

Novell Netware

网络应用软件开发指南

夏洪山 编著

科学出版社
龍門書局

Novell NetWare

网络应用软件开发指南

夏洪山 编著

科学出版社
龙门书局

1 9 9 7

(京)新登字 092 号

内 容 简 介

本书共十五章,系统地介绍了 Novell NetWare 网络操作系统文件服务、同步事务处理、目录服务、计帐服务、对等通信、装订业务、队列管理、事务跟踪处理、连接与工作站管理、网络打印、网络诊断技术以及网络增值服务等基本原理与应用接口技术,并结合具体程序设计实例,介绍了 NetWare 网络应用软件的开发技术和调试策略。

本书取材新颖、论述通俗、技术实用,适合计算机网络、计算机软件以及计算机应用等专业的大专院校师生、科研与计算机软件开发工作者使用与参考。与本书配套的软盘另售,详情见书末订单。

需要购买本书或技术咨询的读者,请与 010-62562329, 010-62541992 或传真 010-62579874 联系。

Novell NetWare 网络应用软件开发指南

夏洪山 编著

责任编辑 朱培华

科学出版社出版
北京 100080

北京东黄城根北街 16 号

邮政编码:100717

兰空印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

*

1997 年 1 月第 一 版 开本:787×1092 1/16

1997 年 1 月第一次印刷 印张:19

印数:1-5000

字数:438 000

ISBN7-03-005586-1/TP·654

定价:23.00 元

前 言

90年代的今天,计算机的普及应用已经进入了“网络计算”时代。独立的单台计算机应用,已经发展成为资源共享、任务分布或采用客户/服务器体系结构的多计算机协同的网络处理。

近10年来,Novell公司的网络操作系统NetWare,在网络技术的应用中发挥了极其重要的先导作用。在今天的网络产品市场上,Novell NetWare产品应用最为广泛,约占全球(包括中国)局域网软件市场的五成之多。NetWare的核心技术之一——IPX/SPX协议,已经成为举世公认的工业标准,为众多的网络硬件产品所支持。

Novell公司在经历了近20年的发展之后,如今是世界上第四大软件制造商。从当初只能支持星型结构和打印共享的网络软件,发展成现在能够支持80x86保护模式多操作系统环境的NetWare 4.1版,使得UNIX,DOS,Windows以及OS/2等系统之间能够平滑连接,共享资源。NetWare系统功能强大,开放性能好。它不仅提供网络通信服务和文件服务,而且开放网络打印服务、队列服务、计帐服务以及网络资源管理等功能,使得用户不仅能够方便地使用NetWare系统本身的功能,而且还能够利用它的应用接口,开发满足实际应用要求的网络应用软件,进一步发挥网络资源的积极作用。

本书共分十五章十三个专题,分别介绍NetWare的基本功能与主要技术特征,以及应用这些技术的应用软件开发技术。本书将对每个专题从三个基本方面进行阐述。首先,将介绍专题技术的实现原理、技术特性和功能特点。然后介绍实现该技术或服务功能的NetWare应用接口。在此基础上,将结合所介绍的技术,通过一个或多个程序范例,介绍利用NetWare应用接口实现具体应用软件功能的程序设计技术、实现方法和调试技巧。本书中介绍的NetWare应用接口函数是一些最常用的和最主要的,并在程序实例中予以应用,以进一步加深对它们的理解。本书从实用出发,对所介绍的一些常用数据和特性力求以表格化归纳,以便在开发网络应用软件的过程中查阅和参考。

