44 Network Programming

"Ignoring an insult is smart."

This chapter will be useful for the people who are working with LAN. Novell Netware and Windows NT are the most widely used Network Operating Systems. These Network Operating Systems help to link the computers present on LAN and support resource sharing.

44.1 Novell Netware

Novell Netware *was* the widely used Network Operating System by many LAN users. Nowadays, Windows NT is getting popularity because of its tight security. And most of the people who use Novell Netware has moved to Windows NT.

Until version 4, Novell Netware uses DOS as a bootstrap loader. One of the interesting programming for Novell Netware is 'Chat' program that helps to communicate with other users on the Network. Quite honestly, now Novell Netware is obsolete. And so explaining Novell Netware Programming will be boring. Actually Novell Netware also uses 'interrupts' like DOS. For the interrupts used by Novell Netware, please refer the Ralf Brown's interrupt list found in CD 💿.

44.1.1 Network Library

I told you, Network Programming is just an interrupt programming. The Network library called **Netware C Library 1.6** by **Adrian Cunnelly** has implemented most of the necessary functions using interrupts. So for the easy programming, we can use this library. The Basic Registration fee is £10.00 which includes the latest version of the library, royalty-free use of all library functions, unlimited technical support, and low-cost upgrades. A disk containing the full source code of the library is also available for £35.00

The library includes:

- Workstation Functions (`GetConnectionID`, `GetDefaultConnectionID`, `GetNetwareShellVersion`, etc.)
- Message Functions (`BroadcastToConsole`, `GetBroadcastMessage`, `GetPersonalMessage`, `LogNetworkMessage`, `SendBroadcastMessage`, `SendPersonalMessage`, etc)
- File Functions (`EraseFiles`, `PurgeAllErasedFiles`, `ScanFileInformation`, etc)
- Directory Functions (`AddTrusteeToDirectory`, `GetDirectoryPath`, etc)

- Print Functions (CancelLPTCapture, GetBannerUserName, GetPrinterStatus, etc)

and many more useful Network functions. It is found in CD.

44.1.2 Example – Toserver.c

The following is the example code that uses the **Netware C Library 1.6**. This code is for sending message to the server. To compile this program, you need the respective header file and library file. Please look into the CD for a complete working version of the program.

```
/********************************************************************/
/* File:            TOSERVER.C                                      */
/*                                                                  */
/* Function:        Send message to the default server             */
/*                                                                  */
/* Usage:           toserver "message"                             */
/*                                                                  */
/* Functions Called: BroadcastToConsole                            */
/*                                                                  */
/********************************************************************/

#include "netware.h"

#include <stdio.h>

int main (int argc,char *argv[]);
int main (int argc,char *argv[])
{
   if (argc !=2)
   {
      printf("Usage is 'toserver message'\n");
      return(-1);
   }
   else
      return(BroadcastToConsole(argv[1]));
}
```

Note
This program would compile only in Tiny memory model.

44.1.3 Example – Ulist.c

This is another example code that uses the **Netware C Library 1.6**. This code is for getting the statistics about the logged in users.

```
/**********************************************************************/
/* File:            ULIST.C                                       */
/*                                                                */
/* Function:        List all users that are currently logged into the*/
/*                  default server, and some useful stats (only if   */
/*                  calling user has console operator rights).     */
/*                                                                */
/* Usage:           ulist                                         */
/*                                                                */
/* Functions Called: GetConnectionNumber                          */
/*                  GetConnectionInformation                      */
/*                  GetConnectionsUsageStatistics                 */
/*                                                                */
/**********************************************************************/
#include <conio.h>
#include <dos.h>
#ifndef TURBOC
#include <search.h>
#endif
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <time.h>
#include "netware.h"

#define    FALSE    0
#define    TRUE     (!FALSE)

static char *days_of_week[] = { "Sun" , "Mon" , "Tue" ,
                                "Wed" , "Thu" , "Fri" ,
                                "Sat" };


/**********************************************************************/

void main()
{
unsigned int    station;
long object_id;
word object_type;
char object_name[OBJECT_LENGTH];
char logintime[7];
int thisone;
long systemelapsedtime;
double bytesread,byteswritten;
long totalrequestpackets;
char c;
```

```
/* Here, we loop through all the possible stations (connections).  */
   if((thisone=GetConnectionNumber()) == 0)
   {
      printf("*** No netware shell loaded ***\n");
      exit(255);
   }

   printf("                            ---Login----");
   printf("      -----file bytes------   request\n");
   printf("conn User Name         day     time");
   printf("      read           written   packets\n");
   printf("=================================");
   printf("      =============================\n");

   for (station=1; station<100; station++)
   {
        GetConnectionInformation( station , object_name,
                                  &object_type,&object_id,
                                  logintime);
        if (object_name[0]!=0)
        {
           if (thisone==station) c='*'; else c=' ';
           printf(" %2u %c%-16s %-3s %02d:%02d:%02d",
                   station , c , object_name ,
                   days_of_week[ logintime[6] ],
                   logintime[3],logintime[4],logintime[5] );
           if(GetConnectionsUsageStatistics( station,
                   &systemelapsedtime ,
                   &bytesread,&byteswritten,&totalrequestpackets)==0)
                   printf("      %-10.0f %10.0f   %7ld\n",
                           bytesread,byteswritten,totalrequestpackets);
           else
                   printf("\n");
        }
    }
}
```

44.2 Windows NT

Windows NT is another famous Network Operating System. We cannot program it from TC/DOS. The fact is Windows NT does **not** have DOS. The 'command prompt' of Windows NT is just a DOS Emulator. Windows NT uses different technologies from other Windows versions like 95/98. Windows 95 and Windows 98 are the GUIs (Graphical User Interface) running above DOS. Whereas Windows NT is a pure 32 bit Operating System. And so programming Windows NT from DOS is not possible.