

网络编程

网络应用随处可见。任何时候浏览 Web、发送 email 信息或是玩在线游戏，你就正在使用网络应用程序。有趣的是，所有的网络应用都是基于相同的基本编程模型，有着相似的整体逻辑结构，并且依赖相同的编程接口。

网络应用依赖于很多在系统研究中已经学习过的概念。例如，进程、信号、字节顺序、内存映射以及动态内存分配，都扮演着重要的角色。还有一些新概念要掌握。我们需要理解基本的客户端-服务器编程模型，以及如何编写使用因特网提供的服务的客户端-服务器程序。最后，我们将把所有这些概念结合起来，开发一个虽小但功能齐全的 Web 服务器，能够为真实的 Web 浏览器提供静态和动态的文本和图形内容。

11.1 客户端-服务器编程模型

每个网络应用都是基于客户端-服务器模型的。采用这个模型，一个应用是由一个服务器进程和一个或者多个客户端进程组成。服务器管理某种资源，并且通过操作这种资源来为它的客户端提供某种服务。例如，一个 Web 服务器管理着一组磁盘文件，它会代表客户端进行检索和执行。一个 FTP 服务器管理着一组磁盘文件，它会为客户端进行存储和检索。相似地，一个电子邮件服务器管理着一些文件，它为客户端进行读和更新。

客户端-服务器模型中的基本操作是事务(transaction)(见图 11-1)。一个客户端-服务器事务由以下四步组成。

- 1) 当一个客户端需要服务时，它向服务器发送一个请求，发起一个事务。例如，当 Web 浏览器需要一个文件时，它就发送一个请求给 Web 服务器。
- 2) 服务器收到请求后，解释它，并以适当的方式操作它的资源。例如，当 Web 服务器收到浏览器发出的请求后，它就读一个磁盘文件。
- 3) 服务器给客户端发送一个响应，并等待下一个请求。例如，Web 服务器将文件发送回客户端。
- 4) 客户端收到响应并处理它。例如，当 Web 浏览器收到来自服务器的一页后，就在屏幕上显示此页。

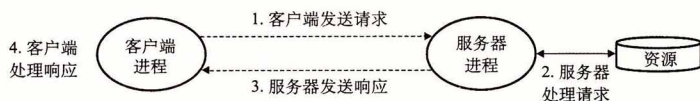


图 11-1 一个客户端-服务器事务

认识到客户端和服务是进程，而不是常提到的机器或者主机，这是很重要的。一台主机可以同时运行许多不同的客户端和服务，而且一个客户端和服务的事务可以在同一台或是不同的主机上。无论客户端和服务是怎样映射到主机上的，客户端-服务器模型都是相同的。