本书在第一章中首先对NetWare的体系结构、基本特性以及网络通信协议特点进行介绍。第二章主要介绍本书中所遵循的一些程序设计规则、NetWare应用接口的引用技术以及软件开发环境的设置。文件服务是NetWare系统中最基本的功能。第三章将介绍NetWare的文件、文件属性、文件路径、文件句柄以及文件安全等资源的管理技术。通过程序实例,介绍如何在应用程序中实现对网络系统中的文件进行管理。第四章介绍同步事务处理技术。这是网络数据库应用的关键技术之一。在这一章中,将介绍六种加锁技术,并通过程序实例,介绍加锁与同步技术的应用方法。目录管理对于DOS用户不太重要。然而,在NetWare网络环境下,如何通过对目录的管理,实现对网络用户和数据资源的管理,则是网络管理的重要任务。因此,第五章将介绍NetWare目录管理功能,以及与目录相关的管理技术、网络托管业务和数据安全保障技术。这一章的应用程序主要用于示范如何实现网络目录安全的管理。在第六章中,将介绍NetWare系统的计帐收费技术以及帐务管理技术。我们将通过程序范例,介绍在应用软件中实现网络计帐收费的方法。第七章和第十三章都是着重介绍NetWare的核心技术之一——IPX/SPX协议及其应用。第七章首先介绍这两种协议的基本原理和基

本结构,以及与通信服务相关的通信事务处理技术。在程序设计实例中,介绍一种基于 IPX 实现点——点通信的方法,以示范如何开发网络通信软件。第十三章重点介绍在 IPX/SPX 通信技术的基础上如何诊断和获得网络硬件配置和软件设置信息,以及它们的运行状态信息。在 NetWare 网络系统中,任何一项资源和服务都是 NetWare 的管理对象,如何将这些对象纳入 NetWare 的管理之中(称装订),这是第八章的内容。在这一章中,我们通过对装订对象特性以及特性值的介绍,了解装订对象管理的原理和技术。所给程序范例将介绍如何应用 NetWare 装订技术,了解系统中指定装订对象的信息,供有关对象处理使用。第九章介绍队列管理技术,以及作业调度与对象服务概念,并设计一个应用程序,将本地作业提交给网络服务处理。网络数据的可靠性与完整性保障是 NetWare 系统的一个重要性能,因此在第十章中将介绍实现这些性能的事务跟踪处理概念与技术,并通过程序实例说明如何在应用程序中实现这种功能。在网络系统中,工作站总是与文件服务器息息相关,它们之间的通信连接管理是第十一章的内容。在这一章中,我们将要介绍工作站、服务器及其关系以及对它们进行管理的技术,并通过程序设计范例,介绍文件服务器如何管理注册的工作站,以及工作站如何向文件服务器申请注册或注销。打印共享是 NetWare 系统最原始的功能之一。我们将在第十二章中介绍网络打印技术,以及如何在应用程序中实现网络打印共享。在实际应用中,一个用户如何向网络提供新的服务(功能)?又如何使用户通过网络来共享这种新的服务?一个用户又如何通过网络说明需要何种服务?这就是第十四章中介绍的网络广播与控制技术,以及实现这些要求的软件开发技术。第十五章将介绍网络增值服务程序的开发技术,首先介绍 NetWare 的进程管理技术,以便掌握进程控制原理,以开发网络增值服务程序,并通过具体程序实例,介绍增值服务程序的设计方法、调试策略与排错技巧。

在本书的编写和出版过程中,得到了北京希望电脑公司以及南京大学张福炎教授的热情支持,熊可宜审校了全稿,谨表示衷心感谢。

由于作者水平所限,书中内容错误难免,敬请读者不吝指正。

编著者

1996 年 2 月 18 日于南京

目 录

前言	
第一章 Novell NetWare 的体系结构	1
1.1 NetWare 网络操作系统	1
1.2 工作站与 SHELL 软件	2
1.3 NetWare 网络通信协议简介	3
1.4 NetWare 的文件服务系统	4
第二章 NetWare 应用软件开发环境	5
2.1 中断调用规则	5
2.2 Intel 数据存储格式	7
2.3 编程语言	7
2.4 函数和变量的命名规则	7
2.5 NetWare 函数的调用规则	8
第三章 文件服务	9
3.1 NetWare 下的文件路径	9
3.2 文件服务应用接口	10
3.3 文件的删除与恢复	14
3.4 文件服务应用举例	16
第四章 同步事务处理	21
4.1 同步问题	21
4.2 加锁及其对性能的影响	22
4.3 同步事务处理应用接口	23
4.4 应用举例	34
第五章 网络文件目录管理	44
5.1 有关目录服务的几种表格	44
5.2 目录服务应用接口	46
5.3 目录管理应用举例	63
第六章 计帐服务	70
6.1 网络资源的管理	70
6.2 计帐参数的设置	70
6.3 计帐服务应用接口函数	78
6.4 计帐服务应用举例	81
第七章 对等通信	88
7.1 IPX/SPX 协议通信原理	88
7.2 通信服务应用接口函数	94

