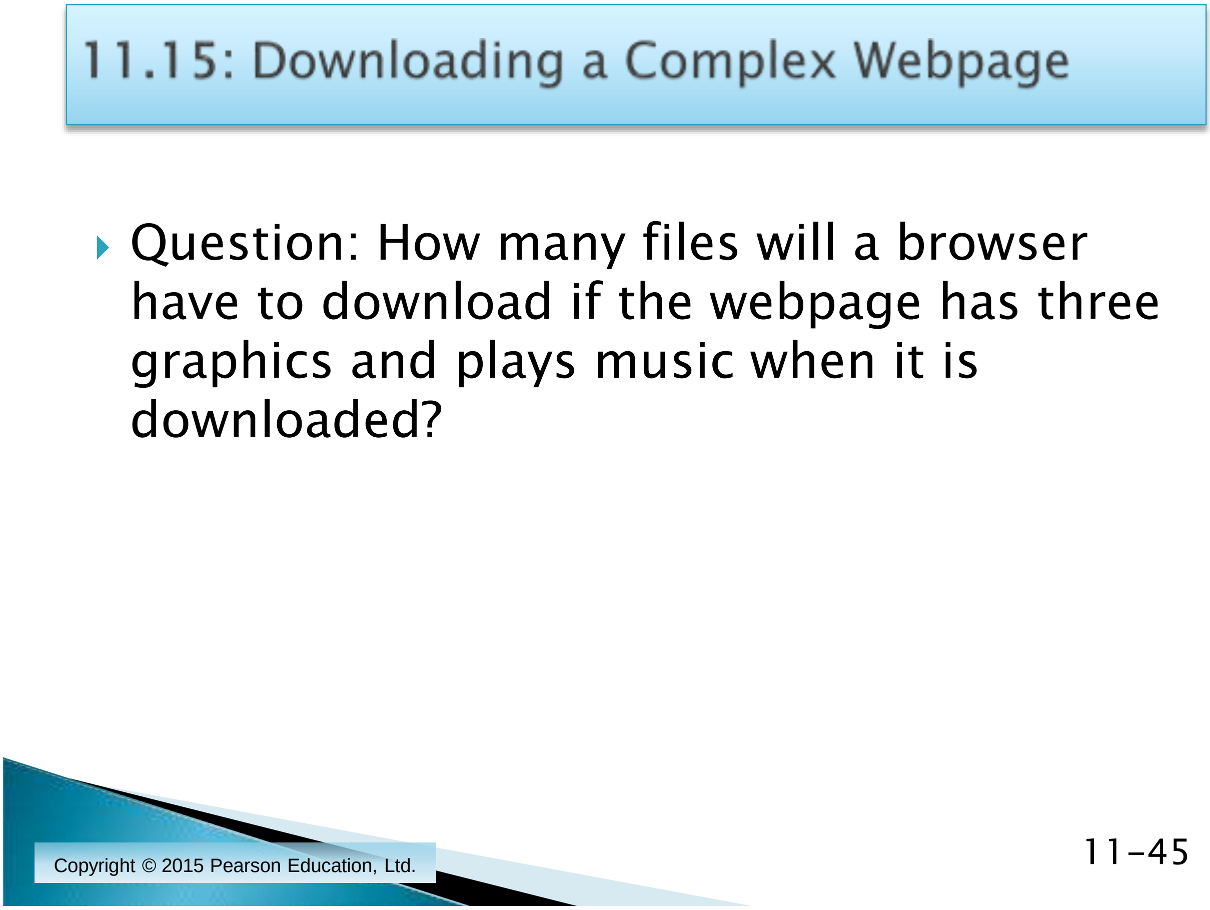


11.15: Downloading a Complex Webpage

Based on tags in the HTML file, the graphics files are downloaded and rendered on the page.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

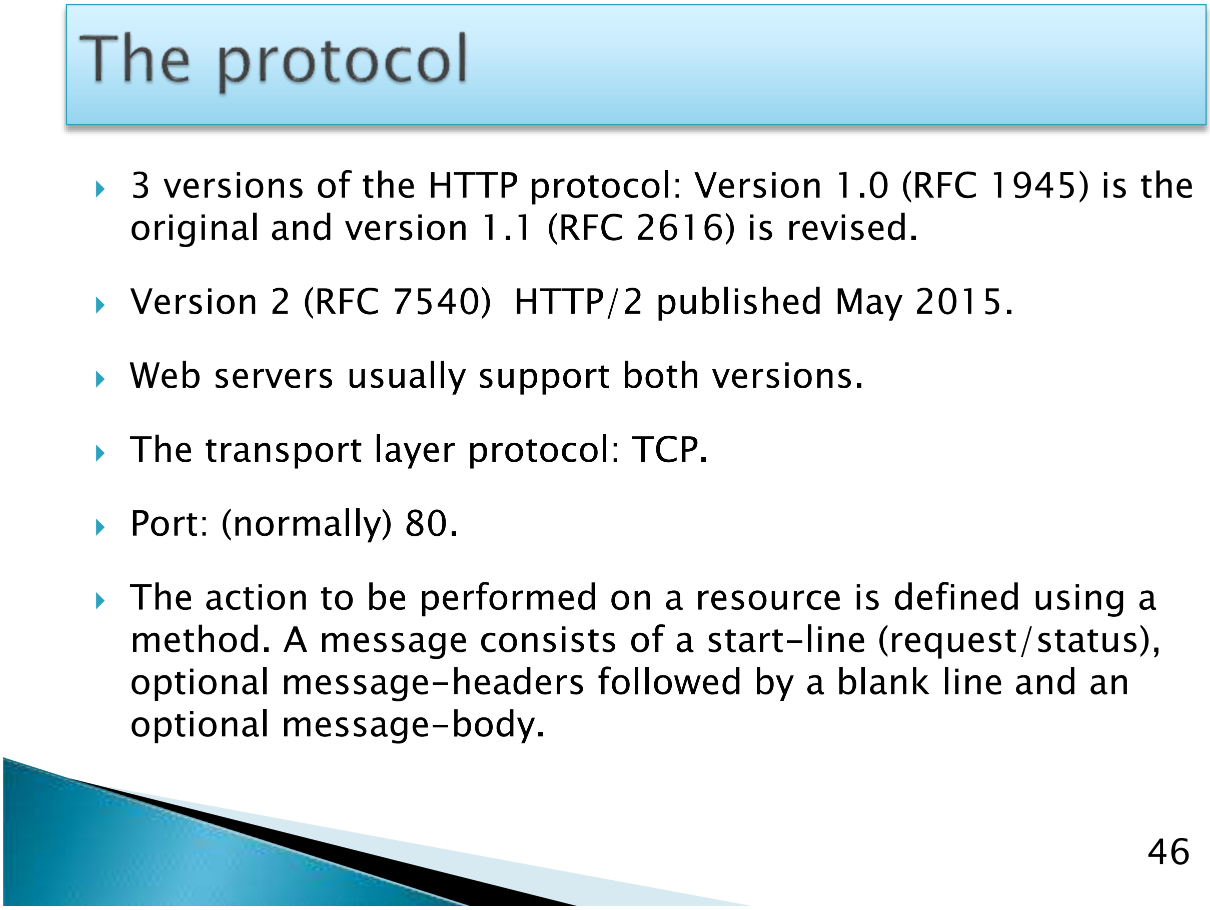


11.15: Downloading a Complex Webpage

- Question: How many files will a browser have to download if the webpage has three graphics and plays music when it is downloaded?

Copyright © 2015 Pearson Education, Ltd.

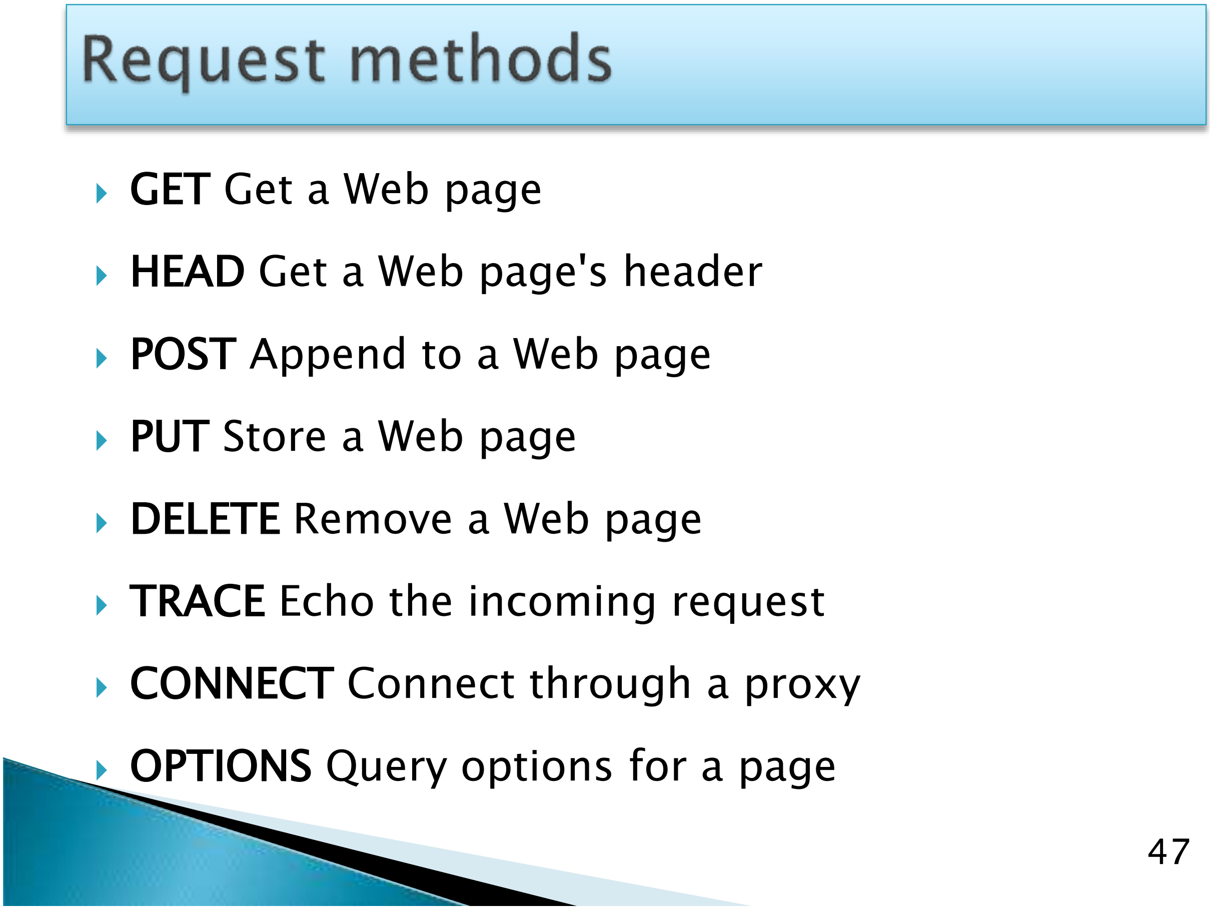
© 2013 Pearson



The protocol

- 3 versions of the HTTP protocol: Version 1.0 (RFC 1945) is the original and version 1.1 (RFC 2616) is revised.
- Version 2 (RFC 7540) HTTP/2 published May 2015.
- Web servers usually support both versions.
- The transport layer protocol: TCP.
- Port: (normally) 80.
- The action to be performed on a resource is defined using a method. A message consists of a start-line (request/status), optional message-headers followed by a blank line and an optional message-body.

