

Difference between HTTP1.1 vs HTTP2

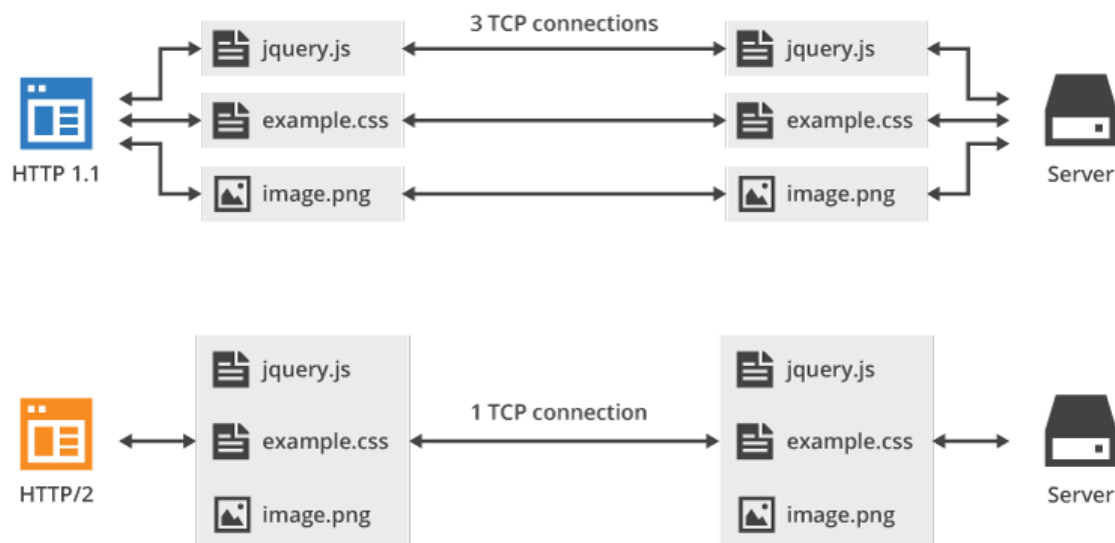
HTTP1.1 was officially released in 1997 and till 2014 HTTP1.1 had many updates and standards were added. Later in 2015 a more efficient version HTTP2.0 was released. HTTP 2.0 is known for its better performance compared to HTTP1.1.

Some of the differences between HTTP2.0 and HTTP1.1 is mentioned below.

- **Multiplexed streams**

Bi-directional sequence of text format frames sent over the HTTP2.0 protocol exchanged between the server and client are known as “streams”. In HTTP1.1 protocol were capable of transmitting only one stream at a time along with some time delay between each stream transmission.

