**Difference between HTTP/1.1 and HTTP/2.0**

**HTTP:**

The Hyper Text Transfer Protocol “HTTP”& it is used in client – server communication. Development of HTTP was initiated by [Tim Berners-Lee](https://en.wikipedia.org/wiki/Tim_Berners-Lee) at [CERN](https://en.wikipedia.org/wiki/CERN) in 1989 and summarized in a simple document describing the behaviour of a client and a server using the first HTTP protocol version that was named 0.9.

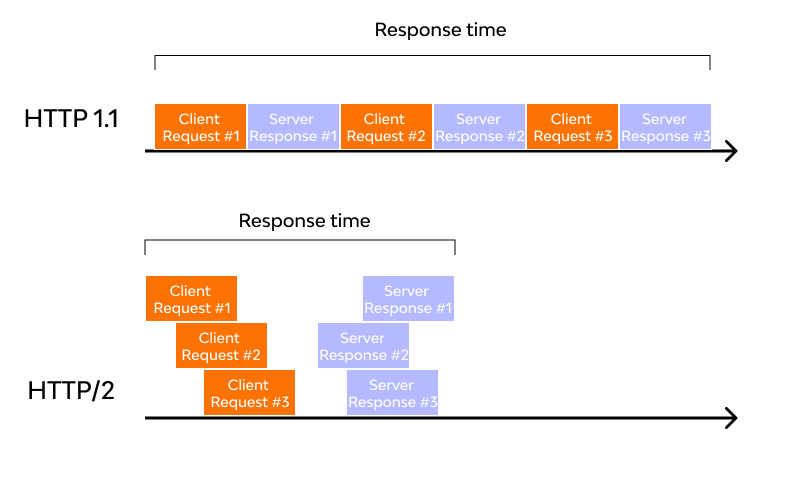
By using the HTTP user sends the request to the server & the server send the response to the user. It is a protocol used to access the data on the World Wide Web (www).

It evolved HTTP 1.1 in 1997 and then its specifications were updated in 1999, 2014, and 2022. [HTTP/2](https://en.wikipedia.org/wiki/HTTP/2), published in 2015, provides a more efficient expression of HTTP's semantics "on the wire".

The HTTP protocol can be used to transfer the data in the form of plain text, hypertext, audio, video, and so on.

[HTTP resources](https://en.wikipedia.org/wiki/Web_resource) are identified and located on the network by “[Uniform Resource Locators](https://en.wikipedia.org/wiki/Uniform_Resource_Locator)” (URLs), using the [Uniform Resource Identifiers](https://en.wikipedia.org/wiki/Uniform_Resource_Identifier) (URI's) schemes *http* and [*https*](https://en.wikipedia.org/wiki/Https). As defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [3986](https://datatracker.ietf.org/doc/html/rfc3986), URIs are encoded as [hyperlinks](https://en.wikipedia.org/wiki/Hyperlink) in [HTML](https://en.wikipedia.org/wiki/HTML) documents, so as to form interlinked [hypertext](https://en.wikipedia.org/wiki/Hypertext) documents.

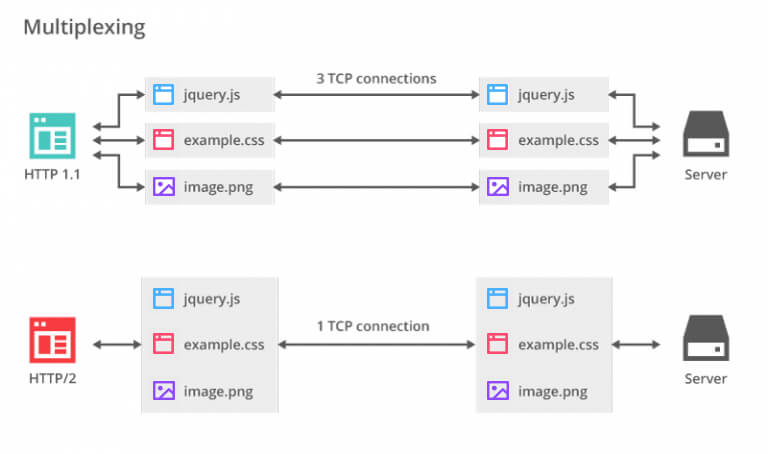
HTTP/2 improved on HTTP/1.1 in a number of ways that allowed for speedier content delivery and improved user experience, including:



**Single connection and multiplexing:**

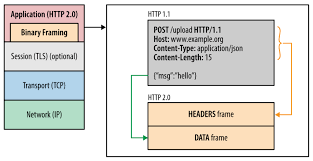
HTTP/1.1 is a sequential protocol, where the browser opens only one TCP connection, requests one file, and only after receiving the file is too large, or the server processing is too slow, the page can crash. To minimize this issue, browsers usually open multiple connections, between 6 and 8, per server. In HTTP/2 on the other hand, a TCP connection will be persistent and only one per source is needed since parallel requests and responses can request/receive all the necessary.

This brings a reduction in processing and memory consumption, a reduction in network operating cost, and increased usability. The result is reduced network latency and lower hardware and software costs.



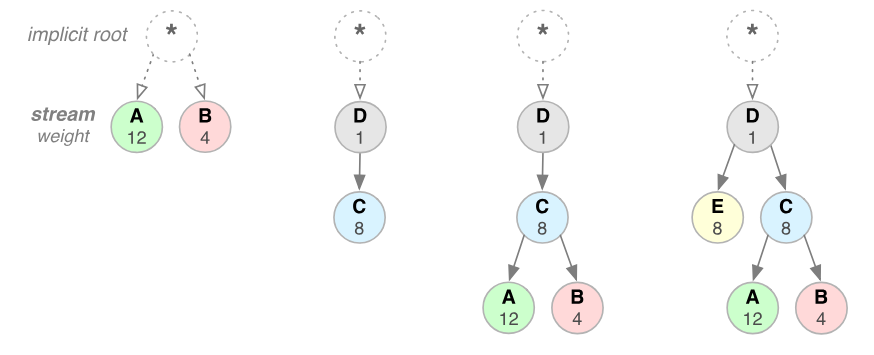
**Binary framing layer:**

This brings is the core of all HTTP/2 performance improvements, determining how HTTP messages are encapsulated and transferred between client and server. The encoding mechanism has been redesigned without changing the semantics of method, verbs, and headers. Communications are broken into frames, over a single TCP connection.



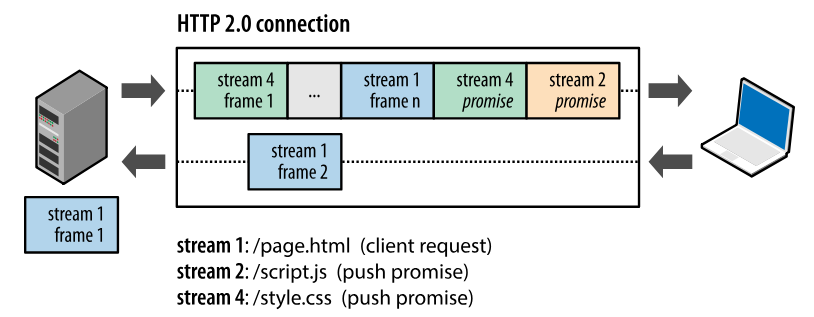
**Request Prioritization:**

It enables the browser to request all elements when discovered, communicating to the server its intention to prioritize any of them. This is done through dependencies and stream weights. An example of this is when CSS files and JS files are requested, that with HTTP/2 the browser will prioritize CSS files first, even if its request in DOM order comes after the JS.



