

单次调用就能获取集合的整个层次结构和其中各个独立实体的所有属性。

19.2.9 PROPPATCH方法

PROPPATCH 方法为对指定资源设置或删除多个属性提供了原子化机制。原子化可以保证要么所有请求都成功，要么跟所有请求都没发出一样。

PROPPATCH 方法的 XML 基元素是 `<propertyupdate>`。它作为一个容器使用，容纳了需要修改的属性。XML 的 `<set>` 和 `<remove>` 元素用于描述操作。

- `<set>`
指定要设置的属性值。`<set>` 含有一个或多个 `<prop>` 子元素，它们依次包含了该资源上要设置的属性的名/值对。如果属性已存在，其值就被覆盖。
- `<remove>`
指定要删除的属性。与 `<set>` 不同的是，在 `<prop>` 容器中只列出了属性的名称。

438

下面这个小例子设置并删除了 owner 属性：

```
<d:propertyupdate xmlns:d="DAV:" xmlns:o="http://name-space/scheme/">
  <d:set>
    <d:prop>
      <o:owner>Author A</o:owner>
    </d:prop>
  </d:set>

  <d:remove>
    <d:prop>
      <o:owner/>
    </d:prop>
  </d:remove>
</d:propertyupdate>
```

对 PROPPATCH 请求的响应和对 PROPFIND 请求的响应非常像。参见 RFC 2518 以获取更多信息。

表 19-2 总结了 PROPFIND 和 PROPPATCH 方法的状态码。

表19-2 PROPFIND与PROPPATCH方法的状态码

状态码	定义者	方法	效果
200 OK	HTTP	PROPFIND, PROPPATCH	命令成功
207 Multi-Status	WebDAV	PROPFIND, PROPPATCH	作用于一个或多个资源（或者集合）时，每个对象的状态都被封装到一个 207 响应中。这是一种常见的成功响应
401 Unauthorized	HTTP	PROPPATCH	需要授权才能完成对属性的修改操作