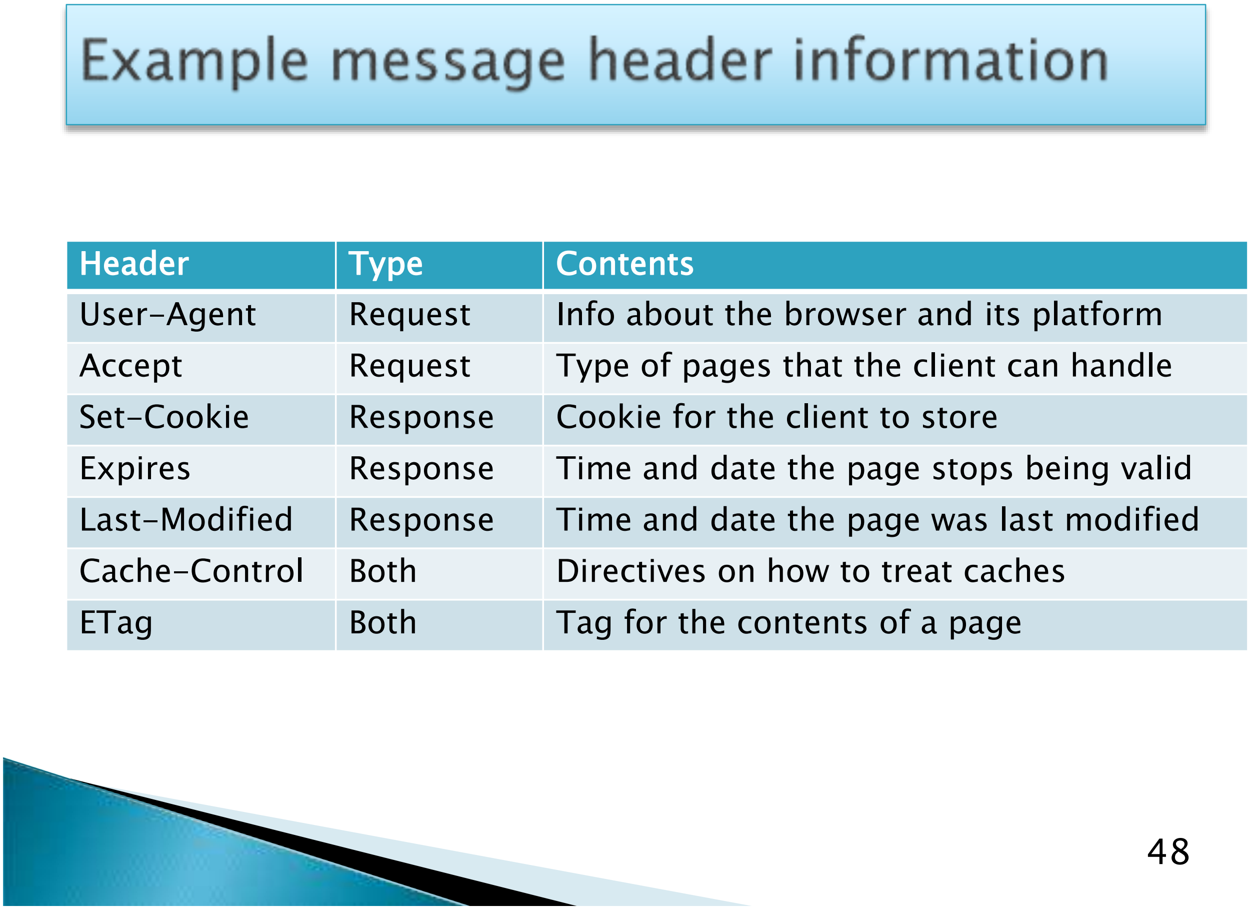
© 2013 Pearson



Request methods

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

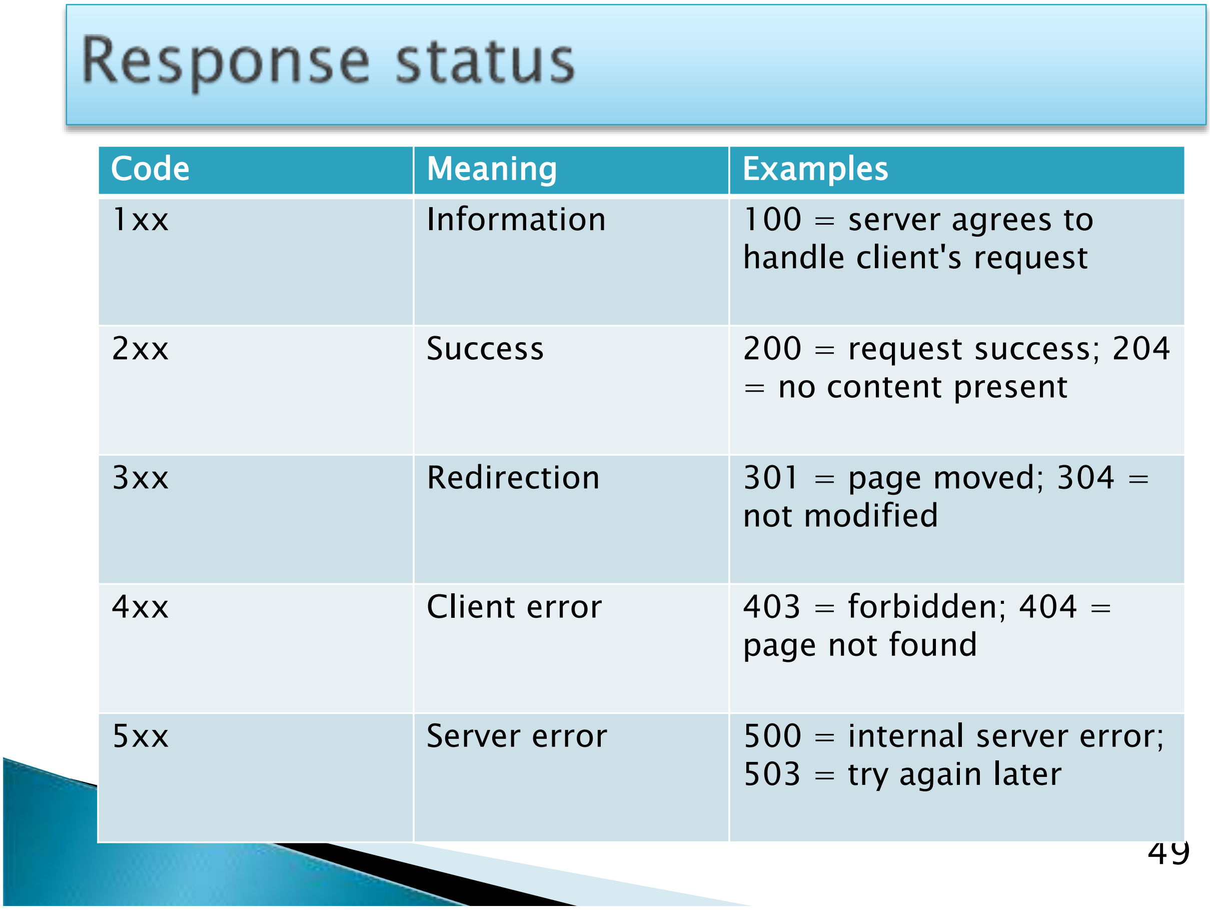
© 2013 Pearson



Example message header information

Header	Type	Contents
User-Agent	Request	Info about the browser and its platform
Accept	Request	Type of pages that the client can handle
Set-Cookie	Response	Cookie for the client to store
Expires	Response	Time and date the page stops being valid
Last-Modified	Response	Time and date the page was last modified
Cache-Control	Both	Directives on how to treat caches
ETag	Both	Tag for the contents of a page

