

**E-Content of
INTERNET TECHNOLOGY AND WEB DESIGN**

Chapter: 8.13 HTTP

Topic: 8.13.1 HTTP

HTTP

- The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems.
- This is the foundation for data communication for the World Wide Web (ie. internet) since 1990.
- HTTP is a generic and stateless protocol which can be used for other purposes as well using extension of its request methods, error codes and headers.
- Basically, HTTP is a TCP/IP based communication protocol, which is used to deliver data (HTML files, image files, query results etc.) on the World Wide Web.
- The default port is TCP 80, but other ports can be used.
- It provides a standardized way for computers to communicate with each other.
- HTTP specification specifies how clients request data will be constructed and sent to the server, and how servers respond to these requests.

Basic Features of HTTP

- There are following three basic features which makes HTTP a simple but powerful protocol.
 - **HTTP is connection-less**
 - ❖ The HTTP client i.e. browser initiates an HTTP request and after a request is made, the client disconnects from the server and waits for a response.
 - ❖ The server process the request and re-establish the connection with the client to send response back.
 - **HTTP is media independent**
 - ❖ This means, any type of data can be sent by HTTP as long as both the client and server know how to handle the data content.

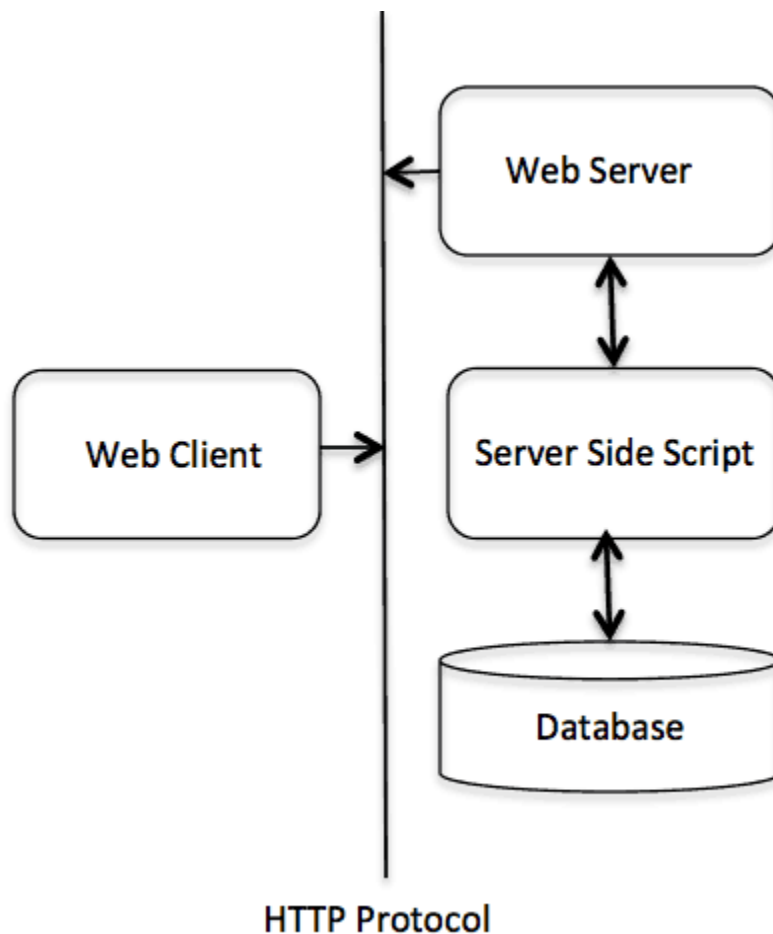
E-Content of

INTERNET TECHNOLOGY AND WEB DESIGN

- ❖ This is required for client as well as server to specify the content type using appropriate MIME-type.
- **HTTP is stateless**
 - ❖ As mentioned above, HTTP is a connection-less and this is a direct result that HTTP is a stateless protocol.
 - ❖ The server and client are aware of each other only during a current request. Afterwards, both of them forget about each other.
 - ❖ Due to this nature of the protocol, neither the client nor the browser can retain information between different requests across the web pages.

HTTP Architecture

- The HTTP protocol is a request/response protocol based on client/server based architecture where web browser, robots and search engines, etc. act like HTTP clients and Web server acts as server.



