

浏览器基数很大，那么需要进行修改的时候，重新配置每个浏览器是非常困难，甚至是不可能的。

PAC 文件是一些小型的 JavaScript 程序，可以在运行过程中计算代理设置，因此，是一种更动态的代理配置解决方案。访问每个文档时，JavaScript 函数都会选择恰当的代理服务器。

142

要使用 PAC 文件，就要用 JavaScript PAC 文件的 URI 来配置浏览器 [配置方式与手工配置类似，但要在“automatic configuration”（自动配置）框中提供一个 URI]。浏览器会从这个 URI 上获取 PAC 文件，并用 JavaScript 逻辑为每次访问计算恰当的代理服务器。PAC 文件的后缀通常是 .pac，MIME 类型通常是 application/x-ns-proxy-autoconfig。

每个 PAC 文件都必须定义一个名为 FindProxyForURL(url, host) 的函数，用来计算访问 URI 时使用的适当的代理服务器。函数的返回值可以是表 6-1 列出的任意值。

表6-1 代理自动配置脚本的返回值

FindProxyForURL的返回值	描 述
DIRECT	不经过任何代理，直接进行连接
PROXY host:port	应该使用指定的代理
SOCKS host:port	应该使用指定的 SOCKS 服务器

例 6-1 中的 PAC 文件为 HTTP 事务处理指定了一个代理，为 FTP 事务处理指定了另一个代理，并为所有其他类型的事务处理使用直连方式。

例 6-1 代理自动配置文件示例

```
function FindProxyForURL(url, host) {
    if (url.substring(0,5) == "http:") {
        return "PROXY http-proxy.mydomain.com:8080";
    } else if (url.substring(0,4) == "ftp:") {
        return "PROXY ftp-proxy.mydomain.com:8080";
    } else {
        return "DIRECT";
    }
}
```

更多有关 PAC 文件的细节，请参见第 20 章。

6.4.3 客户端代理配置：WPAD

另一种浏览器配置机制是 WPAD 协议。WPAD 协议的算法会使用发现机制的逐