CMPE 344 Computer Networks
Fall 2019

Applications

Reading: Peterson and Davie, §9.1, 9.3, and selected topics

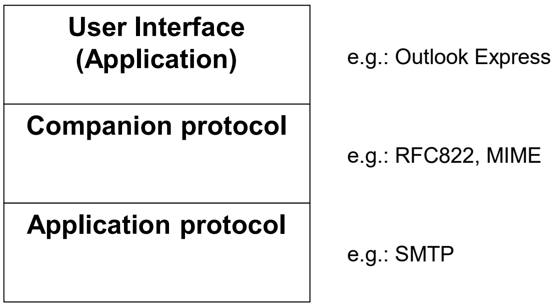
Application-Layer Protocols

- Network applications run on end systems
  - They depend on the network to provide a service
  - … but cannot run software on the network elements
- Network applications run on multiple machines
  - Different end systems communicate with each other
  - Software is often written by multiple parties
- Leading to a need to explicitly define a protocol
  - Types of messages (e.g., requests and responses)
  - Message syntax (e.g., fields, and how to delineate)
  - Semantics of the fields (i.e., meaning of the information)
  - Rules for when and how a process sends messages

Application vs. Application-Layer Protocols

- Application-layer protocol is just one piece
  - Defining how the end hosts communicate
- Example: World Wide Web
  - HyperText Transfer Protocol is the protocol used to retrieve web pages from remote servers
  - But the Web includes other components, such as document formats (HTML), Web browsers, servers,..
- Example: electronic mail {see next slide}
  - Simple Mail Transfer Protocol (SMTP) is the protocol used to exchange electronic mail
  - But e-mail includes other components, such as mail servers, user mailboxes, mail readers,..

