

快了，因为多个组件对象同时出现在屏幕上时，用户能够看到加载的进展。¹⁵ 如果整个屏幕上有很多动作在进行，即使实际上秒表显示整个页面的下载时间更长，人们也会认为 Web 页面加载得更快一些。

4.5 持久连接

Web 客户端经常会打开到同一个站点的连接。比如，一个 Web 页面上的大部分内嵌图片通常都来自同一个 Web 站点，而且相当一部分指向其他对象的超链通常都指向同一个站点。因此，初始化了对某服务器 HTTP 请求的应用程序很可能在不久的将来对那台服务器发起更多的请求（比如，获取在线图片）。这种性质被称为站点本地性（site locality）。

因此，HTTP/1.1（以及 HTTP/1.0 的各种增强版本）允许 HTTP 设备在事务处理结束之后将 TCP 连接保持在打开状态，以便为未来的 HTTP 请求重用现存的连接。在事务处理结束之后仍然保持在打开状态的 TCP 连接被称为持久连接。非持久连接会在每个事务结束之后关闭。持久连接会在不同事务之间保持打开状态，直到客户端或服务器决定将其关闭为止。

90

重用已对目标服务器打开的空闲持久连接，就可以避开缓慢的连接建立阶段。而且，已经打开的连接还可以避免慢启动的拥塞适应阶段，以便更快速地进行数据的传输。

4.5.1 持久以及并行连接

我们看到，并行连接可以提高复合页面的传输速度。但并行连接也有一些缺点。

- 每个事务都会打开 / 关闭一条新的连接，会耗费时间和带宽。
- 由于 TCP 慢启动特性的存在，每条新连接的性能都会有所降低。
- 可打开的并行连接数量实际上是有限的。

持久连接有一些比并行连接更好的地方。持久连接降低了时延和连接建立的开销，将连接保持在已调谐状态，而且减少了打开连接的潜在数量。但是，管理持久连接时要特别小心，不然就会累积出大量的空闲连接，耗费本地以及远程客户端和服务器的资源。

持久连接与并行连接配合使用可能是最高效的方式。现在，很多 Web 应用程序都会打开少量的并行连接，其中的每一个都是持久连接。持久连接有两种类型：比较老

注 15：渐进式图片会先显示低分辨率的近似图形，然后再逐渐增加图片的分辨率，而随着渐进式图片应用的逐步增加，这种效果就更加明显了。