

# **SX.25 Administrator's Guide**

Document Number 007-2179-002

## CONTRIBUTORS

Production by Gloria Ackley  
Engineering contributions by Inna Liou and Mary Artibee

© Copyright 1994, Silicon Graphics, Inc.— All Rights Reserved

This document contains proprietary and confidential information of Silicon Graphics, Inc. The contents of this document may not be disclosed to third parties, copied, or duplicated in any form, in whole or in part, without the prior written permission of Silicon Graphics, Inc.

## RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure of the technical data contained in this document by the Government is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 52.227-7013 and/or in similar or successor clauses in the FAR, or in the DOD or NASA FAR Supplement. Unpublished rights reserved under the Copyright Laws of the United States. Contractor/manufacturer is Silicon Graphics, Inc., 2011 N. Shoreline Blvd., Mountain View, CA 94039-7311.

Silicon Graphics and IRIS are registered trademarks and IRIX is a trademark of Silicon Graphics, Inc.

---

# Contents

	<b>About This Guide</b>	xi
<b>1.</b>	<b>Introduction to SX.25 Administration</b>	<b>1</b>
	Network Administration	2
	Installation	2
	Maintenance	3
	Bringing Changes into Effect	3
	Kernel Configuration	4
	<i>Pad</i> Configuration	6
	X.25 Addresses	6
<b>2.</b>	<b>Configuration Using the <i>x25adm</i> Utility</b>	<b>9</b>
	Network Management Options	10
	Accessing Menus in <i>x25adm</i>	10
	Kernel Configuration	14
	IXE Kernel Parameters	14
	LAPB Kernel Parameters	14
	<i>Pad</i> Kernel Parameters	15
	SLD Kernel Parameters	16
	WAN Kernel Parameters	16
	X.25 Kernel Parameters	17
	XTY Kernel Parameters	18

- Network Configuration 18
  - Configuring a WAN Subnetwork 20
  - Configuring a Loopback Driver 21
  - Configuring a LAN Subnetwork 22
  - Configuring the FDDI Interface 24
- Database Tables Configuration 26
  - Menu Options 27
  - Configuring the Pad Daemon 28
  - Configuring the Pad Database 30
    - Adding a Pad Entry 30
    - Facilities 32
    - Altering a Pad Entry 34
    - Deleting a Pad Entry 35
    - Listing Pad Database Entries 36
  - Configuring the PVC Database 36
  - Configuring the Subnetworks Database 38
    - Adding a Subnetwork Entry 38
    - Deleting or Altering a Subnetwork Entry 39
    - Listing Subnetwork Entries 40
  - Configuring the Wanmap Database 40
    - Changing a Wanmap Table 41
    - Adding a Wanmap Entry 42
    - Deleting or Altering a Wanmap Entry 42
    - Listing Wanmap Entries 43
    - Deleting a Wanmap Table 43
    - Displaying a Wanmap Table 43
- Bringing Changes into Effect 44

- 3. Parameter Configuration 47**
  - X.25 Configuration Files 47
    - Modifying Files 50
      - Creating a template 51
      - Modifying an existing template file 52
    - X.25 Parameters 53
      - Specifying Logical Channel Ranges 55
      - Sequence Numbering 56
      - Packet and Window Sizes 57
      - Setting Timer Values 59
      - Setting Counter Values 61
      - Transit Delay Parameters 62
      - Throughput Class Parameters 62
      - Closed User Groups 64
      - Subscription Options 65
      - Localization Information 67
      - D-Bit Control 69
      - Throughput Class Window/Packet Parameters 71
    - X.25 Level II (LAPB) Parameters 74
      - Link Level Conformance 77
    - WAN Parameters 78
  - Changing a Stack File 82
- 4. Running SX.25 with TCP/IP 87**
  - IP to X.25 Configuration 88
  - IXE Kernel Configuration 89
  - IXE Parameter Configuration 91
  - IXE Network Configuration 95

- IXE Database Configuration 99
  - Address Mapping 99
    - CCITT Mode 99
    - DDN Mode 100
    - Making Changes 101
  - Adding an IXEmap Database Entry 101
  - Altering an IXEmap Entry 104
  - Deleting an IXEmap Entry 104
  - Listing IXEmap Database Entries 106
- 5. **UUCP Configuration** 107
  - Configuring the XTY System 107
  - Files Associated with UUCP 108
    - Devices 109
    - Dialcodes 109
    - Permissions 109
    - Dialers 110
    - Systems 110
- 6. **Troubleshooting** 111
  - STREAMS Logging Facilities 111
    - STREAMS Trace Messages 111
      - X.25 PLP Driver 112
      - LAPB Driver 114
      - LLC2 Driver 116
    - Tracing Summary 118
  - STREAMS Error Messages 118
  - Error Messages 119
    - Network Board-Loading (nload) Error Messages 119
    - Network Daemon (snetd) Error Messages 119
- Glossary** 123
- Index** 127

---

# Figures

<b>Figure 1-1</b>	SX.25 and Related Modules	5
<b>Figure 2-1</b>	x25adm Menu Tree	13
<b>Figure 2-2</b>	Activating Changes	44
<b>Figure 4-1</b>	Module Relationships	87



---

## Tables

<b>Table 2-1</b>	Call User Data Values and Uses	29
<b>Table 3-1</b>	Parameter Defaults	55
<b>Table 3-2</b>	Handling CUG Requests	64
<b>Table 3-3</b>	Tuning Options for a Subnetwork	65
<b>Table 3-4</b>	Tuning Parameters	67
<b>Table 6-1</b>	Tracing Levels	118



---

## About This Guide

The *SX.25 Administrator's Guide* is designed for those responsible for the configuration of networking equipment and software.

The guide is organized into the following chapters, followed by the Glossary and the Index.

Chapter 1, "Introduction to SX.25 Administration," is an introduction giving some background information and a brief overview of the features of SX.25. It also outlines some of the main issues in network addressing.

Chapter 2, "Configuration Using the x25adm Utility," shows how to set up a network using the *x25adm* system administration utility, which provides an easy-to-use interface for configuring SX.25 for your network.

Chapter 3, "Parameter Configuration," goes through the sets of parameters available at the **paramconfig** option of *x25adm*, detailing the defaults and permitted values.

Chapter 4, "Running SX.25 with TCP/IP," presents information on additional configuration tasks that you must perform if you plan to run SX.25 with TCP/IP.

Chapter 5, "UUCP Configuration," shows how to configure XTY and describes the UUCP files.

Chapter 6, "Troubleshooting," describes STREAMS trace and error logging, and lists error messages.

Other manuals related to using SX.25 include:

- *SX.25 User's Guide*
- *SX.25 Release Notes*
- *IRIS Advanced Site and Server Administration Guide*
- *IRIS Software Installation Guide*

See also the online reference manual pages: *dlpi(7)*, *llctemplate(4)*, *lltune(1M)*, *snetd(1M)*, *snetd.options(4)*, and *llc2(7)*.

## Introduction to SX.25 Administration

This chapter contains the following sections:

- “Network Administration”
- “X.25 Addresses”

SX.25 is a Silicon Graphics communications software product based upon the X.25 protocols specified by the International Consultative Committee for Telegraphy and Telephony (CCITT). It is a STREAMS implementation of the X.25 protocol suite.

SX.25 implements the CCITT recommendations, including:

- X.3/X.28/X.29 packet assembler/disassembler (*pad*) program.
- X.25 Level III, '80, '84, and '88 versions. Both support basic (modulo 8) and extended (modulo 128) packet sequence numbering.
- X.25 Level II LAPB, '80, '84, and '88 versions. The '84 and '88 versions support basic and extended packet sequence numbering.

SX.25 also supports the following:

- ISO 8208 support for DTE-DTE operation
- DTE and DCE operation
- IP to X.25 encapsulation (IXE)
- LLC-2 over IEEE 802.3, used to run X.25 over the LAN
- both virtual call (VC) and permanent virtual circuit (PVC) services

SX.25 supports per-call facilities. More information on these is given in the following chapters of this guide.

## Network Administration

As network administrator, you first have to install the network, then maintain it.

### Installation

Installation is covered in detail in the *SX.25 Release Notes* and *IRIS Software Installation Guide*. Initial installation involves the following tasks:

- Setting up the network hardware. This includes the installation of cabling and transceivers, and the hardware network interfaces on each host on the network. This is outside the scope of this manual.
- Installing SX.25 network software on your systems, using the SGI *inst* command for installation.
- Configuring the SX.25 drivers for the networks and subnetworks you want, using the *x25adm* system administration program. You normally complete this configuration before the network is started. You then reboot the system with the new kernel and start the network.

Configuration involves:

- Configuring the kernel to create streams and memory blocks for each network connection. SX.25 can operate through several external interfaces, driven through front-end boards.
- Tuning X.25 communications for your network, by selecting the file of X.25 parameters or creating a new file with customized values.
- Creating the tables of hosts available on the network and their X.25 addresses and facilities.
- Configuring the *pad* (packet assembler/disassembler) daemon.
- Creating the network. This involves linking the various stack modules and drivers, and initializing their values. The network daemon (*snetd*) uses the contents of the *snetd.options* file to produce the STREAMS stack.

## Maintenance

This refers to the changes you make to an existing network. You may have to stop and restart the network to effect some types of change to the configuration; refer to “Bringing Changes into Effect” described next. The following are tasks you need to perform:

- Add new lines, hosts, and so on.
- Maintain the tables of hosts available on the network and their X.25 addresses and facilities.
- Monitor the behavior of the network.
- Change the configuration of the kernel or other parameters, to improve performance as circumstances change. You can create different files of parameters for different circumstances, and switch between these.

## Bringing Changes into Effect

You can make and activate some configuration changes without stopping the network. Other configuration changes require you to stop and restart the network to bring them into effect. These include:

- Changes in the kernel configuration, including the combination of boards and the range of facilities they offer, do not take effect until the kernel is rebuilt and the machine rebooted.
- Changes in the network configuration do not take effect until you reconfigure and restart the network.
- Some changes in parameter configuration require network reconfiguration; for others you need only restart the network. Changes to parameter values that are made “currently,” rather than to the template file, only survive until the next time the network is stopped.
- Changes in tables take effect while the network is up.

For example, to change the kernel memory sizes, you must rebuild the kernel, then restart the system and the network.

**Note:** The X.25 network will start automatically on reboot if you enable the restart option of the *x25adm* Start/Stop menu; refer to “Bringing Changes into Effect” in Chapter 2.

## Kernel Configuration

Figure 1-1 shows the structure of SX.25 modules. All the modules that you have to configure, except for the *pad* program, are part of the kernel. *Pad* program configuration is described in the following section. The kernel modules are as follows:

X.25	The X.25 driver needs a list of parameters specifying the default settings for each subnetwork. In some cases, these settings are overridden by what the user sets when making a call.
LAPB	To run X.25 over the WAN, the LAPB driver requires one set of the parameters for each subnetwork.
LLC-2	To run X.25 over the LAN, the LLC-2 driver requires one set of parameters for each subnetwork.
IXE	If you plan to run TCP/IP over X.25, refer to Chapter 4, “Running SX.25 with TCP/IP,” for details of the configuration you need to carry out.
XXX	The X.3, X.28, and X.29 <i>pad</i> modules for both incoming and outgoing <i>pad</i> calls.
SLD	SGL line discipline module for incoming <i>pad</i> calls.
XTY	The pseudo TTY driver, for running existing tty-based applications.

Chapter 2, “Configuration Using the x25adm Utility,” describes how to carry out all necessary configuration using the system administration utility *x25adm*.

