

HTTP1.0

HTTP 協議老的標準是 HTTP/1.0，為了提高系統的效率，HTTP 1.0 規定瀏覽器與伺服器只保持短暫的連線，瀏覽器的每次請求都需要與伺服器建立一個 TCP 連線，伺服器完成請求處理後立即斷開 TCP 連線，伺服器不跟蹤每個客戶也不記錄過去的請求。頻寬和延遲也是影響一個網路請求的重要因素。在網路基礎建設已經使得頻寬得到極大的提升的當下，大部分時候都是延遲在於響應速度。基於此會發現，http1.0 被抱怨最多的就是連線無法複用，和 head of line blocking 這兩個問題。理解這兩個問題有一個十分重要的前提：客戶端是依據域名來向伺服器建立連線，一般 PC 端瀏覽器會針對單個域名的 server 同時建立 6~8 個連線，手機端的連線數則一般控制在 4~6 個。顯然連線數並不是越多越好，資源開銷和整體延遲都會隨之增大。連線無法複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，慢啟動則對檔案類大請求影響較大。head of line blocking 會導致頻寬無法被充分利用，以及後續健康請求被阻塞。

HTTP1.1

為了克服 HTTP 1.0 的這個缺陷，HTTP 1.1 支援持久連線，在一個 TCP 連線上可以傳送多個 HTTP 請求和響應，減少了建立和關閉連線的消耗和延遲。一個包含有許多影像的網頁檔案的多個請求和應答可以在一個連線中傳輸，但每個單獨的網頁檔案的請求和應答仍然需要使用各自的連線。HTTP 1.1 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，但伺服器端必須按照接收到客戶端請求的先後順序依次回送響應結果，以保證客戶端能夠區分出每次請求的響應內容，這樣也顯著地減少了整個下載過程所需要的時間。

HTTP 1.1 在繼承了 HTTP 1.0 優點的基礎上，也克服了 HTTP 1.0 的效能問題。HTTP 1.1 通過增加更多的請求頭和響應頭來改進和擴充 HTTP 1.0 的功能。在 HTTP 1.1 中增加 Host 請求頭欄位後，WEB 瀏覽器可以使用主機頭名來明確表示要訪問伺服器上的哪個 WEB 站點，這才實現了在一臺 WEB 伺服器上可以在同一個 IP 地址和埠號上使用不同的主機名來建立多個虛擬 WEB 站點。HTTP 1.1 的持續連線，也需要增加新的請求頭來幫助實現。HTTP 1.1 還提供了與身份認證、狀態管理和 Cache 快取等機制相關的請求頭和響應頭。HTTP/1.0 不支援檔案斷點續傳。

由上，HTTP/1.1 相較於 HTTP/1.0 協議的區別主要體現在：

- 1 快取處理
- 2 頻寬優化及網路連線的使用
- 3 錯誤通知的管理
- 4 訊息在網路中的傳送
- 5 網際網路地址的維護

6 安全性及完整性

HTTP2.0

使用 HTTP2.0 測試便可看出 HTTP2.0 比之前的協議在效能上有很大的提升。下面總結了 HTTP2.0 協議的幾個特性。

多路複用 (Multiplexing)

多路複用允許同時通過單一的 HTTP/2 連線發起多重的請求-響應訊息。在 HTTP/1.1 協議中瀏覽器客戶端在同一時間，針對同一域名下的請求有一定數量限制，超過限制數目的請求會被阻塞。這也是為何一些站點會有多個靜態資源 CDN 域名的原因之一，拿 Twitter 為例，<http://twimg.com>，目的就是變相的解決瀏覽器針對同一域名的請求限制阻塞問題。而 HTTP/2 的多路複用(Multiplexing) 則允許同時通過單一的 HTTP/2 連線發起多重的請求-響應訊息。因此 HTTP/2 可以很容易的去實現多流並行而不用依賴建立多個 TCP 連線，HTTP/2 把 HTTP 協議通訊的基本單位縮小為一個一個的幀，這些幀對應著邏輯流中的訊息。並行地在同一個 TCP 連線上雙向交換訊息。

首部壓縮

HTTP/1.1 並不支援 HTTP 首部壓縮，為此 SPDY 和 HTTP/2 應運而生，SPDY 使用的是通用的 DEFLATE 演算法，而 HTTP/2 則使用了專門為首部壓縮而設計的 HPACK 演算法。

服務端推送

服務端推送是一種在客戶端請求之前傳送資料的機制。在 HTTP/2 中，伺服器可以對客戶端的一個請求傳送多個響應。Server Push 讓 HTTP1.x 時代使用內嵌資源的優化手段變得沒有意義；如果一個請求是由你的主頁發起的，伺服器很可能會響應主頁內容、logo 以及樣式表，因為它知道客戶端會用到這些東西。這相當於在一個 HTML 文件內集合了所有的資源，不過與之相比，伺服器推送還有一個很大的優勢：可以快取！也讓在遵循同源的情況下，不同頁面之間可以共享快取資源成為可能。