

(续)

状态码	定义者	方法	效果
403 Forbidden	HTTP	PROPFIND, PROPPATCH	对于 PROPFIND 来说, 客户端不允许访问该属性。对于 PROPPATCH 来说, 客户端不允许修改该属性
404 Not Found	HTTP	PROPFIND	属性不存在
409 Conflict	HTTP	PROPPATCH	与修改语义冲突——例如, 试图修改只读的属性
423 Locked	WebDAV	PROPPATCH	目标资源被锁定, 并且没有提供锁定令牌, 或者锁定令牌不匹配
507 Insufficient Storage	WebDAV	PROPPATCH	没有足够的空间登记修改的属性

## 19.2.10 集合与名字空间管理

集合是指对预定义的层次结构中的资源进行的逻辑或物理上的分组。集合的一个典型的例子就是目录。就像文件系统中的目录一样, 集合作为其他资源 (也包括其他集合, 和文件系统中的目录一样) 的容器使用。

WebDAV 使用了 XML 的名字空间机制。与传统的名字空间不同, XML 名字空间的分区在阻止所有名字空间冲突的同时, 还允许进行精确的结构控制。

WebDAV 提供了 5 种方法对名字空间进行操作: DELETE、MKCOL、COPY、MOVE 以及 PROPFIND。本章前面已经讨论过 PROPFIND 了, 下面来讨论其他方法。

## 19.2.11 MKCOL 方法

MKCOL 方法允许客户端在服务器上指定的 URL 处创建集合。乍一看, 仅仅为了创建集合而定义一个新方法好像有点多余。在 PUT 或 POST 方法之上加以修饰看起来就是个完美的替代方案。WebDAV 协议的设计者确实考虑过这些替代方案, 但最终还是选择定义一个新方法。决策背后的一些理由如下所述。

- 为了使用 PUT 或 POST 来创建集合, 客户端要随请求发送一些额外的“语义黏胶”。这当然是可以做到的, 但定义这种特别的东西总是乏味且易错的。
- 大多数访问控制机制都是建立在方法类型之上的——只有少数能在库中创建和删除资源。如果给其他方法过多的功能, 这些访问控制机制就无法运作了。

下面是请求的例子:

```
MKCOL /publishing HTTP/1.1
```