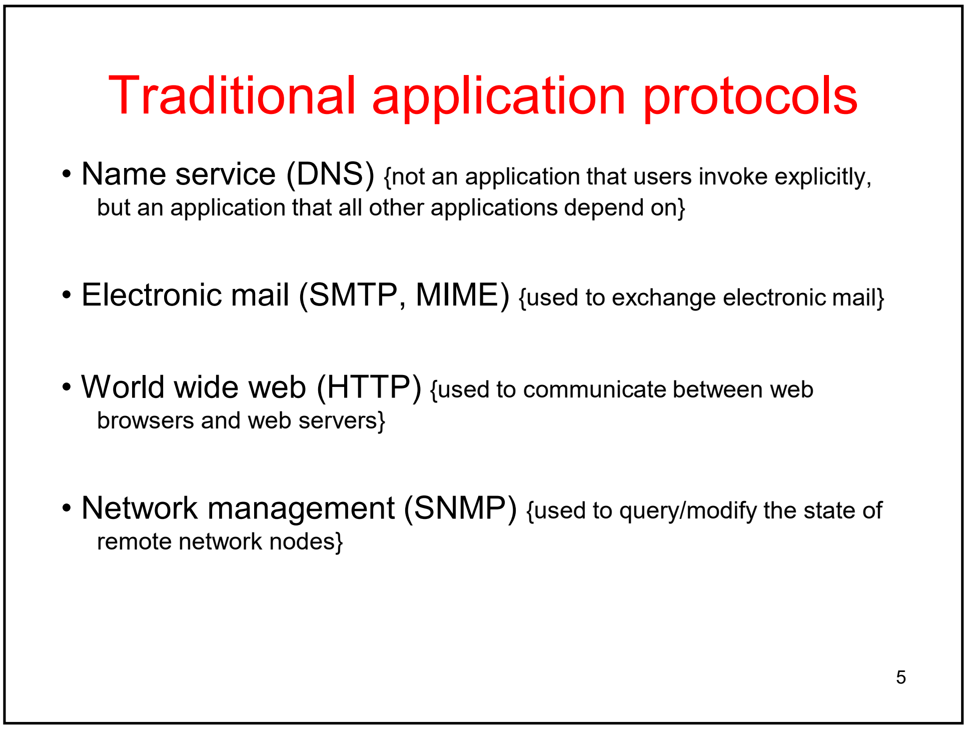
E-mail example



User Interface (Application) e.g.: Outlook Express

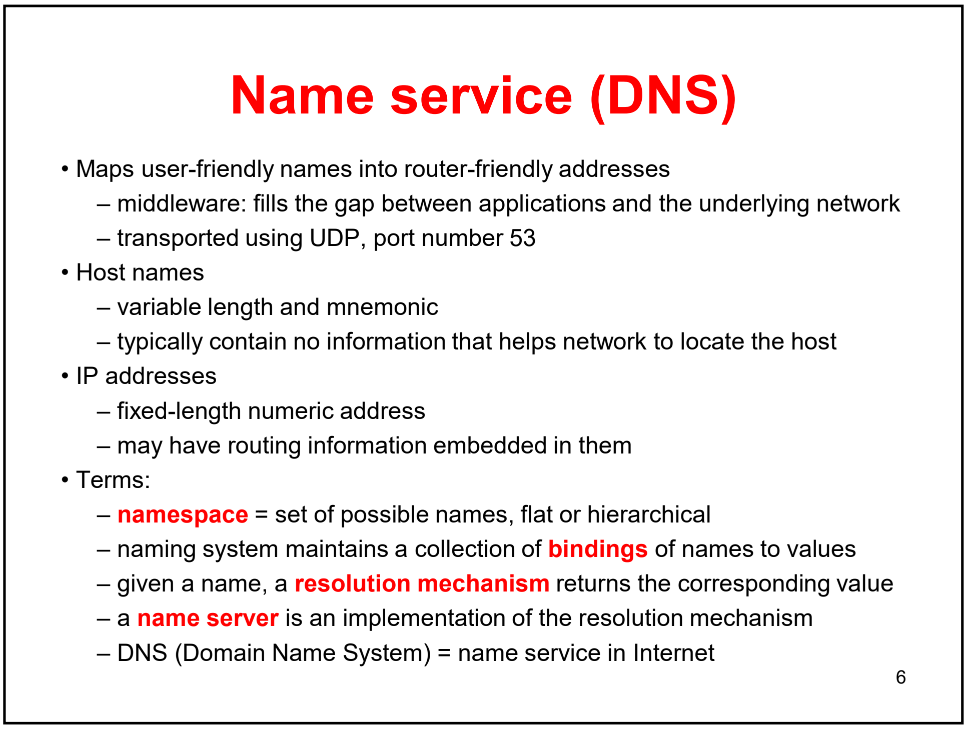
Companion protocol e.g.: RFC822, MIME

Application protocol e.g.: SMTP



Traditional application protocols

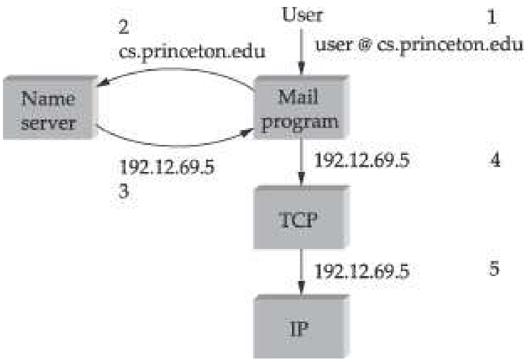
- Name service (DNS) {not an application that users invoke explicitly, but an application that all other applications depend on}
- Electronic mail (SMTP, MIME) {used to exchange electronic mail}
- World wide web (HTTP) {used to communicate between web browsers and web servers}
- Network management (SNMP) {used to query/modify the state of remote network nodes}



Name service (DNS)

- Maps user-friendly names into router-friendly addresses
  - middleware: fills the gap between applications and the underlying network
  - transported using UDP, port number 53
- Host names
  - variable length and mnemonic
  - typically contain no information that helps network to locate the host
- IP addresses
  - fixed-length numeric address
  - may have routing information embedded in them
- Terms:
  - **namespace** = set of possible names, flat or hierarchical
  - naming system maintains a collection of **bindings** of names to values
  - given a name, a **resolution mechanism** returns the corresponding value
  - a **name server** is an implementation of the resolution mechanism
  - DNS (Domain Name System) = name service in Internet