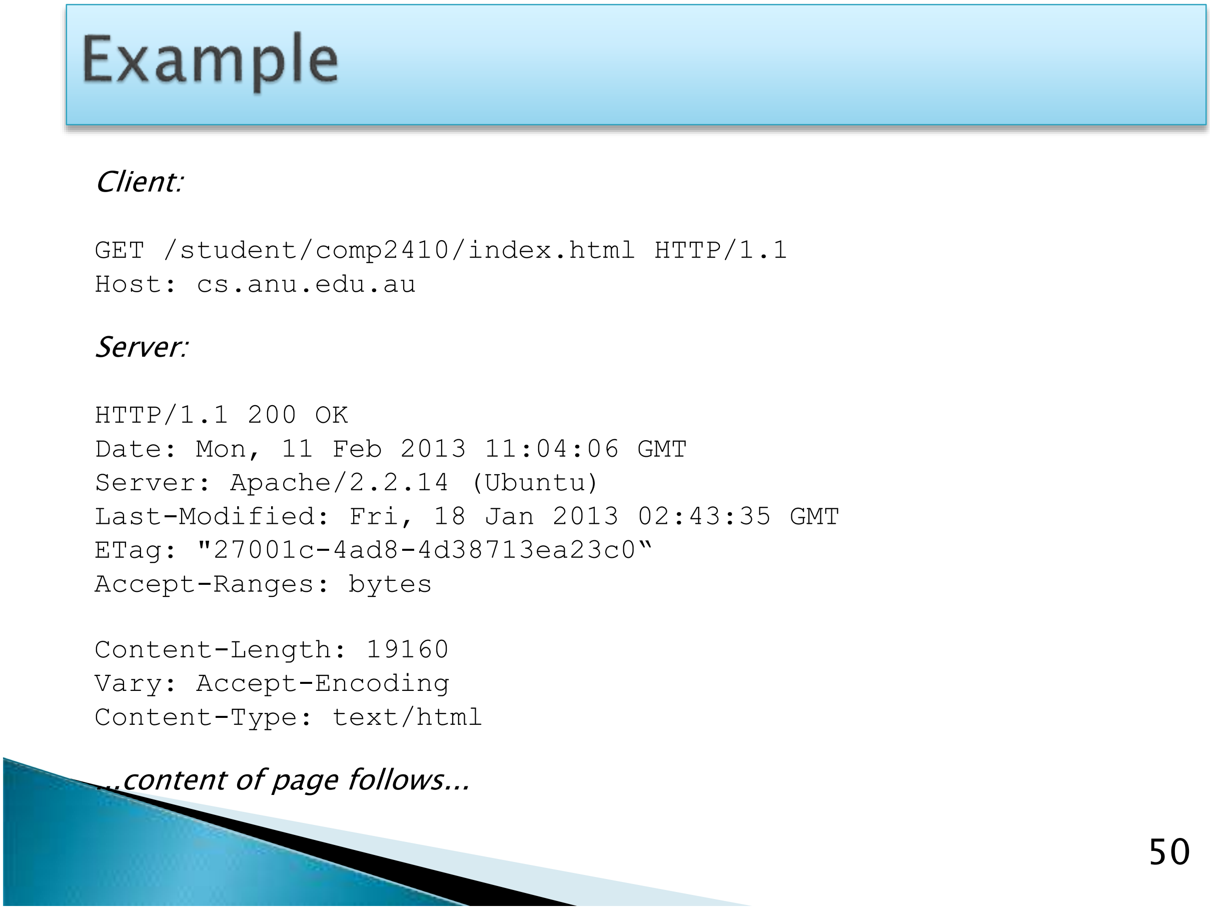
© 2013 Pearson



Response status

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

© 2013 Pearson



Example

Client:

```
GET /student/comp2410/index.html HTTP/1.1
Host: cs.anu.edu.au
```

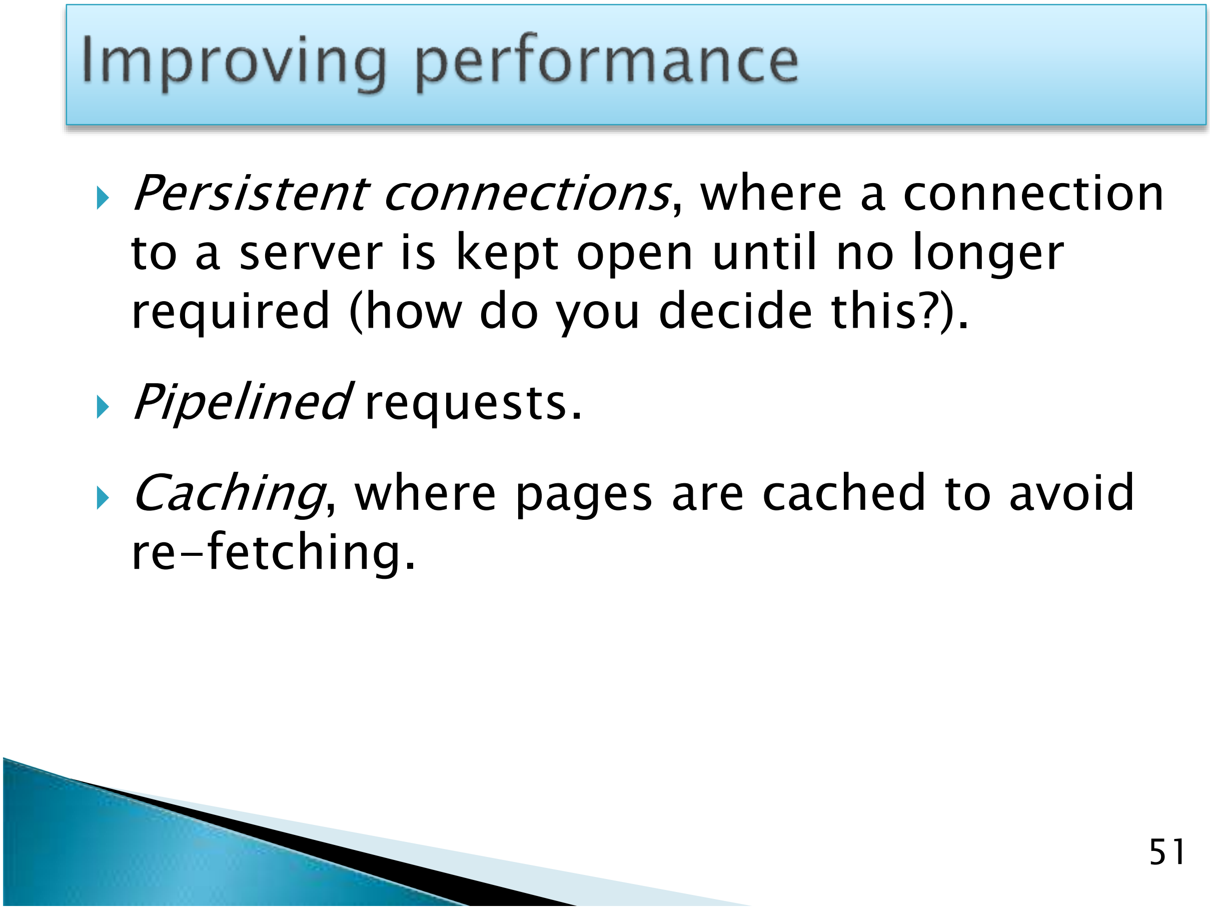
Server:

```
HTTP/1.1 200 OK
Date: Mon, 11 Feb 2013 11:04:06 GMT
Server: Apache/2.2.14 (Ubuntu)
Last-Modified: Fri, 18 Jan 2013 02:43:35 GMT
ETag: "27001c-4ad8-4d38713ea23c0‟
Accept-Ranges: bytes

Content-Length: 19160
Vary: Accept-Encoding
Content-Type: text/html
```

...content of page follows...

© 2013 Pearson

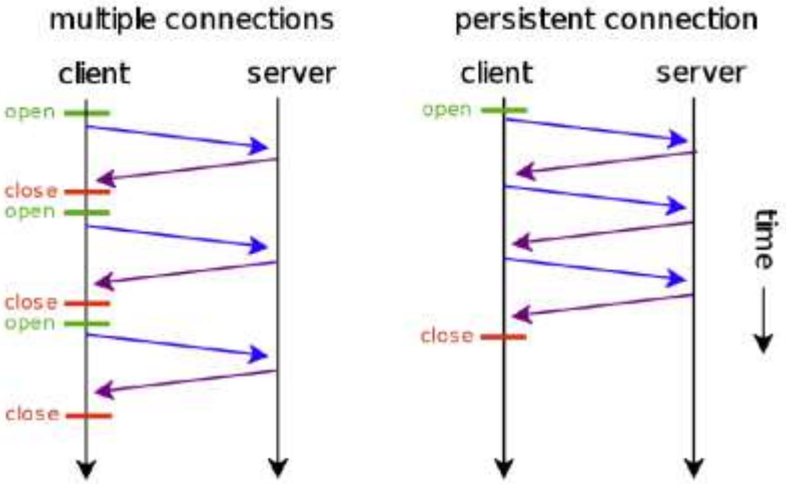


Improving performance

- *Persistent connections*, where a connection to a server is kept open until no longer required (how do you decide this?).
- *Pipelined* requests.
- *Caching*, where pages are cached to avoid re-fetching.

© 2013 Pearson

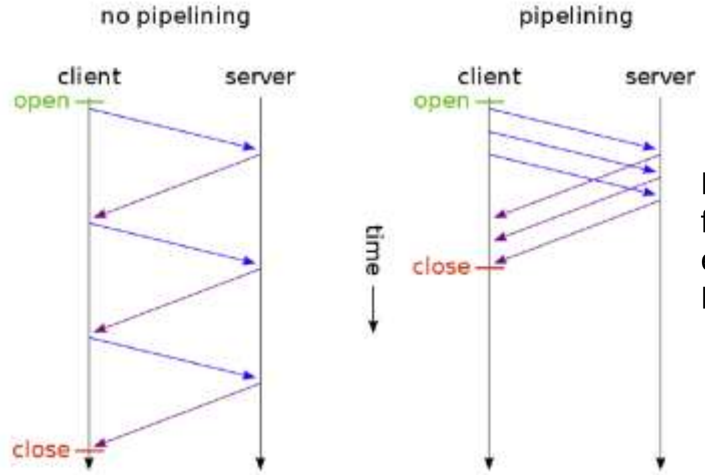
Persistent connection



Wikimedia Commons

© 2013 Pearson

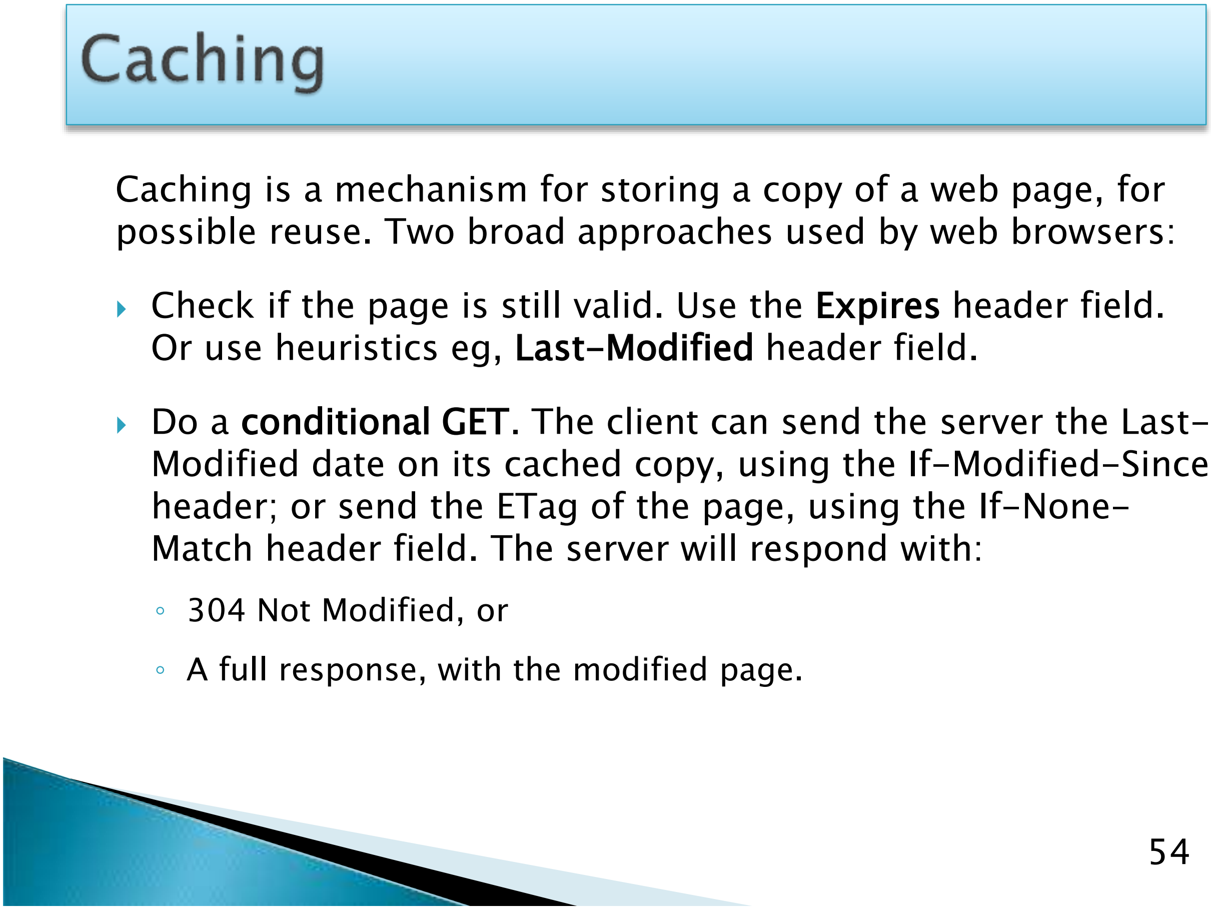
Pipelining



Revised, formalised and emphasised in HTTP/2

By Mwhitlock (Own work) [Public domain], via Wikimedia Commons

© 2013 Pearson

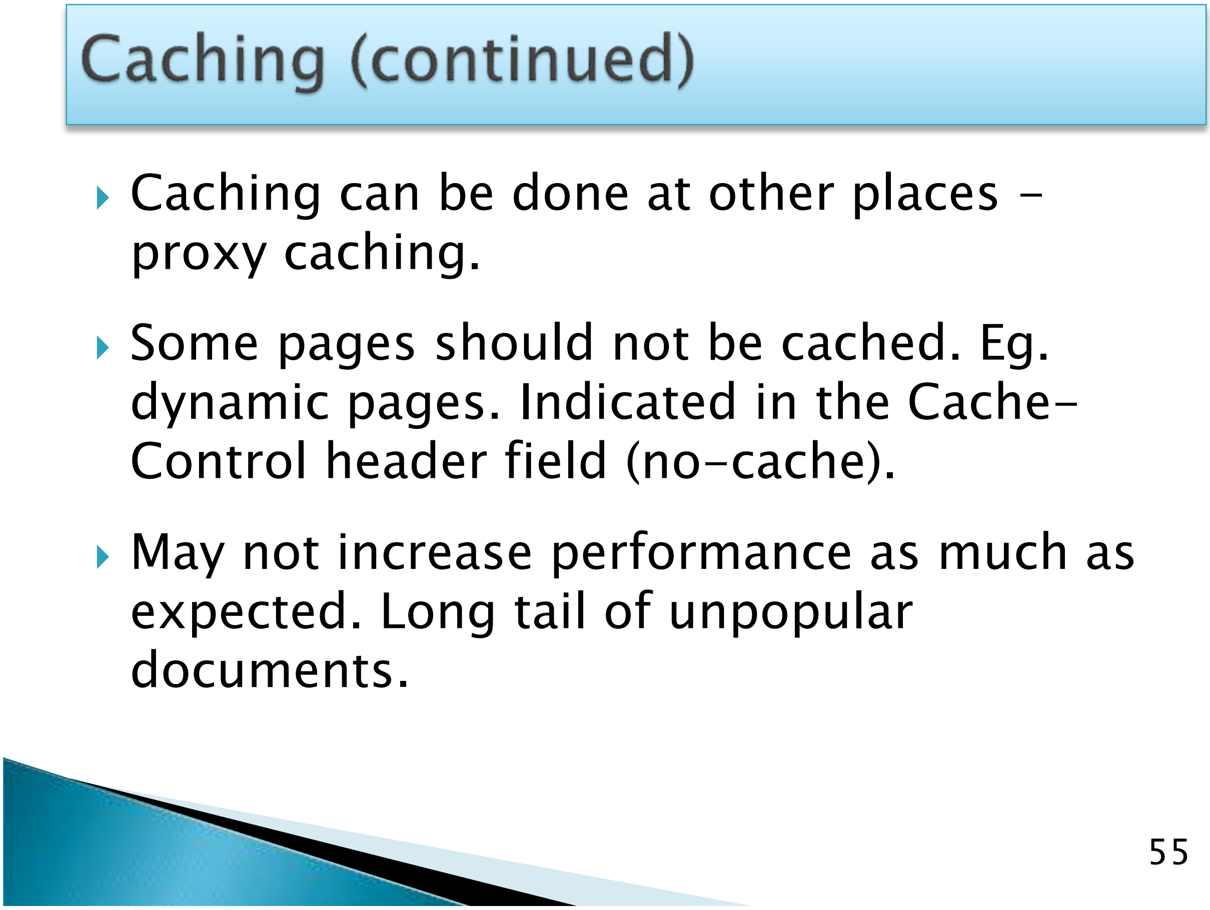


Caching

Caching is a mechanism for storing a copy of a web page, for possible reuse. Two broad approaches used by web browsers:

- Check if the page is still valid. Use the **Expires** header field. Or use heuristics eg, **Last-Modified** header field.
- Do a **conditional GET**. The client can send the server the Last-Modified date on its cached copy, using the If-Modified-Since header; or send the ETag of the page, using the If-None-Match header field. The server will respond with:
  - 304 Not Modified, or
  - A full response, with the modified page.

