

计算机与网络的出现是继语言、文字、造纸与印刷术、电报与电话之后的第五次信息技术革命，计算机与网络已经深入到社会生活的各个方面。在计算机网络技术中，通常会涉及两张网——Network 和 Web。Net 的原意为渔网，Network 主要指硬件网络，它包括了 TCP/IP (Transmission Control Protocol/Internet Protocol) 四层网络体系结构中的三层，即主机-网络层、互联层、传输层，是由实现了这些层级协议的硬件设备，如网线、网卡、中继器、集线器、交换机、路由器、计算机（安装网络操作系统，即实现 TCP/IP 的操作系统）等组成的。Web 的原意为蜘蛛网，它是 World Wide Web (WWW) 的简称，中文称为万维网，主要指软件网络。Web 是 TCP/IP 网络体系结构中最顶层——应用层的应用之一。应用层中有远程登录 (Telnet)、文件传输 (FTP)、Web 应用 (HTTP)、电子邮件 (SMTP 和 POP3) 等应用，在计算机网络发展的早期，Telnet 和电子邮件曾先后是最广泛的网络应用，Web 技术出现后即呈现出爆炸式的发展趋势，目前 Web 已成为人们共享信息的主要手段，是最流行的网络应用，其他的网络应用也都趋向于通过 Web 来实现。就像网络的发展和普及形成计算机即网络的概念一样，WWW 几乎成为 Internet 的代名词。但互联网 (Internet) 并不等同于万维网 (Web)，Web 只是互联网上基于超文本的软件应用系统。

## 1.1 互联网简介

计算机网络是使用通信线路和遵循特定技术规范（即实现特定网络协议）的设备将多台计算机连接起来，并能利用软件实现通信和资源共享的系统。计算机网络按照连接设备的范围分为局域网、城域网和广域网。局域网与广域网在地理范围上的巨大差异导致其通信方式的不同，局域网的通信方式为共享网络，广域网的通信方式为点对点。网络体系结构及其通信协议和共享网络通信是局域网技术的核心内容，网络互连是广域网技术的核心内容，这些都是计算机网络课程的范畴，而点对点通信技术是通信专业的主要内容。以太网是应用最广泛的局域网。局域网使用通信线路和网络设备（一般为路由器）连接起来就成为了广域网。城域网介于局域网与广域网之间，以提供接入、媒体、信息服务为主，强调业务功能和服务质量。目前城域网仍没有完全统一的技术标准。全世界范围内的局域网、城域网、广域网互相连接而成的一个超级广域网就是 Internet，称为互联网、因特网，Internet 是网络的网络。

因特网使用 TCP/IP 协议让不同的设备可以彼此通信。但使用 TCP/IP 协议的网络并不一定是因特网，一个局域网也可以使用 TCP/IP 协议。互联网的应用方式早期主要基于大

型主机系统，目前采用客户机/服务器体系结构。小型网络中的对等网络，各台计算机既可以作为服务器，也可以作为客户机，可身兼二职，是一种特殊的客户机/服务器结构模式。Web 应用基于 Internet，其安全机制，运行、访问方式都以 Internet 环境为基础，Web 技术中的许多主题与 Internet 相关。Web 应用的应用层协议为 HTTP (Hyper Text Transfer Protocol)，底层通常采用 TCP/IP 协议。Web 应用是典型的客户机/服务器结构，服务器程序称为 Web 服务器，或 HTTP 服务器，客户端程序采用统一的浏览器，所以又称为 B/S (Browser/Server) 应用程序。

## 1.2 万维网

万维网，即 World Wide Web，简称 WWW 或 Web。万维网是互联网的功能之一。万维网是建立在客户机/服务器模型之上，以超文本标记语言 (HTML, Hypertext Markup Language) 与超文本传输协议 (HTTP) 为基础，提供面向 Internet 服务的、用户界面一致的主从结构分布式超媒体系统。其中的服务器为 Web 服务器，又称 WWW 服务器。Web 服务器提供由 HTML 编写的超文本文件 (称为网页) 和其他资源，这些网页及资源采用超链接互相连接，它们既可放置在同一台服务器上，也可放置在不同地理位置的服务器上。用户使用统一的客户端软件 (浏览器) 浏览服务器上的网页和资源，客户端浏览器的主要功能是向服务器发送 HTTP 请求并显示服务器返回的 HTTP 应答信息页。万维网的工作方式如图 1-1 所示。



