Application Layer Protocol

- Types of Messages
- Message Syntax
- Message Semantics
- Rules for when and how processes send and respond to messages
Transport Services

- Data Integrity
- Timing
- Throughput
- Security
Internet Transport Layer

- No throughput guarantees
- No delay guarantees
- No security built into TCP or UDP
- SSL
  - Secure socket layer
  - Application layer services built on top of TCP
Applications

- http
- Domain name server (DNS)
- Video streaming and content distribution networks
HyperText Transfer Protocol (http)

- Overview
- Connections
- Message Format
- Cookies
- Web Caching
Overview

- Created in the early 1990s by Tim Berners-Lee
- Client/Server Architecture
  - http client/http server
  - Request/response
- Web Page
  - Web page consist of one or more objects
- URL: hostname/pathname
  - cs.uwlax.edu/~gendreau/cs471
- Stateless
Hypertext

- Vannevar Bush: Memex (1954)
- Ted Nelson (1963)
- Douglas Engelbart (1962)
- Hypercard (MacIntosh 1987)
Request/Response
Connections

- TCP connections
- Persistent
- Non-Persistent
Message Format

• Syntax

• Semantics

• Standard ASCII Text

• Methods
  — GET
  — POST
  — HEAD
  — PUT
  — DELETE
HTTP request message

- two types of HTTP messages: request, response
- HTTP request message:
  - ASCII (human-readable format)

  GET /index.html HTTP/1.1
  Host: www-net.cs.umass.edu
  User-Agent: Firefox/3.6.10
  Accept: text/html,application/xhtml+xml
  Accept-Language: en-us,en;q=0.5
  Accept-Encoding: gzip, deflate
  Accept-Charset: ISO-8859-1, utf-8;q=0.7
  Keep-Alive: 115
  Connection: keep-alive

* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/
HTTP request message: general format

```
method   sp   URL   sp   version   cr   lf
header field name   value   cr   lf
header field name   value   cr   lf
entity body
```
HTTP response message

status line (protocol status code status phrase)

HTTP/1.1 200 OK
Date: Sun, 26 Sep 2010 20:09:20 GMT
Server: Apache/2.0.52 (CentOS)
Last-Modified: Tue, 30 Oct 2007 17:00:02 GMT
ETag: "17dc6-a5c-bf716880"
Accept-Ranges: bytes
Content-Length: 2652
Keep-Alive: timeout=10, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=ISO-8859-1

data data data data data data data data ...

* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/
HTTP response message: general format

```
+---+----+----------------+----+---+
| version | sp | Status code | sp | phrase | cr  | lf |
| header field name | sp | value | cr  | lf |
| header field name | sp | value | cr  | lf |
| cr | lf |
| entity body |
```

status line
header lines
body
Message Format

• Status codes
  – 200 OK:
  – 301 Moved Permanently
  – 400 Bad Request
  – 404 Not Found:
  – 505 HTTP Version Not Supported:
Cookies

- Client maintains cookie file
- Server maintains database matching id number with information about the user
- Once the id is established it can be included in response messages
Cookies: keeping “state” (cont.)

client

usual http request msg

usual http response msg

cookie: 1678

usual http response msg

cookie: 1678

usual http request msg

usual http response msg

one week later:

usual http request msg

cookie: 1678

usual http response msg

server

Amazon server creates ID 1678 for user

create entry

backend database

access
cookie-specific action

access
cookie-specific action
Web Cache

- Proxy Server
- Conditional Get
Web Cache