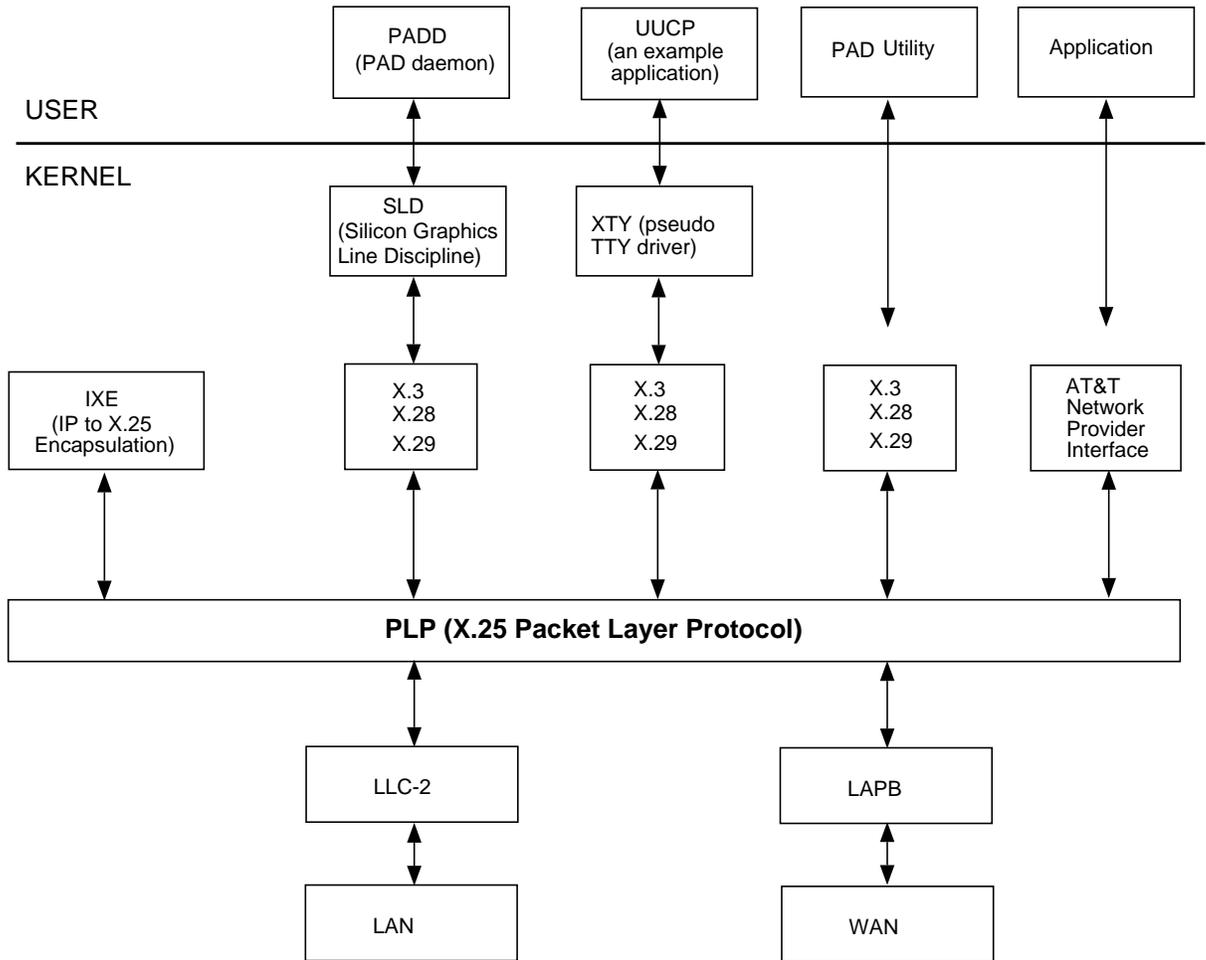


Figure 1-1 SX.25 and Related Modules

### ***Pad* Configuration**

Hosts that can provide *pad* facilities need to be registered in the *pad* database. You also need to configure the values for the *pad* daemon listen parameters for incoming calls. Set up values for the window and packet sizes and facilities to be used on *pad* calls to the specified address. For details, refer to Chapter 2, “Configuration Using the x25adm Utility.”

### **X.25 Addresses**

A host on the network is identified by its X.25 address. The X.25 address contains up to four fields. For more information on addressing, refer to the *X25addr(5)* manual page. The format of this address depends on the configuration used, as follows:

#### **LAN 1984/88:**

```
id.LSAP
id.LSAP[.N].OSI NSAP
id.LSAP.X.non-OSI address extension
```

#### **WAN 1980:**

```
id.DTE
```

Or, for a permanent virtual circuit:

```
id.LCI.P
```

#### **WAN 1984/88:**

```
id.DTE[.N].OSI NSAP
id.DTE.X.non-OSI address extension
```

Or, for a permanent virtual circuit:

```
id.LCI.P
```

## Where:

id	is a single-character (alphanumeric) subnetwork identifier. This identifies a connection to or from the WAN or LAN.
DTE	defines the Subnetwork Point of Attachment (SNPA) for each WAN line. This is entered as an address of up to 15 decimal digits, or if X.25(88) TOA/NPI address format is being used, then up to 17 decimal digits.
LSAP	defines the SNPA for each LAN line. It comprises the MAC address plus the DSAP address, 14 hexadecimal digits in all:  The MAC address is a unique 12-digit address identifying hosts and devices on the Ethernet; it is also called an Ethernet address. An example of a MAC address is:  <code>08004e011f37</code>  The DSAP address is the Destination Service Access Point, which is a two-digit address indicating the destination Link Layer service access point.
N or X	is an indicator that tells you whether the next part of the address is an OSI NSAP address or a non-OSI address extension.
OSI NSAP	is the Network Service Access Point, the point at which the Network Service is provided. The NSAP address can be up to 40 hexadecimal digits.
non-OSI address extension	indicates that the given address (up to 40 hexadecimal digits) is encoded in the X.25 Call Request packet as an “eXtended” address, that is, non-OSI, as opposed to an NSAP address (OSI).
LCI	is the logical channel identifier, three hexadecimal digits. Pad in front with zeroes if necessary, for example, enter “009” rather than “9”. (The “P” that follows indicates that the LCI describes a permanent virtual circuit).



---

## Configuration Using the *x25adm* Utility

This chapter contains the following sections:

- “Network Management Options”
- “Accessing Menus in *x25adm*”
- “Kernel Configuration”
- “Network Configuration”
- “Database Tables Configuration”
- “Bringing Changes into Effect”

*SX.25* provides a comprehensive set of network management utilities to help you configure and manage your network.

Tasks you need to perform include the following:

- Configure the list of *pad* hosts on the local area network (LAN) and the wide area network (WAN), and set the default *pad* program options.
- Choose suitable X.25 and LAPB parameter files for each subnetwork. You can also customize files of X.25 parameters if the default values do not suit the type of network you plan to use. LLC-2 and its LAN interfaces are configured through the data link provider interface (DLPI) modules. Refer to the online reference manual page DLPI(7) for details.
- Alter the system network configuration.
- Start up the network.

If you plan to run X.25 with TCP/IP, you will also need to configure the IP parameters. For more information about configuring IP, refer to Chapter 4, “Running *SX.25* with TCP/IP.”

## Network Management Options

This section describes the options that comprise the SX.25 Network Management facility *snetmgmt*. The following section shows you how to access these menus under *x25adm*. The *snetmgmt* menu has five components, the Network Management submenus, with options that let you carry out all the necessary configuration procedures described above. The main functions associated with each of these five submenus are as follows:

- kernelconfig**    accesses the Kernel Configuration menu  
Options available at this menu let you access the kernel parameters to tune SX.25 for a particular configuration.
- netconfig**        accesses the Network Configuration menu  
This menu lets you change the fields of the *netconf* file and also lets you recreate the network after making changes.
- paramconfig**    accesses the Parameter Configuration menu  
Options available at this menu let you select the X.25 parameter files that suit your network and configure options for *pad* hosts and the *pad* listen daemon.
- startstop**        accesses the Network Start, Stop, and Status menu  
Options available at this menu let you start and stop the network, and display the network status.
- tables**            accesses the Database Tables Configuration menu  
Options available at this menu let you modify the names and addresses of hosts, subnetworks, IXEmap, PVCmap, WANmap, and *pad* facilities on the network.

## Accessing Menus in x25adm

Configuration is performed using the options of the SX.25 Network Management menu. To access an option from the command line, enter one of the following:

`x25adm`

Or to go directly to the specified Network Management menu subcommand, for example, *kernelconfig*:

**x25adm snetmgmt\_subcommand**

The first method, typing **x25adm**, displays the top menu of the *x25adm* menu tree:

NETWORK MANAGEMENT

```
1 kernelconfig   kernel configuration menu
2 netconfig     network daemon configuration menu
3 paramconfig   parameter configuration menu
4 startstop     network start, stop, and status menu
5 tables        database tables menu
```

Enter a number, a name, the initial part of a name, or ? or <number>? for HELP, ^ to GO BACK, q to QUIT:

The second method takes you directly to one of the Network Management menu options. An example is:

**x25adm tables**

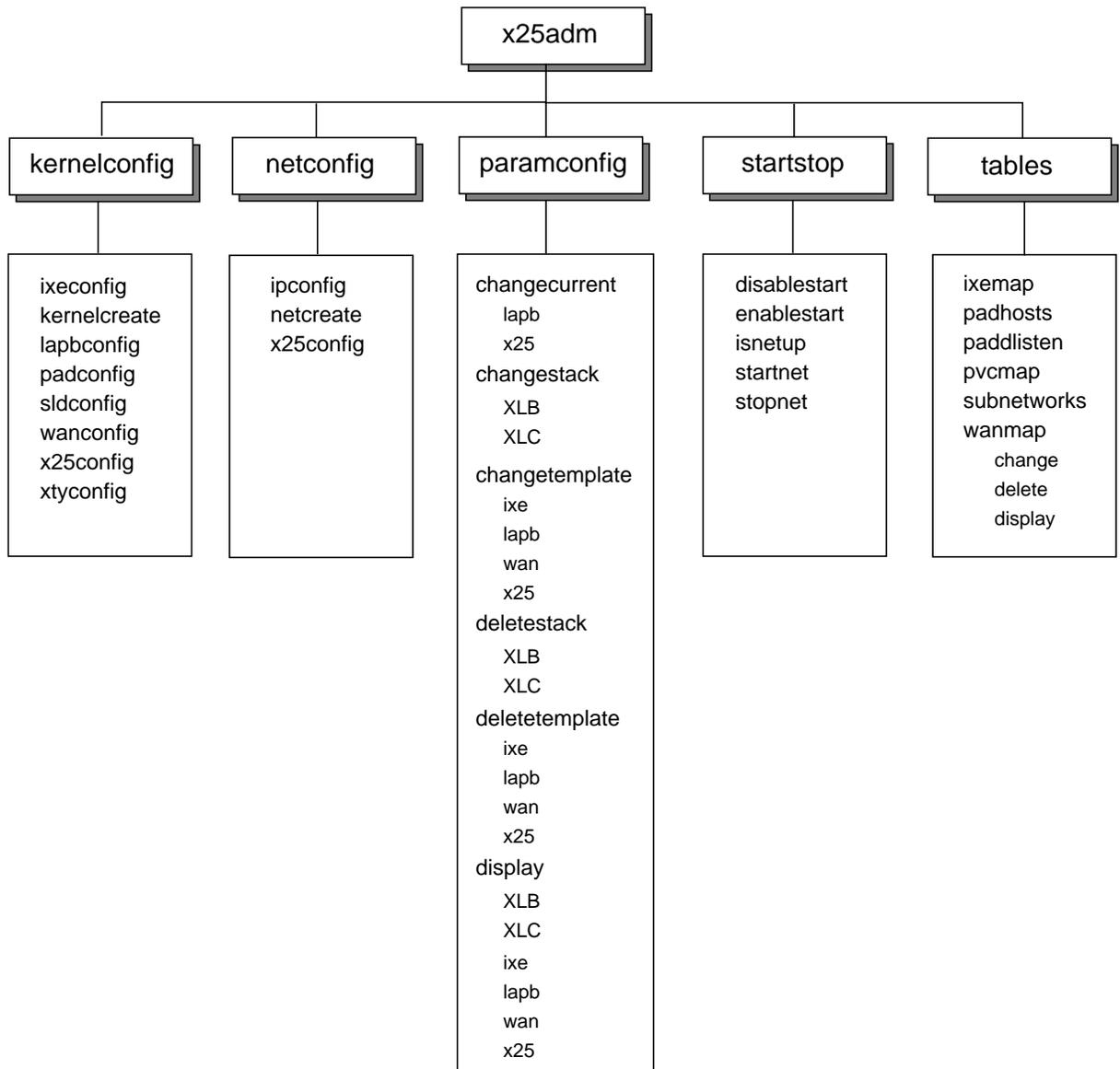
which takes you directly to the tables subcommand of the Network Management menu.