图 1-1 Web 工作方式

万维网起源于欧洲粒子物理研究室 CERN (European Organization for Nuclear Research)。1989 年 3 月，欧洲粒子物理研究室的 Tim Berners-Lee (蒂姆·贝纳斯·李) 提出 Web 计划。1980 年代后期，超文本技术已经出现，但没有人能想到把超文本技术应用到计算机网络上来，超文本只是一种新型的文本而已。Tim 创造性地将超文本技术应用于计算机文件中，通过在文件中嵌入超链接，把网络上不同计算机 (主机) 上的信息连接起来，并且通过超文本传输协议 (HTTP) 在 Web 服务器和客户机之间传输。Tim 成功开发出了世界上第一个 Web 服务器 (httpd) 和第一个 Web 客户端浏览编辑程序 (World Wide Web)。1989 年 12 月，他的发明正式定名为 World Wide Web，即 WWW。1991 年 8 月 6 日，Tim 建立的第一个网站 (也是世界上第一个网站) <http://info.cern.ch/> 上线。Web 在互联网上的出现，立即引起轰动，获得了极大的成功，并被迅速推广应用。Web 出现之前，互联网上内容的表现形式极其单调枯燥，资源的访问权限也很严格，网络操作较为复杂。而 Web 服务器能发布图文并茂的信息，甚至还可以发布音频和视频信息。人们只要通过使用简单的方法，就可以很迅速方便地获取丰富的信息资料。此外，Internet 的许多其他功能，如 E-mail、Telnet、FTP 等都可以通过 Web 实现。Web 是 Internet 上取得的最激动人心的成就，为 Internet 的普及迈出了开创性的一步。Web 技术给 Internet 赋予了强大的生命力，Web 浏览的方式带来了互联网靓丽的青春。Web 技术的发明改变了全球信息化的传统模式，开启了一个信息交流的全新时代。

Tim 在发明万维网之后，又相继制定了互联网的 URI、HTTP、HTML 等技术规范。1994 年，他在美国麻省理工学院成立了非营利性的万维网联盟 W3C (World Wide Web Consortium)，邀请 Microsoft、Netscape、Sun、Apple、IBM 等共 155 家互联网著名公司，致力于 WWW 技术协议的标准化，进一步推动 Web 技术的发展。W3C 对互联网技术的发展和作用起到了基础性和根本性的支撑作用，是 Web 技术领域最具权威和影响力的国际中立性技术标准机构。Tim 被业界公认为“互联网之父”。

Web 是互联网上由许多互相链接的超文本组成的系统，在这个系统中，每个信息单元称为一个“资源”，由一个全局“统一资源定位符”(URL)来定位；这些资源主要是 HTML 编写的超文本文件(网页)；通过 HTTP 传送给用户。Web 的三个基本要素分别是：资源寻址定位的统一资源定位器 (URL)；资源访问方式的超文本传输协议 (HTTP)；以及资源表达编辑的超文本标记语言 (HTML)。

### 1.3 统一资源定位符

统一资源定位符 (URL, Uniform Resource Locator)，又称为统一资源定位器，是 Web 资源的全局地址表示，用于在互联网上寻址定位 Web 资源。与 URL 类似的还有一个统一资源标识符 (URI, Uniform Resource Identifier)，用来在互联网上唯一地标识一个资源。URL 是一种具体的 URI，URL 不仅可以用来标识一个资源，同时还能够定位这个资源。另外与 URL 并列的一个术语是统一资源命名 (URN, Uniform Resource Name)，它通过名字来标识资源，比如 `mailto:java-net@java.sun.com`。概括地说，URI 是以一种抽象的高层次概念定义统一资源标识，它可以是相对的；而 URL 和 URN 则是具体的资源标识方式，它们一般是绝对的。URL 和 URN 都是一种 URI。