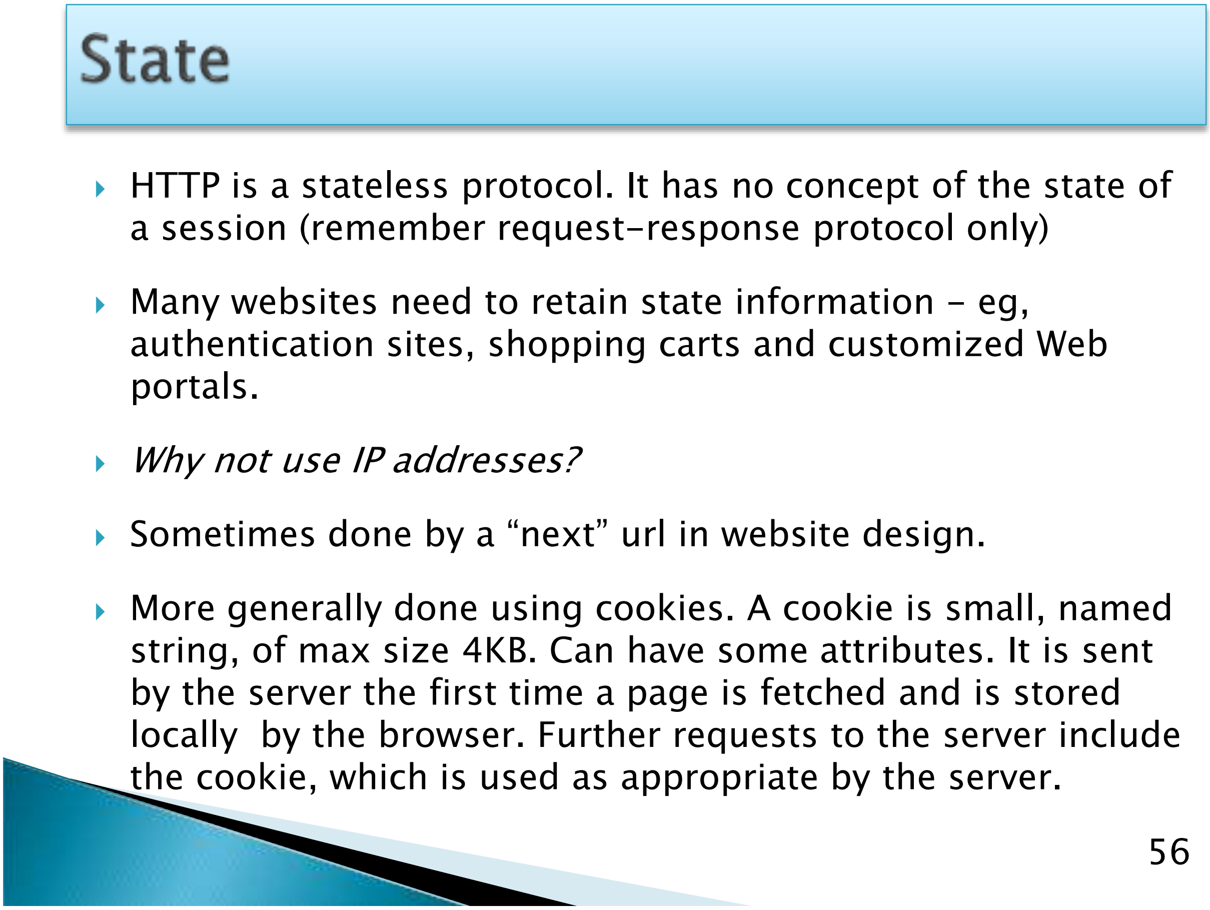
© 2013 Pearson



Caching (continued)

- Caching can be done at other places – proxy caching.
- Some pages should not be cached. Eg. dynamic pages. Indicated in the Cache-Control header field (no-cache).
- May not increase performance as much as expected. Long tail of unpopular documents.

© 2013 Pearson

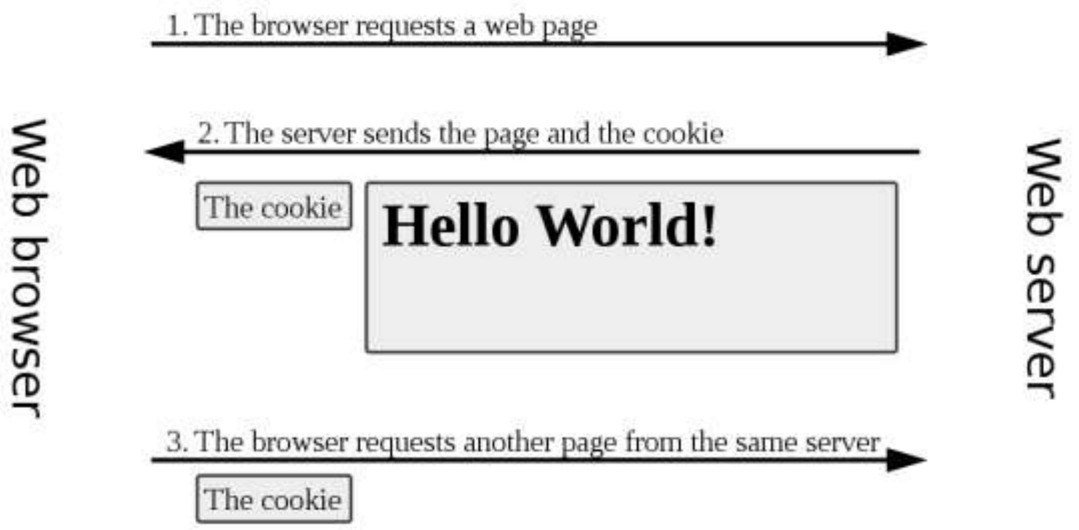


State

- HTTP is a stateless protocol. It has no concept of the state of a session (remember request-response protocol only)
- Many websites need to retain state information – eg, authentication sites, shopping carts and customized Web portals.
- *Why not use IP addresses?*
- Sometimes done by a “next” url in website design.
- More generally done using cookies. A cookie is small, named string, of max size 4KB. Can have some attributes. It is sent by the server the first time a page is fetched and is stored locally by the browser. Further requests to the server include the cookie, which is used as appropriate by the server.

© 2013 Pearson

Cookie example



Web browser

1. The browser requests a web page

2. The server sends the page and the cookie

The cookie

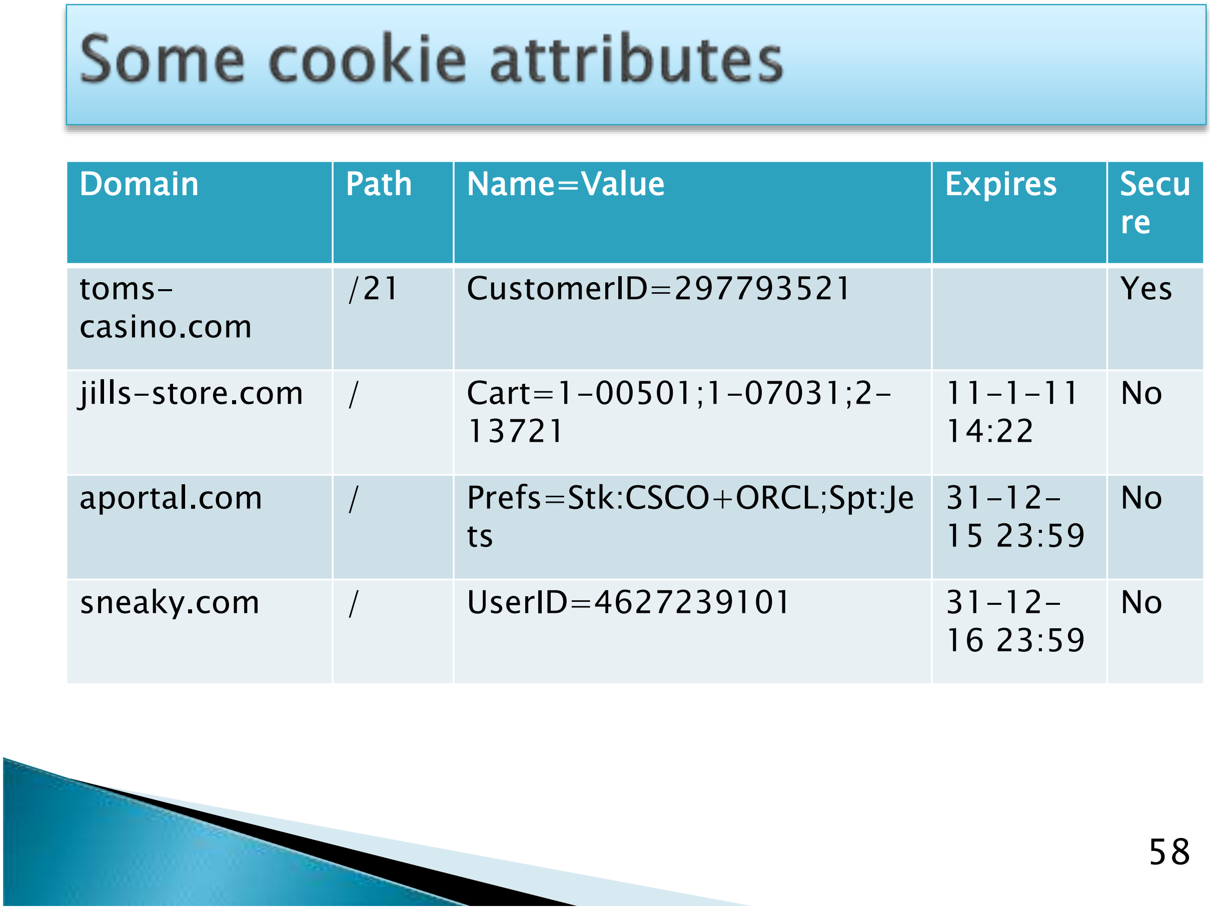
Hello World!