7.3	IPX 通信应用举例	106
第八章	装订业务	127
8.1	装订对象	127
8.2	装订对象特性	129
8.3	装订对象特性值	132
8.4	装订文件	133
8.5	装订业务应用接口函数	133
8.6	装订业务应用举例	148
第九章	队列管理	156
9.1	队列管理的特点	156
9.2	队列与装订对象管理	157
9.3	队列管理应用接口函数	160
9.4	队列管理应用举例	173
第十章	事务跟踪处理	188
10.1	基本原理	188
10.2	应用接口函数	189
10.3	应用举例	192
第十一章	连接与工作站管理	198
11.1	连接服务	198
11.2	工作站管理	199
11.3	连接与工作站管理应用接口函数	203
11.4	应用举例	212
第十二章	网络打印技术	222
12.1	网络打印进程的捕获	222
12.2	捕获打印标志	223
12.3	网络打印控制	225
12.4	打印服务应用接口函数	226
12.5	网络打印应用举例	232
第十三章	NetWare 网络诊断技术	236
13.1	基于 IPX 的网络诊断	236
13.2	基于 SPX 的网络诊断	240
13.3	诊断服务应用接口函数	241
13.4	NetWare 网络诊断技术的应用	248
第十四章	网络广播控制	259
14.1	SAP 的标识广播	259
14.2	定位增值服务	261
14.3	网络广播应用举例	262
第十五章	网络增值服务	267
15.1	增值服务程序设计	267

15.2	增值服务程序结构.....	269
15.3	进程控制问题.....	271
15.4	增值服务程序的调试策略.....	271
15.5	增值服务应用接口函数.....	272
15.6	网络增值服务应用举例.....	278
附录 NetWare 应用接口函数返回码.....		291
参考文献.....		296

第一章 Novell NetWare 的体系结构

早期的 Novell 网络操作系统是一个基于 MC68000 微处理器的网络软件 ShareNet,它仅仅支持星型结构的局域网。自从微型计算机 PC/XT 问世之后,Novell 逐步转向开发以 DOS^①为基础的网络文件服务系统,即 NetWare。从组成结构上来看,NetWare 包含三大基本部分:

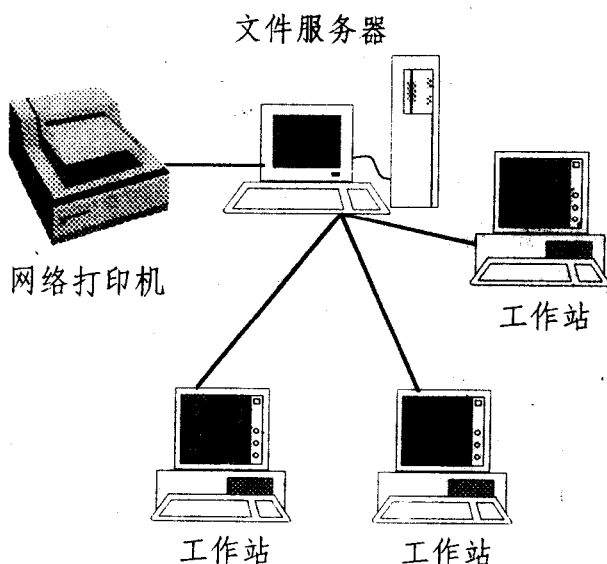
(1)网络操作系统——这是 NetWare 软件系统的管理核心,提供开放的网络操作服务,支持不同软硬件平台的文件服务,充分适应“网络计算”的技术发展趋势。