**URL 的格式为：**`protocol://hostname(or IP)[:port]/website/path/[file][?query][#fragment]`

协议://主机域名 (或 IP 地址) :端口号/网站/目录/文件名?查询参数#信息片断

例如：`http://www.lyu.edu.cn:80/chpage/index.html?str=abc#a1`

URL 主要由四部分组成：

第一部分是协议 (protocol)，或称为服务方式，主要的协议或者服务方式为 `http`、`ftp`、`file`、`gopher`、`https`、`mailto`、`news`。第一部分和第二部分之间用“://”符号隔开。

第二部分是服务器的域名 (或 IP 地址) 与端口号，域名是 IP 地址的文字表示，通过 DNS 进行转换。主机域名组成形式通常为：机器名+域名+域树+域林。HTTP 的默认端口号为 80，URL 中可以省略此默认的端口号。常用协议的默认端口号分别是 `telnet: 23`；`ftp: 21`；`smtp: 25`；`pop3: 110`；`dns: 53`。

第三部分是主机上资源的具体地址，如网站名 (或虚拟目录名)、目录和文件名等。第二部分和第三部分用“/”符号隔开。

第四部分是参数，HTTP 请求时可以向服务器传递参数，参数格式为：`?名 1=值 1&名 2=值 2`，与第三部分以“?”分隔，参数之间以“&”符号分隔。

另外，在同一网站内寻址，URL 的后面还可以加上网页锚点，网页锚点是页面内部信息片断的地址，是页面内部的定位寻址机制。

## 1.4 超文本传输协议

HTTP 是 Hypertext Transfer Protocol 的缩写，即超文本传输协议。顾名思义，HTTP 提供了访问超文本信息的功能，是 Web 浏览器和 Web 服务器之间的应用层通信协议。Web 使用 HTTP 协议传输各种超文本页面和数据。HTTP 可以使浏览器更加高效，使网络传输减少。它不仅保证计算机正确快速地传输超文本文档，还确定传输文档中的哪一部分，以及哪部分内容首先显示（如文本先于图形）等。HTTP 包含命令和传输信息，不仅可用于 Web 访问，也可以用于其他因特网/内联网应用系统之间的通信，从而实现各类应用资源超媒体访问的集成。

HTTP 协议底层通常采用 TCP/IP 协议，但它并没有规定必须使用和基于的支持层。事实上，HTTP 可以在任何其他互联网协议上，或者在其他网络上实现。HTTP 只假定其下层协议提供可靠的传输，任何能够提供这种保证的协议都可以被其使用。

基于 HTTP 协议的客户端/服务器模式会话（信息交换）分四个过程：建立连接、发送请求信息、发送响应信息、关闭连接。

(1) 建立连接：客户端的浏览器向服务器端发出建立连接的请求，服务器端给出响应就可以建立连接了。

(2) 发送请求：客户端按照协议的要求通过连接向服务器端发送自己的请求。

(3) 给出应答：服务器端按照客户端的要求给出应答，把结果（HTML 文件）返回给客户端。

(4) 关闭连接：客户端接到应答后关闭连接。

HTTP 协议采用了请求/响应模型。客户端向服务器发送一个请求，请求头包含请求的方法、URL、协议版本，以及包含请求修饰符、客户信息和内容的类似于 MIME 的消息结构。服务器以一个状态行作为响应，响应的内容包括消息协议的版本，成功或者错误编码加上包含服务器信息、实体元信息以及可能的实体内容。通常 HTTP 消息有请求和响应两种类型。这两种类型的消息都由一个起始行、一个或者多个头域、一个指示头域结束的空行和可选的消息体组成。

### HTTP 请求的格式：

```
HTTP 请求方法 请求的目标资源 HTTP/版本号
HTTP 头域
空行
HTTP 消息主体
```

### 简单的请求消息示例：

```
GET/chpage/index.html.HTTP/1.1
Host:www.lyu.edu.cn
Cache-Control: no-cache
```

```
<h1>Hello</h1>
```