Notes:

1. Only the first of the entry methods described lets you go back up to higher levels in the menu tree. After using the other method, you have to exit and re-enter *x25adm*.
2. At most menus, you can also enter the following responses:
  - ? to display help information
  - <number>? to display help information on a specified subcommand
  - ^ to display the previous menu
  - q to exit to the command line

3. To select a value displayed at a menu, you can:
  - Enter the number that precedes the parameter value in the menu. At the Network Management menu, for example, enter 5 for tables.
  - Enter the name of the parameter in full.
  - Enter a unique abbreviation of the name. At the Network Management menu, for example, enter **ta** for tables.

Figure 2-1 shows the structure of the *x25adm* menu tree.



**Figure 2-1** x25adm Menu Tree

## Kernel Configuration

This section shows how to configure the X.25 kernel using the *kernelconfig* option of the Network Management menu. Alternatively, you can enter the *kernelconfig* option directly from the command line:

```
x25adm kernelconfig
```

The following menu is displayed:

```
KERNEL CONFIGURATION MENU
```

```
1 ixecfg          IXE configuration
2 kernelcreate    kernel creation
3 lapbconfig      LAPB configuration
4 padconfig       PAD configuration
5 sldconfig       SLD configuration
6 wanconfig       WAN configuration
7 x25config       X25 configuration
8 xtyconfig       XTY configuration
```

Enter a number, a name, the initial part of a name, or ? or <number>? for HELP, ^ to GO BACK, q to QUIT:

### IXE Kernel Parameters

Refer to Chapter 4, “Running SX.25 with TCP/IP.”

### LAPB Kernel Parameters

Select the **lapbconfig** option of the Kernel Configuration menu to change the LAPB configuration. There is just one parameter:

#### Maximum Number of LAPB Devices - LAPB\_DEVS

This is the maximum number of upper streams that can be opened to the LAPB driver. The prompt is:

```
LAPB_DEVS
The current value is 8
Type: integer {4-255}
Or  : RETURN = 8
```

? = HELP, ^ = GO UP A LEVEL, q = QUIT

New value:

Enter the number of LAPB devices on your system.

## Pad Kernel Parameters

Select the **padconfig** option of the Kernel Configuration menu to change the two *pad* parameters:

### Maximum Number of XXX Connections - MAX\_CONNS

This is the maximum number of connections to X.3, X.28, and X.29.

The current value is 16

Type: integer {0-256}

Or : RETURN = 16

? = HELP, ^ = GO UP A LEVEL, q = QUIT

New value:

Enter the number of connections on your system.

### Default X.29 Version - X29\_DEFAULT

This states whether the 1980, 1984, or 1988 specification is to be used as the default version.

The current value is 88

Type: integer {80, 84, 88}

Or : RETURN = 84

? = HELP, ^ = GO UP A LEVEL, q = QUIT

New value:

Enter **80**, **84**, or **88**.

You are asked to confirm any changes you have made.

### SLD Kernel Parameters

Select the **sldconfig** option of the Kernel Configuration menu to change the number of SGI line discipline (SLD) modules. One SLD module is used for each incoming *pad* call. The prompt is:

```
NUM_SLDS
The current value is 16
Type: integer {0-256}
Or  : RETURN = 16
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

Enter a value greater than, or equal to, the maximum number of incoming calls allowed. You are asked to confirm your selection.

### WAN Kernel Parameters

Select the **wanconfig** option of the Kernel Configuration menu to change the number of wide area network (WAN) boards you can use in your system. The following prompt is displayed:

```
WANS_LOWER
The current value is 1
Type: integer {0-2}
Or  : RETURN = 1
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value: 1
```

Enter the number of WAN boards. You are asked to confirm your selection before returning to the **kernelconfig** menu.

## X.25 Kernel Parameters

Select the **x25config** option of the Kernel Configuration menu to carry out any X.25 kernel configuration required before building the kernel. There are three parameters:

### Number of Connections - X25\_CON

This specifies the number of STREAMS connections to X.25. This must exceed the next parameter, which is the number of virtual circuits. The default when you first install SX.25 is 68. This is four more than the default number of virtual circuits. The following screen is displayed:

```
X25 Level III Kernel Parameter Configuration
-----

X25_CON

The current value is 68

Type: integer {0-256}
Or  : '<CR>' = 68
      '?' = HELP, '^' = GO UP A LEVEL, 'q' = QUIT

New value:
```

Enter a value or press Return to confirm the default.

### Number of VC Elements - X25\_VC\_ELEMENTS

This specifies the number of simultaneous X.25 connections (virtual circuits) that the driver can handle. It is the number of virtual circuit database elements available to the driver. This value should be less than X25\_CON. The default value when you first install SX.25 is 64.

```
X25_VC_ELEMENTS

The current value is 64

Type: integer {0-256}
Or  : RETURN = 64
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT

New value:
```

Enter a value or press Return to confirm the default.

### Number of LSAP Elements - X25\_LSAP\_ELEMENTS

This is the number of LSAP database elements available to the driver. It specifies the number of links X.25 can run over. Allocate one for each available physical WAN link. Additionally, if you are running X.25 over LLC-2, add to this value the number of elements available for “logical” LLC links.

The default value is 16, the range of values is 0-256.

This is the last parameter. Press ^ to return to the Network Management menu or q to quit *x25adm*.

### XTY Kernel Parameters

The following parameter governs the maximum number of simultaneous pseudo TTY driver (XTY) connections that can be handled. The *xyt* driver is used by *uucp/cu* type calls; refer to the section “Configuring the XTY System” in Chapter 5.

XS.25 supports XTY\_OUT system, which uses the XTY device driver only for outgoing *uucp/cu* calls. You must set up this parameter:

```
XTY_OUT
The current value is 0
Type: integer {0-128}
Or : RETURN = 0
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

Enter the number of connections.

## Network Configuration

To access the Network Configuration menu, select the **netconfig** option from the Network Management menu, or enter the **netconfig** option directly from the command line:

```
x25adm netconfig
```

**The following menu is displayed:**

## NETWORK DAEMON CONFIGURATION

```

1 ipconfig      IP network configuration
2 netcreate     netconf creation
3 x25config     X25 network configuration

```

Enter a number, a name, the initial part of a name, or ? or <number>? for HELP, q to QUIT:

To configure IP, refer to Chapter 4, "Running SX.25 with TCP/IP."

To configure lines below X.25, select **x25config**. A screen similar to the following is displayed:

## CONFIGURATION OF LINES BELOW X25

-----

The following commands may be used throughout:

Command	Action
-----	-----
?	Give specific help information
??	Give this general help information
RETURN	Use current value (in square brackets)
q	Quit

Please wait..... checking configuration files

The current configurations are:

'Wide Area Network' (WAN) board			'X25 over LAPB' stack		
Line	Subnetwork	DTE Address	Stack	Extended	Link Type
----	-----	-----	----	-----	-----
0	A	23423130014263	def.dte80	no	DTE
1	B	23423130014264	def.dte80	no	DTE

Do you want to reconfigure a line on the Wide Area Network (WAN) board (there are 2 lines in total)?

## Configuring a WAN Subnetwork

Enter `y` in response to the above prompt to configure a line on the WAN board. On a two-line board, the lines are numbered 0 and 1. You are prompted for the number of the line you want to configure:

```
Enter the number of the line (0 - 1) on the board (or '=' to list current configurations):
```

Enter `0` or `1`. You are now prompted to enter the subnetwork ID, which is a single character identifier for this line; the default is shown in square brackets:

```
Enter the one character letter or number (the subnetwork ID) by which you want to identify this line (or '-' to clear whole entry).
```

```
Subnetwork IDs in use: A B C
```

```
Subnetwork ID [A]
```

Enter the subnetwork ID or press Return to retain the default. You are prompted for the DTE address for this line:

```
Enter the 1 to 15 digit decimal DTE address of this subnetwork ('-' for no address) [23423130014263]
```

Enter the DTE address or press Return to retain the default.

You are asked for the name of the stack file that holds the parameter values you want to use on this subnetwork. The available stack files are listed. To modify a stack file, use the **paramconfig** option of the Network Management menu; refer to Chapter 3, "Parameter Configuration."

```
Enter the name of the 'X25 over LAPB' stack for this subnetwork.
```

```
Stacks Available
```

```
-----
```

```
def.dce80          def.dce84          def.dce88
```

```
def.dte80          def.dte84          def.dte88
```

```
Stack [def.dte88]
```

Enter the name of a stack file or press Return to retain the default.

You are asked to specify the Link procedures to be used:

Which link procedures do you want to use (LAPB, LAP)? [LAPB

Enter the relevant procedure or press Return to keep the default. If you have an 1984 network and selected the LAPB link procedure, the following prompt is displayed:

Do you want to use extended LAPB addressing? [n]

Specify whether extended addressing is to be used. This is the last parameter. A message similar to the following is displayed:

You have finished configuring this line.

You have created the following entry:

'Wide Area Network' (WAN) board 'X25 over LAPB' stack

```
-----
Line Subnetwork DTE Address      Stack      Extended Link Type
-----
0    A           23423130014263 def.dte88  no         DTE
```

Are you sure you want to install these values?

Enter **y** to confirm the entries you have made. You are prompted to configure another board. When you have finished configuring WAN lines, enter **n** in response to this prompt.

## Configuring a Loopback Driver

When you finish configuring WAN lines, if your system is running LAPB in the kernel, you are prompted to configure the WAN loopback driver:

Do you want to configure the WAN loopback driver (wloop)?

Enter **y** if you do want to configure the loopback driver. If not, enter **n**. You can skip the next section and go on to “Configuring a LAN Subnetwork” later in this chapter.

The existing configuration is displayed:

```
'WAN loopback driver (wloop)' 'X25 over LAPB' stack
-----
Line Subnetwork DTE Address Stack Extended Link Type
-----
0 not set not set not set not set not set
1 not set not set not set not set not set
Do you want to reconfigure a line on the wloop driver?
(there are 2 lines in total, yet to be configured)
```

The sequence of parameters is the same as for the WAN line. You must ensure that:

- one line is configured as a DTE, the other as a DCE
- both lines match with respect to extended addressing, 80/84/88 networks, and LAPB/LAP operation

A message similar to the following is displayed:

```
You have finished configuring this line.
You have created the following entry:
'WAN loopback driver (wloop)' 'X25 over LAPB' stack
-----
Line Subnetwork DTE Address Stack Extended Link Type
-----
0 X 23423130014263 def.dte88 no DTE
Are you sure you want to install these values?
```

Enter **y** to confirm the entries you have made. It will then prompt you to configure another line. When you have finished configuring WAN loopback driver lines, enter **n** in response to this prompt.

### Configuring a LAN Subnetwork

The script checks your system to see whether your system has an Ethernet interface or an FDDI interface. If your system has either of these, it prompts you to configure them. If it has neither, then the following prompts will not show up.

If your system has an Ethernet interface, you are prompted to configure it:

The current configuration is:

```
Ethernet board      'X25 over LLC2' stack
-----
Subnetwork  MAC Address  Local SAP  Loopback SAP  Stack
-----
C           standard    7E         70             lan84
```

Do you want to reconfigure the Ethernet board?

To configure the Ethernet board, enter **y** at this prompt. A screen similar to the following is displayed:

Enter the one character letter or number (the subnetwork id) by which you want to identify this board (or '-' to clear whole entry).