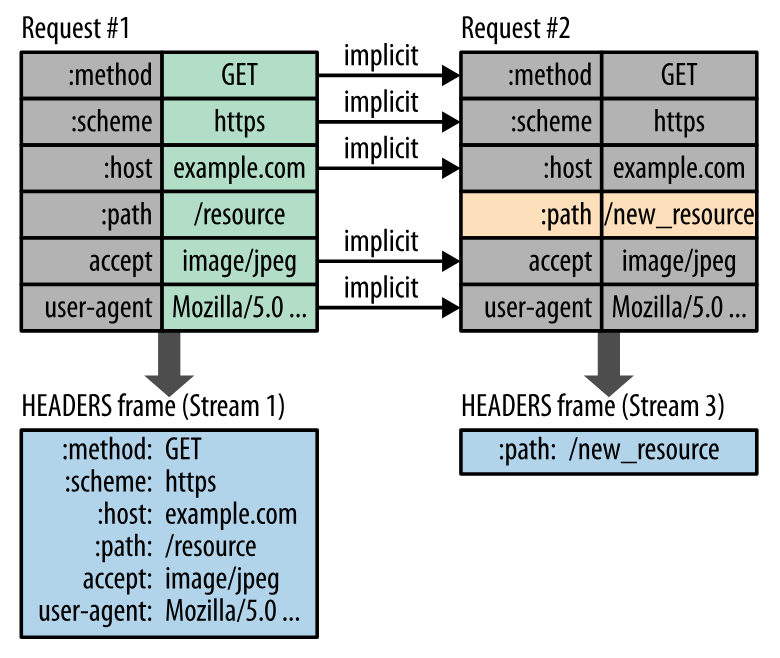
**Server Push:**

It allows the server to send multiple responses to the client, from a single request, without the client explicitly asking for it. Imagine the following scenario: a web browser requests a page’s HTML file; the server then responds with the requested file and also sends the CSS file, JavaScript, icons, and other things.



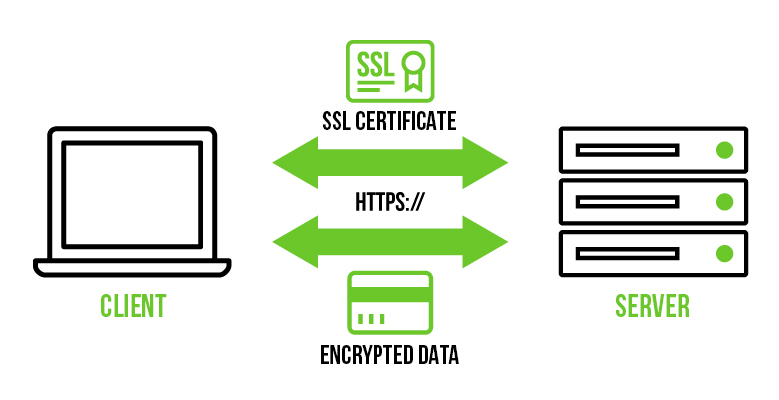
**Automatic compression:**

HTTP requests have headers with important information about the resource and its properties. With HTTP/2 the headers are compressed using the HPACK algorithm, thus reducing the size of each transfer and maintaining and updating an indexed list of the header fields seen earlier. In addition, data compression via GZIP, which needs to be enabled in HTTP/1.1, became standard in version 2.



**Data security and encryption:**

The HTTP/2 protocol has been implemented to work with or without encryption. However, all major browsers declared that they will only support HTTP/2 with encryption, requiring the use of an SSL certificate.



List of the few differences of HTTP/1.1 and HTTP/2.0

|  |  |
| --- | --- |
| HTTP/1.1 | HTTP/2.0 |
| * It use to works on the textual format * There is head of line blocking that blocks all the requests behind it until it doesn’t get it’s all resources. * It uses request resources inlining for use getting multiple pages. * It compresses data by itself. | * It works on the binary protocol.’ * It allows multiplexing so one TCP connection is required for multiple requests. * It uses PUSH frame by server that collects all multiple pages. * It uses HPACK for data compression. |