HTTP 请求方法包括 GET、HEAD、POST、OPTIONS、PUT、DELETE、TRACE。方法 GET 和 HEAD 应该被所有的通用 WEB 服务器支持，其他方法的实现是可选的。GET 方

法用于获取请求的目标资源。HEAD 方法也是获取请求的目标资源，只是可以在响应时，不返回消息体。POST 方法请求服务器接收包含在请求中的实体信息，可以用于提交表单，向新闻组、BBS、邮件群组 and 数据库发送消息。OPTIONS 方法返回服务器针对特定资源所支持的 HTTP 请求方法，可以利用该请求方法来测试服务器的功能性。PUT 方法向指定资源位置上传其最新内容。DELETE 方法请求服务器删除目标资源。TRACE 方法回显服务器收到的请求，主要用于测试或诊断。

浏览器中最常用的请求方式是 GET 和 POST。直接在浏览器地址栏输入访问地址所发送的请求，或提交表单发送请求时，没有设置 method 属性，或设置 method 属性为“get”，这几种都是 GET 方式的请求。GET 方式的请求会将请求参数的名和值转换成字符串，并附加在原 URL 之后，因此可以在地址栏中看到请求参数名和值。且 GET 请求传送的数据量较小，一般不能大于 2KB。POST 方式的请求通常需要设置 form 元素的 method 属性为“post”。传送的数据量较大，用户不能在地址栏里看到请求参数值，安全性相对较高。

HTTP 请求中的头域包括通用头域、请求头域和实体头域三个部分。每个头域由一个域名，冒号（:）和域值三部分组成。域名是大小写无关的，域值前可以添加任何数量的空格符，头域可以被扩展为多行，在每行开始处，使用至少一个空格或制表符。

通用头域包含请求和响应消息都支持的头域，通用头域包含 Cache-Control、Connection、Date、Pragma、Transfer-Encoding、Upgrade、Via。对通用头域的扩展要求通信双方都支持此扩展，如果存在不支持的通用头域，一般将会作为实体头域处理。

请求头域可能包含下列字段 Accept、Accept-Charset、Accept-Encoding、Accept-Language、Authorization、From、Host、If-Modified-Since、If-Match、If-None-Match、If-Range、If-Range、If-Unmodified-Since、Max-Forwards、Proxy-Authorization、Range、Referer、User-Agent。对请求头域的扩展要求通信双方都支持，如果存在不支持的请求头域，一般将会作为实体头域处理。

请求和响应消息都可以包含实体头域，实体头域包含关于实体的原信息，实体头域包括 Allow、Content-Base、Content-Encoding、Content-Language、Content-Length、Content-Location、Content-MD5、Content-Range、Content-Type、Etag、Expires、Last-Modified、Extension-Header。Extension-Header 允许客户端定义新的实体头，但是这些域可能无法被接收方识别。

### HTTP 响应的格式:

```
HTTP/版本号 状态代码 状态描述
响应头部字段
空行
响应主体
```

简单的响应消息示例:

```
HTTP/1.1 200 OK
Content-Type: text/plain
```

```
<h1>Hello World!</h1>
```

状态代码主要用于机器自动识别，状态描述主要用于帮助用户理解。状态代码由三个数字组成，第一个数字定义响应的类别，后两个数字没有分类的作用。第一个数字可能取 5 个

不同的值:

- 1xx: 信息响应类, 表示接收到请求并且继续处理;
- 2xx: 处理成功响应类, 表示动作被成功接收、理解和接受;
- 3xx: 重定向响应类, 为了完成指定的动作, 必须接受进一步处理;
- 4xx: 客户端错误, 客户端请求包含语法错误或者是不能正确执行;
- 5xx: 服务器端错误, 服务器不能正确执行一个正确的请求。

404 Not Found 是一个最为常见的状态码, 它意味着无法找到请求的资源, 通常是 URL 不正确。200 OK 是用户想要得到的, 它意味着请求被正确处理, 没有什么错误。500 Server Error 则意味着服务器遇到了问题, 不能完成请求, 通常是动态页面中的代码发生错误。