(2)工作站接口——一种 Shell 软件,在网络通信控制和 DOS 应用之间提供一个通信界面,使它们之间的网络通信连接接口标准化。

(3)网桥/路由器——实现多个网络的互联。

1.1 NetWare 网络操作系统

NetWare 网络操作系统的核心服务,是一个面向多用户的多任务管理系统。虽然它面向的,大部分应用都是基于 DOS 系统,但是 NetWare 的控制内核已经完全独立于 DOS,并充分利用 CPU 的特性(如保护方式),提供网络文件服务、通信服务和打印服务等增值服务功能。换言之,NetWare 系统通过文件服务器,对连接在局域网上的计算机系统提供一系列的服务操作,例如数据的集中存储管理、网络数据存取的协调控制、某些共享资源(打印机)的分配以及通信控制等,如图 1.1 所示。



① 注:PC 机泛指以 Intel 80x86 CPU 为处理器的个人计算机。DOS 泛指支持 PC 机的微型计算机磁盘操作系统。

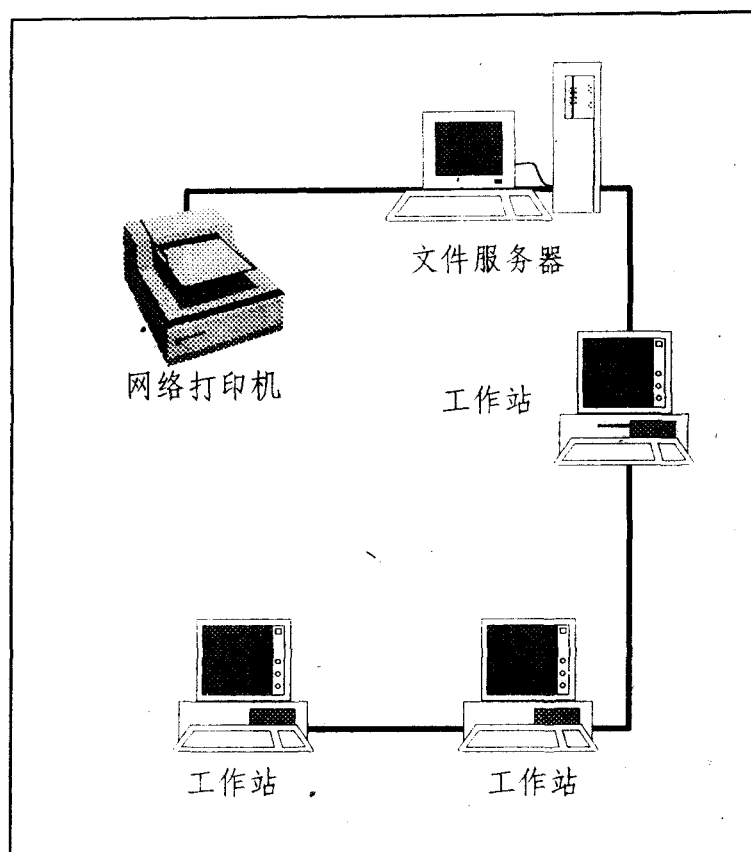


图 1.1 NetWare 文件服务器与局域网的关系

1.2 工作站与 SHELL 软件

如图 1.1 所示, NetWare 支持的局域网结构是文件服务器——工作站方式。工作站是用户应用程序运行的基本环境, 是接收文件服务器各种服务、共享网络资源的“客户”(Client)。NetWare 网络操作系统在工作站一端, 通过它的用户接口 NetWare SHELL 软件, 完成 DOS 应用或其他类型应用与 NetWare 系统的平滑连接。NetWare 的 SHELL 软件包含以下几部分:

(1) 网络通信驱动——NetWare 网络操作系统支持若干种不同厂家和不同协议的网络通信接口设备。通过不同的网络通信驱动程序, 实现各种不同网络通信设备之间的协议转换。

(2) 网络通信协议控制——在 NetWare 网络操作系统内部, 通过 IPX 协议或 SPX 协议, 进行局域网上应用系统之间的数据交换。

(3) DOS 接口——将 DOS 应用的各种网络操作要求转换成 NetWare 系统的内部操作, 实现 DOS 应用与网络软件的无缝连接。

图 1.2 描绘了 DOS 与 NetWare SHELL 软件之间的关系。其中, NIC 为网络接口控制, 即网络接口设备驱动程序。NETx.COM 为 NetWare 网络操作系统的 DOS 接口软件。IPX.COM 为 NetWare 的网络通信协议(IPX 和 SPX)控制程序。

