

Hypertext Transfer Protocol (HTTP)

Overview

- It is an application level protocol used to transfer data over the Internet.
- The system of clients and servers transferring data via HTTP is known as the World Wide Web (WWW).
- The Web was invented by Tim Berners-Lee and colleagues in 1991.
- The original purpose of the Web was to enable physicists to easily and readily share research papers.
- The original protocol was HTTP 1.0, and it was revised to HTTP 1.1 in 1999. It was specified in RFC 2616, which has been replaced with RFC's 7230-7235.
- As a practical matter, HTTP 1.1 remains the standard, but a new standard, HTTP 2.0, has been adopted. HTTP 2, however, mainly deals with performance issues; it leaves the basic approach of HTTP 1.1 intact.

Specification-- General

- HTTP does not ensure reliability in communication; instead, it relies on the underlying transport protocol. Consequently, HTTP is transmitted over TCP/IP.
- The default port for HTTP is 80, although alternates can be used.
- The protocol is stateless, meaning that HTTP servers do not maintain any state information about previous transfers to clients.
- Originally, as part of this stateless paradigm, socket connections were not maintained after a transfer completed. Because of the rise of interaction between clients and servers, socket connections are now often maintained for some period of time; this approach is known as persistent connections.

Specification-- Structure

- HTTP takes a basic transaction-oriented approach, meaning that transfers occur through request-response pairs.
- The HTTP server, commonly referred to as a Web server, waits for connections from clients, which are typically browsers.
- Once the client and the server are connected, it is the responsibility of the client to transmit an HTTP request, to which the server will respond.
- A request is provided to a client using a Uniform Resource Locator (URL); however, the client must transform the URL into an HTTP request message.
- **NOTE:** A URL is a type of URI, and the terms are often used interchangeably (see, e.g., URI on Wikipedia).

- Each HTTP message takes the form:

```
Request-Line | Status-Line  
(message-header CRLF)*  
CRLF  
message-body?
```

- A request takes the form:

```
Method URI HTTP-Version CRLF
```

- Request Methods:

```
OPTIONS  
GET  
HEAD  
POST  
PUT  
DELETE  
TRACE  
CONNECT
```

- Request methods are explained in Sec. 4 of RFC 7231.

- Example **GET** request:

GET /home.html HTTP/1.1

- In HTTP 1.1, the **HOST** header is required:

Host: cdm.depaul.edu

- Specific request headers are explained in Sec. 5 of RFC 7231, Sec. 3 of RFC 7232, and Sec. 4 of RFC 7235.

A response begins with a Status-Line. The form for Status-Lines:

HTTP-Version Status-Code Reason-Phrase CRLF

- The Status-Code is a three digit number, while the Reason-Phrase is a short text description of the status.
- Status codes are primarily explained in Sec. 6 of RFC 7231.
- Response, or "entity," headers are primarily explained in Sec. 7 of RFC 7231.
- A few status codes and entity headers are explained in the other HTTP RFC's.