Example



⇒ Application (mail) uses DNS to translate the name into IP address

DNS domain hierarchy

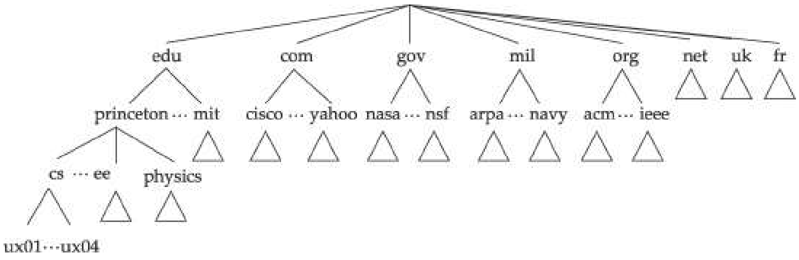
- First level hierarchy

– domains for each country + edu, com, gov, mil, org, net

– DNS first level managed by Internet Corporation for Assigned Names and Numbers (ICANN), also manages address allocations

- Hierarchy is partitioned into subtrees, zones

– zone corresponds to fundamental implementation unit in DNS (i.e., a name server)



DNS domain hierarchy (cont)

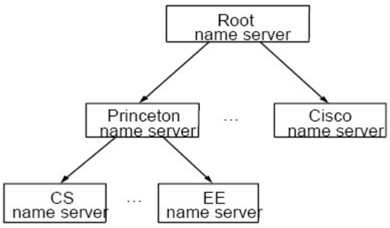
- Zones implemented in two or more name servers (redundancy)
  - – clients send queries to name servers
  - – servers response with final answer or pointer to another server
- Name binding database consists of resource records
  - – format: <Name, Value, Type, Class, TTL>
  - – Type: how Value is interpreted,
    - A: means that Value is an IP address, name-address mapping
    - NS: Value contains name for host that knows how to resolve the name
    - CNAME: Value is a canonical name for host, used to define aliases
    - MX: Value gives the domain name for a host running a mail server
  - – Class: only widely used class IN (Internet)
  - – TTL: how long resource record is valid (used by servers that cache resource records from other servers)
  - – can use alias for company web server ⇒ web server to be changed without remote users being affected
  - – MX allows administrators to change the mail host without changing user email addresses

DNS domain hierarchy (cont)

- Root name server: NS record for each 2nd level server + A record that translates name into IP address

  <princeton.edu, cit.princeton.edu, NS,IN>

  <cit.princeton.edu, 128.196.128.233, A, IN>



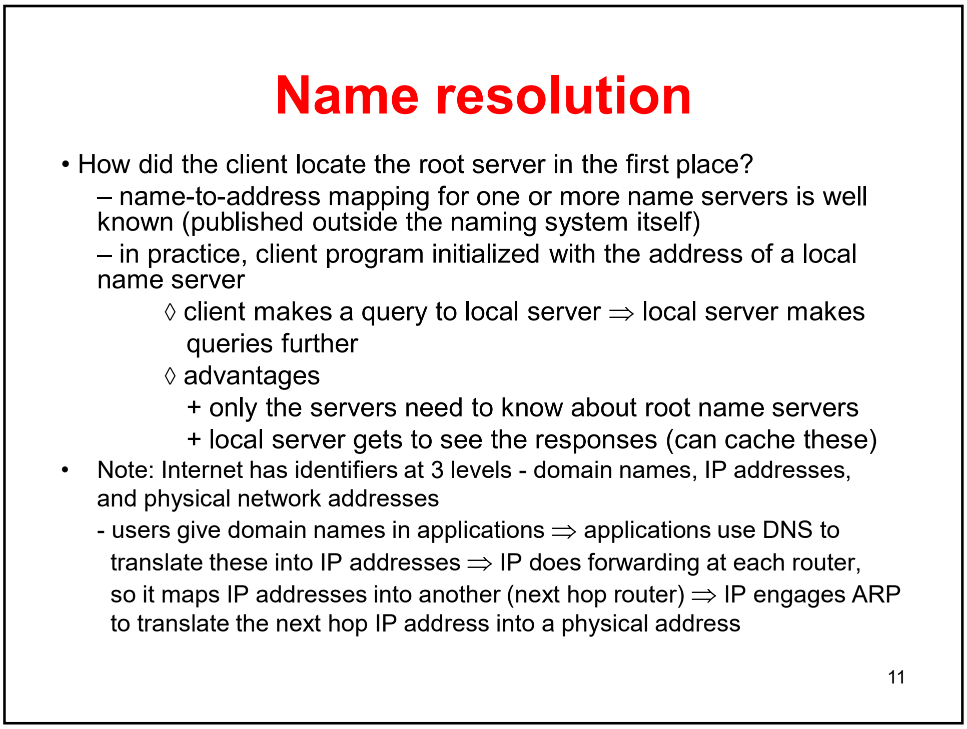
- At 2nd level, records contain either final answers or pointer to 3rd level name servers