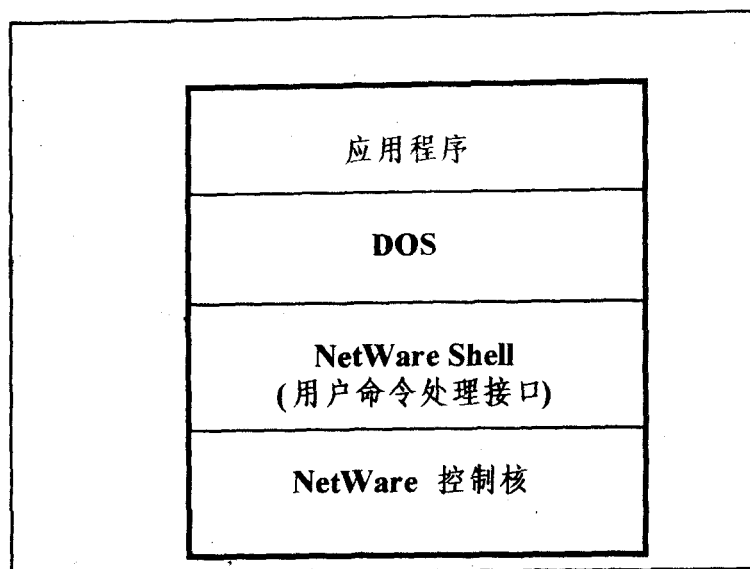


图 1.2 工作站上 NetWare SHELL 与 DOS 之间的关系

1.3 NetWare 网络通信协议简介

在 NetWare 网络操作系统中,不管采用的网络拓扑结构类型是总线型、令牌环型还是星型,局域网上各工作站之间以及工作站与文件服务器之间的数据通信,都可以采用 NetWare 系统提供的 IPX 或 SPX 协议来实现。

1.3.1 IPX 协议

IPX 协议,即网间数据包交换协议(Internet Packet Exchange),是一种快捷简便的通信协议。根据 IPX 的协议要求,将一组数据按 IPX 协议格式组成数据包(Packet),文件服务器或工作站可以将它们发往目的站,或者从其他节点站(node)接收这些数据包。NetWare 网络操作系统通过提供桥接或路由功能,使得位于不同网段的工作站之间按 IPX 协议进行数据的透明交换。

IPX 协议以最快最简洁的方式向目的站传递数据包,而不管对方是否已经正确地接收。换言之,IPX 协议的通信过程不需要接收方的确认。由于发送方不需要等待对方的确认,从而减少了发送过程因等待确认而消耗的时间,相对而言,通信速度就快了。之所以采用这种协议,出于两种最主要的原因。第一,IPX 协议具有 CRC 校验功能,可以保证数据包本身的正确性。理论与实践表明,IPX 协议中的 CRC 校验码可以保证数据传输的误码率低于 10^{-6} 。第二,如果接收方在规定的时限内不作出响应,IPX 将会自动重发。在使用 IPX 协议通信过程中不建立收发之间的“连接”。另一个原因是,对于要求高的通信应用,可以采用下面将要介绍的 SPX 协议来实现。

1.3.2 SPX 协议

SPX 协议,即有序数据包交换协议(Sequenced Packet Exchange),是 IPX 协议的扩展。它要

求在数据通信的收—发方之间建立通信“连接”，保证数据包有序地、确定地进行传输。这种面向“连接”的通信协议，要求数据接收方在接收到数据包之后向发送方发送“确认”信息，从而保证数据包有序地正确传送。SPX 协议还具有流量控制功能，避免信道堵塞。有关 IPX/SPX 的详细介绍，参见第七章。

1.3.3 NetBIOS 协议

NetBIOS(Network Basic Input/Output System)协议，是 IBM 公司开发的“站—站”通信标准协议。由于这种协议被广泛应用，Novell 公司考虑到 NetWare 网络系统与那些基于 NetBIOS 的应用之间的兼容性，在 NetWare SHELL 中提供了一个 NetBIOS 接口程序 (NetBIOS.COM)，用于 NetBIOS 协议与 IPX 协议之间的功能转换。IPX、SPX 以及 NetWare NetBIOS 三者与 ISO 的 OSI 协议模型之间的对应关系如图 1.3 所示。