Subnetwork IDs in use: A B C

Subnetwork ID [C]

Enter the subnetwork ID of the line you want to configure or press Return to retain the default. The following screen is displayed:

You are strongly advised to use only the standard (i.e. Manufacturer's) MAC address, but if you need to change it then enter the 12-digit hexadecimal MAC address of this subnetwork ('-' for standard) [standard address]

Press Return to retain the current value or enter a dash (-) for the standard MAC address supplied by the manufacturer, as is recommended. You are prompted for the SAP of this subnetwork:

Enter the 2-digit hexadecimal local SAP address of this subnetwork [7E]

Enter the SAP of the subnetwork or press Return to retain the default. You are prompted to enter the loopback SAP of the subnetwork:

Enter the 2-digit hexadecimal loopback SAP address of this subnetwork [70]

Enter the loopback SAP of the subnetwork, or press Return to retain the default. You are prompted to enter the name of the stack file that holds the parameter values you want to use on this subnetwork. The available stack

files that you have set up using the **paramconfig** option of the Network Management menu are displayed, for example:

Enter the name of the 'X25 over LLC2' stack for this subnetwork.

```
Stacks Available
-----
def.lan   lan84   lan88
Stack [lan88]
```

Enter the name of the stack file you want to use, or press Return to retain the default. A message similar to the following is displayed:

You have finished configuring this board.

You have created the following entry:

```
Ethernet board      'X25 over LLC2' stack
-----
Subnetwork  AC Address  Local SAP  Loopback SAP  Stack
-----
C           standard  7E         70            lan88
```

Are you sure you want to install these values?

Enter **y** to confirm these values. You are prompted to configure another Ethernet board. When you have finished configuring the Ethernet boards, enter **n** in response to the prompt.

### Configuring the FDDI Interface

When you finish configuring the Ethernet interface, and you have an FDDI interface, you are prompted to configure the FDDI interface:

current configuration is:

```
FDDI board      'X25 over LLC2' stack
-----
Subnetwork  MAC Address  Local SAP  Loopback SAP  Stack
-----
D           standard  7E         70            lan84
```

Do you want to reconfigure the FDDI board?

To configure the FDDI board, enter **y** at this prompt. A screen similar to the following is displayed:

```
Enter the one character letter or number (the subnetwork id)
by which you want to identify this board (or '-' to clear
whole entry).
```

```
Subnetwork IDs in use: A B C D
```

```
Subnetwork ID [D]
```

Enter the subnetwork ID of the line you want to configure or press Return to retain the default. The following screen is displayed:

```
You are strongly advised to use only the standard (i.e.
Manufacturer's) MAC address, but if you need to change it
then enter the 12-digit hexadecimal MAC address of this
subnetwork ('-' for standard) [standard address]
```

Press Return to retain the current value or enter a dash (-) for the standard MAC address supplied by the manufacturer, as is recommended. You are prompted for the SAP of this subnetwork:

```
Enter the 2-digit hexadecimal local SAP address of this
subnetwork [7E]
```

Enter the SAP of the subnetwork or press Return to retain the default. You are prompted to enter the loopback SAP of the subnetwork:

```
Enter the 2-digit hexadecimal loopback SAP address of this
subnetwork [70]
```

Enter the loopback SAP of the subnetwork, or press Return to retain the default. You are prompted to enter the name of the stack file that holds the parameter values you want to use on this subnetwork. The available stack files that you have set up using the **paramconfig** option of the Network Management menu are displayed, for example:

```
Enter the name of the 'X25 over LLC2' stack for this
subnetwork.
```

```
Stacks Available
```

```
-----
```

```
def.lan    lan84    lan88
```

```
Stack [lan84]
```

Enter the name of the stack file you want to use, or press Return to retain the default. A message similar to the following is displayed:

You have finished configuring this board.

You have created the following entry:

FDDI board            'X25 over LLC2' stack

-----

Subnetwork	MAC Address	Local SAP	Loopback	SAP	Stack
D	standard	7E	70		lan84

Are you sure you want to install these values?

Enter **y** to confirm these values. You are prompted to configure another FDDI interface. When you have finished configuring the FDDI boards, enter **n** in response to the prompt.

When you finish entering values, the following message is displayed:

You have finished configuring X25.

Do you want to save the changes? [y]

Press the RETURN key to see the netconfig menu [?, q]:

After making network configuration changes, you must select the **netcreate** option at the Network Daemon Configuration menu in order to recreate the network with the new configuration. The changes you have made will not take effect until you do this and restart the network.

## Database Tables Configuration

The Database Tables menu gives access to the group of subcommands that let you set up network information tables. These tables form a database of information about the characteristics of your network.

Access the Database Tables menu from the Network Management menu by selecting the **tables** subcommand, or directly from the command prompt:

```
x25adm tables
```

The following menu is displayed:

DATABASE TABLES

```

1 ixemap          update ixemap database
2 paddlisten     update pad daemon listen database
3 padhosts       update pad database
4 pvcmap         update pvc database
5 subnetworks    update subnetworks database
6 wanmap         update wanmap database

```

Enter a number, a name, the initial part of a name, or ? or <number>? for HELP, ^ to GO BACK, q to QUIT:

The following options are used to configure X.25:

- ixemap**            Configures the IP to X.25 address map; refer to “IXE Database Configuration” in Chapter 4 .
- paddlisten**      Configures listen information for the *pad* daemon; refer to “Configuring the Pad Daemon” later in this chapter.
- padhosts**        Configures the list of X.25 hosts on your network; refer to “Configuring the Pad Database” later in this chapter.
- pvcmap**            Configures facility information for a subnetwork’s PVC range; refer to “Configuring the PVC Database” later in this chapter.
- subnetworks**      Configures the calling address, name, and description for each of your subnetworks; refer to “Configuring the Subnetworks Database” later in this chapter.
- wanmap**            Configures a number of tables holding mappings between either X.121 destination addresses or names, such as “uk.co.host”, and interface addresses; refer to “Configuring the Wanmap Database” later in this chapter.

## Menu Options

The available options are listed in parentheses on *pad* database menus. In addition to the normal ‘?’, ‘^’ and ‘q’ options, you can enter the following options to most menus:

- k**                    Kill the changes made to this line and begin altering a new one.

- l Use default values, including those for parameters that you have not yet seen. This displays the values of the parameters you have set up to this point in the menu sequence.
- r Re-enter: abort any changes made so far and retain initial values.
- \r This has a special meaning for facilities; do not confuse it with r.
- s Skip: leave this parameter unchanged and move to the next.
- t Truncate this line: remove values from remaining parameters.
- w Give all remaining parameters default values.
- \x Treat x as a non-command character, where x is any single character.

**Note:** All options of the Tables menu let you add, alter, or delete an entry, as well as list all entries. After “Configuring the Pad Daemon”, the descriptions concentrate on the **addentry** option, because this is the most wide-ranging.

### Configuring the Pad Daemon

The **paddlisten** option of the Database Tables menu lets you configure the *pad* daemon. To access the *pad* Daemon Listen menu, select the **paddlisten** option from the Database Tables menu, or enter the **paddlisten** option directly from the command line:

```
x25adm paddlisten
```

The following menu is displayed:

```
PAD DAEMON LISTEN MENU
1 addentry          add listen entry
2 alterentry        alter listen entry
3 delentry          delete listen entry
4 listentries       list listen entries

Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

The *pad* daemon listen database holds information about the X.25 addresses and/or Call User Data (CUD) fields for which the daemon handles incoming calls. To create a new entry, enter **1** or **addentry** or an abbreviation, for example, **ad**. The following screen is displayed:

```
ADD LISTEN ENTRY
Enter Listen_String
[string, r, \x, ?, ^, q]:
```

The Listen String specifies the called addresses and CUD field values for which this application is prepared to accept calls. The format is as follows:

```
< CUD field > , < called addresses field > ;
< [ (I|S) [bytes] ] | X > , < [ (I|S) (D|N) [semi-octets] ] | X > ;
```

where Table 2-1 shows the values and their use:

**Table 2-1** Call User Data Values and Uses

Value	Use
I	Precedes a field that must match identically the supplied field
S	Precedes a field that must match the start of the supplied field
X	Does not care
D	Precedes a field referring to an X.25 DTE
N	Precedes a field referring to an X.25 NSAP

The following is an example of a valid listen string:

```
S01000000,SD2342;
```

The CUD field should start with 01000000; the address field should start with 2342, and is an X.25 DTE address.

When you enter a valid listen string, the complete listen entry is displayed for you to confirm.

```
Ok? [y/n]:
```

Enter **y** or **yes**. The following message is displayed:

```
Added listen entry
```

You are prompted for the listen string of the next listen entry you want to make:

```
Enter Listen_String  
[string, r, \x, ?, ^, q]:
```

When you have finished entering listen entries, enter `q`. You are asked to confirm that you want to save the changes. Enter `y`. You are prompted to press Return to display the Database Tables menu, or `q` to quit.

### Configuring the Pad Database

The network information required by *SX.25* includes the names, addresses, and call information for each of the X.25 hosts on your network. You can associate the names and addresses of these hosts with aliases. This information is held in the *pad* database. The *pad* database menu lets you add, alter, and delete information about the X.25 hosts on your network. To access this menu, select the **pad** option of the Database Tables menu or enter the **pad** option directly from the command line:

```
x25adm padhosts
```

The following menu is displayed:

```
PADHOSTS DATABASE  
  
1 addentry          add padhost entry  
2 alterentry       alter padhost entry  
3 delentry         delete padhost entry  
4 listentries      list padhost entries  
  
Enter a number, a name, the initial part of a name, or ? or  
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

#### Adding a Pad Entry

To add a *pad* entry, select the **addentry** option or **1**, or an abbreviation, for example, **ad**. You are presented with a series of prompts. Your replies to these produce the new entry.

**Note:** You can quit the session at any time by entering `q`. You are asked whether you want to save the changes or not. You then exit *x25adm* and return to the command prompt.

The first prompt asks for the *pad* hostname:

```
ADD PAD HOST ENTRY
```

```
Enter Name
```

```
[string, r, \x, ?, ^, q]:
```

Enter a *pad* hostname and press Return. The hostname is a string of any characters except control characters and double quotes. The next prompt is:

```
Enter Address
```

```
[X25 Address, r, \x, ?, ^, q]:
```

An X.25 address comprises four fields (refer to “X.25 Addresses” in Chapter 1). The field entries should be separated by a dot. Only the first field is mandatory for a WAN address. Two fields are mandatory for a LAN address. Trailing dots may be omitted. An example of a valid X.25 address is:

```
A.234231500010.N.38826110002345678941
```

Enter the X.25 address of the host and press Return. It is checked to see if it is a valid address. If the check fails, a message similar to the following is displayed and you are prompted to re-enter the address:

```
“C.0123456789876543.X.234567890-” is not a valid X.25 Address
```

```
Enter Address
```

```
[X25 Address, r, \x, ?, ^, q]:
```

Once you have entered a valid X.25 address, you are prompted to enter the type of X.29 that is used for a call to this host, either the version conformant with the CCITT recommendations made in 1980, 1984, or with those made in 1988:

```
Enter X29_Type
```

```
[integer(80,84,88), l, r, w, \x, ?, ^, q]:
```

Enter a two-digit year number, 80, 84, or 88. Alternatively, press Return to use the configured default.

Once you have made a valid entry at this prompt, you are prompted to enter X.25 facilities to be requested on *pad* calls to this host.

```
Enter Facilities
```

```
[Facilities, l, r, w, \x, ?, ^, q]:
```

You may set values for the size of the incoming (remote-to-local) and the outgoing (local-to-remote) packet and window sizes, and for fast select and reverse charge request. These are described in the following paragraphs.