3. The browser requests another page from the same server

The cookie

Web server

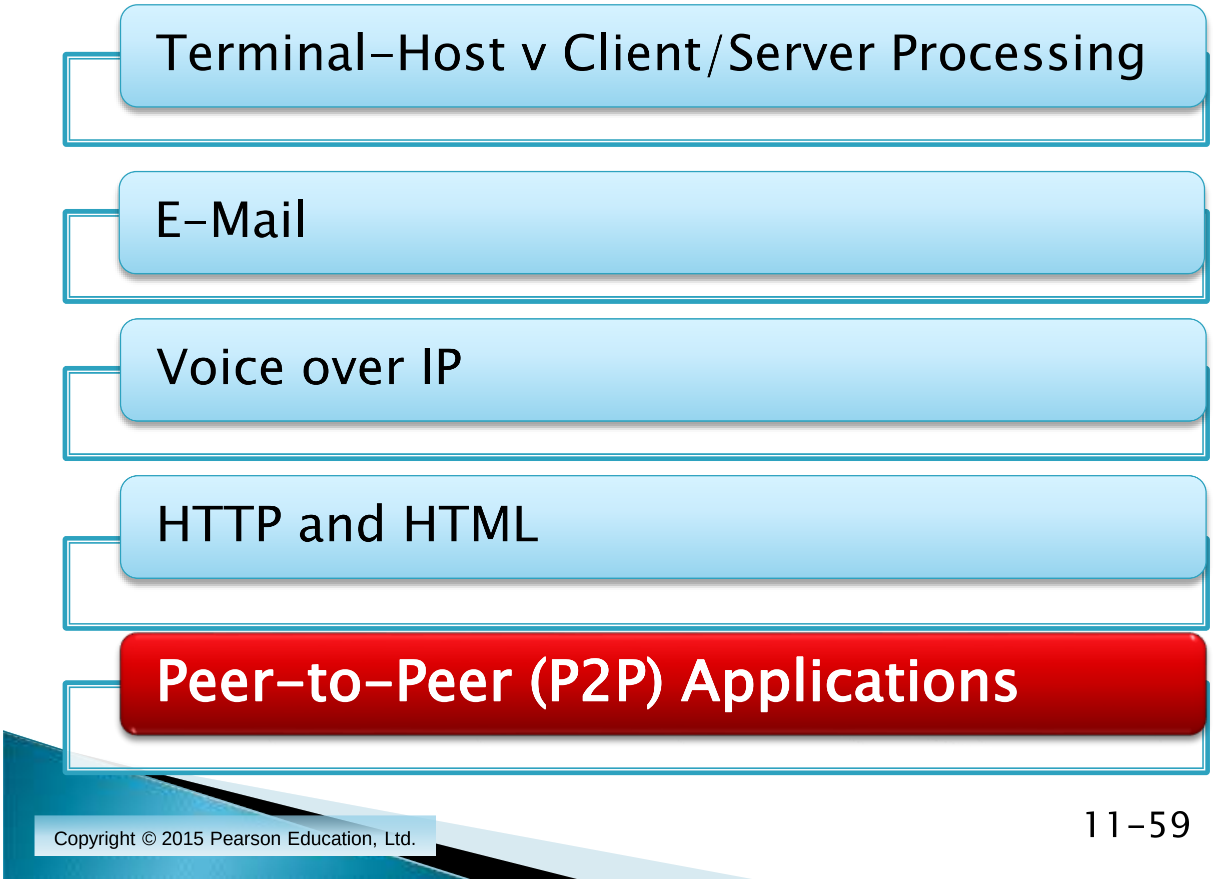
© 2013 Pearson



Some cookie attributes

Domain	Path	Name=Value	Expires	Secure
toms-casino.com	/21	CustomerID=297793521		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	31-12-15 23:59	No
sneaky.com	/	UserID=4627239101	31-12-16 23:59	No

© 2013 Pearson



Terminal–Host v Client/Server Processing

E–Mail

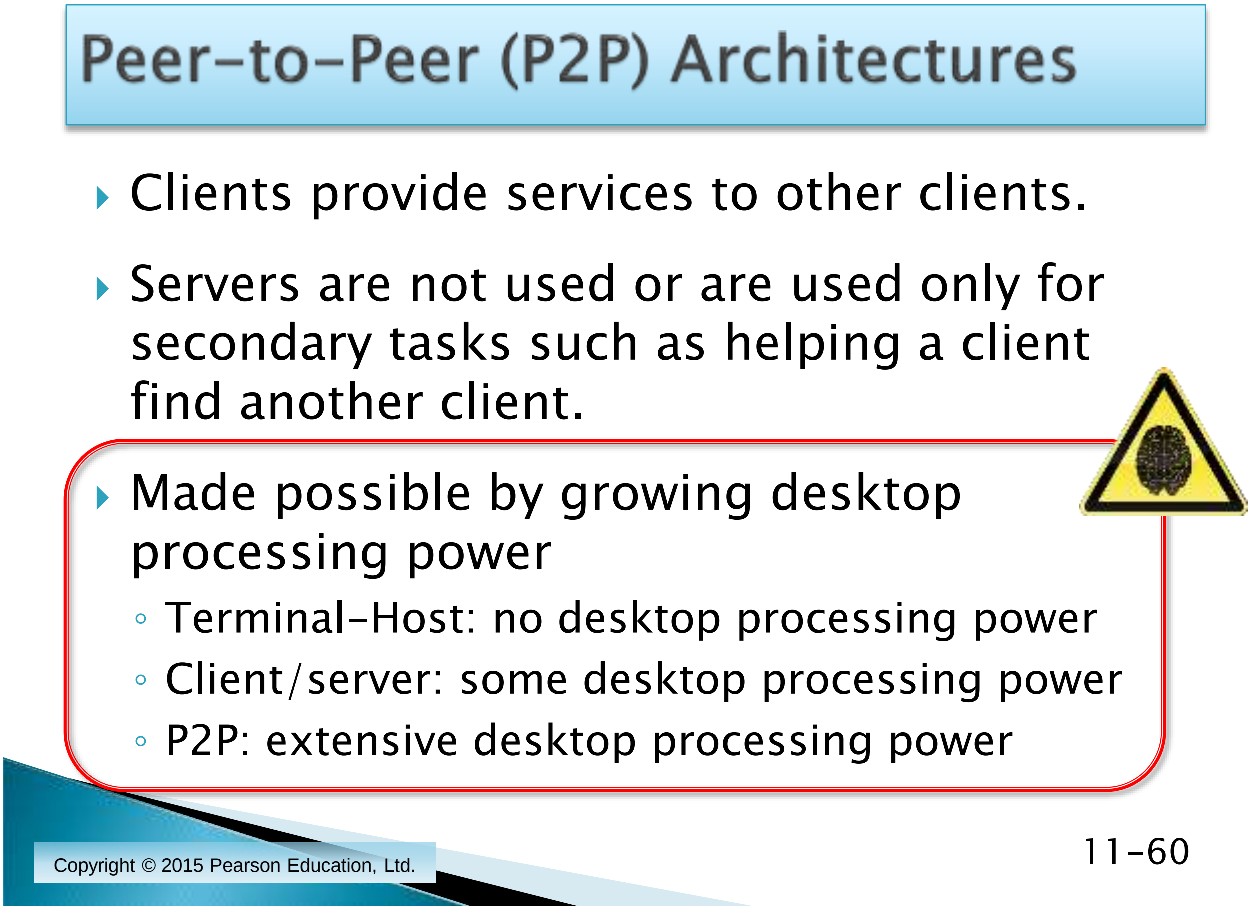
Voice over IP

HTTP and HTML

Peer–to–Peer (P2P) Applications

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

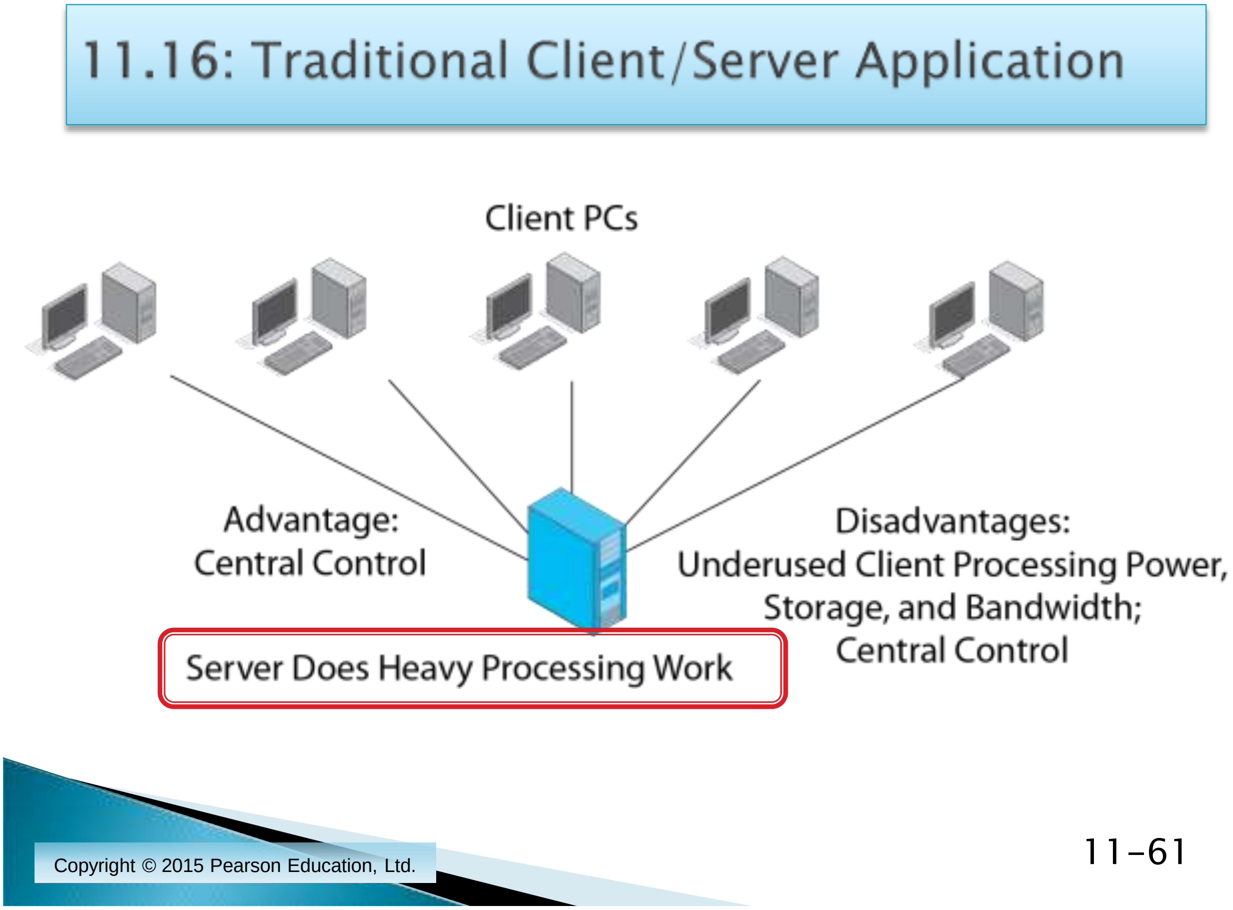


Peer-to-Peer (P2P) Architectures

- Clients provide services to other clients.
- Servers are not used or are used only for secondary tasks such as helping a client find another client.
- Made possible by growing desktop processing power
  - Terminal-Host: no desktop processing power
  - Client/server: some desktop processing power
  - P2P: extensive desktop processing power

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.16: Traditional Client/Server Application

