

- `<depth>`  
它是 Depth 首部的值的副本。
- `<timeout>`  
指明锁的超时时间。在上面的响应中，超时值是 180 秒。

## 1. opaque-locktoken 方案

opaque-locktoken 是设计用来在所有时间内对所有资源提供唯一令牌方案的。为了确保唯一性，WebDAV 规范规定采用 ISO-11578 中描述的 UUID 机制。

434

在实际实现的时候，有一定的回旋余地。服务器可以选择为每个 LOCK 请求生成一个 UUID，或者生成单个 UUID 并通过在结尾附加额外的字符来维护唯一性。从性能角度衡量，选择后面那种方法更好。不过，如果服务器选择实现后面那种方法，就必须保证附加的扩展部分永远不会重用。

## 2. XML 元素 <lockdiscovery>

XML 元素 <lockdiscovery> 提供了发现活跃的锁的机制。如果有人试图去给已经被锁定的文件上锁，他会收到指明当前拥有者的 XML 元素 <lockdiscovery>。<lockdiscovery> 元素列出了所有未解除的锁和相应的属性。

## 3. 锁的刷新和 Timeout 首部

为了刷新锁，客户端需要重新提交锁定请求，并把锁定令牌放在 If 首部中。返回的超时值可能和早先的超时值不同。

除了接受服务器给定的超时值，客户端也可以在 LOCK 请求中指明要求的超时值。这可以通过 Timeout 首部做到。Timeout 首部的语法允许客户端在逗号分隔的列表中描述一些选项。例如：

```
Timeout : Infinite, Second=86400
```

服务器没有义务必须满足这些选项。但是，客户端必须在 XML 元素 <timeout> 中提供锁定过期的时间。无论怎样，锁定超时只是一个指导值，服务器不一定受其约束。管理员可以手工重设，某些异常事件也可能导致服务器重设锁。客户端应当避免使锁定时间太长。

尽管有这些原语，图 19-3 中显示的“丢失更新问题”并没有得到完全解决。为了彻底解决这个问题，需要带有版本控制的协作事件系统。