HTTP/1.1

HTTP is a top-level application protocol that exchanges information between a client computer and a local or remote web server.

In this process, the client send a text based message to the server by using Get or post method.

When first response the client receives on an HTTP Get request is not shown fully rendered page instead of it contains links to the additional resources needed by the request page

In HTTP/1.0 the client had to break and remake the TCP connections with every new request but this problem is take care by the HTTP/1.1 through persistent connections and pipe lining and this allows the client would send a multiple requests along the same connections without waiting for a response

But major drawback is multiple data packets cannot pass through the same connections so that the head of the queue cannot retrieve backlog request and it block all the request (HOL)Head-of-line blocking. By adding separate or parallel connections will rectify this issue

In HTTP/1.1, flow control relies on the underlying TCP connection. When this connection initiates both client and server establish their buffer sizes using their system default settings.

If the receiver’s buffer is partially filled with data, it will tell the sender its receive window

In HTTP/1.1, if the developer knows in advance which additional resources the client machine will need to render the page, they can use a technique called resource inlining

HTTP/2.0

HTTP/2.0 is began as seedy protocol developed at google to solve the web page load latency by using the compression ,multiplexing and prioritization

While HTTP/1.1 use a request and response in a plain text format and HTTP/2.0 use the binary framing layer to retrieve in all messages in binary format

Binary framing layer encodes request/response and cut them into smaller packets of info. greatly increase the flexibility of data transfer

The interleaved requeat and ressponce run in parallel without head blocking by use of multiplexing

HTTP/2 multiplexes streams of data within a single TCP connection. As a result, receive windows on the level of the TCP connection are not sufficient to regulate the delivery of individual streams.

HTTP/2 solves this problem by allowing the client and server to implement their own flow controls, rather than relying on the transport layer.

HTTP/2 enables multiple concurrent responses to a client’s initial GET request, a server can send a resource to a client along with the requested HTML page, providing the resource before the client asks for it. This process is called server push.