  <cs.princeton.edu, gnat.cs.princeton.edu, NS, IN>

  <gnat.cs.princeton.edu, 192.12.69.5, A, IN> (pair like above)

  <jupiter.physics.princeton.edu, 128.196.4.1, A, IN> (final record)

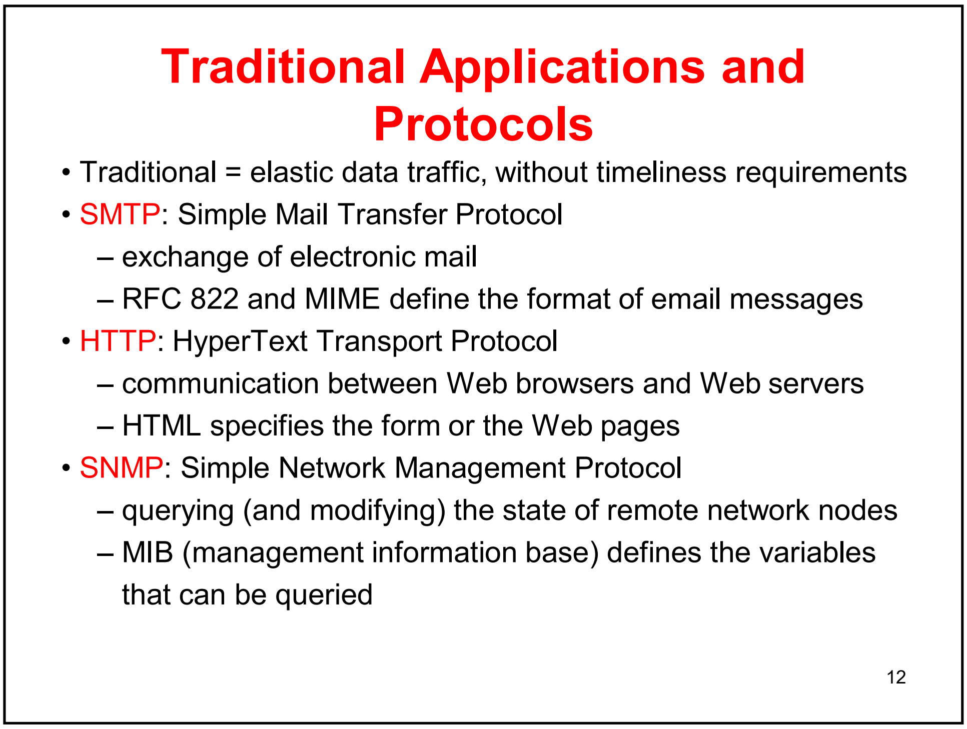
- Lowest level contains final records, aliases for hosts (CNAME) and MX records



Name resolution

- How did the client locate the root server in the first place?
  - – name-to-address mapping for one or more name servers is well known (published outside the naming system itself)
  - – in practice, client program initialized with the address of a local name server
    - ◊ client makes a query to local server ⇒ local server makes queries further
    - ◊ advantages
      - + only the servers need to know about root name servers
      - + local server gets to see the responses (can cache these)
- Note: Internet has identifiers at 3 levels - domain names, IP addresses, and physical network addresses
  - - users give domain names in applications ⇒ applications use DNS to translate these into IP addresses ⇒ IP does forwarding at each router, so it maps IP addresses into another (next hop router) ⇒ IP engages ARP to translate the next hop IP address into a physical address

11



Traditional Applications and Protocols

- Traditional = elastic data traffic, without timeliness requirements
- SMTP: Simple Mail Transfer Protocol
  - – exchange of electronic mail
  - – RFC 822 and MIME define the format of email messages
- HTTP: HyperText Transport Protocol
  - – communication between Web browsers and Web servers
  - – HTML specifies the form or the Web pages
- SNMP: Simple Network Management Protocol
  - – querying (and modifying) the state of remote network nodes
  - – MIB (management information base) defines the variables that can be queried