E-Content of
INTERNET TECHNOLOGY AND WEB DESIGN
FIG 8.15 : HTTP Protocol

- **Client**

- The HTTP client sends a request to the server in the form of a request method, URI, and protocol version, followed by a MIME-like message containing request modifiers, client information, and possible body content over a TCP/IP connection.

- **Server**

- The HTTP server responds with a status line, including the message's protocol version and a success or error code, followed by a MIME-like message containing server information, entity Meta information, and possible entity-body content.

HTTP Version

- HTTP uses a **<major>.<minor>** numbering scheme to indicate versions of the protocol. The version of an HTTP message is indicated by an HTTP-Version field in the first line.
- Here is the general syntax of specifying HTTP version number.

```
HTTP-Version = "HTTP" "/" 1*DIGIT "." 1*DIGIT
```

- Example

```
HTTP/1.0
```

```
or
```

```
HTTP/1.1
```

Uniform Resource Identifiers (URI)

- Uniform Resource Identifiers (URI) is simply formatted, case-insensitive string containing name, location etc to identify a resource, for example a website, a web service etc.

E-Content of INTERNET TECHNOLOGY AND WEB DESIGN

- A general syntax of URI used for HTTP is as follows.

URI = "http:" "/" host [":" port] [abs_path ["?" query]]

- Here if the **port** is empty or not given, port 80 is assumed for HTTP and an empty **abs_path** is equivalent to an **abs_path** of "/".
- The characters other than those in the **reserved** and **unsafe** sets are equivalent to their ""%" HEX HEX" encoding.
- EXAMPLE

Date/Time Formats

- All HTTP date/time stamps must be represented in Greenwich Mean Time (GMT), without exception.
- HTTP applications are allowed to use any of the following three representations of date/time stamps:

Sun, 06 Nov 1994 08:49:37 GMT ; RFC 822, updated by RFC 1123
Sunday, 06-Nov-94 08:49:37 GMT ; RFC 850, obsoleted by RFC 1036
Sun Nov 6 08:49:37 1994 ; ANSI C's asctime() format

Character Sets

- User use character set to specify the character sets that the client prefers.
- Multiple character sets can be listed separated by commas. If a value is not specified, the default is US-ASCII.
- Example, following are valid character sets

E-Content of INTERNET TECHNOLOGY AND WEB DESIGN

US-ASCII
or
ISO-8859-1
or
ISO-8859-7

Content Encodings

- A content encoding values indicate an encoding algorithm has been used to encode the content before passing it over the network.
- Content coding's are primarily used to allow a document to be compressed or otherwise usefully transformed without losing the identity.
- All content-coding values are case-insensitive. HTTP/1.1 uses content-coding values in the Accept-Encoding and Content-Encoding header fields which we will see in subsequent chapters.

Example, following are valid encoding schemes.

Accept-encoding: gzip
or
Accept-encoding: compress
or
Accept-encoding: deflate

Media Types

- HTTP uses Internet Media Types in the **Content-Type** and **Accept** header fields in order to provide open and extensible data typing and type negotiation.
- All the Media-type values are registered with the Internet Assigned Number Authority ((IANA).
- Following is a general syntax to specify media type.

E-Content of INTERNET TECHNOLOGY AND WEB DESIGN

Media-type = type "/" subtype *(";" parameter)

- The type, subtype, and parameter attribute names are case- insensitive.
- Example

Accept: image/gif

Language Tags

- HTTP uses language tags within the **Accept-Language** and **Content-Language** fields.
- A language tag is composed of 1 or more parts. A primary language tag and a possibly empty series of subtags.

Language-tag = primary-tag *("-" subtag)

- White space is not allowed within the tag and all tags are case- insensitive.
- Example tags include,

En, en-US, en-cockney, i-cherokee, x-pig-latin

HTTP request header

- The information, in the form of a text record, that a user's browser sends to a Web Server containing the details of what the browser wants and will accept back from the server.
- The requestheader also contains the type, version and capabilities of the browser that is making the request so that server returns compatible data.
- Upon receipt of the request header, the server will return an HTTP response header to the client that is attached to the file(s) being sent.

HTTP response header

E-Content of
INTERNET TECHNOLOGY AND WEB DESIGN

- The information, in the form of a text record that a Web Server sends back to a client's browser in response to receiving an HTTP request.
- The response header contains the date, size and type of file that the server is sending back to the client and also data about the server itself. The header is attached to the files being sent back to the client.
- Contrast with an HTTP request header.