### Facilities

To configure facilities, enter a facility string, specifying a range of facilities. You can separate the values for facilities by commas, as shown in the following example:

```
p7/7,w2/2,f,r
```

The following description of the available facilities shows how to form the string:

**Packet sizes** Enter the incoming and outgoing packet sizes as part of the facilities string. The values of the packet sizes are expressed as a number **n**, where “two to the power of n” gives the packet size. The range for the parameter is 4 to 12, giving a packet size of 16 to 4096 bytes. The values of the sizes are preceded by a **p** and separated by a slash (/). The first number is the incoming packet size, the second is the outgoing packet size. An example of an incoming/outgoing packet size is: p7/7.

**Window sizes** Enter the values of the incoming and outgoing window sizes as decimal numbers. The range for the window sizes is 1-127. The values of the window sizes are preceded by a **w** and separated by a slash (/). The first number is the incoming window size, the second is the outgoing window size. An example of an incoming/outgoing window size is: w2/2.

**Fast select** Include **f** in the facilities string, as in the following example: p7/7w2/2f.

**Reverse charge call** Include **r** in the facilities string, as in the following example: p7/7w2/2r. Alternatively, enter **\r** (not just **r**) on its own to make reverse charging the only facility.

**Fast Select and Reverse Charge** Include **fr** in the facilities string, as in the following example: p7/7w2/2fr.

After facilities, set the following parameters:

### **Closed User Group - CUG**

The use of a Closed User Group (CUG) number indicates that the call is to be made to the specified Closed User Group on the specified host. There are two classes of closed user group:

- multi-user      Enter the CUG (a number of up to four digits) preceded by **G**.
- bilateral      This type of CUG has only two members. Users may be members of more than one bilateral closed user group. Enter the CUG (up to four digits) preceded by **B**.

### **Recognized Private Operating Agency - RPOA**

This provides transit networks with additional routing information about calls to this host. Enter up to four RPOA numbers, each of four digits preceded by **T**. There should be no spaces between each RPOA. Press Return if no RPOAs are required.

### **Call User Data String - CUD**

The Call User Data string is sent in a call request. It may be of any length up to 124 ASCII characters. The prompt is:

```
Enter CUD  
[string, l, r, w, \x, ?, ^, q]:
```

If you enter more than 12 characters, Fast Select is automatically used. Enter any CUD required and press Return.

### **Alias**

An alias can be any alternative name for the host and should be a string of permitted characters (not control characters or double quotes).

```
Enter Alias  
[string, l, r, w, \x, ?, ^, q]:
```

Enter one or more aliases. When you have entered the last alias name, press Return in response to the alias prompt. All the entries for the host you have set up are displayed for you to confirm:

```
Name           = CentralX
Address        = C.080069021CC57e
X29_Type      = 80
Facilities    = p8/8,w4/4
CUG           = g1234
RPOA         = t43211234
Call_User_Data = hello
Alias         = cent
```

Ok? [y/n]:

Enter **y** to confirm these. The following message is displayed, and you are prompted to enter the next *pad* host:

```
Added pad entry
Enter Name
[String, r, \x, ?, ^, q]:
```

Enter details of another *pad* host or type **q** to exit.

### Altering a Pad Entry

To alter a *pad* entry, select the **alterentry** option of the *pad* database menu (or **2**, or an abbreviation, for example: **a1**).

**Note:** The sequence of menus is similar to those described in the previous section, “Adding a Pad Entry” on page 30. For more information on the meaning of parameters, refer to that section.

```
ALTER PADHOST ENTRY
Enter Name           to be altered
[string, ?, ^, q]:
```

Enter the name of the *pad* entry you want to alter. The existing name, address and parameter values are displayed, as follows:

```
Name           = CentralX
Address        = C.080069021CC57e
X29_type      = 80
Facilities    = P8/8W4/4
```

```
CUG          = -
RPOA         = -
Call_User_Data = hello
Alias        = cent
```

Alter? [y/n]:

Enter **y** to display all the existing parameters, in turn. You can change any of the values. The prompts include the following:

```
Enter new Name
Current value: "CentralX"
[string, s, l, r, k, t, \x, ?, ^, q]:
```

and:

```
Enter new Facilities
[Facilities, s, l, r, k, t, w, \x, ?, ^, q]:
```

When you finish making changes, the following message is displayed:

```
Ok? [y/n]:
```

Then, if you enter **y**:

```
Altered padhost entry
```

Accepting the altered parameters here is not final. You are asked whether you want to make changes to another host entry. When you have finished changing all the relevant *pad* host entries, the following message lets you undo the changes you have made since selecting the **alterentry** option:

```
Do you wish to save existing changes? [y/n]:
```

Enter **y** to accept all the changes you have made, or **n** to leave the *pad* hosts unaltered.

### Deleting a Pad Entry

To delete an entry, select the **delentry** option of the *pad* database menu. If you delete an entry, you are prompted to confirm the deletion before the entry is removed from the database.

### Listing Pad Database Entries

When you have created entries for all the X.25 hosts on the network, you can return to the *pad* hosts database menu by entering ^ at the prompt. Select **listentries** to check the changes you have made. The hostname and address information for all hosts in the *pad* database are displayed.

### Configuring the PVC Database

The PVCmap Database associates the packet and window sizes for use on a subnetwork's permanent virtual circuit (PVC) range.

**Note:** You only need to specify a mapping here if the packet and window sizes for a particular logical channel differ from the configured default sizes for the subnetwork. You can specify values for both directions, local to remote and remote to local.

To access the PVCmap Database menu, select the **pvcmap** option from the Database Tables menu or enter the **pvcmap** option directly from the command line:

```
x25adm pvcmap
```

The following menu is displayed:

```
PVCMAP DATABASE MENU
```

```
1 addentry          add pvcmap entry
2 alterentry        alter pvcmap entry
3 delentry          delete pvcmap entry
4 listentries       list pvcmap entries
```

```
Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

Only the procedure for adding an entry is described here. This shows all the parameters used to configure a PVC. Select **addentry** to display the following:

```
ADD PVCMAP ENTRY
```

```
Enter Subnetwork_Identifier
[string, r, ?, ^, q]:
```

Enter the subnetwork identifier. The following prompt is displayed:

```
Enter Logical_Channel  
[string, r, ?, ^, q]:
```

Enter the Logical Channel Identifier (LCI) as three hexadecimal digits, zero-filled if necessary (that is, "009" rather than "9").

```
Enter Loc_Packet_Size  
[integer(16,32,64,128,256,512,1024,2048,4096), r, ?, ^, q]:
```

This is the packet size to be used on the PVC in the direction local to remote. Enter one of the listed values.

```
Enter Rem_Packet_Size  
[integer(16,32,64,128,256,512,1024,2048,4096), r, ?, ^, q]:
```

This is the packet size to be used on the PVC in the direction remote to local. Enter one of the listed values.

```
Enter Loc_Window_Size  
[integer(1-127), r, ?, ^, q]:
```

This is the window size to be used on the PVC in the direction local to remote. Enter the desired value, or press Return to keep the default.

```
Enter Rem_Window_Size  
[integer(1-127), r, ?, ^, q]:
```

This is the window size to be used on the PVC in the direction remote to local. Enter the desired value, or press Return to keep the default. This is the last parameter. Your selections are displayed:

```
Subnetwork_Identifier = B  
Logical_Channel      = 009  
Loc_Packet_Size      = 128  
Rem_Packet_Size      = 128  
Loc_Window_Size      = 2  
Rem_Window_Size      = 2
```

```
Ok? [y/n]:
```

Enter **y** to confirm these values.

## Configuring the Subnetworks Database

You can supply SX.25 with the names and calling addresses of the subnetworks you want on your network. A subnetwork may be a connection to the WAN or to the Ethernet. You can associate the address of each subnetwork with a name and a description. To access the Subnetwork Database menu, select the **subnetworks** option from the Database Tables menu or enter the **subnetworks** option directly from the command line:

```
x25adm subnetworks
```

The following menu is displayed:

```
SUBNETWORK DATABASE MENU
1 addentry add subnetwork entry
2 alterentry alter subnetwork entry
3 delentry delete subnetwork entry
4 listentries list subnetwork entries

Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

This menu lets you manage the entries in the subnetwork database.

### Adding a Subnetwork Entry

To add a subnetwork entry, select the **addentry** option or **1**, or an abbreviation, for example **ad**. You are prompted for a series of parameters for the first subnetwork you want to set up. The parameters are as follows:

#### Subnetwork Calling Address

The calling address must be appropriate for the type of subnetwork. It contains either an LSAP or a DTE address, according to whether the subnetwork is on the LAN or the WAN. The calling address is an X.25 address entered as up to four fields; refer to “X.25 Addresses” in Chapter 1. The prompt is:

```
ADD SUBNETWORK ENTRY
Enter Calling Address
[X25 Address, r, \x, ?, ^, q]:
```

The field entries should be separated by dots. Trailing dots may be omitted. Only the first field is mandatory for a WAN address. Two fields are mandatory for a LAN address. An example of a valid X.25 address is:

```
A.234231500010.N.38826
```

After entering a calling address, press Return. Your entry is checked. If it is invalid, a message similar to the following is displayed and you are prompted to re-enter the address:

```
"0800390030N" is not a valid X25 Address
Enter Address
[X25 Address, r, \x, ?, ^, q]:
```

### Subnetwork Name

The name of the subnetwork is a string of at least two permitted characters (not control characters or double quotes). The prompt is:

```
Enter Name
[string, r, \x, ?, ^, q]:
```

Enter the name of the subnetwork and press Return. The name you have entered is checked to see that it does not duplicate an existing entry in the subnetwork table. If the name exists, a message is displayed and you are prompted to re-enter the subnetwork name.

### Subnetwork Description

You are prompted to enter a description for the subnetwork. The description can be a string of any number of characters terminated by pressing Return. It is used by the **x25stat(1M)** utility to display network statistics.

When you have created entries for all the subnetworks on the network, you can return to the Subnetworks Database menu by entering **^** at the prompt.

### Deleting or Altering a Subnetwork Entry

To delete or alter an entry, access the Subnetwork Database menu and select the **delentry** option or the **alterentry** option, respectively.

If you delete an entry, you are prompted to confirm the deletion before the entry is removed from the database.

### Listing Subnetwork Entries

To check the entries you have made, use the **listentries** option: type **4**, or **listentries**, or an abbreviation, for example **li**. The hostname and address entries in the subnetwork database are displayed, for example:

```
LIST SUBNETWORK ENTRIES

Calling Address      = A.01234567890123
Name                 = sginet1
Description          = PSS subnetwork A

Calling Address      = B.123456789012345.N.22444
Name                 = sginet2
Description          = LAN subnetwork B

Calling Address      = C.080069021CC57e.X.112468900
Name                 = sginet3
Description          = Development subnetwork C

Press the RETURN key to see the Subnetwork Menu [?, q]:
```

### Configuring the Wanmap Database

The Wanmap Database provides a mapping between X.121 destination addresses (or names) and interface addresses. To access the Wanmap Database menu, select the **wanmap** option from the Database Tables menu. The following menu is displayed:

```
WANMAP TABLES

1 change             change or create wanmap table
2 delete             delete a wanmap table
3 display            display wanmap table

Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

This menu lets you manage a number of Wanmap database tables.

### Changing a Wanmap Table

To change a wanmap table, select the **change** option, or an abbreviation, for example **ch**. This option allows you to:

- Create a new wanmap table.
- Modify an existing wanmap table.
- Modify the current wanmap entries being used on a particular subnetwork.

After selecting the **change** option, you are prompted to enter a wanmap table or an existing subnetwork. The valid tables and subnetworks are listed before the prompt. The table you enter, or the current entries for the subnetwork you specify are used as the basis for the “new” table. This “new” table can be another table or a subnetwork to be updated. Thus, this allows you to:

- Create a new table.
- Update a table.
- Update a subnetwork’s table of wanmap entries.
- Use a table to update a subnetwork’s table of wanmap entries.
- Copy a table to another table.
- Copy a subnetwork’s table of wanmap entries to a table.
- Copy a subnetwork’s table of wanmap entries to another subnetwork.
- Update a subnetwork’s table of wanmap entries, if a table was originally specified.

Once you have entered both table/subnetwork names, the following menu is displayed:

#### WANMAP DATABASE MENU