HTTP 响应中的头域包括通用头域、响应头域和实体头域三个部分。通用头域和实体头域类似于 HTTP 响应。响应头域包含 Age、Location、Proxy-Authenticate、Public、Retry-After、Server、Vary、Warning、WWW-Authenticate。对响应头域的扩展要求通信双方都支持, 如果存在不支持的响应头域, 一般将会作为实体头域处理。

超文本传输协议已经演化出了很多版本, 它们中的大部分都是向下兼容的。客户端在请求的开始告诉服务器它采用的协议版本号, 而后者则在响应中采用相同或者更早的协议版本。HTTP/0.9 已过时, 因为它只接受 GET 一种请求方法, 没有在通信中指定版本号, 且不支持请求头。由于该版本不支持 POST 方法, 所以客户端无法向服务器传递太多信息。

HTTP/1.0 这是第一个在通信中指定版本号的 HTTP 协议版本, 至今仍被广泛采用, 特别是在代理服务器中。

HTTP/1.1 是当前版本, 持久连接被默认采用, 并能很好地配合代理服务器工作, 还支持以管道方式同时发送多个请求, 以便降低线路负载, 提高传输速度。

## ➡ 1.5 Web 浏览器

浏览器是 Web 系统中的客户机程序, 用户利用浏览器来访问服务器中的网页等资源。简单地说, 浏览器是一个超文本文件解析程序, 这个程序实现了 HTTP 协议。最初的浏览器是基于文本的, 不能显示任何类型的图形信息, 也没有图形用户界面 (GUI, Graphical User Interface), 这在很大程度上限制了 Web 的使用。1993 年, 美国伊利诺伊大学 (University of Illinois) 的国家超级计算机应用中心 (NCSA, National Center for Supercomputing Applications) 发布了具有图形用户界面的浏览器 Mosaic。普通用户借助于这个工具可以方便地访问 Web, Mosaic 带来的这种强大功能和便利, 直接导致了 Web 应用的爆炸性增长。

1994 年 4 月, Mosaic 的核心开发人员 Marc Andreessen (马克 安德森) 和 Jim Clark (吉姆 克拉克, 硅图 SGI, Silicon Graphics Incorporation 创始人) 在美国加州设立了 “Mosaic Communication Corporation”。由于伊利诺伊大学拥有 Mosaic 的商标权, 且已将其转让给望远娱乐公司 (Spyglass Entertainment), 因此 Andreessen 开发团队彻底重新撰写浏览器程序的源代码, 并将浏览器名称更改为 “Netscape Navigator”, 公司名字也改为 “Netscape Communication Corporation”, 中译为 “网景”。网景公司获得了巨大的成功, Navigator 浏览器成为当时最热门、市场占有率为首位的浏览器。

1995 年, 微软向 Spyglass 公司买下 Mosaic 的授权, 以此为基础开发了 Internet Explorer, 进军浏览器市场。之后网景公司的 Netscape Navigator 与微软公司的 Internet Explorer 之间进行了“浏览器大战”。1997 年, 微软推出功能更先进的 IE 4.0。1998 年, 微软发布 Windows 98 操作系统时捆绑 IE 5, 这一系列技术攻势和市场策略使网景公司 Netscape Navigator 市场占有率从 90% 急剧下降, 最后被美国在线 (AOL) 收购。

1998 年, 网景公开了它的浏览器源码, 并重新命名为 Mozilla, 全部程序进行了重写, 2002 年发布了第一个版本。2004 年, 基于 Mozilla 源码的 Firefox 发布。Firefox 浏览器很成功, 不断地蚕食微软的 IE 浏览器市场。目前市场上的主流浏览器有谷歌公司 (Google) 的 Chrome, 挪威欧普拉软件公司 (Opera Software ASA) 的 Opera, 苹果公司 (Apple) 的 Safari, Mozilla 基金会的 Firefox, 微软的 IE (Internet Explorer)。