12

Electronic mail (SMTP, MIME)

- Mail service consists of
  - a mail reader,
  - a message transfer protocol (SMTP) and
    - ◊ SMTP = Simple Message Transfer Protocol
  - companion protocols RFC 822 (request for comments) & MIME (multipurpose internet mail extensions)
- Mail access protocol: retrieval from server
  - reader programs: Netscape Messenger, Outlook, etc..
  - POP3: Post Office Protocol (RFC 1939)
    - ◊ authorization (agent ⇔server) and download
    - ◊ downloads mails to your own local host
  - IMAP: Internet Mail Access Protocol (RFC 1730)
    - ◊ more features (more complex)
    - ◊ manipulation of inbox and stored messages on server
  - HTTP: Hotmail , Yahoo! Mail, gmail, etc…

Electronic mail (cont)

- Message format:
  - RFC 822: message has two parts - a header and a body
    - both in ASCII text
  - MIME: extends RFC 822 so that message can contain all sorts of data
    - data still presented as ASCII text
  - ASCII format ⇒ human can pretend to be an smtp client
- Message header:
  - series of <CRLF>-terminated lines (carriage-return + line-feed)
  - separated from message body by blank line
  - each header line contains a Type and a Value separated by a colon
    - To: student@emu.edu.tr
    - Subject: CMPE344

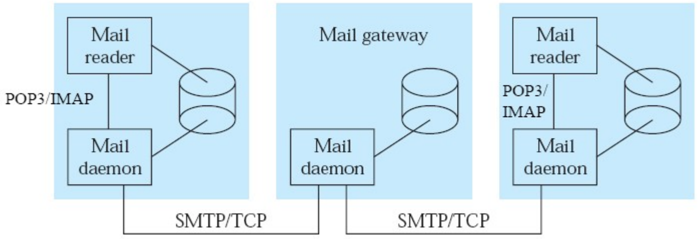
MIME

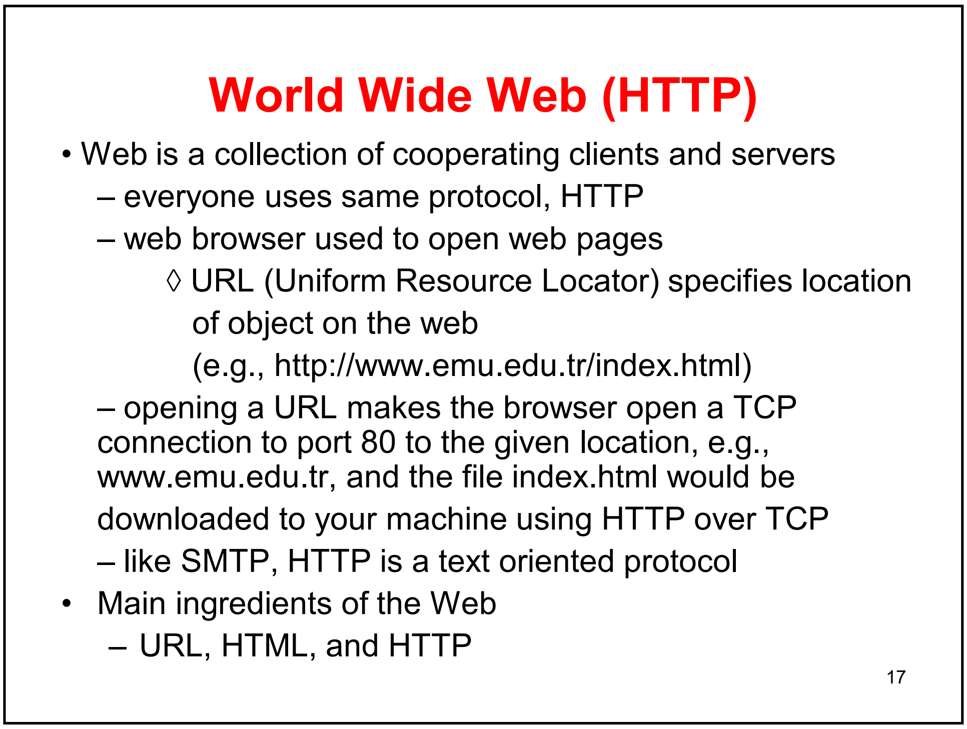
- Extends RFC 822 to allow email messages to carry audio, video, images, Word documents etc.
- Consists of 3 basic pieces
  - – collection of header lines
    - ◊ extend the original set defined in RFC 822
    - ◊ ex. MIME-version, Content-Description, Content-Type, Content-Transfer-Encoding..
  - – definitions for a set of content types
    - ◊ ex. image/gif, image/jpeg, text/plain, text/richtext, application/postscript, application/msword
  - – a way to encode various data types so that they can be shipped in an ASCII mail message
- base64 coding of binary data into ASCII: map every 3 bytes of the original binary data into 4 ASCII characters