```
1 addentry      add wanmap entry
2 alterentry    alter wanmap entry
3 delentry      delete wanmap entry
4 listentries   list wanmap entries
```

Enter a number, a name, the initial part of a name, or ? or <number>? for HELP, ^ to GO BACK, q to QUIT:

This menu lets you manage the entries in the wanmap database.

### Adding a Wanmap Entry

To add a wanmap entry, select the **addentry** option, **1**, or an abbreviation, for example, **ad**. You are prompted for a series of parameters. They are as follows:

#### WAN Remote Address

This is typically an X.121 destination address. However, this may be a name, such as "uk.co.host." The prompt is:

```
ADD WANMAP ENTRY
Enter Remote_address
[String, r, \x, ?, ^, q]:
```

An example of a valid X.121 address is:

```
12345678901234
```

After entering an X.121 destination address, press Return. The validity of your entry is checked.

#### Interface Address

The name of the interface address that the X.121 address or name is mapped to. The prompt is:

```
Enter Interface_address
[string, r, \x, ?, ^, q]:
```

Enter the interface address and press Return.

### Deleting or Altering a Wanmap Entry

To delete or alter an entry, access the Wanmap Database menu and select the **delentry** option or the **alterentry** option, respectively.

If you delete an entry, you are prompted to confirm the deletion before the entry is removed from the database. To alter a wanmap entry, you will be prompted for the entry you wish to alter, that is, the **Rem\_Address** field. The entry that you have selected is shown and you may change this entry after confirming that this is the entry you wish to alter.

### Listing Wanmap Entries

To check the entries you have made, select the **listentries** option, **4**, or an abbreviation, for example, **li**. The hostname and address entries in the wanmap database are displayed, for example:

```
LIST WANMAP ENTRIES
Remote_address          =12345678901234
Interface_address      =0101444568765
Remote_address          =us.ca.host
Interface_address      =2152,1334244,2233442332323
Remote_address          =12342342432
Interface_address      =us.ba.host
```

Press the RETURN key to see the Wanmap Menu [?, q]:

### Deleting a Wanmap Table

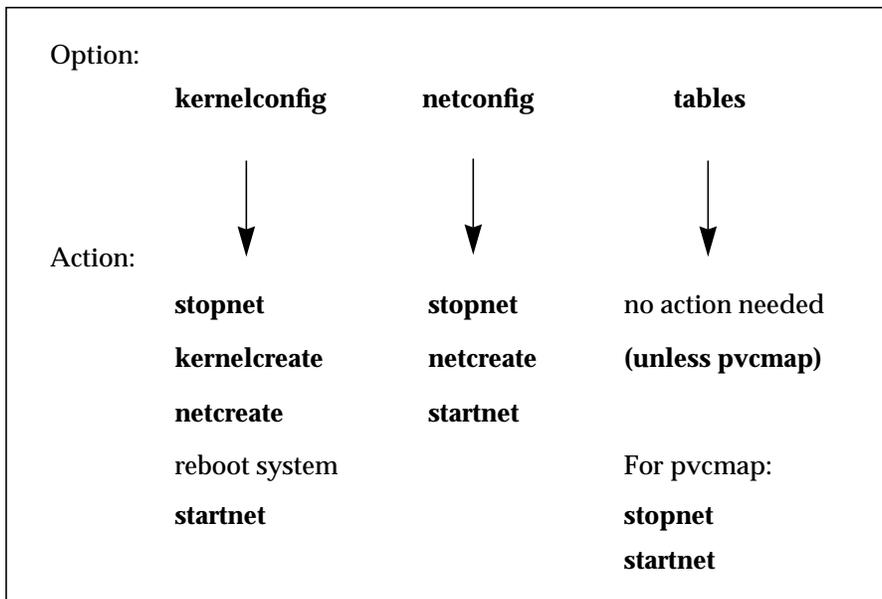
To delete a wanmap table, select the **delete** option, or an abbreviation, for example, **de**. You are prompted for the table you wish to delete. After you enter the table name, it is removed, if the table exists.

### Displaying a Wanmap Table

To display a wanmap table, select the **display** option, or an abbreviation, for example, **di**. You are prompted for the table you wish to display. After you enter the table name, it is displayed, if the table exists.

## Bringing Changes into Effect

This section shows what you have to do to activate any changes you have made, whichever area these affect. If you have modified the kernel using *kernelconfig*, you have to rebuild the kernel. If you have modified aspects of your network using *netconfig*, you have to recreate the network configuration. If you have only modified database tables, there is no need for further action here. See Figure 2-2 for available options.



**Figure 2-2** Activating Changes

When using *x25adm* it allows you to start and stop the network, and also to check its current status. In addition, you can choose to have the network start up automatically whenever you reboot the system.

At the Network Management menu, select **startstop** to display the following menu:

```
NETWORK START/STOP MENU
```

```
1 disablestart  disable the starting of the network on reboot
2 enablestart   enable the starting of the network on reboot
3 isnetup       determine whether or not network is running
4 startnet      start network
5 stopnet       stop network
```

```
Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

To prevent the network restarting on reboot, select option **disablestart**.

The following message is displayed:

```
The network will now NOT start when the system is rebooted.
Press the Return key to see the startstop menu [?, ^, q]:
```

The **enablestart** option has the opposite effect. By default, when you first install SX.25, the network will not start on re-boot.

Select **isnetup** to display the status of the network:

```
Network is up
```

Select **startnet** to start the network according to the parameters that you have configured. If you have already started the network, the following message appears:

```
Network was already up
```

```
Press the Return key to see the startstop menu [?, ^, q]:
```

Select **stopnet** to stop the network.



---

## Parameter Configuration

This chapter contains the following sections:

- “X.25 Configuration Files”
- “Changing a Stack File”

SX.25 implements a comprehensive range of configurable parameters for X.25, LAPB, WAN, IXE, XLB, and XLC. Four main sets of configurable parameters are covered in turn in this chapter: X.25, LAPB, and WAN. IXE is covered in Chapter 4, “Running SX.25 with TCP/IP.” The XLB and XLC stack files are described in “Changing a Stack File” later in this chapter. LLC-2 is covered in the reference manual pages.

These sets of parameters are held in files. Most of these parameters are configurable and can be changed by using the options of the *x25adm* utility to modify the relevant file.

### X.25 Configuration Files

**Note:** If you are running SX.25 with TCP/IP, you need to configure the parameters for the IXE driver. Refer to Chapter 4, “Running SX.25 with TCP/IP,” for a full explanation of IXE configuration.

The use of files to store parameters has two advantages:

- You can switch quickly between two different network setups, if the files already exist, without having to directly modify any parameters.
- If you do have to change one parameter, there’s no need to set all the rest. You can keep the existing values for these.

Two types of configuration files are used: template files and stack files. Template files contain parameter values. Stack files associate two or more template files.

The files come from two sources:

- A number of default files are provided with SX.25. These all have the prefix *def*.
- You can create your own files, based on the default files, to suit your own network topologies. These files cannot have the *def* prefix.

Once you reach a specific menu, there is no need to supply the suffix when you specify a file. The supplied LAPB file, for example, is specified as *def*, rather than *def.lapb*. To specify the file *def.dce80.x25*, just enter *def.dce80*. The default files of parameter values are as follows:

#### **X.25 template files**

Under X.25 there are several template files set up to suit the type of network (80, 84, or 88, DCE or DTE) that you wish to run:

- *def.dce80.x25*
- *def.dte80.x25*
- *def.lan.x25*
- *def.dce84.x25*
- *def.dte84.x25*
- *def.dce88.x25*
- *def.dte88.x25*

#### **LAPB template file**

The following template file is available:

- *def.lapb*

#### **WAN template file**

The following template file is available:

- *def.wan*

**IXE template files**

Under IXE, there are two default template files for running IP over X.25 on the WAN or the LAN, respectively:

- *def.wan.ixe*
- *def.lan.ixe*

If you plan to run SX.25 with TCP/IP, refer to Chapter 4, “Running SX.25 with TCP/IP.”

**XLB stack files**

These stack files associate X.25, LAPB (XLB), and WAN template files. There are six default files to suit the type of network (1980, 1984, 1988, DTE and DCE). The default XLB files are:

- *def.dte80.XLB*
- *def.dce80.XLB*
- *def.dte84.XLB*
- *def.dce84.XLB*
- *def.dte88.XLB*
- *def.dce88.XLB*

**XLC stack files**

These stack files associate X.25 and LLC-2 (XLC) template files. LLC-2 template files are handled with the DLPI module. The stack files associate two files together to provide parameters for an interface. There is one default XLC stack file:

- *def.lan.XLC*

## Modifying Files

If the values of any of the parameters in the provided files are inappropriate for your network, you can create new files containing the values you need. You cannot change or delete the default template files that are supplied with SX.25 and that have the *def.* prefix. You cannot use this prefix when naming new files.

From the Network Management menu, select the **paramconfig** option, or enter the **paramconfig** option directly from the command line:

```
x25adm paramconfig
```

The following menu is displayed:

```
PROTOCOL PARAMETER CONFIGURATION
```

```
1 changecurrent  change running configuration
2 changestack    change stack file
3 changetemplate change template files
4 deletestack    delete stack file
5 deletetemplate delete template files
6 display        display configuration
```

```
Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, q to QUIT:
```

The **display** option lets you look at existing parameter values. The **deletestack** and **deletetemplate** options let you remove unwanted stack and template files from your system.

The procedure for changing stack files is described in “Changing a Stack File,” the next section of this chapter.

There are two main ways of changing configuration parameters:

1. Select **changecurrent** to modify the current configuration for a subnetwork for X.25, LAPB, or LLC-2. Any changes you make here last only until the next time you stop the network. The following menu is displayed:

```
CHANGE CURRENT CONFIGURATION
```

```
1 lapb          change current LAPB configuration
2 x25           change current X.25 configuration
```

Enter a number, a name, the initial part of a name, or ?  
or <number>? for HELP, ^ to GO BACK, q to QUIT: 1

Select the group of parameters you want to modify. Your choice is displayed and you are asked to specify which subnetwork to change:

Choose which subnetwork you wish to reconfigure.

Current Subnetworks:

A            B

From:

Enter the identifier of the subnetwork you want to modify. The parameters are displayed in turn for you to change or skip. When you reach the end of the set, you are asked to confirm the changes you have made:

Do you wish to save existing changes? [y/n]: y

2. Select **changetemplate** to create a new template or modify one of your existing templates (not one of the default files). You have to specify which group of parameters to change: LAPB, LLC-2, WAN, X.25, or IXE. The IXE is covered in Chapter 4, "Running SX.25 with TCP/IP." The two possibilities are as follows.

### Creating a template

This can be based either on a current configuration, once the parameter values for a subnetwork have been set up, or one of the template or default files. When you specify which type of template file you want to create, say LAPB, a message like the following is displayed:

Changing a LAPB template file.

-----

Choose which template or subnetwork you wish to edit from.

Template files:

testfile            def

Current Subnetworks:

A            B

From:

Enter the name of the template file or the ID of the subnetwork that is to provide the initial parameter settings. This is confirmed for you, then the following is displayed:

```
Please enter the name of the file to write to (File name
should have between 2 and 9 characters [A-Za-z0-9._]).
```

```
To:
```

Enter the name of the new file which is to contain the modified values. If you enter the name of an existing file, you are warned:

```
That file already exists.
```

```
Are you sure you want to overwrite it? [y/n] n
```

The parameters are displayed in turn for you to change or skip. When you reach the end of the set, you are asked to confirm the changes you have made.

When you create a new file of parameters, rather than modifying existing files, you have to implement the new file. For example, if you are running X.25 over the LAN you have to set parameter values for the LLC layer of each of the X.25 links. This involves the following steps:

1. Use the new template file for each of the links from the X.25 module.
2. Enter `x25adm netconfig` to access the Network Management menu and do the following:
  - Enter the name of the new template file at the x25 module option.
  - Use the `netcreate` option to bring the changes you have made into effect.