## 1.6 Web 服务器

Web 服务器本质上是驻留于因特网上某种类型计算机的程序, 通常将运行 Web 服务器程序的计算机称为 Web 服务器。Web 服务器一般具有独立的 IP 地址, 当 Web 浏览器 (客户端) 连到服务器上并发送 HTTP 请求时, 服务器将处理该请求并将请求的资源以 HTTP 应答返回到该浏览器上。Web 服务器是另一个实现 HTTP 协议的程序, 它的基本功能是响应和处理 HTTP 请求, 即接收、解析、应答客户端的 HTTP 请求。Web 服务器的工作原理并不复杂, 可分成 4 个步骤: 连接过程、请求过程、应答过程以及关闭连接。服务器是一种被动程序, 只有当 Internet 上其他计算机发出请求时, 服务器才会响应。

传统 Web 服务器的作用是整理和储存各种 WWW 资源, 响应客户端软件的请求, 把用户所需的资源传送到客户端机器上。目前大部分 Web 服务器除能完成这些基本功能外, 还能运行网页中嵌入的服务器端脚本程序, 这样的 Web 服务器又称为应用程序服务器, 应用程序服务器是 Web 服务器的扩展, Web 服务器可当成应用程序服务器的一个子集。HTML 文件中嵌入的服务器端脚本程序为 VB 代码的是 ASP 网页; 服务器端脚本程序为 Java 代码的是 JSP 网页; 服务器端脚本程序为 PHP 代码的是 PHP 网页。服务器端脚本程序通常具有事务处理 (Transaction Processing), 数据库连接 (Database Connectivity) 和消息传输 (Messaging) 等功能。服务器端脚本程序的强大功能使得 Web 应用可以随心所欲地实现各种复杂的商业逻辑。目前 Web 服务器成为 Internet 上最大的计算机群, Web 文档之多、功能之强大、链接的网络之广泛, 令人难以想象。

在 UNIX 和 Linux 平台下使用最广泛的 Web 服务器是 Apache, 而 Windows 平台使用的 Web 服务器为 IIS (Internet Information Server), 其他著名的 Web 服务器还有 IBM 的 WebSphere、美国在线的 AOL Server、W3C 组织的 Jigsaw、俄罗斯的 Nginx。Apache 与 Nginx 是纯静态网页服务器, 支持服务器端动态程序需要外部的功能模块, IIS 是一个强大的应用程序服务器, 支持 ASP 和 ASP.NET 动态技术, 其余几个都是基于 Java 的应用程序服务器。在选择使用 Web 服务器时应考虑的因素有: 性能、安全性、日志和统计、虚拟主机、代理服务、缓冲服务和集成应用程序等。

Apache 是世界上用得最多的 Web 服务器, 市场占有率达 60% 左右。它源于 NCSA 的 httpd

服务器，当 NCSA WWW 服务器项目停止后，那些使用 NCSA WWW 服务器的人们开始交换用于此服务器的补丁，这也是 Apache 名称的由来（a patch 补丁的异体）。世界上很多著名的网站都是 Apache 的产物，它的成功之处主要在于它的源代码开放，有一支开放的开发队伍，支持跨平台的应用（可以运行在几乎所有的 UNIX、Linux、Windows 系统平台上）以及它的可移植性等方面。

Microsoft 的 Web 服务器产品为 Internet Information Services (IIS)，IIS 是允许在公共 Intranet 或 Internet 上发布信息的 Web 服务器。IIS 是目前最流行的 Web 服务器产品之一，很多著名的网站都是建立在 IIS 平台上的。IIS 提供了一个图形界面的管理工具，称为 **Internet 服务管理器**，可用于监视配置和控制 Internet 服务。

