

HTTP1.0為了提高系統的效率, HTTP1.0規定瀏覽器與服務器只保持短暫的連接, 瀏覽器的每次請求都需要與服務器建立一個TCP連接, 服務器完成請求處理後立即斷開TCP連接。客戶端和服務器端每次建立和關閉連接是一個相對費時的過程, 並且會嚴重影響客戶機和服務器的性能。被抱怨最多的就是連接無法復用, 和head of line blocking這兩個問題。連接無法復用會導致每次請求都經歷三次握手和慢啟動。三次握手在高延遲的場景下影響較明顯, 慢啟動則對文件類大請求影響較大。head of line blocking會導致帶寬無法被充分利用, 以及後續健康請求被阻塞。

HTTP1.1為了剋服HTTP1.0的缺陷, 支持且默認持久連接, 在一個TCP連接上可以傳送多個HTTP請求和響應, 減少了建立和關閉連接的消耗和延遲。一個包含有許多圖像的網頁文件的多個請求和應答可以在一個連接中傳輸, 但每個單獨的網頁文件的請求和應答仍然需要使用各自的連接。HTTP1.1還允許客戶端不用等待上一次請求結果返回, 就可以發出下一次請求, 這樣也顯著地減少了整個下載過程所需要的時間, 但是同一個TCP連接裡面, 所有的數據通信是按次序進行的, 如果前面的處理特別慢, 後面就會有許多請求排隊等著。HTTP1.1也通過增加更多的請求頭和響應頭來改進和擴充HTTP1.0的功能。

HTTP1.1相較於 HTTP1.0協議的區別主要體現在:緩存處理、帶寬優化及網路連接的使用、錯誤通知的管理、消息在網路中的發送、網際網路地址的維護、安全性及完整性。

HTTP2.0多了幾個特性, 包含多路復用(Multiplexing)、二進制分幀層、首部壓縮(Header Compression)和服務端推送(Server Push)。多路復用允許同時通過單一的連接發起多重的請求-響應消息。HTTP2.0在應用層(HTTP2.0)和傳輸層(TCP or UDP)之間增加一個二進制分幀層, 改進傳輸性能, 實現低延遲和高吞吐量。在二進制分幀層中, HTTP2.0會將所有傳輸的信息分割為更小的消息和幀(frame), 並對它們採用二進制格式的編碼, 這些幀對應著邏輯流中的消息, 並行地在同一個 TCP 連接上雙向交換消息。首部壓縮(Header Compression)。服務端推送(Server Push)是一種在客戶端請求之前發送數據的機制。如果一個請求是由你的主頁發起的, 服務器很可能會響應主頁內容、logo 以及樣式表, 也讓在遵循規範的情況下, 不同頁面之間可以共用緩存資源成為可能。

参考：

[超文本傳輸協定 - 維基百科, 自由的百科全書 \(wikipedia.org\)](#)

[如何优雅的谈论HTTP / 1.0 / 1.1 / 2.0 - 简书 \(jianshu.com \)](#)

[面试官：说说 HTTP1.0/1.1/2.0 的区别？| web前端面试 - 面试官系列 \(vue3js.cn \)](#)