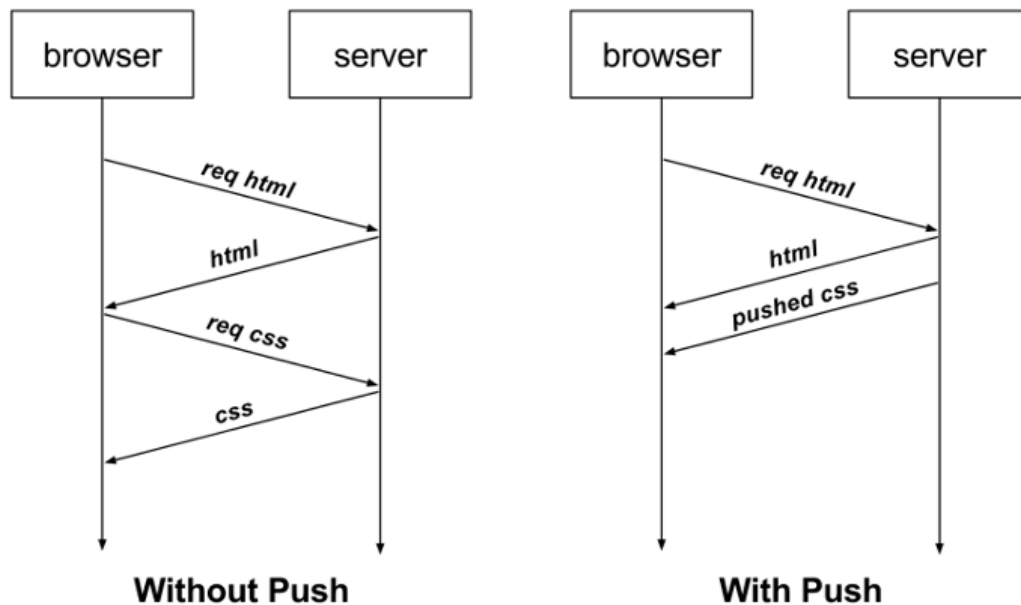
Multiplexing



- **Server Push**

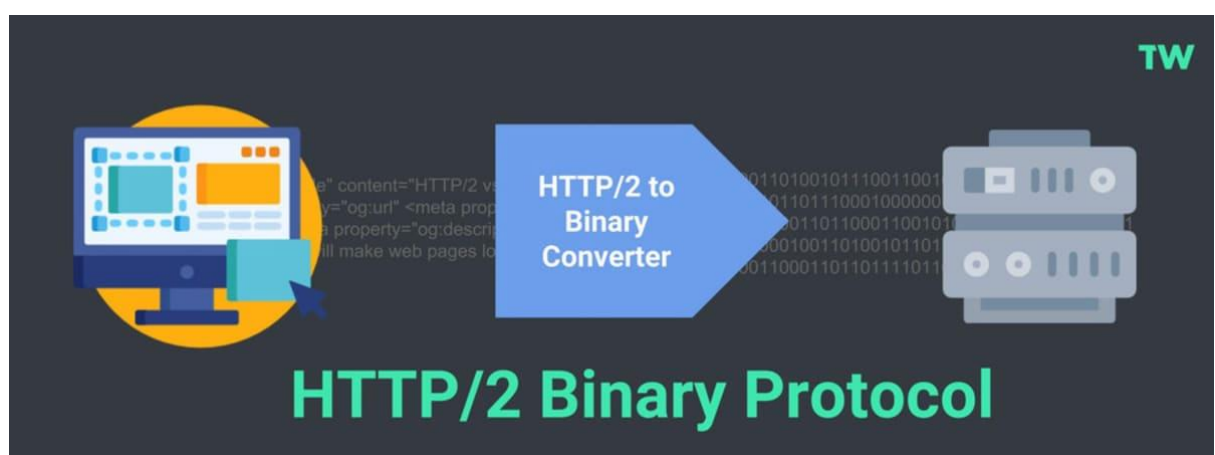
The capability allows the server to send additional cacheable information to the client that isn't requested but is anticipated in future requests. The capability to respond to the requested was only present in the HTTP1.1.

For example, if the client requests for the resource X and it is understood that the resource Y is referenced with the requested file, the server can choose to push Y along with X instead of waiting for an appropriate client request.



- **Binary Protocols**

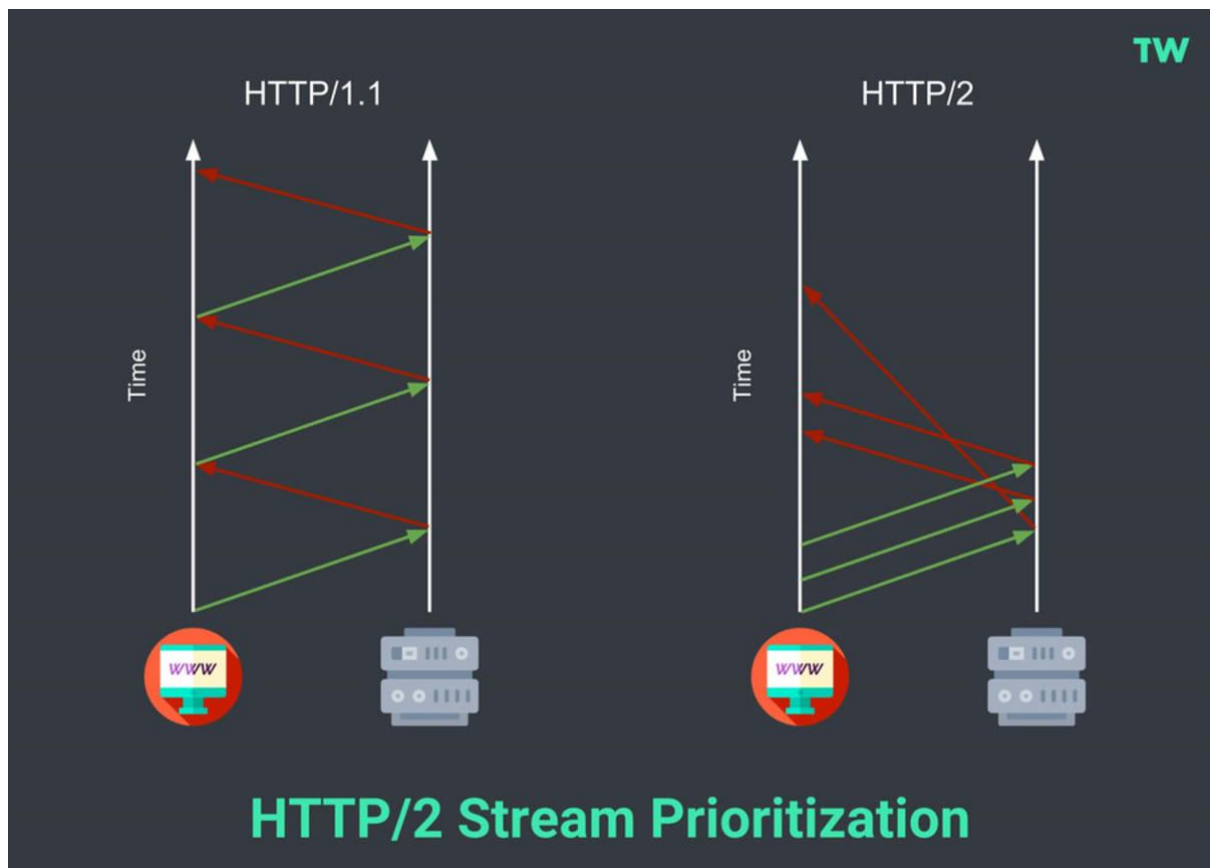
The latest HTTP version has been updated significantly in terms of capabilities and attributes, such as transforming from a text protocol to a binary protocol. HTTP1.0 used to process text commands to complete request-response cycles. HTTP2.0 will use binary commands (in 1s and 0s) to execute the same tasks.



- **Stream prioritization**

HTTP2.0 implementation allows the client to provide preference to particular data streams. Although the server is not bound to follow these instructions from the client, the mechanism

allows the server to optimize network resource allocation based on end-user requirements. In the HTTP1.1 only after a response is got the next request would be sent.



- **Header Compression**

Both HTTP1.1 and HTTP2.0 compress HTTP messages to make them smaller. However, HTTP2.0 uses a more advanced compression method called HPACK that eliminates redundant information in HTTP header packets making the packets smaller in size resulting in faster loading of data.