Client PCs

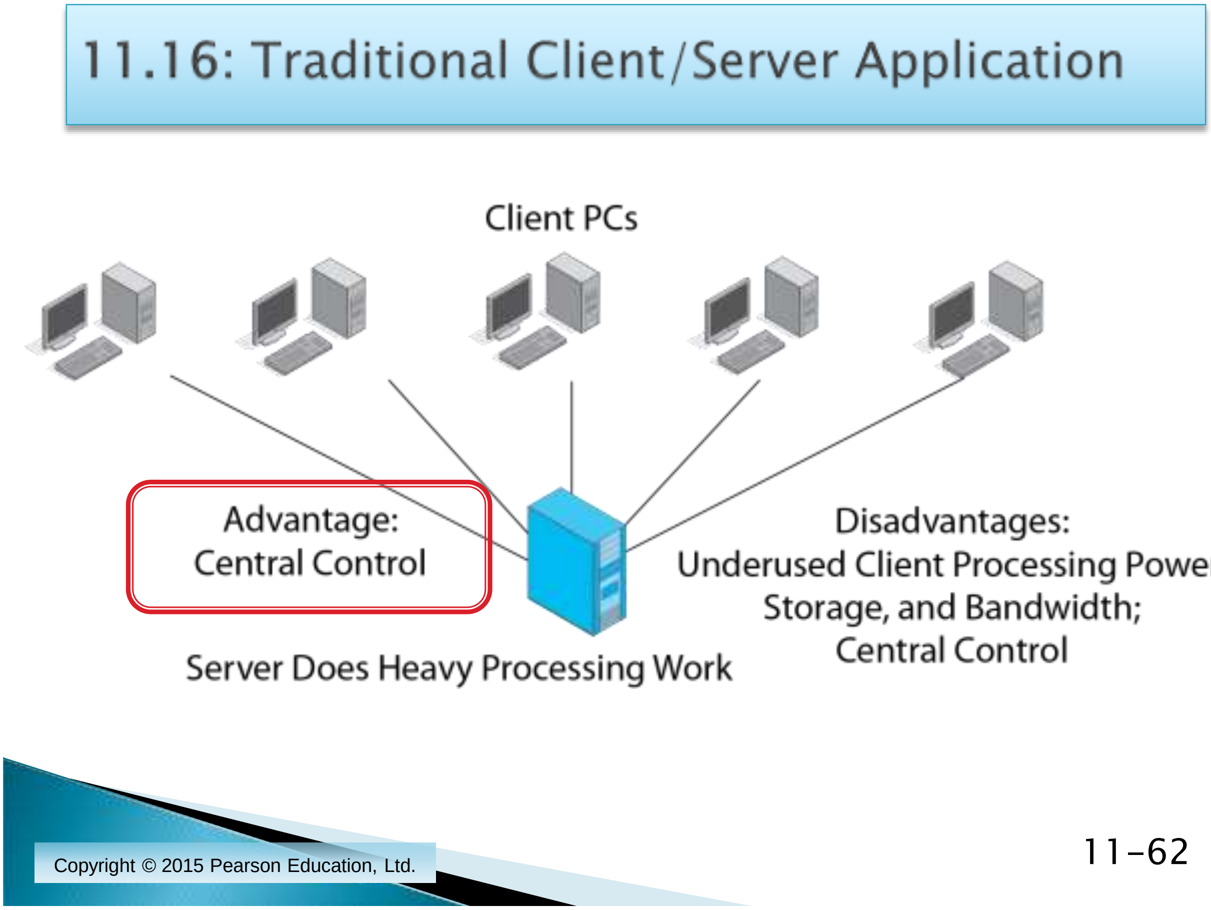
Advantage:
Central Control

Disadvantages:
Underused Client Processing Power,
Storage, and Bandwidth;
Central Control

Server Does Heavy Processing Work

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.16: Traditional Client/Server Application

Client PCs

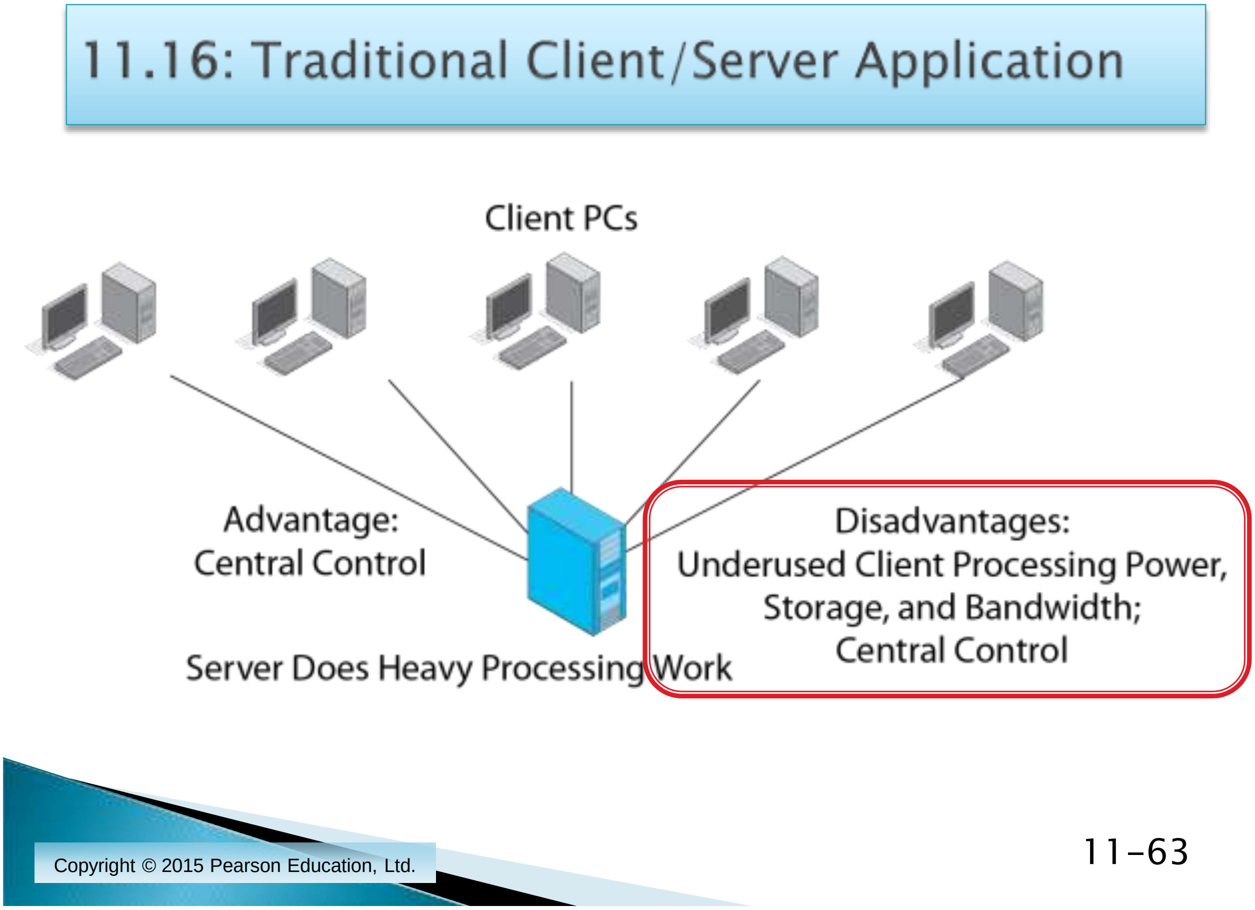
Advantage:
Central Control

Server Does Heavy Processing Work

Disadvantages:
Underused Client Processing Power,
Storage, and Bandwidth;
Central Control

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.16: Traditional Client/Server Application

Client PCs

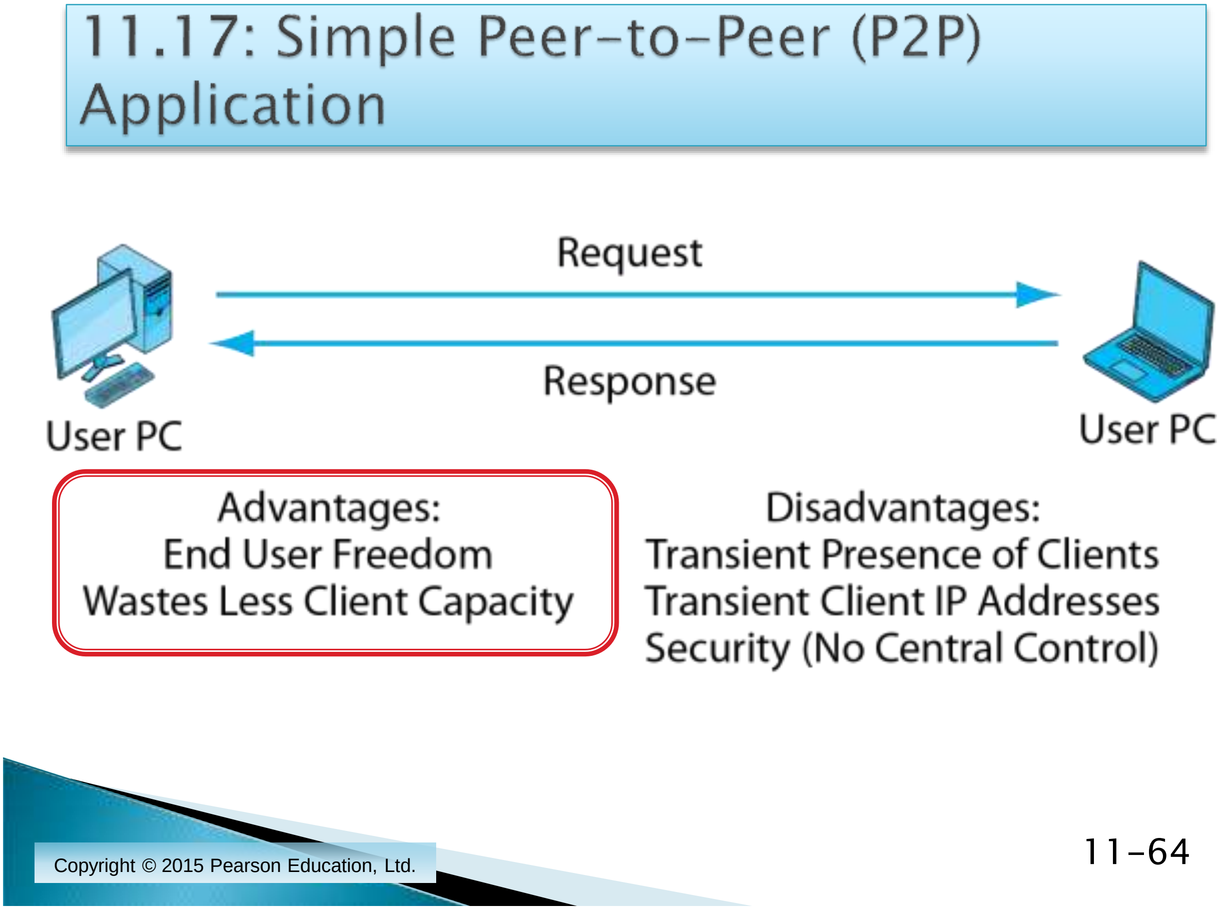
Advantage:
Central Control

Server Does Heavy Processing Work

Disadvantages:
Underused Client Processing Power,
Storage, and Bandwidth;
Central Control

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.17: Simple Peer-to-Peer (P2P) Application

