【CN】Day14

Created	@June 5, 2022 8:03 PM
Class	
Type	
Materials	HTTP Request and Response Message
Reviewed	□

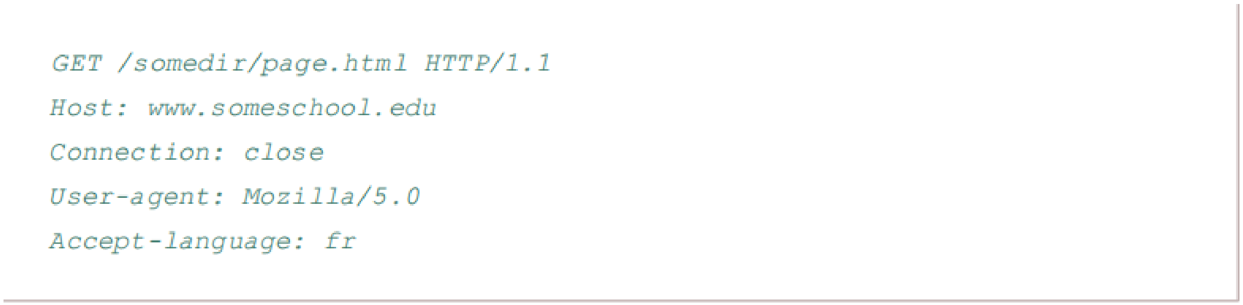
【Ch2】Application Layer

2.2.3 HTTP Message Format

There are two types of HTTP messages, request messages and response messages.

HTTP Request Message

The following is a typical HTTP request message:



```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
Connection: close
User-agent: Mozilla/5.0
Accept-language: fr
```

The first line is called the request line; the subsequent lines are called the header lines.

The request line has three fields: the method field, the URL field, and the HTTP version field.

The method field can take on several different values, including `GET`, `POST`, `HEAD`, `PUT`, and `DELETE`.

The GET method is used when the browser requests and object, with the requested object identified in the URL field.

In this example, the browser implements `HTTP/1.1`.

The header line Host: `www.someschool.edu` specifies the host on which the object resides.

By including the `Connection: close` header line, the browser wants the server to close the connection after sending the requested object.

The `User-agent:` header specifies the user agent, which is the browser type that is making the request to the server. This is important because the server can send different versions of Web pages to different browsers.

Finally, the `Accept-language:` header indicates that the user prefers to receive a French version of the object.

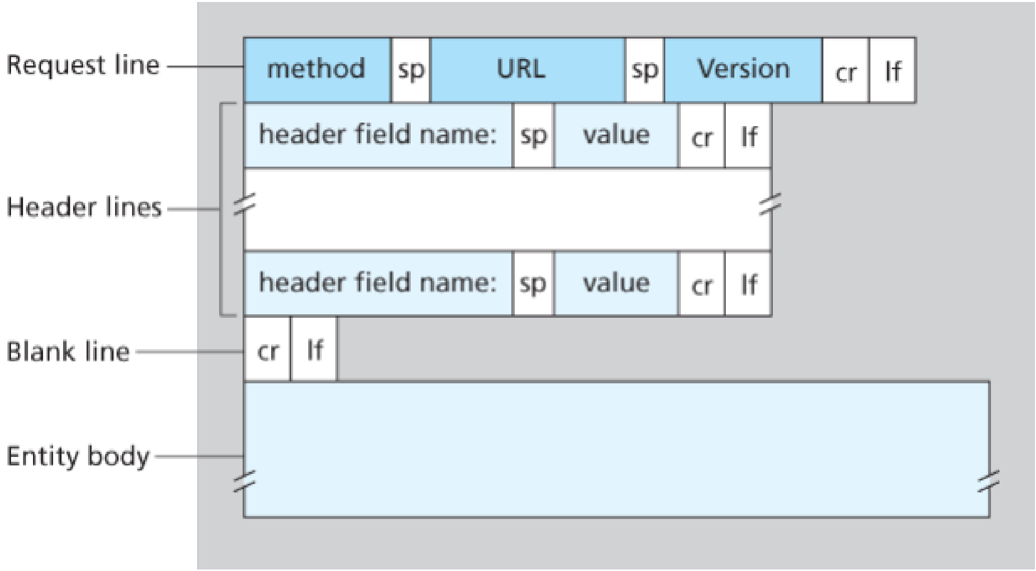


Figure 2.8 General format of an HTTP request message

The general format closely follows our earlier example. However, there is an “entity body”.

The entity body is empty with the `GET` method, but is used with the `POST` method. An HTTP client often uses the `POST` method when user fills out a form-for example, when a user search words to a search engine. If the value of the method field is `POST`, then the entity body contains when the user entered into the form fields.

A request generated with a form does not necessarily use the POST method. Instead, HTML forms often use the `GET` method and include the inputted data in the requested URL.

For example, if a form uses the GET method, has two fields, and the inputs to the two fields are `monkeys` and `bananas`, then the URL will have the structure `www.somesite.com/animalssearch?monkey&bananas`.