1.3.4 NCP 协议

NetWare 网络操作系统中定义了一种服务协议，用于表达通信过程中对文件服务器的文件服务请求，如磁盘访问请求、目录服务请求等。这种服务协议就是 NetWare Core Protocol，它是 NetWare 核心服务协议，用于 NetWare 网络操作系统内部的控制与管理。

OSI	NetWare		TCP/IP	NetBIOS
应用层	Netware 应用 Netware 核心服务 Netware 增值服务			
表示层		NCP		
会话层	NetBIOS 仿真			NetBIOS
传输层	SPX		TCP	
网络层	IPX		IP	
链接层	Open Data Link Interface			
物理层	802.3, 802.5, ...			

图 1.3 IPX/SPX 以及 NetBIOS 与 OSI 模型的对应关系

1.4 NetWare 的文件服务系统

NetWare 的文件服务系统是通过文件服务器实现的。为了快速而有效地响应来自工作站的文件服务请求，NetWare 自身有一个完全独立于 DOS 的操作系统，用于负责文件的读/写、目录管理、磁盘空间映射等服务，使工作站与文件服务器之间透明地“连接”在一起，为应用提供了一个功能远远超过应用载体——工作站本身的运行环境。有关“文件服务”的内容，我们将在第三章中详细介绍。

第二章 NetWare 应用软件开发环境

NetWare 应用软件,是指利用 NetWare 软件系统提供的各种功能来实现自己的应用目标而开发并运行在 NetWare 环境下的用户程序啊荒远闲颯▷庵衷啊 etWare 环境下的程序设计,必须遵循 NetWare 系统的一些重要规则来调用 NetWare 系统的功能。

2.1 中断调用规则

在 NetWare 系统中,面向应用程序开放的 IPX.COM 程序和 NETx.COM 程序使用了许多系统中断,如表 2.1 所示。

表 2.1 NetWare 系统使用的中断一览表

中断调用程序	中 断 号	作 用
IPX.COM	INT 08H	IPX 用于定时
	INT 2FH	IPX 经 INT 2FH 转向调用 INT 7AH
	INT 64H	IPX 的内部中断,同 INT 7AH
	INT 7AH	IPX 中断,实现 IPX 的不同功能
	INT F4H INT F6H	IPX 的内部中断
NETx.COM	INT 10H	BIOS 的 INT 10H,用于显示
	INT 17H	BIOS 的 INT 17H,用于并行打印
	INT 1BH	Ctrl-Break, Ctrl-C 中断处理
	INT 20H	程序中止
	INT 21H	NetWare 的 API 调用
	INT 24H	显示网络出错信息
	INT 27H	程序驻留中断
	INT F5H	内部使用

从表 2.1 可以看出,NetWare 系统的功能通过两种中断方式向应用程序开放:INT 7AH 和 INT 21H。

2.1.1 INT 7AH

NetWare 系统中有关 IPX 或 SPX 协议的各种功能处理,主要是通信控制,应用程序均通过 INT 7AH 来调用。根据 NetWare 网络系统软件的特性,考虑到在不同机器中的安装,INT 7AH 的处理程序地址(即中断向量)因系统而异。因此,一般来说,应用程序必须首先通过 INT 2FH 获得 INT 7AH 的中断向量指针,然后进行引用功能处理。其过程如下列几行语句所示:

```

;Entry of IPX/SPX Processor
IPXAddress dw 2dup(?)
; int IPX Initialize(void);
;returns TRUE if IPX is installed, or FALSE.
_IPXInitialize proc near
move     ax, 7A00H
INT      2Fh
mov      IPXAddress,di      ; offset of IPXprocessor
mov      IPXAddress+2,ES    ; segment of IPXprocessor
and      ax, 00FFh
ret
_IPXInitialize endp

