HTTP and XHTML

CS174

Chris Pollett

Aug. 27, 2008.

Outline

- Apache
- .htaccess files
- MIME
- HTTP
- XHTML

Apache

- As we said last day this is the most common web server.
- It has three configuration files, but in modern usage only the httpd.conf file is used.
- This file consists of a bunch of lines in a pseudo-tagged based language. # is used for comments
- Here is a brief list of common directives:
    - ServerName --- names the server system.
    - ServerRoot --- path to the server root.
    - DocumentRoot --- path to the document root.
    - LoadModule --- loads a module (like PHP to be used with the server) .
    - AddModule --- makes the added module available for use.
    - AddType --- associate a file name extension with a MIME type.
    - Alias -- alias a URL path to a physical directory path .
    - <Directory> </Directory> --- control how the contents of a particular directory can be served.

.htaccess files

- Even if you don't have access to the httpd.conf file, if the webserver admin allows it, you might be able to use .htaccess files in the directories you control.

- An .htaccess file allows you to do configurations similar to what can be done in <Directory></Directory> tags.

- Example:

```
AuthUserFile  "/Volumes/Library/WebDocuments/library/.htpasswd"
AuthGroupFile /dev/null
AuthName library
AuthType Basic

<Limit GET POST>
require valid-user
</Limit>
```

MIME

- MIME stands for Multipurpose Internet Mail Extensions.
- It was originally developed as a way to send attachments containing non-text documents over the internet.
- Nowadays, it provides a way for a browser to determine the format of the documents it receives from the server.
- A MIME specification has the format is:
  type/subtype.
- The most common types are *text*, *image*, *video*.
- Some example complete specifications are: text/plain, text/html, image/gif, image/jpeg, video/mpeg.
- In addition to the above, you sometimes see types such as application/xhtml+xml, application/xml, or experimental types which begin with an x-. For example, video/x-msvideo.

HTTP

- The details of HTTP (hypertext transfer protocol) can be found in the RFC 2616 available at http://www.w3.org.
- HTTP consists of two phases, the request and the response.
- A request has the format:
  1. HTTP method Domain part of URL HTTP version.
  2. Header fields.
  3. Blank line.
  4. Message Body.
- For example, at a Unix prompt trying typing:

  telnet www.cs.sjsu.edu 80 ◄──────────── Not part of request

  GET /index.shtml HTTP/1.0

More HTTP

- The most commonly used HTTP requests are:
  - GET --- request contents of specified document.
  - HEAD --- request header contents of specified document.
  - POST --- Executes the specified document, using the enclosed data.
  - PUT --- replaces the specified document with the enclosed data.
  - DELETE ---deletes the specified document.
- Of these, GET and POST are the most common.

Header Fields

- After the first line of the HTTP request one can have any number of header fields.

- There are four of headers: general, those used in the request, those used in the response, and entity headers which are used in both.

- To common headers are:

  - Accept: says what MIME types the browser can handle.

  - Host: (required in HTTP/1.1) Says the name of the host that you are trying to contact. Multiple hosts www.widgets.com, www.sprockets.com might be hosted on the same machine (virtual hosting). This says which one you are trying to contact.

- For example:

  GET /respond.html HTTP/1.1

  Host: blanca.uccs.edu

  Accept: text/*

  Accept: image/gif

Response Phase

- An HTTP response has the following format:
    1. Status line
    2. Response header fields
    3. Blank line
    4. Response body
- An example status line might be:

HTTP/1.1   200   OK.

- In general, one will have a 3-digit status code and a message. The codes come in one of five categories: 1 - information, 2 success, 3 - redirection, 4 - client error, 5 - server error.

An example response header

HTTP/1.1 200 OK

Date: Mon, 28 Aug 2006 19:09:45 GMT

Server: Apache/2.2.2 (Fedora)

Accept-Ranges: bytes

Connection: close

Content-Type: text/html

- In the above, Date:, Server:, Accept-Ranges:, etc are all response fields.

XHTML

- HTML was original derived from the Standard Generalized Markup Language (SGML) which is an ISO standard for specifying text-formatting languages.
- HTML was designed to specify content rather than how it was formatted (presentation).
- The reason was to allow it to be displayed in as many different kinds of computers and browsers as possible.
- As a specification language for simple tag based languages, SGML is severe overkill, so a stripped-down specification language called XML was developed in the late 90's.
- XHTML is HTML specified using XML.
- The most common variant of HTML today are:

  ISO HTML (HTML 4.01) which is specified using SGML and XHTML 1.0 Transitional, XHTML 1.0 STRICT, XHTML 1.1.
- XHTML 2.0 has been under development for several years, but so far isn't ready for general use.
- HTML 5 is also currently under development and looks like it might be used.

Basic Syntax

- The fundamental structural unit of HTML is a pair of tags.

- For example, a paragraph might be specified with: <p> The quick brown fox..</p>

- Here <p> is called the *open tag* and </p> is called the *close tag*. "The quick brown fox" is the *content* of the tag.

- Unlike earlier variants of HTML, all tags names in XHTML are lower case.

- To be a legal document, every open tag must be closed. Further, the nesting must be legal.

   For example,

   <img src="my_picture.png"></img> <!--this is okay. BTW, this is an example HTML comment -->

    <!--this is also okay and is an abbreviation for the line above -->

   <img src="my_picture.png" > <!--not okay by itself -->

   <P>old style paragraph</P> <!--not okay -->

   <p><i>Hello</i></p> <!--okay -->

   <p><i>Hello</p></i> <!--not okay -->

- Open tags may have attributes. For example, src in the image tag above. The value of an attribute must be given in double quotes.

Standard XHTML Document Structure

- XML declarations:

  <?xml version = "1.0" encoding = "utf-8" ?>

  <!-- as not supported by some old browsers validators doesn't usually check this -->

- SGML DOCTYPE. This says which Document Type Definition will be used:

  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
      "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

- The XHTML document:

  <html xmlns = "http://www.w3.org/1999/xhtml" >

      <!-- might have namespaces for other things like SVG -->

      <head><!--what an HTTP head request gives you-->

        <title>name of my document</title></head>

      <body><!--actually page stuff--></body>

      </html>