HTTP/1.1 loads resources one after the other, so if one resource cannot be loaded, it blocks all the other resources behind it. In contrast, HTTP/2 is able to use a single TCP connection to send multiple streams of data at once so that no one resource blocks any other resource. HTTP/2 does this by splitting data into binary-code messages and numbering these messages so that the client knows which stream each binary message belongs to.

**Difference between HTTP/1.1 and HTTP/2 are:**

| **HTTP/1.1** | **HTTP/2** |
| --- | --- |
| It’s works on the textual format. | It works on the binary protocol. |
| There is head of line blocking that blocks all the requests behind it until it doesn’t get its all resources. | It allows multiplexing so one TCP connection is required for multiple requests. |
| It uses requests resource In lining for use getting multiple pages | It uses PUSH frame by server that collects all multiple pages |
| It compresses data by itself. | It uses HPACK for data compression. |

These are the high-level differences between HTTP1 and HTTP2:

• HTTP2 is binary, instead of textual

• HTTP2 is fully multiplexed, instead of ordered and blocking

• HTTP2 can, therefore, use one connection for parallelism

• HTP2 uses header compression to reduce overhead

• HTTP2 allows servers to “push” respzonses proactively into client caches