```

在 _IPXInitialize 子程序中,IPX 处理程序的地址存放在 IPXAddress 中。在引用 INT 7AH 处理 IPX 的功能时,采用下列等效调用。例如打开插口:

```

mov al,socket_type      ;socket type
mov dx,socket_num       ;socket number in hi-low
mov bx,0                 ;open socket
call dword ptr IPXAddress ;INT 7AH

```

_IPXInitialize 子程序有两种调用方式。一种方式是用汇编语言,如上所示,通过 Call 语句调用,返回值在 AX 寄存器内:

```
call _IPXInitialize
```

另一种方式是用 C 语言调用,返回值是一个整型数:

```
int IPXInitialize();
```

在后面的章节中我们会看到用汇编语言编写的子程序,同时还定义了对应的 C 语言函数格式,它们都可以在 C 语言程序中调用。

2.1.2 INT 21H

NetWare 网络系统中除 IPX/SPX 处理之外的所有功能调用,如文件服务、队列管理等等,均称为 API(应用程序接口)调用,都需要通过 INT 21H 来引用。NetWare 系统中的 INT 21H 调用是在 DOS 的 INT 21H 功能之上扩展的,并与 DOS 系统兼容。例如,NetWare 系统中用于同步处理的加锁服务功能子程序 SetLockMode,可以采用以下两种方式之一来实现:

方法一,用汇编语言编程,例如:

```

_SetLockMode proc near
mov AH, ocbh      ;功能号
mov AL, 001h      ;lock mode
int 21H
ret
_SetLockMode endp

```

方法二,用 C 语言编程:

```

void SetLockMode(void){
    inregs.h.ah=0xc6          /* 功能号 */
    inregs.h.al=0x01          /* lock mode */
    intdos(& inregs,&outregs); /* int 21H */
    return(outregs.h.al);}     /* return */

```

尽管实现 SetLockMode 的方法不同,但它们的 C 语言调用定义都应是统一的:

```
void SetLockMode(void);
```

2.2 Intel 数据存储格式

我们知道,在 PC 系统中,字结构的数据存储格式不同于普通的书写格式。例如,字 (WORD)结构十进制数据 256 的十六进制书写格式为 0X0100,但存放在存储器中则是以下形式:

低地址 0001 高地址

而十六进制数 0X12345678 则存放成两个字的结构:

低地址 78 56 34 12 高地址

这就是 low-high 顺序的 Intel 数据存放格式,即数据的最低有效位存放在低地址端,最高有效位存放在高地址端。

令人遗憾的是,NetWare 系统中使用 high-low 的顺序存放数据,这是因为早期 NetWare 受 MC68000 CPU 的影响而沿袭至今。因此,我们在调用 NetWare 的功能时,有的参数值要做 low-high 到 high-low 存放格式的变换。

2.3 编程语言

迄今为止,用于开发 NetWare 应用程序的程序设计语言有好多种,如 Microsoft C, Borland C, Lattice C 和 WATCOM C 等。在本书的例子程序中,我们使用 Microsoft C(6.0 版以上,小模式)和 MASM 宏汇编两种语言。尽管使用的编程语言可能不同,但是调用 NetWare 的功能来实现网络应用的方法、原理、规则等都是通用的。

2.4 函数和变量的命名规则

在应用程序中,我们可能同时采用汇编程序和 C 语言程序来实现子程序或程序段。其中,一些调用频度高的子程序,尤其是调用 NetWare 功能的子程序,其命名方法将遵循匈牙利符号规则,即,函数名用其功能的英文单词组合而成(参见程序范例)。此外,我们将尽可能使用 NetWare 系统中推荐的函数名称和相应的参数。

例如,用于打开 IPX 插口的函数调用可以定义成:

```
int IPXOpenSocket(BYTE * nSocketNumber, BYTE nSocketType);
```

如果我们用 C 语言来实现,对应的函数定义如下:

```
int IPXOpenSocket(BYTE * nSocketNumber, BYTE nSocketType)
```

```
{  
.....  
}
```