Request

Response

User PC

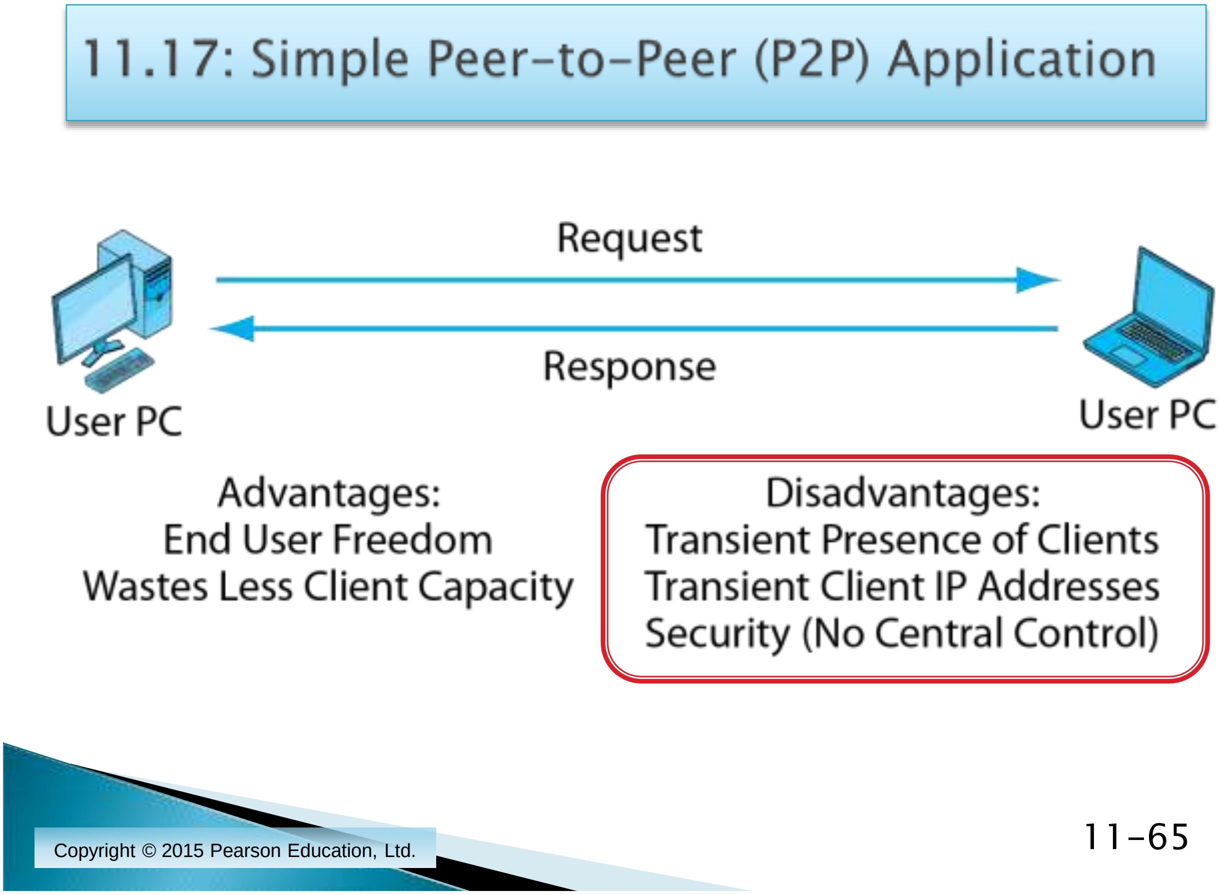
User PC

Advantages:
End User Freedom
Wastes Less Client Capacity

Disadvantages:
Transient Presence of Clients
Transient Client IP Addresses
Security (No Central Control)

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



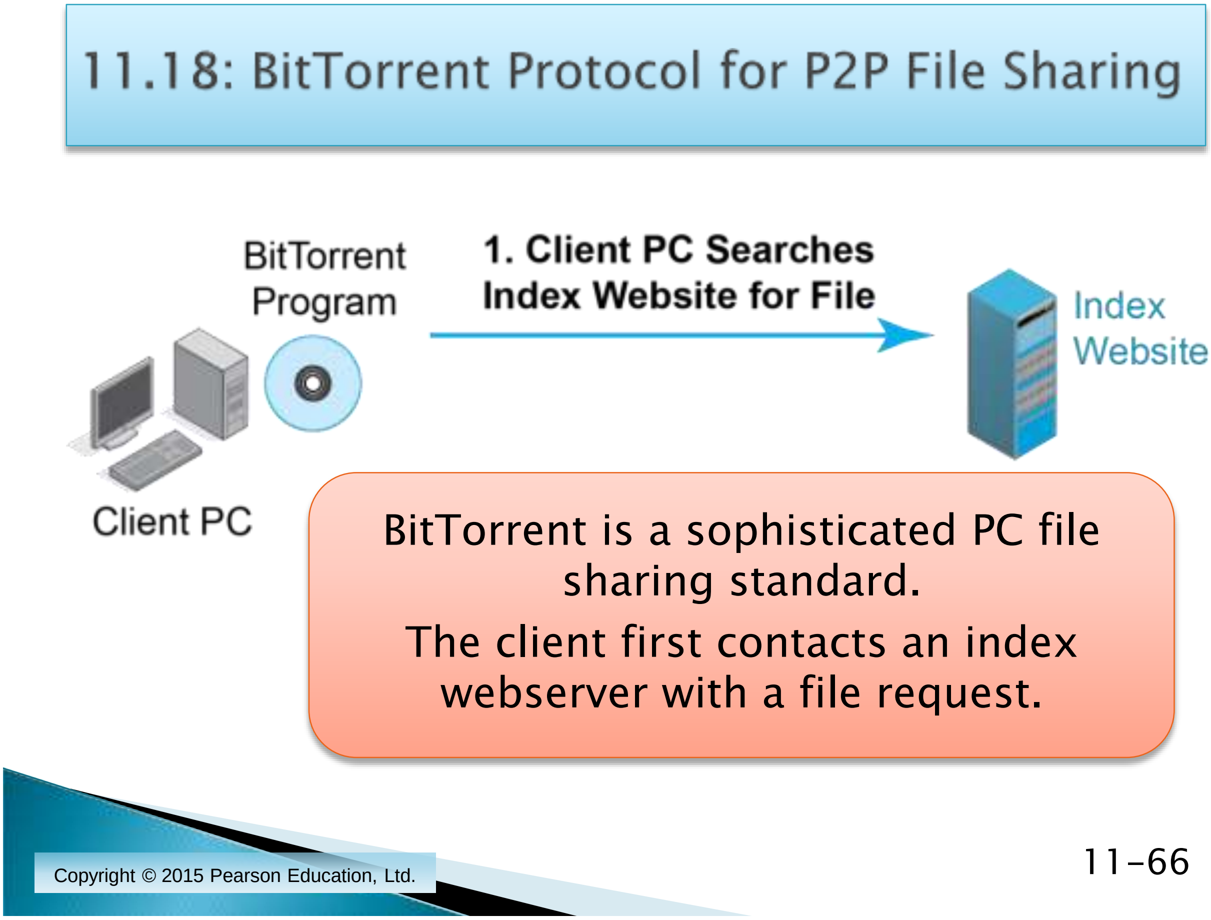
11.17: Simple Peer-to-Peer (P2P) Application

Advantages:
End User Freedom
Wastes Less Client Capacity

Disadvantages:
Transient Presence of Clients
Transient Client IP Addresses
Security (No Central Control)

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



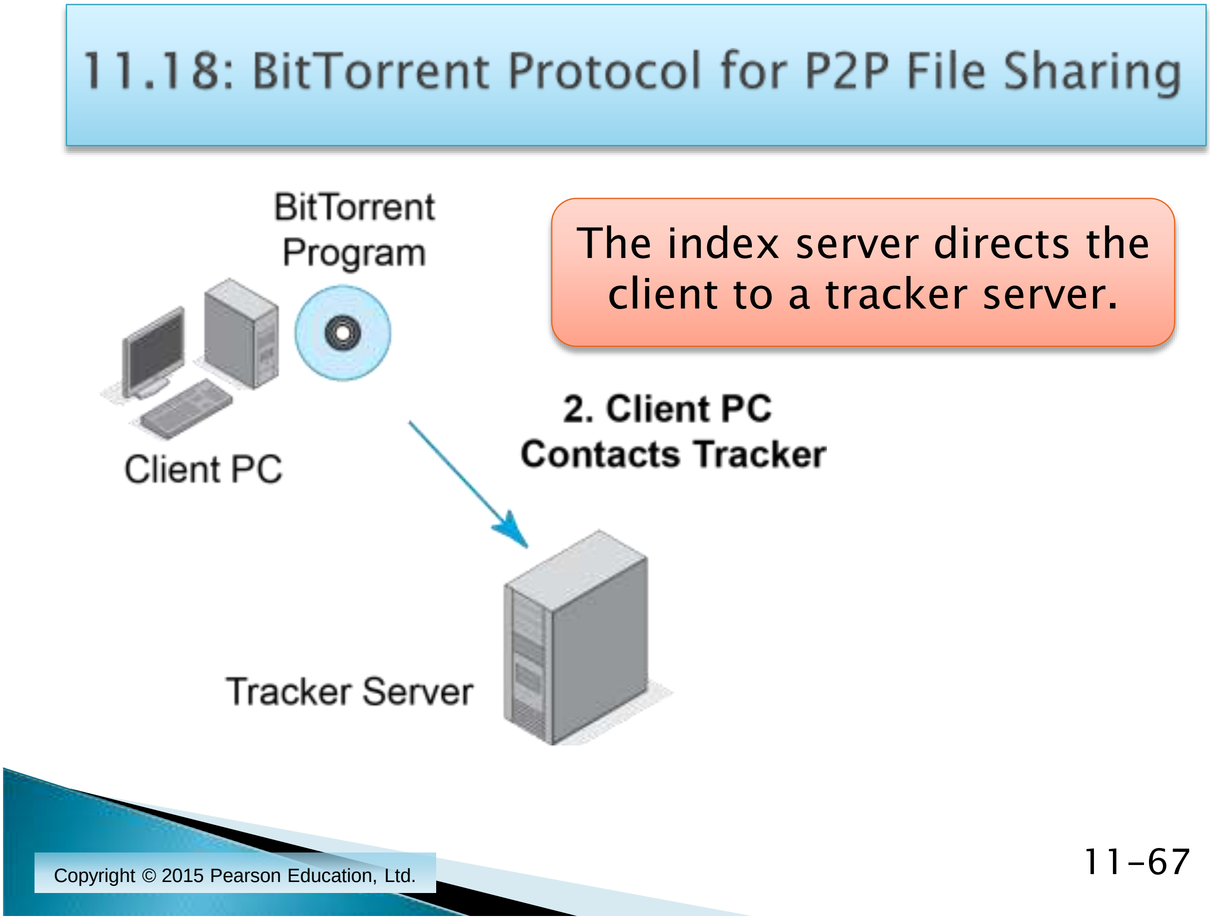
11.18: BitTorrent Protocol for P2P File Sharing

BitTorrent is a sophisticated PC file sharing standard.

The client first contacts an index webserver with a file request.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



11.18: BitTorrent Protocol for P2P File Sharing

The index server directs the client to a tracker server.

