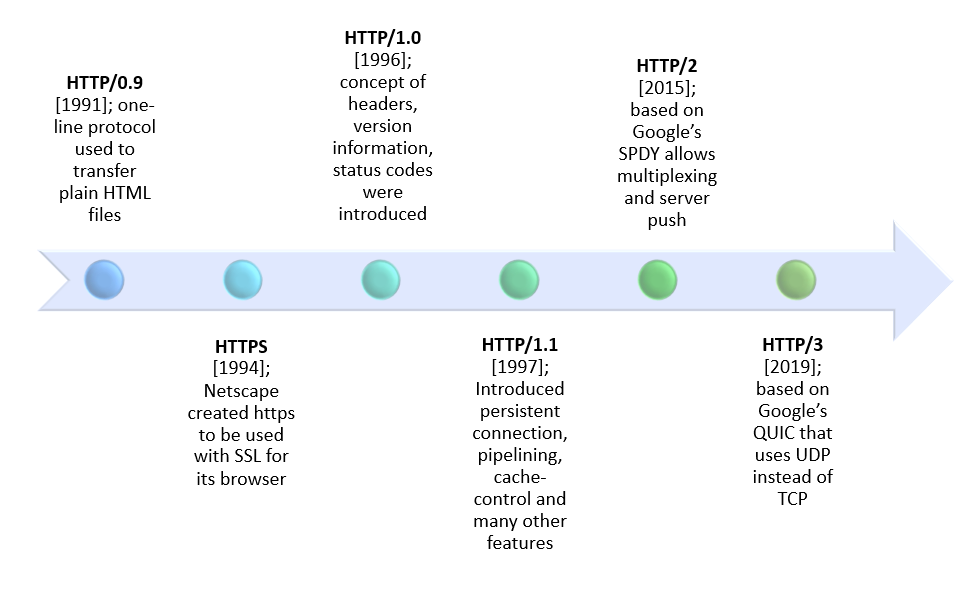
History of HTML

**HTTP** (HyperText Transfer Protocol) is the underlying protocol of the World Wide Web. Developed by **Tim Berners-Lee** and his team between 1989-1991, HTTP has seen many changes, keeping most of the simplicity and further shaping its flexibility. HTTP has evolved from an early protocol to exchange files in a semi-trusted laboratory environment, to the modern maze of the Internet, now carrying images, videos in high resolution and 3D.

* A textual format to represent hypertext documents, the *HyperText Markup Language* (HTML).
* A simple protocol to exchange these documents, the *HypertText Transfer Protocol*(HTTP).
* A client to display (and accidentally edit) these documents, the first Web browser called *WorldWideWeb*.
* A server to give access to the document, an early version of *http*.



## HTTP/0.9 – The one-line protocol

The initial version of HTTP had **no version number**; it has been later called *0.9 to differentiate it from the later versions*.

HTTP/0.9 is extremely simple:

* Requests consist of a single line
* Start with the only possible method GET followed by the PATH to the resource (not the URL as both the protocol, server, and port are unnecessary once connected to the server).

## HTTP/1.0 – Building extensibility

HTTP/0.9 was very limited and both browsers and servers quickly extended it to be more versatile:

* Versioning information is now sent within each request (HTTP/1.0 is appended to the GET line)
* A status code line is also sent at the beginning of the response, allowing the browser itself to understand the success or failure of the request and to adapt its behaviour in consequence (like in updating or using its local cache in a specific way)
* The notion of HTTP headers has been introduced, both for the requests and the responses, allowing metadata to be transmitted and making the protocol extremely flexible and extensible.
* With the help of the new HTTP headers, the ability to transmit other documents than plain HTML files has been added (thanks to the Content-Type header).

## HTTP/1.1 – The standardized protocol

In parallel to the somewhat chaotic use of the diverse implementations of HTTP/1.0, and since 1995, well before the publication of HTTP/1.0 document the next year, proper standardization was in progress. The first standardized version of HTTP, **HTTP/1.1 was published in early 1997**, only a few months after HTTP/1.0.

HTTP/1.1 clarified ambiguities and introduced numerous improvements:

* A connection can be reused, saving the time to reopen it numerous times to display the resources embedded into the single original document retrieved.
* Pipelining has been added, allowing to send a second request before the answer for the first one is fully transmitted, lowering the latency of the communication.
* Chunked responses are now also supported.
* Additional cache control mechanisms have been introduced.
* Content negotiation, including language, encoding, or type, has been introduced, and allows a client and a server to agree on the most adequate content to exchange.
* Thanks to the Host header, the ability to host different domains at the same IP address now allows server colocation.

## [HTTP/2 – A protocol for greater performance](https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/Evolution_of_HTTP#http2_%E2%80%93_a_protocol_for_greater_performance)

The HTTP/2 protocol has several prime differences from the HTTP/1.1 version:

* It is a binary protocol rather than text. It can no longer be read and created manually. Despite this hurdle, improved optimization techniques can now be implemented.
* It is a multiplexed protocol. Parallel requests can be handled over the same connection, removing the order and blocking constraints of the HTTP/1.x protocol.
* It compresses headers. As these are often similar among a set of requests, this removes duplication and overhead of data transmitted.
* It allows a server to populate data in a client cache, in advance of it being required, through a mechanism called the server push.

## Post-HTTP/2 evolution

HTTP didn't stop evolving upon the release of HTTP/2. Like with HTTP/1.x previously, HTTP's extensibility is still being used to add new features. Notably, we can cite new extensions of the HTTP protocol appearing in 2016:

* Support of cAlt-Svc allows the dissociation of the identification and the location of a given resource, allowing for a smarter CDN caching mechanism.
* The introduction of Client-Hints allows the browser, or client, to proactively communicate information about its requirements, or hardware constraints, to the server.
* The introduction of security-related prefixes in the Cookie header, now helps guarantee a secure cookie has not been altered.

This evolution of HTTP proves its extensibility and simplicity, liberating creation of many applications and compelling the adoption of the protocol. The environment in which HTTP is used today is quite different from that seen in the early 1990s. HTTP's original design proved to be a masterpiece, allowing the Web to evolve over a quarter of a century, without the need of a mutiny.

By healing flaws, yet retaining the flexibility and extensibility which made HTTP such a success, the adoption of HTTP/2 hints at a bright future for the protocol.