

注意，范围请求也属于一类实例操控，因为它们是在客户端和服务端之间针对特定的对象实例来交换信息的。也就是说，客户端的范围请求仅当客户端和服务端拥有文档的同一个版本时才有意义。

364

## 15.10 差异编码

我们曾把网站页面的不同版本看作页面的不同实例。如果客户端有一个页面的已过期副本，就要请求页面的最新实例。如果服务器有该页面更新的实例，就要把它发给客户端，哪怕页面上只有一小部分发生了改变，也要把完整的新页面实例发给客户端。

若改变的地方比较少，与其发送完整的新页面给客户端，客户端更愿意服务器只发送页面发生改变的部分，这样就可以更快地得到最新的页面。差异编码是 HTTP 协议的一个扩展，它通过交换对象改变的部分而不是完整的对象来优化传输性能。差异编码也是一类实例操控，因为它依赖客户端和服务端之间针对特定的对象实例来交换信息。RFC 3229 描述了差异编码。

图 15-10 更清楚地展示了差异编码的结构，包括请求、生成、接收和装配文档的全过程。客户端必须告诉服务器它有页面的哪个版本，它愿意接受页面最新版的差异 (delta)，它懂得哪些将差异应用于现有版本的算法。服务器必须检查它是否有这个页面的客户端现有版本，计算客户端现有版本与最新版之间的差异（有若干算法可以计算两个对象之间的差异）。然后服务器必须计算差异，发送给客户端，告知客户端所发送的是差异，并说明最新版页面的新标识 (ETag)，因为客户端将差异应用于其老版本之后就会得到这个版本。

365

客户端在 If-None-Match 首部中使用的是它所持有页面版本的唯一标识，这个标识是服务器之前响应客户端时在 ETag 首部中发送的。客户端是在对服务器说：“如果你那里页面的最新版本标识和这个 ETag 不同，就把这个页面的最新版本发给我。”如果只有 If-None-Match 首部，服务器将会把该页面的最新版本完整地发给客户端。（假设最新版和客户端持有的版本不同。）

366

不过，如果客户端想告诉服务器它愿意接受该页面的差异，只要发送 A-IM 首部就可以了。A-IM 是 Accept-Instance-Manipulation（接受实例操控）的缩写。形象比喻的话，客户端相当于这样说：“哦对了，我能接受某些形式的实例操控，如果你会其中一种的话，就不用发送完整的文档给我了。”在 A-IM 首部中，客户端会说明它知道哪些算法可以把差异应用于老版本而得到最新版本。服务端发送回下面这些内容：一个特殊的响应代码——226 IM Used，告知客户端它正在发送的是所请求对象