### Modifying an existing template file

The procedure for modifying an existing template file is similar to that for creating a new file, but you supply the name of an existing file as the destination file.

## X.25 Parameters

From the Protocol Parameter Configuration menu, select one of the following to change X.25 parameters:

### **changetemplate**

Change the values in the files you are using. You have to specify from which template or subnetwork you want to edit; refer to “Modifying Files” in the previous subsection.

If you create a new file, you will have to implement the changes using the **x25config** option of the Network Daemon Configuration menu. To do this, reinitialize the network using the **netcreate** option of the Network Daemon Configuration menu, then start up the network using the **startnet** option of the Start/Stop menu.

### **changecurrent**

Change the current configuration. You have to specify which subnetwork you want to reconfigure; refer to “Modifying Files” in the previous subsection.

### **changestack**

Change the files held in the XLC and XLB stack files. Because you do not change individual parameters when you do this, it is not covered in detail here. Refer to “Changing a Stack File” (the next section) for more information.

Whether you select **changecurrent** or **changetemplate**, you are taken step-by-step through the parameters described here. If you do not want to change the value displayed, press Return to move to the next parameter. When you have changed all the parameters you need to, you can skip the remainder. The changes you have made are checked for consistency, and you are asked to confirm that you want to implement them.

At each prompt, you can enter ? to display an explanation of its use and the range of values permitted. Enter ^ to return to the **changetemplate** menu or q to quit **paramconfig**.

The parameters are:

**Network Protocol Mode - NET\_MODE**

This determines the characteristics of the network protocol. For example, a value "X25\_LLC" specifies X.25(84) or X.25(88) over LLC-2 procedures. Set NET\_MODE according to the protocol type you plan to use on your network. Currently valid protocols are:

1 X25_LLC	9 TELENET	17 DATAPAK
2 X25_88	10 TRANSPAC	18 DATANET
3 X25_84	11 TYMNET	19 DCS
4 X25_80	12 DATEX_P	20 TELEPAC
5 PSS	13 DDX_P	21 F_DATAPAC
6 AUSTPAC	14 VENUS_P	22 FINPAC
7 DATAPAC	15 ACCUNET	23 PACNET
8 DDN	16 ITAPAC	24 LUXPAC

Enter the appropriate number at the menu prompt.

**X.25 Version - X25\_VERSION**

This lets you specify the version of the X.25 protocol used over the network, either X.25(80), X.25(84), or X.25(88).

X25\_VERSION

The current value is 80

Type: integer {80, 84, 88}

Or : RETURN = 80

? = HELP, ^ = GO UP A LEVEL, q = QUIT

Enter a new value or press Return to retain the current value.

**Note:** The NET\_MODE of X25\_LLC overrides an X.25(80) value in this field to X.25(84).

### Level 3 Packet Level Protocol Mode - L3PLPMODE

This indicates either the DTE/DCE nature of the link's Packet Level Protocol (PLP), or how that nature is to be resolved. A value of 0 indicates DCE, value 1 means DTE, while 2 indicates that this is to be resolved using the procedures described in ISO 8208 for DTE-DTE operation. For example, for a NET\_MODE of X25\_LLC, this parameter is set to 2.

### Specifying Logical Channel Ranges

The following set of parameters assign ranges for permanent virtual circuits, as well as incoming, two-way, and outgoing logical channels. The parameters are presented in pairs specifying the highest and lowest channel numbers of each type. Table 3-1 shows the parameter defaults and meanings.

**Table 3-1** Parameter Defaults

Parameter	Default	Meaning
LPC	0	Lowest Permanent Virtual Circuit
HPC	0	Highest Permanent Virtual Circuit
LIC	0	Lowest Incoming Logical Channel
HIC	0	Highest Incoming Logical Channel
LTC	400	Lowest Two-way Logical Channel
HTC	43F	Highest Two-way Logical Channel
LOC	0	Lowest Outgoing Logical Channel
HOC	0	Highest Outgoing Logical Channel

In a DTE/DTE environment, one DTE sees the range of LICs as presented here. The other DTE sees these ranges from the viewpoint of a DCE, for example, it views the range from LIC to HIC as one-way *outgoing*.

The numbers of logical channels of each type are calculated from the values of the pairs of parameters:

$$HPC - LPC + 1 = \text{Number of PVCs.}$$

**Note:**  $H*C = L*C = 0$  denotes no channels in that grouping.

All the prompts for these parameter values are structured in the same way. The following example shows the prompt for a value for the lowest permanent virtual circuit:

```
LPC
The current value is 0
Type: string
Or  : RETURN = 0
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

Enter a new value or press Return to retain the current value. You can press ? to get help, ^ to return to the **changetemplate** menu, or q to quit **paramconfig**.

When you have finished making changes to the X.25 parameters, a series of checks is performed before the file of parameters is written. If the channel ranges you have specified are inconsistent, an error message is displayed. For example, you should not have any of the highest logical channel numbers set lower than the lowest channel number. Also, if any of the channel ranges calculated from the channel number parameters overlap, this is reported as an error. You must correct the parameter values or they will be ignored and all your other changes abandoned.

### Sequence Numbering

#### Sequence Numbering Option - THISGFI

This indicates whether Modulo 8 or Modulo 128 sequence numbering operates on the network. It takes one of two values:

8	Modulo 8
128	Modulo 128

## Packet and Window Sizes

### Maximum Packet Size - LOCMAXPKTSIZE, REMMAXPKTSIZE

These two parameters define the maximum packet sizes for the two transmission directions. LOCMAXPKTSIZE, for example, specifies the maximum local-to-remote packet size.

Use this parameter to set the maximum size of data packets that are acceptable on the X.25 link. On any incoming call, a value for the packet size parameter greater than LOC or REMMAXPKTSIZE is negotiated down to this value when the call is accepted. The data packet size, in octets, is two to the power of the number you enter here. Enter a number in the range 7 to 12, inclusive. This gives a range for the data packet size of 128 to 4096 octets ( $2^{12}$ , that is,  $1 << 12$ ).

**Note:** Ensure that LOC or REMMAXPKTSIZE is greater than the frame size that you have set for the underlying protocol; the LAPB I frame on the WAN or the LLC-2 I frame on the LAN. The I frame size is set in the LAPB or LLC-2 template of configurable parameters.

### Default Packet Size - LOCDEFPKTSIZE, REMDEFPKTSIZE

These two parameters define the default packet sizes for the two transmission directions. LOCDEFPKTSIZE, for example, specifies the default local-to-remote packet size.

Use this parameter to specify the default packet size for a particular network. The usual standard value is 7, which gives a default data packet size for each direction of transmission of 128 octets ( $2^7$ , that is,  $1 << 7$ ). The value may be nonstandard, if it is agreed between all communicating parties on the LAN or between the DTE and DCE.

### Maximum Window Size - LOCMAXWSIZE, REMMAXWSIZE

These two parameters define the maximum window sizes for the two transmission directions. LOCMAXWSIZE, for example, specifies the maximum local-to-remote window size.

Use this parameter to select the maximum window size acceptable on the subnetwork. On any incoming call, a value for the window size parameter greater than LOC or REMMAXWSIZE is negotiated down to this value when the call is accepted. For Modulo 8 networks, the range for LOC or REMMAXWSIZE is 2 to 7, inclusive, while for Modulo 128 the range is 2 to 127, inclusive.

**Default Window Size - LOCDEFWSIZE, REMDEFWSIZE**

These two parameters define the default window sizes for the two transmission directions. LOCDEFWSIZE, for example, specifies the default local-to-remote window size.

Use this parameter to specify the value of the default window size for a particular network. The usual standard value is 2. The value may be nonstandard provided it is agreed between all communicating parties on the LAN, or between the DTE and DCE.

**Note:** The sequence numbering scheme, modulo 8 or modulo 128, affects the range of this parameter. For modulo 8 networks, the range for LOC or REMDEFWSIZE is 1 to 7, inclusive, while for modulo 128 the range is 1 to 127, inclusive.

**Maximum NSDU Length - MAXNSDULEN**

SX.25 tries to concatenate data packets marked with the “More Data” mark (M-bit) into a single network service data unit (NSDU). However, to avoid running out of buffers, you can specify a maximum length. When this maximum is exceeded, concatenation stops and the data currently held is passed to the NS user (with the More parameter set). Applications can override this parameter for specific circuits using the **nsdulimit** field of the Quality Of Service parameter on Connect Requests and Connect Confirmations. The default value for this parameter is 256, the maximum 32000.

### Setting Timer Values

All timer parameters are displayed in the same format, for example:

```
T20VALUE
The current value is 1800
Type: integer {0-32000}
Or : RETURN = 1800
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

At each menu, enter a new value or press Return to retain the current value. You can also enter ?, ^ or q.

All timer values are measured in tenths of a second. The use, defaults, and permitted ranges are as follows:

#### Acknowledgment Delay - ACKDELAY

SX.25 attempts to suppress the generation of Level 3 Receive Ready control packets. Acknowledgment carried by data or multiple acknowledgments is preferred to each data packet being explicitly and separately acknowledged. ACKDELAY specifies the maximum time, in tenths of a second, during which a pending acknowledgment is withheld. The permitted range is 1 - 32000.

The remaining timer parameters, some of which are interdependent, specify the values of various timers.

<b>T20value</b>	This specifies, in number of tenths of a second, the DTE timer parameter T20, the Restart Request Response Timer. The default value is 1800, the permitted range 0-32000.
<b>T21value</b>	This specifies, in number of tenths of a second, the DTE timer parameter T21, the Call Request Response Timer. The default value is 2000, the permitted range 0 - 32000.
<b>T22value</b>	This specifies, in number of tenths of a second, the DTE timer parameter T22, the Reset Request Response Timer. The default value is 1800, the permitted range 0 - 32000.

<b>T23value</b>	<p>This specifies, in number of tenths of a second, the DTE timer parameter T23, the Clear Request Response Timer.</p> <p>The default value is 1800, the permitted range 0 - 32000.</p>
<b>Tvalue</b>	<p>This field is related to, but does not match exactly, the DTE Window Status Transmission Timer, T24. It specifies the maximum time, in tenths of a second, for which data received from the remote transmitter can remain unacknowledged due to a flow control condition. After expiration of this timer, any outstanding acknowledgments are carried by an X.25 Level 3 Receive-Not-Ready packet. Use this timer to ensure that if the remote transmitter does not receive an acknowledgment, this does not cause resets within the virtual circuit. This timer should be approximately half the value set for <b>T25value</b>.</p> <p>The default value is 750, the permitted range 0 - 32000.</p> <p><b>Note:</b> The implementation of this timer does not mean that the window status is transmitted every <b>Tvalue</b> tenths of a second.</p>
<b>T25value</b>	<p>This timer specifies, in number of tenths of a second, the value of DTE timer parameter T25, the Window Rotation Timer. This timer should be approximately twice the value set for Tvalue.</p> <p>The default value is 1500, the permitted range 0 - 32000.</p>
<b>T26value</b>	<p>Use this parameter to specify, in tenths of a second, the DTE timer parameter T26, the Interrupt Response Timer.</p> <p>The default value is 1800, the permitted range 0 - 32000.</p>
<b>Idlevalue</b>	<p>Use this parameter to specify the period, in tenths of a second, over which a link-level connection associated with no connections is maintained. If the link is to a WAN, then this should be zero (that is, infinity). This timer is only appropriate when running X.25 over the LAN.</p> <p>The default LAN value is 600, the default WAN value is 0. For both, the permitted range is 0 - 32000.</p>

**Connectvalue**

Use this parameter to specify the number of tenths of a second over which the DTE/DCE resolution should be completed. This prevents two packet-level entities failing to resolve their DTE/DCE nature. When this timer expires, the link connection is disconnected and all pending connections aborted.

The default value is 2000, the permitted range 0 - 32000.