Message transfer (SMTP)

- E-mail delivery
  - – mail reader ⇒ message to mail daemon ⇒ daemon uses SMTP running over TCP to get message to a daemon in another machine ⇒ this daemon puts the message into user’s mailbox
  - – SMTP uses TCP on port 25
- Mail traverses many mail gateways that store and forward email msgs

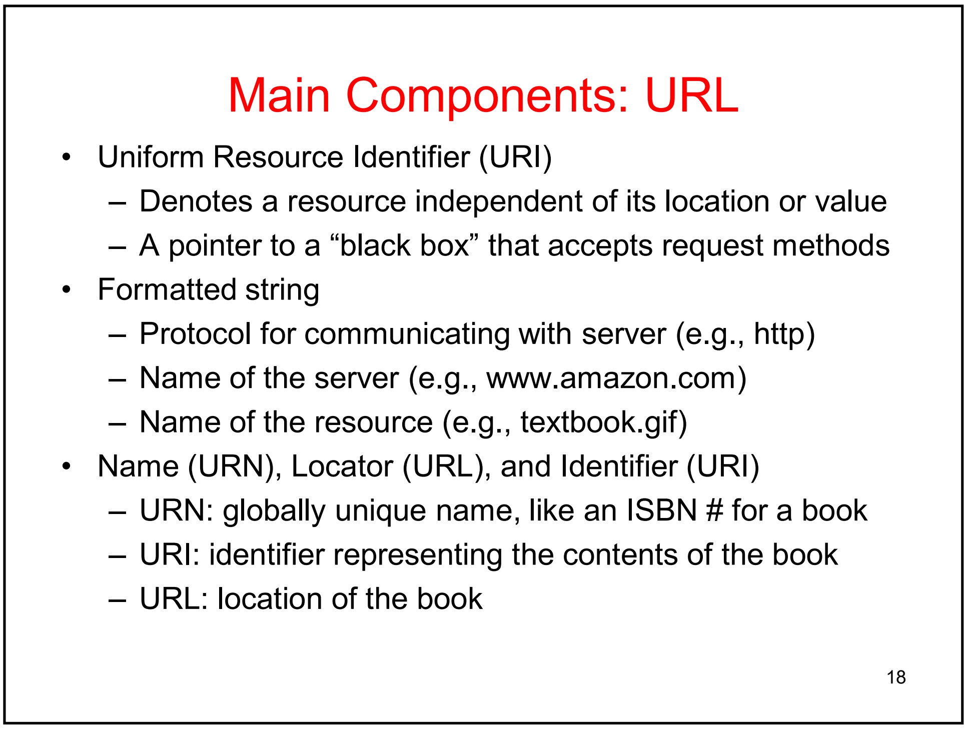
– mail gateway vs. IP router? IP router stores datagrams in memory and tries to retransmit them for a short period of time (fraction of seconds), mail gateway buffer messages on disk and try resending for days





World Wide Web (HTTP)

- Web is a collection of cooperating clients and servers
  - everyone uses same protocol, HTTP
  - web browser used to open web pages
    - ◊ URL (Uniform Resource Locator) specifies location of object on the web (e.g., http://www.emu.edu.tr/index.html)
  - opening a URL makes the browser open a TCP connection to port 80 to the given location, e.g., www.emu.edu.tr, and the file index.html would be downloaded to your machine using HTTP over TCP
  - like SMTP, HTTP is a text oriented protocol
- Main ingredients of the Web
  - URL, HTML, and HTTP