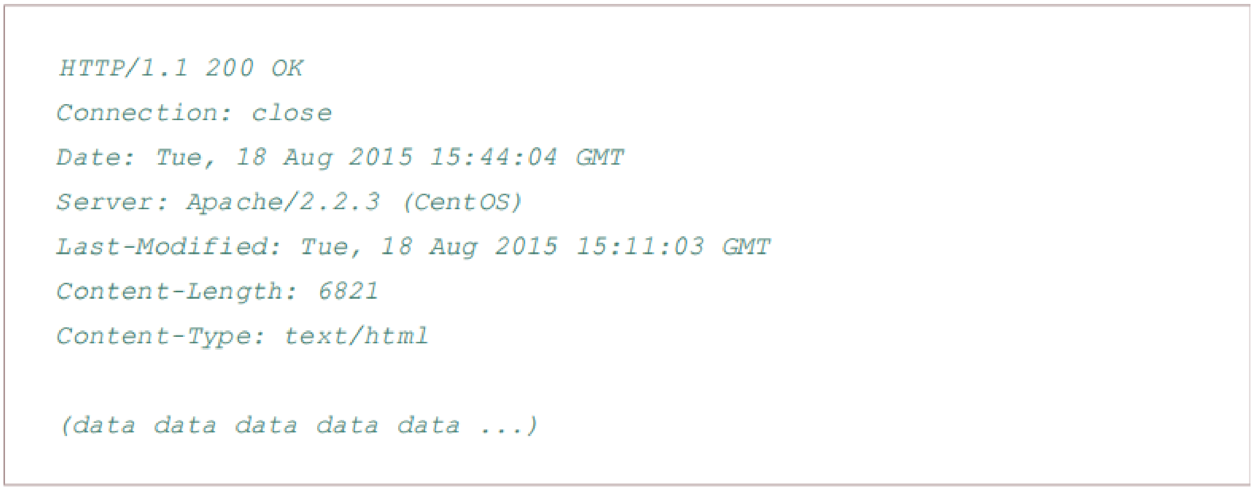
The `HEAD` method is similar to GET method. When a server receives a request with the `HEAD` method, it responds with an HTTP message but it leaves out the requested object. `HEAD` is often used for debugging.

The `PUT` method is often used in conjunction with Web publishing tools. It allows a user to upload an object to a specific path on a specific Web server.

The `DELETE` method allows a user to delete an object on a Web server.

HTTP Response Message

The following is a typical HTTP response message.



```
HTTP/1.1 200 OK
Connection: close
Date: Tue, 18 Aug 2015 15:44:04 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Tue, 18 Aug 2015 15:11:03 GMT
Content-Length: 6821
Content-Type: text/html

(data data data data data ...)
```

It has three sections: an initial status line, six header lines, and then the entity body.

The entity body contains the requested object itself.

The status line has three fields: the protocol version field, a status code, and a corresponding status message. In this example, the status line indicates that the server is using `HTTP/1.1` and that everything is OK.

Now let’s look at the headers lines. The server uses the Connection: close header line to tell client that it is going to close the TCP connection after sending the message.

The `Date:` header line indicates the time and date when the HTTP response was created and sent by the server.

The `Server:` header line indicates that the message was generated by an Apache Web server

The `Last-Modified:` header is critical for object caching, both in the local client and in network cache servers.

The `Content-Length:` header line indicates the number of bytes in the object being sent.

The `Content-Type` header line indicates that the object in the entity body is HTML text.

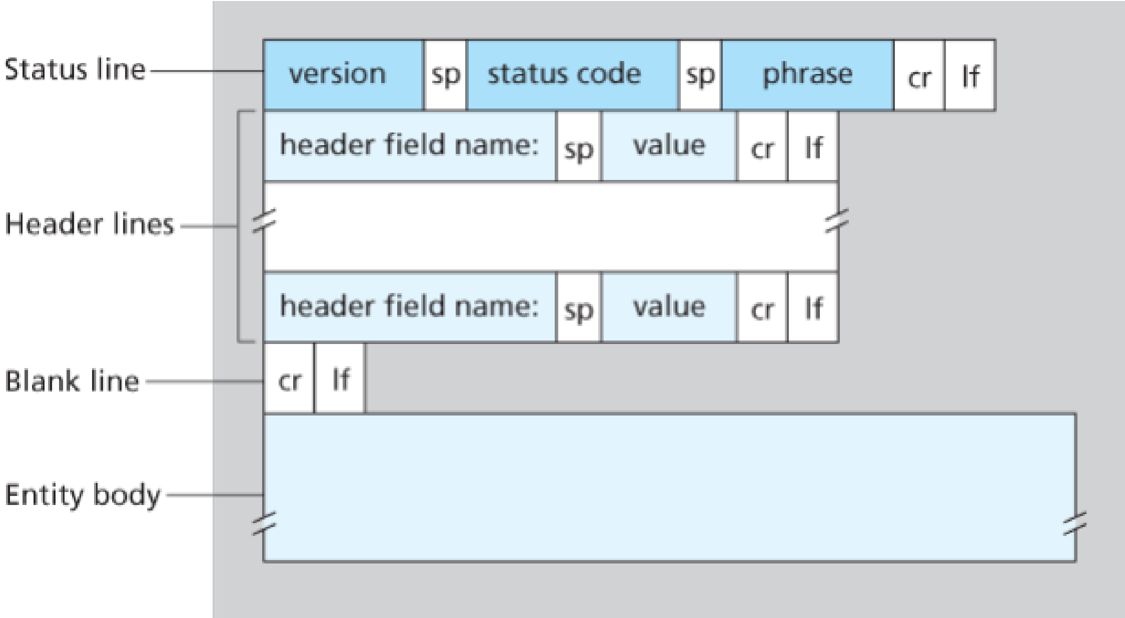


Figure 2.9 General format of an HTTP response message

The status code and associated phrase indicate the result of the request. Some common status codes and associated phrases:

- 200 OK: Request succeeded and the information is returned in the response

- 301 Moved Permanently: Requested object has been permanently moved; the new URL is specified in `Location:` header of the response message. The client will automatically retrieve the new URL
- 400 Bad Request: This is a generic error code indicating that the request could not be understood by the server.
- 404 Not Found: The requested document does not exist on this server.
- 505 HTTP Version Not Supported: The requested HTTP protocol version is not supported by the server.