## 1.7 资源类型标识

电子邮件初期只能传送 US-ASCII 字符的文本，为了在邮件中包含非 US-ASCII 字符的文本，以及图像、音频、视频等媒体内容，互联网应用开发者采取了多种方法，多用途网际邮件扩充协议（MIME，Multipurpose Internet Mail Extensions）设计的目标就是允许各种不同类型的文档都可能通过 Internet 邮件进行发送。MIME 通过统一的标准定义邮件内容的信息格式，使邮件发送者、邮件代理、邮件接收者能够正确地解析邮件的内容，从而达到传送多媒体的通信目的。目前 MIME 是解决邮件中包含复合型数据问题的成熟机制，而且被用于互联网上标注资源的类型。Web 中也采用了 MIME 来指定所传递的文档类型。Web 服务器在一个将要发送到浏览器的文档头部附加了 MIME 的格式说明。当浏览器从 Web 服务器中接收到这个文档时，它就根据其中包含的 MIME 格式说明来确定下一步的操作，比如，如果文档内容为文本，则 MIME 格式说明将通知浏览器文档的内容是文本，并指明具体的文本类型；如果文档内容为音频，那么 MIME 格式说明将通知浏览器文档的内容是音频，并给出音频格式的具体描述，以便于浏览器选用合适的播放程序，正确播放接收的音频文件。

MIME 信息类型（Content-Type）定义的基本格式为：媒体类型/子类型，媒体类型被用来声明数据的一般类型，而子类型则指明了数据的细节格式。顶层媒体类型有 7 种：文本媒体类型（text）、图像媒体类型（image）、音频媒体类型（audio）、视频媒体类型（video）、应用媒体类型（application）、多部分媒体类型（multipart）、报文媒体类型（message）。其中前 5 个为离散媒体类型，后 2 个为复合媒体类型。媒体类型值可能会扩张，通过扩张，子类型集合不断增长。为了确保媒体类型的子属性值发展有序、规范、风格一致，MIME 媒体类型机制建立了一个登记程序，使用 IANA（Internet Assigned Numbers Authority）作为 MIME 各个方面扩展的中心登记处，见表 1-1。

表 1-1 常见 MIME 类型

类型/子类型	扩展名	类型/子类型	扩展名
text/plain	.txt(.c .h)	application/octet-stream	.bin(.class .exe)
text/html	.html(.htm .stm)	application/postscript	.ps(.ai .eps)
text/css	.css	application/hta	.hta

(续表)

类型/子类型	扩展名	类型/子类型	扩展名
image/bmp	.bmp	application/msword	.doc(.dot)
image/gif	.gif	application/pdf	.pdf
image/jpeg	.jpeg(.jpg .jpe)	application/rtf	.rtf
image/png	.png	application/vnd.ms-excel	.xls(.xla .xlc .xlm .xlt)
image/tiff	.tiff(.tif)	application/x-javascript	.js
image/png	.png	application/zip	.zip
image/x-icon	.ico	multipart/alternative	
audio/basic	.au(.snd)	multipart/parallel	
audio/mid	.mid(.rmi)	multipart/digest	
audio/mpeg	.mp3	message/rfc822	
audio/x-wav	.wav	message/partial	
video/mpeg	.mpeg(.mp2 .mpe .mpg)		
video/quicktime	.mov(.qt)		
video/x-msvideo	.avi		

## 本章小结

本章介绍了 Web 与网络及互联网的关系、Web 的三要素、与 Web 相关的各种概念与术语。Web 是万维网 (World Wide Web) 的简称, 是互联网之上基于超文本的软件系统。Web 的三个基本要素是: 统一资源定位器 (URL); 超文本传输协议 (HTTP); 超文本标记语言 (HTML)。与 Web 相关的技术术语有: 网络 (Network)、互联网 (Internet)、万维网 (WWW World Wide Web)、W3C (World Wide Web Consortium)、统一资源定位符 (URL, Uniform Resource Locator)、统一资源命名 (URN, Uniform Resource Name)、统一资源标识符 (URI, Uniform Resource Identifier)、超文本传输协议 (HTTP)、Web 浏览器、Web 服务器、多用途网际邮件扩展协议 (MIME, Multipurpose Internet Mail Extensions)。

## 思考题

1. 什么是 Web? 简述 Web 与 Internet 的关系。
2. 简述 Web 的三个基本要素。
3. “互联网之父”是谁? 简述 W3C 组织及其作用。
4. 写出 5 个主流的 Web 浏览器。
5. 写出 4 个常用的 Web 服务器。