如果我们用汇编程序来实现对应于上述定义的 IPXOpenSocket 函数,则具体程序如下。需要说明的是,为了从 C 语言程序中调用这个函数,要保证子程序入口参数的一致性。因此,下列程序中开头的几行关于 BP 指针的运算,其作用是调整入口参数指针:

```
-IPXOpenSocket proc far  
    push bp  
    mov bp,sp  
    mov bx,[bp+6]  
    mov dl,[bx]  
    mov dh,[bx+1]  
    mov al,[bp+8]  
    mov bx,0  
    call far ptr IPXAddress  
    pop bp  
    and ax,0FFh  
    retf  
-IPXOpenSocket endp
```

从上面的例子可以看到,函数或子程序名 IPXOpenSocket 就是“打开插口”的意思,便于记忆和拼写。

变量的命名规则也是如此,例如上面的 nSocketType 用于存放插口类型。一般地,字符变量以小写 c 打头,整型数以小写 n 打头,布尔量以小写 b 打头,指针以小写 p 打头,长指数以 lp 打头等。

2.5 NetWare 函数的调用规则

正如本章第一节中所介绍的,NetWare 的功能是通过 INT 7AH 或 INT 21H 引用的。这些 NetWare 功能的引用,就是应用程序要调用的 NetWare 函数。在调用 NetWare 函数过程中,函数的入口参数或返回参数通常由具体的函数定义确定。在使用汇编语言编程时,入口参数或返回参数的设置具有以下规则:

参 数 性 质	寄存器及其参数用途
入 口 参 数	BX=函数调用号
	ES: SI=源数据单元指针
	ES: DI=目标数据单元指针
返 回 参 数	AL=返回码

当参数较多时,具体的函数还要使用其他几个寄存器,这需要参考相关的技术手册。

第三章 文件服务

文件服务是 NetWare 的主要功能之一。虽然 NetWare 的 SHELL 软件与用户的接口均为 DOS 命令,但是 NetWare 已经摆脱了 DOS 的约束。它不仅提供面向 DOS 用户的多任务网络操作环境,而且具有类似 UNIX 的许多特点,以保证各用户的文件服务请求能够得到高效的并发处理。NetWare SHELL 提供的类似 DOS 的文件服务,全部面向用户开放,同时还向用户开放涉及文件服务器一级的网络文件服务,供用户开发适合自己的应用。

通过 NetWare 的文件服务接口,应用程序可以对网络文件进行拷贝、换名、删除以及恢复等操作。

3.1 NetWare 下的文件路径

在 NetWare 系统下,当我们需要调用系统的文件服务功能时,系统需要提供完整的文件路径。NetWare 系统中的文件路径与 DOS 环境下的文件路径不完全相同,因此,我们在这里需要加以特别说明。

在 DOS 环境下,磁盘名是路径名的最高级存储装置,然后才是文件所在的目录。DOS 下完整的文件路径名结构如下:

磁盘名:\目录名\文件名.文件扩展名

在 NetWare 环境下,磁盘已经被动态分配了,可能被分配给若干个文件服务器,然后又被某一文件服务器分成若干个文件卷,若干个目录等等。这种环境下的完整路径名结构是:

文件服务器名/卷名:目录名\文件名.文件扩展名

由此我们可以看出 NetWare 和 DOS 这两种环境下的文件路径名的差异了。

在 NetWare 系统中,关于文件路径名的应用有时可以简化。对于每一台工作站来说,它们在文件服务器上有一张对应于卷名或目录名的表格,表格中的指针就是目录(或卷)句柄(handle),可以与一个完整的文件名一起使用。例如,假设文件服务器为“FILESERVICE”,目录句柄 4 对应的目录名是“SYS:EXAMPLE\ABC”,文件名是“XYZ.DAT”。当我们进行网络文件服务操作时,向文件服务器发送一个请求包,指定文件路径名为:

FILESERVICE/SYS:\EXAMPLE\ABC\XYZ.DAT

当然,对于用户来说,并不需要知道这个目录句柄。如果我们省略了文件服务器名,则默认为当前文件服务器。目录名或文件名可以使用通配符 * 或?,其含义与 DOS 系统下的完全一样。

在对具体文件进行查找时,NetWare 提供了一个“查找属性字节”。通过对属性字节的不同设置,指示文件服务器按不同的要求进行查找。文件“查找属性字节”如表 3.1 所示。