Main Components: URL

- Uniform Resource Identifier (URI)
  - Denotes a resource independent of its location or value
  - A pointer to a “black box” that accepts request methods
- Formatted string
  - Protocol for communicating with server (e.g., http)
  - Name of the server (e.g., www.amazon.com)
  - Name of the resource (e.g., textbook.gif)
- Name (URN), Locator (URL), and Identifier (URI)
  - URN: globally unique name, like an ISBN # for a book
  - URI: identifier representing the contents of the book
  - URL: location of the book

HTTP and TCP connections

- HTTP version 1.0 made a separate TCP connection for each data item
  - waste of resources, especially when most items are small sized
- HTTP version 1.1 allows persistent connections: client and server can exchange multiple request/response messages over the same TCP connection
  - good:
    - ◊ eliminates the connection setup overhead
    - ◊ client can send multiple request messages ⇒ TCP's congestion window mechanism operates more efficiently (not necessary to do slow start for each request)
  - bad:
    - ◊ neither the client nor server knows how long to keep a particular TCP connection open (problem for servers with thousands of connections)
    - ◊ client and server must watch if the other side has elected to close the connection (recall, both sides need to close the TCP connection)

Caching

- WWW cache = web proxy
- Benefits:
  - pages from nearby cache can be displayed quickly
  - can reduce servers' load
- Implementation at several (hierarchical) layers:
  - in user's browser
  - user's site can support a single sitewide cache (takes advantage of pages previously downloaded by other users)
  - ISPs may have their own caches
- Cache needs to make sure it is not responding with an out-of-date version of the page
  - server may assign an expiration date (Expires header field) to each page
  - HTTP conditional requests by using, i.e., If-Modified-Since message header

Network Management (SNMP)

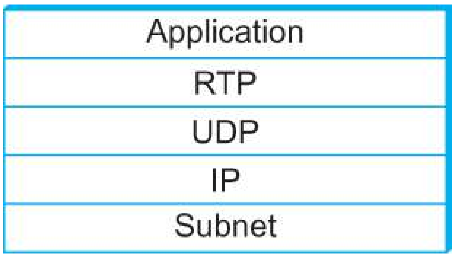
- Possible tasks:
  - monitor faulty equipment in the network
  - keep track of the load on various nodes (need for new routers or links?) etc., etc.
- Nodes in the network are distributed ⇒ use the network to manage the network
  - need a protocol for reading (and writing) state information on different network nodes
- Simple Network Management Protocol (SNMP)
  - request/reply protocol that supports GET and SET messages
  - runs on top of UDP
  - client program uses SNMP to request information, SNMP server running on a node replies
  - depends on companion specification Management Information Base (MIB) that describes object structure of network elements

Interactive multimedia applications

- Voice over IP (VOIP)
  - Phone calls over IP
  - Computer to computer
  - Analog phone to/from computer
  - Analog phone to analog phone
- Enabling protocols:
  - RTP: Real-time Transport Protocol
  - RTCP: Real-time Transport Control Protocol
  - SIP: Session Initiation Protocol