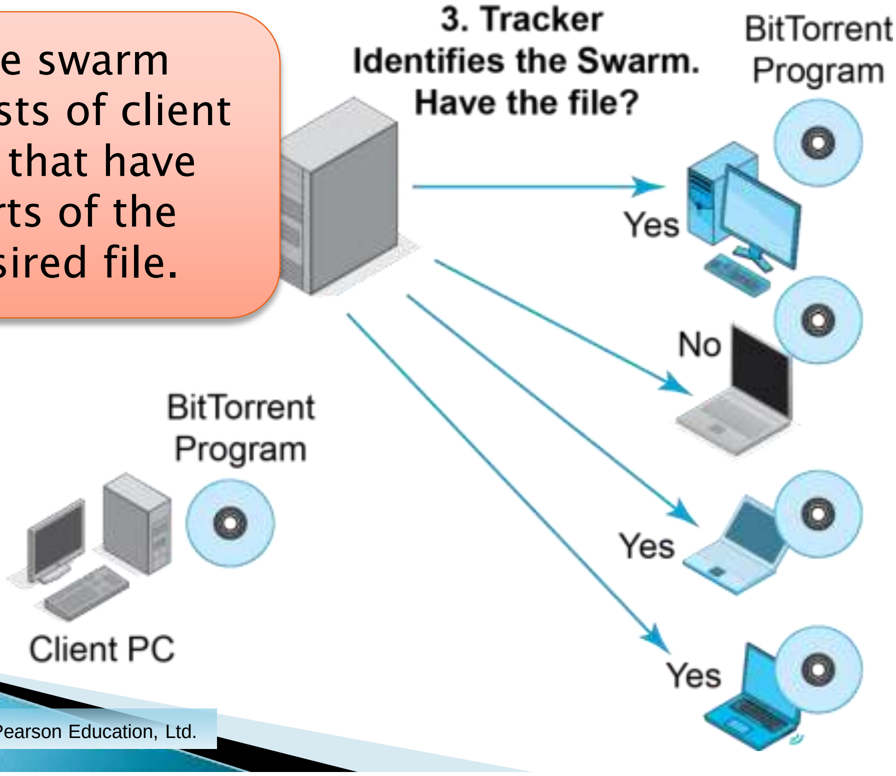
2. Client PC Contacts Tracker

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

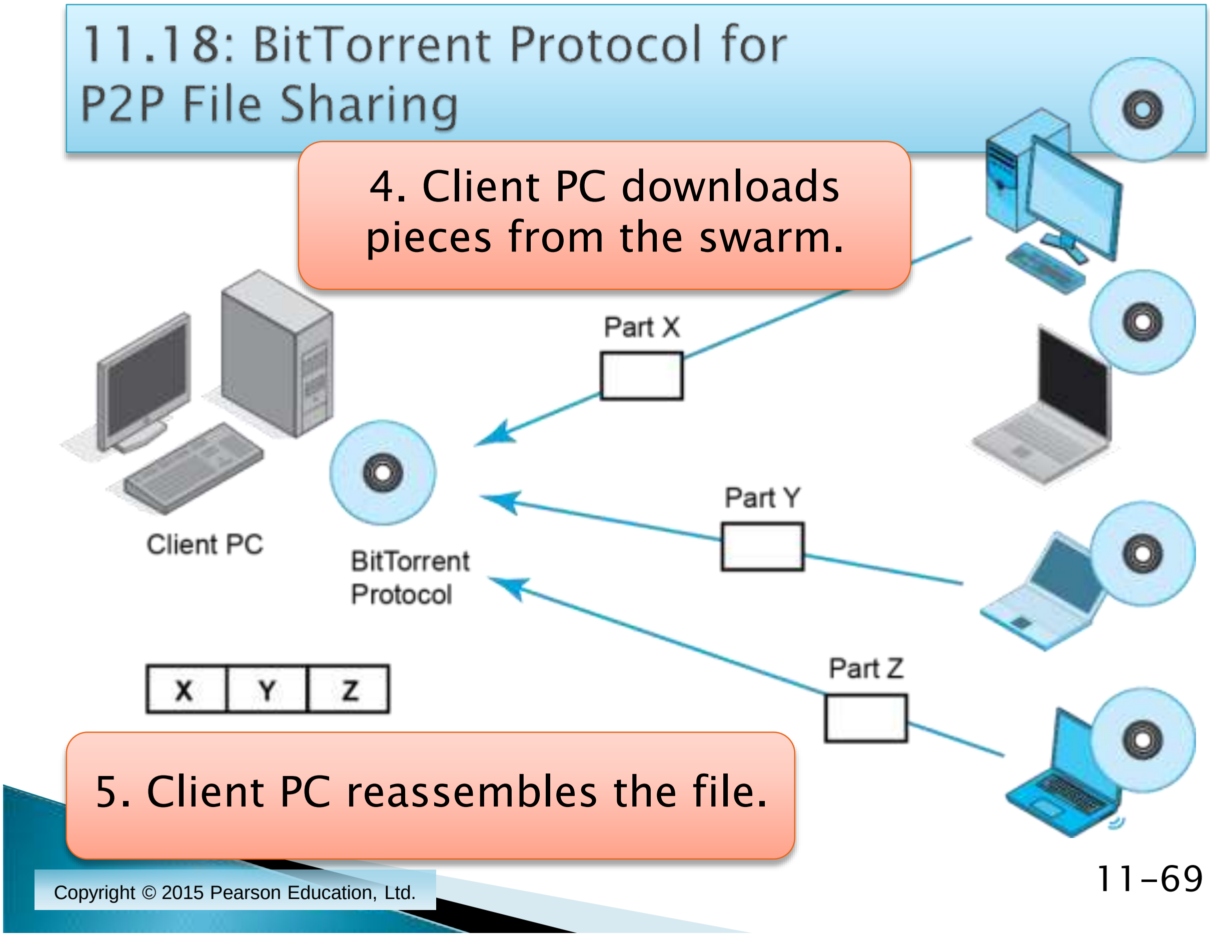
11.18: BitTorrent Protocol for P2P File Sharing

The swarm consists of client PCs that have parts of the desired file.



Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



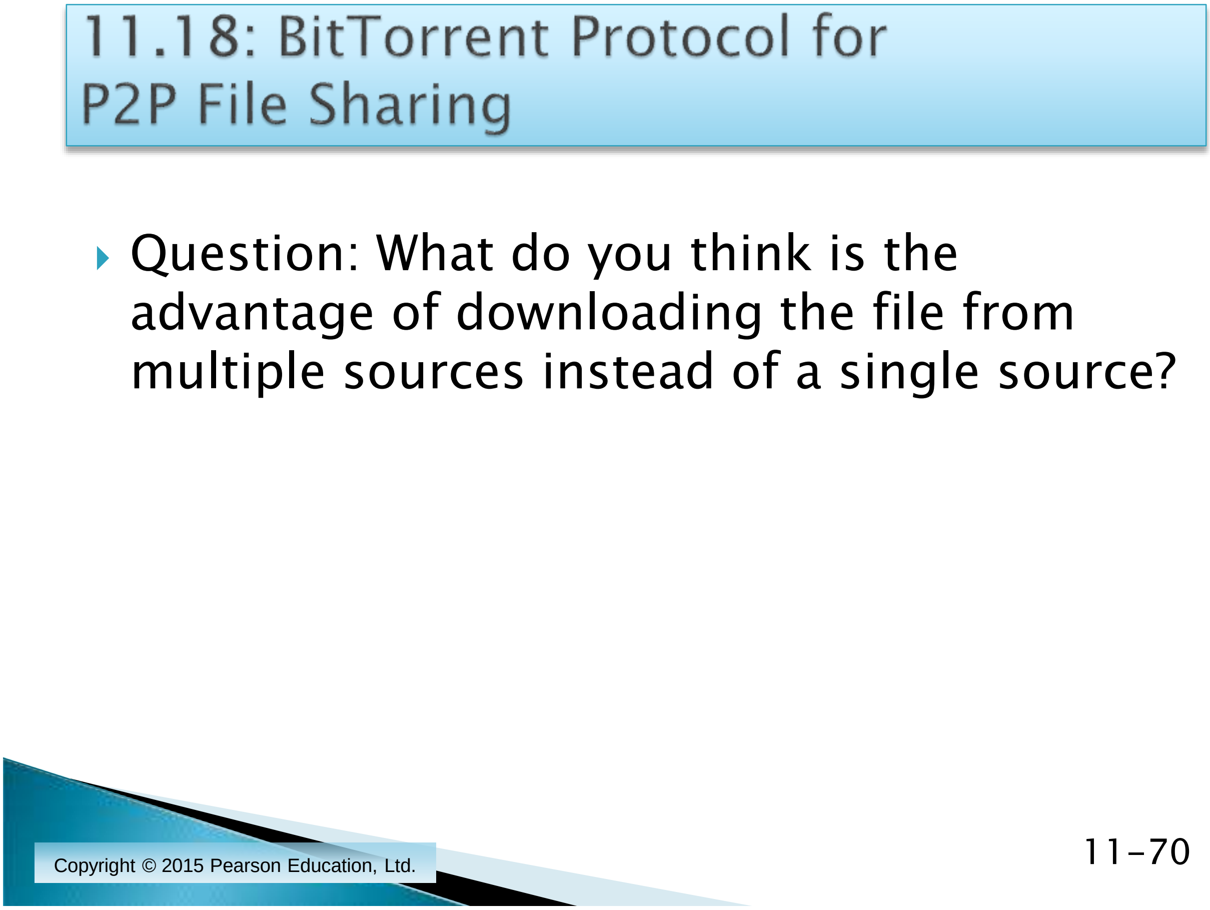
11.18: BitTorrent Protocol for P2P File Sharing

4. Client PC downloads pieces from the swarm.

5. Client PC reassembles the file.

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson

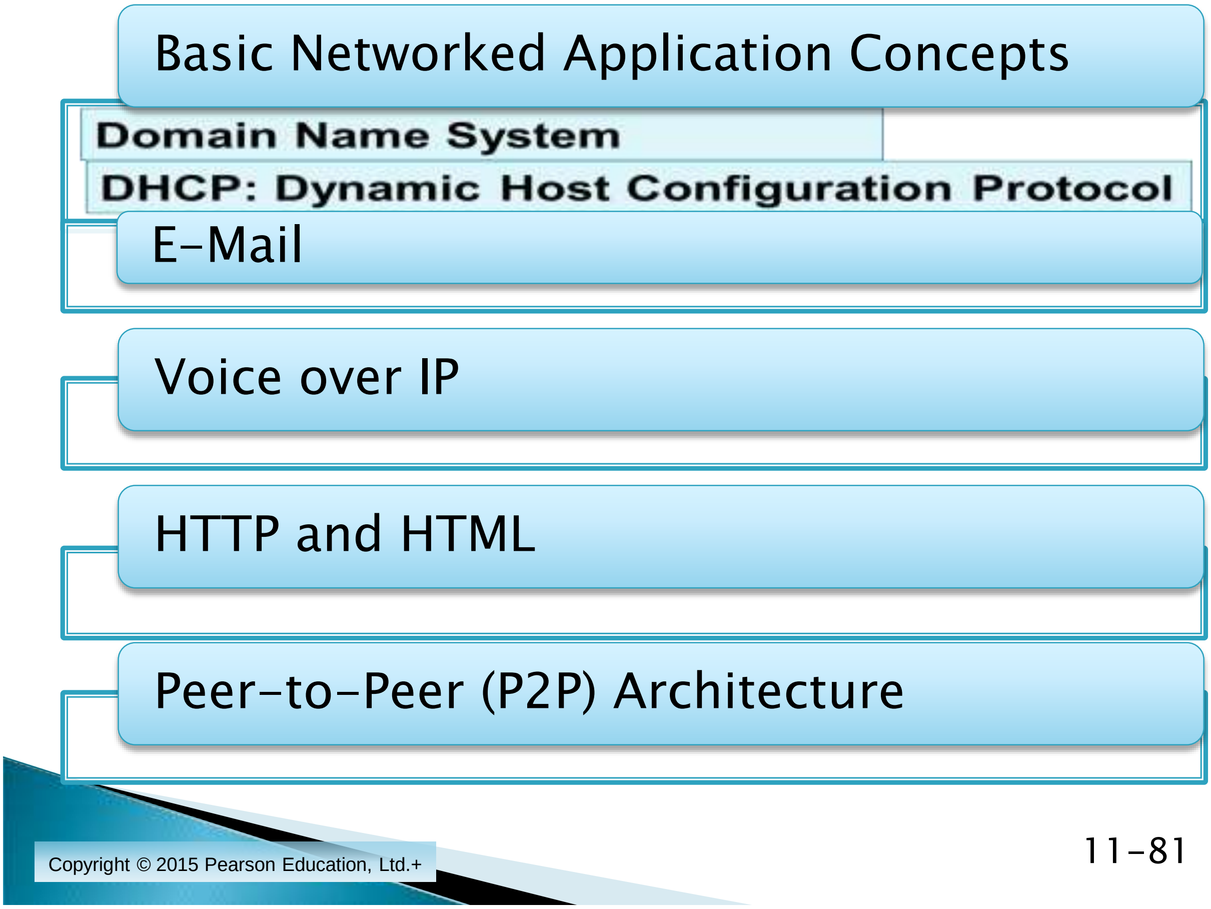


11.18: BitTorrent Protocol for P2P File Sharing

- Question: What do you think is the advantage of downloading the file from multiple sources instead of a single source?

Copyright © 2015 Pearson Education, Ltd.

© 2013 Pearson



Basic Networked Application Concepts

Domain Name System

DHCP: Dynamic Host Configuration Protocol

E-Mail

Voice over IP

HTTP and HTML

Peer-to-Peer (P2P) Architecture

Copyright © 2015 Pearson Education, Ltd.+

© 2013 Pearson