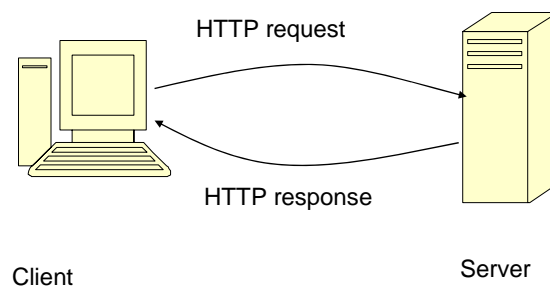


Http overview



■ HTTP requirement



HTTP request

- An HTTP request, which is the message sent from a Web client to a Web server, is comprised of three basic elements:
 - Request line
 - HTTP headers
 - Content



Example:

```
GET /search?hl=en&q=HTTP&btnG=Google+Search HTTP/1.1
Host: www.google.com
User-Agent: Mozilla/5.0 Galeon/1.2.0 (X11; Linux i686; U;) Gecko/20020326
Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,
        text/plain;q=0.8, video/x-mng,image/png,image/jpeg,image/gif;q=0.2,
        text/css,*/*;q=0.1
Accept-Language: en
Accept-Encoding: gzip, deflate, compress;q=0.9
Accept-Charset: ISO-8859-1, utf-8;q=0.66, */*;q=0.66
Keep-Alive: 300
Connection: keep-alive
```



Request line

- The first line of an HTTP request is always the request line.
- The request line specifies the
 - request method
 - the location of the resource (URI)
 - the version of HTTP being used
- These three elements are delimited by spaces



Request Methods

- This method indicates the overall intent of the Web client's request
- Eight request methods in HTTP/1.1:
 - GET, POST, PUT, DELETE, HEAD, TRACE, OPTIONS, and CONNECT.
- Three methods in HTTP/1.0:
 - GET, HEAD, and POST



Request methods -cont

■ GET Method

- ☐ A GET request is basically a request to receive the content located at a specific URL.
- ☐ Obtaining a URL using the GET method allows users to bookmark the URL, create a link to the URL, email the URL to a friend, and the like.
- ☐ A GET request traditionally contains no content and thus no entity headers

■ POST Method

- ☐ Form information was communicated in the content of the request.



Request methods -cont

■ PUT Method

- ☐ It allows the Web client to send content that will be stored on the Web server.

■ DELETE Method

- ☐ It specifies content on the Web server to be deleted as the resource in the request line

■ HEAD Method

- ☐ It is supposed to behave exactly like GET, except that the content is not present.



Request methods -cont

- **TRACE Method**
 - TRACE is a diagnostic request method. This method allows the client to gain more perspective into any intermediary proxies that lie between the client and the server.
- **OPTIONS Method**
 - Sometimes it is helpful to simply identify the capabilities of the Web server you want to interact with prior to actually making a request
- **CONNECT Method**
 - The CONNECT request method is reserved explicitly for use by intermediary servers to create a tunnel to the destination server.



HTTP headers

- General headers
- Request headers
- Entity headers



HTTP Responses

- The Web server will then respond to the client with an HTTP response
 - Status line
 - HTTP headers
 - Content



Status line

- The status line contains three elements:
 - The version of HTTP being used, in the format HTTP/x.x
 - The status code
 - A short description of the status code
- Example:
 - HTTP/1.1 200 OK



Response Status Codes

- Status codes are grouped into the following ranges:
 - ☐ Informational (100-199)
 - ☐ Successful (200-299)
 - ☐ Redirection (300-399)
 - ☐ Client error (400-499)
 - ☐ Server error (500-599)



HTTP header in response

- three types of HTTP headers
 - ☐ General headers
 - ☐ Response headers
 - ☐ Entity headers



Header Formatting

- All HTTP headers follow a general format:
 - *Header-Name: value*
- Example:
 - *Content-type: text/html*



General Headers

- Connection
 - Connection: Keep-Alive
 - Connection: Close
- Date
 - Date: Tue, 21 May 2002 12:34:56 GMT



Response Headers

- Server
 - The Server header relays information to the Web client about the software being used by the Web server



Entity Headers

- Entity headers relay information about the content of an HTTP message
 - Content-Length
 - Content-Type
 - Last-Modified



Media Type Format

- */* (matches any media type)
- type/* (for example, image/*)
- type/subtype (for example, image/png)



Media Type Categories

- | | |
|---------------|-----------------------------------|
| ■ Application | Example: application/vnd.ms-excel |
| ■ Audio | Example: audio/mpeg |
| ■ Image | Example: image/png |
| ■ Message | Example: message/http |
| ■ Model | Example: model/vrml |
| ■ Multipart | Example: multipart/form-data |
| ■ Text | Example: text/html |
| ■ Video | Example: video/quicktime |