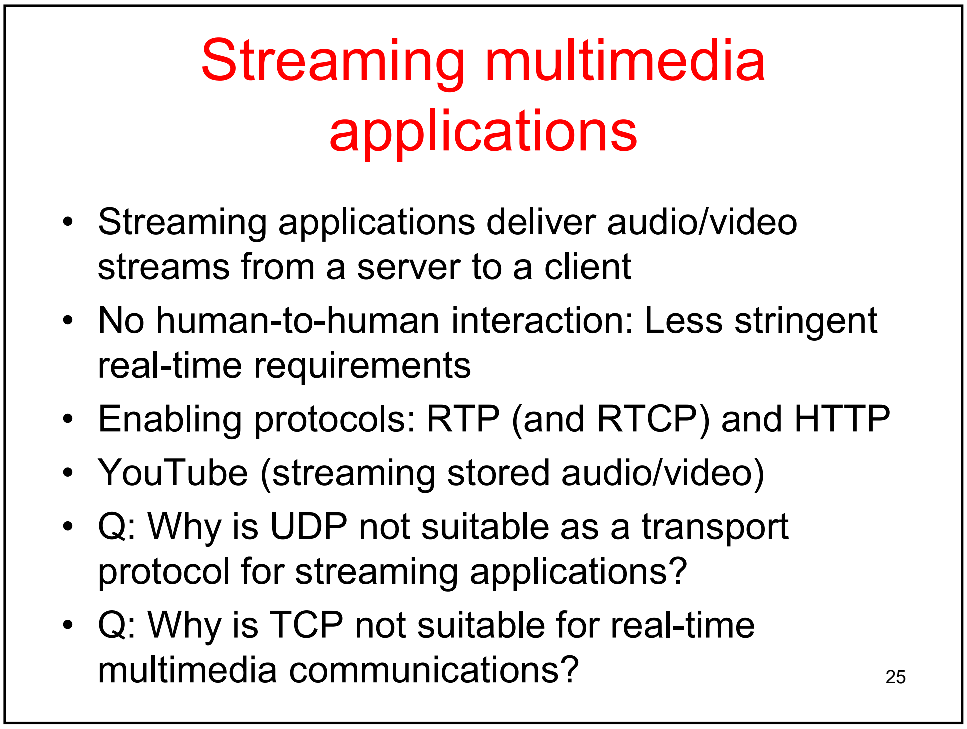
RTP

- An end-to-end protocol used by multimedia applications that have real-time constraints



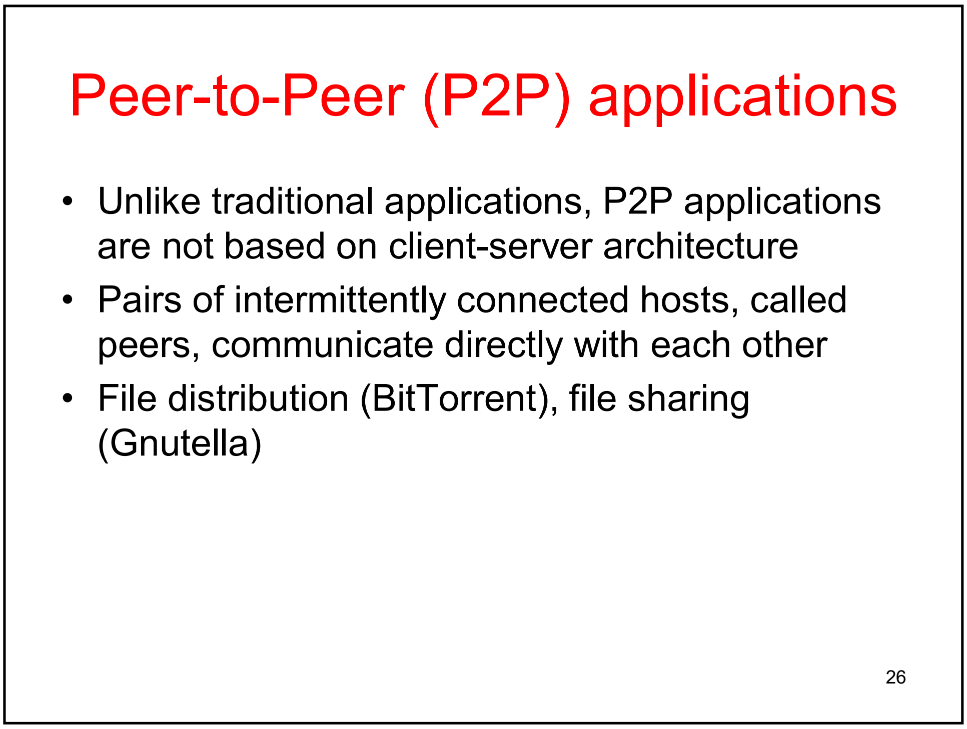
SIP

- An application layer protocol
  - Determines the correct device with which to communicate to reach a particular user
  - Determines if the user is willing to or able to take part in a particular communication
  - Determines the choice of media and coding scheme to use
  - Establishes session parameters (e.g., port numbers)



Streaming multimedia applications

- Streaming applications deliver audio/video streams from a server to a client
- No human-to-human interaction: Less stringent real-time requirements
- Enabling protocols: RTP (and RTCP) and HTTP
- YouTube (streaming stored audio/video)
- Q: Why is UDP not suitable as a transport protocol for streaming applications?
- Q: Why is TCP not suitable for real-time multimedia communications?



Peer-to-Peer (P2P) applications

- Unlike traditional applications, P2P applications are not based on client-server architecture
- Pairs of intermittently connected hosts, called peers, communicate directly with each other
- File distribution (BitTorrent), file sharing (Gnutella)