**Setting Counter Values**

The following set of parameters specifies the values of counters. All these parameters are displayed in the same format, for example:

```
R20VALUE
```

```
The current value is 1
```

```
Type: integer {1-255}
```

```
Or : RETURN = 1
```

```
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

```
New value:
```

Enter a new value or press Return to retain the current value. Each prompt also lets you enter ? for help, ^ to return to the **changetemplate** menu, or q to quit **paramconfig**.

- |                 |   |
|-----------------|---|
| <b>R20value</b> | This value specifies the DTE Restart Request Retransmission Count. The default value is 1, the permitted range 1 - 255. |
| <b>R22value</b> | This value specifies the DTE Reset Request Retransmission Count. The default value is 1, the permitted range 1 - 255.   |
| <b>R23value</b> | This specifies the DTE Clear Request Retransmission Count. The default value is 1, the permitted range 1 - 255.         |

### Transit Delay Parameters

LOCALDELAY and ACCESSDELAY are the values, in milliseconds, of the transit delay attributed, respectively, to internal processing and the effect of the line transmission rate. These parameters are displayed in turn for you to enter a new value or press Return to retain the default:

```
LOCALDELAY
The current value is 5
Type: integer {0-32000}
Or : RETURN = 5
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

The default for both parameters is 5, the permitted range 0 - 32000.

### Throughput Class Parameters

Ensure that you observe the following points when you set these parameters:

- The value of LOC or REMMAXTHCLASS should be greater than or equal to that of LOC or REMDEFTHCLASS, respectively.
- The value of LOC or REMDEFTHCLASS should, in turn, be greater than or equal to that of LOC or REMMINTHCLASS.

### Maximum Throughput Class - LOCMAXTHCLASS, REMMAXTHCLASS

These two parameters define the maximum packet sizes for the two transmission directions. LOCMAXTHCLASS, for example, specifies the maximum local-to-remote packet size.

This is the maximum supported value of the throughput class Quality Of Service parameter. ISO 8208 specifies a range for LOC or REMMAXTHCLASS of greater than or equal to 3, and less than or equal to 12, corresponding to a range of 75 to 48000 bps. However, the valid range supported here is 0 to 15. This caters for non-standard X.25 implementations, which use the Throughput Class parameters (see Throughput Class Type) to map to packet/window sizes.

**Default Throughput Class - LOCDEFTHCLASS, REMDEFTHCLASS**

These two parameters define the default packet sizes for the two transmission directions. LOCDEFTHCLASS, for example, specifies the default local-to-remote packet size.

In some networks, for example, TELENET, negotiation of throughput class is towards a configured default throughput class. In such cases, the flag THCLASS\_NEG\_TO\_DEF, described later in this section, is non-zero and LOC or REMDEFTHCLASS is the default. In other PSDNs, you should set LOC or REMDEFTHCLASS to the value of LOC or REMMAXTHCLASS, described previously.

**Minimum Throughput Class - LOCMINTHCLASS, REMMINTHCLASS**

These two parameters define the minimum packet sizes for the two transmission directions. LOCMINTHCLASS, for example, specifies the minimum local-to-remote packet size.

According to ISO 8208, the throughput class parameter is defined in the range greater than or equal to 3, and less than or equal to 12. Some PSDNs may provide a different mapping, in which case LOC or REMMINTHCLASS is the minimum value.

When you have finished changing all the X.25 parameters, the consistency of the changes you made here is checked. If the value of LOC or REMMAXTHCLASS is less than the respective LOC or REMDEFTHCLASS, an error message is displayed. The *x25adm* program then lets you go back and change the parameter values which are invalid.

### Closed User Groups

Table 3-2 shows how six parameters govern how requests for Closed User Group (CUG) facilities are handled.

**Table 3-2** Handling CUG Requests

Parameter	Usage
SUB_CUG	Subscribe to CUGs without incoming or outgoing access.
SUB_PREF	Subscribe to preferential CUGs.
SUB_CUGOA	Subscribe to CUGs with outgoing access.
SUB_CUGIA	Subscribe to CUGs with incoming access.
CUG_FORMAT	Basic or extended CUG format.
BAR_CUG_IN	Reject incoming calls containing any CUG facility.

The first four parameters specify the *type* of CUG you subscribe to. The prompts are like this:

```
SUB_CUG
The current value is N
Type: string
Or : RETURN = N
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

Specify the appropriate values for your network for these four parameters.

The fifth parameter specifies the *format* of CUG you want to subscribe to:

```
CUG_FORMAT
The current value is 0
Type: integer {0-1}
Or : RETURN = 0
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

Enter **0** if you subscribe to Basic Format CUGs, or **1** if you subscribe to Extended Format CUGs.

Set this flag to reject incoming calls carrying any Closed User Group optional facility. This is necessary in some networks, for example, DDN.

BAR\_CUG\_IN

The current value is N

Type: string

Or : RETURN = N

? = HELP, ^ = GO UP A LEVEL, q = QUIT

New value:

When it is non-zero, such calls are rejected, otherwise incoming Closed User Group facilities are handled with the facilities in the incoming call.

**Note:** When you have finished making configuration changes, the consistency of these changes is checked. If incompatible values are found, for example, if X25\_VERSION is 80 but you have specified extended format CUGs (CUG\_FORMAT = 1), an error message is displayed and *x25adm* lets you go back to make the appropriate changes.

### Subscription Options

Table 3-3 shows the parameters (SUB\_MODES) used to tune options for a particular subnetwork.

**Table 3-3** Tuning Options for a Subnetwork

Parameter	Usage
SUB_EXTENDED	Subscribe to extended call packets (Window and Packet Size negotiation is permitted).
BAR_EXTENDED	Treat Window and Packet Size negotiation in incoming packets as a procedure error.
SUB_FSELECT	Subscribe to Fast Select with no restriction on response.
SUB_FSRRESP	Subscribe to Fast Select with restriction on response.
SUB_REVCHARGE	Let incoming calls specify the Reverse Charging facility.

<b>Table 3-3</b> (continued)      Tuning Options for a Subnetwork	
<b>Parameter</b>	<b>Usage</b>
SUB_LOC_CHG_PREV	Subscribe to local charging prevention.
SUB_TOA_NPI_FMT	Subscribe to using TOA/NPI address format.
BAR_TOA_NPI_FMT	Treat incoming TOA/NPI address formats as a procedure error.
SUB_NUI_OVERRIDE	Subscribe to NUI override.
BAR_INCALL	Bar incoming calls.
BAR_OUTCALL	Bar outgoing calls.

Each parameter is displayed in turn for you to supply a value or retain the default. For example:

```

SUB_FSELECT
The current value is Y
Type: string
Or : RETURN = Y
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
    
```

Enter **y** to subscribe to Fast Select with no restriction on response, or **n** if you do want not to subscribe.

**Note:** The use of extended call packets allows window and packet size negotiation. The SUB\_EXTENDED parameter, if set, permits the use of extended CALL REQUEST and CALL ACCEPT packets. The BAR\_EXTENDED parameter, if not set, permits the use of extended INCOMING CALL and CALL CONFIRM packets.

## Localization Information

### Subnetwork Modes

Table 3-4 shows the parameters used to tune options for a particular subnetwork.

**Table 3-4** Tuning Parameters

Parameter	Usage
ACC_NODIAG	Allow the omission of the diagnostic byte in incoming Restart, Clear, and Reset Indication.
USE_DIAG	Use diagnostic packets.
CCITT_CLEAR_LEN	(This applies to 1980 networks only.) Restrict the length of a Clear Indication to 5 bytes and a Clear Confirm to 3 bytes.
BAR_DIAG	Disallow diagnostic packets.
DISC_NZ_DIAG	Discard diagnostic packets on a non-zero LCN.
ACC_HEX_ADD	Allow hexadecimal digits in DTE addresses.
BAR_NONPRIV_LISTEN	Allow only superusers to listen for incoming calls.

**Note:** BAR\_DIAG and DISC\_NZ\_DIAG control the treatment of incoming diagnostic packets. When BAR\_DIAG is set, incoming diagnostic packets are not allowed. When DISC\_NZ\_DIAG is set, incoming diagnostic packets are discarded on non-zero logical channel numbers.

### International call recognition - INTL\_ADDR\_RECOGN

This governs whether outgoing international call requests are to be recognized and how. Set the value according to the following:

- 0 International calls are not distinguished.
- 1 The DNIC (see below) of the called DTE address is examined and compared that held in psdn\_local members dnic1 and dnic2. A mismatch implies an international call.

- 2 International calls are distinguished by having a 1111 prefix on the DTE address. For example, DATAPAC has this feature.
- 3 International calls are distinguished by having a 0000 prefix on the called DTE address.

This feature is generally used in conjunction with the parameter INTL\_PRIORITISED.

**Data Network Identification Code - DNIC**

This contains the first four BCD digits of the DNIC (Data Network Identification Code) and is only used when INTL\_ADDR\_RECOGN = 1.

**International Call Priority - INTL\_PRIORITISED**

This determines whether some prioritization method is to be used for international calls (assuming that the PSDN supports such a feature) and is used in conjunction with the parameters PRTY\_ENCODE\_CONTROL and PRTY\_PKT\_FORCED\_VAL. INTL\_PRIORITISED has two values:

- N - No priority.
- Y - Prioritize according to ptry\_encode\_control.

**Encode Control - PRTY\_ENCODE\_CONTROL**

This governs how the priority request is to be encoded for this PSDN. Set the value according to the following:

- 0 The priority is encoded according to X.25(88).
- 1 Encode the priority request using the DATAPAC Priority Bit (1976 version).
- 2 Encode the priority request using the DATAPAC Traffic Class (1980 version employing the Calling Network facility marker).

**Priority Packet Forced Value - PRTY\_PKT\_FORCED\_VAL**

If this parameter has a non-zero value, then it implies that all priority call requests and incoming calls should have the associated packet size parameter forced to this value.

**Note:** The actual packet size is two to the power of this parameter. For example, 7 implies 128-byte packets. The permitted range is 4 to 12, inclusive, or zero, implying that the default packet size is to be used.

**Source Address Control - SRC\_ADDR\_CONTROL**

This lets you override or set the calling address in outgoing call requests for this PSDN. Set the value according to the following:

- |   |  |
|---|--|
| 0 | No special action. Calling DTE addresses are encoded as if provided by the network service user.   |
| 1 | Force omission of the calling DTE address, even if the network service user supplied one.  |
| 2 | If the network service user does not supply a DTE address, use the configured DTE address (LOCAL_ADDRESS) for this PSDN (which can, of course, be NULL). |
| 3 | Force the calling DTE address to that contained in LOCAL_ADDRESS, even if the network service user supplied one.   |

**D-Bit Control**

The X.25 standard does not give specific recommendations for handling the D-bit. To provide flexibility to cope with local PSDN requirements, SX.25 lets you specify how the D-bit should be handled. The following group of parameters defines what is done when the communicating parties do not agree on the use of the D-bit. There are two pairs of parameters:

- DBIT\_ACCEPT\_IN and DBIT\_ACCEPT\_OUT control the handling of disagreement in D-bit support in the call setup phase
- DBIT\_DATA\_IN and DBIT\_DATA\_OUT do the same for the data transfer phase.

All menus take the same form, for example:

DBIT\_ACCEPT\_IN

The current value is 2

Type: integer {0-2}

Or : RETURN = 2

The first pair of D-bit parameters have a permitted range of 0-2 with the following meanings:

- 0                    Leave the D-bit set and pass the packet on.
- 1                    Zero the D-bit and pass the packet on.
- 2                    Clear the call (the default).

DBIT\_ACCEPT\_IN defines what happens when a Call Accept is received and there is no local D-bit support. DBIT\_ACCEPT\_OUT defines what happens when a Call Accept is sent and there is no request for D-bit from the other end. This parameter has the same permitted range, usage, and default as DBIT\_ACCEPT\_IN, above.

The second pair of D-bit parameters control the handling of disagreement in D-bit support in data packets. These parameters have a permitted range of 0-2 with the following meanings:

- 0                    Leave the D-bit set and pass the packet on.
- 1                    Zero the D-bit and pass the packet on.
- 2                    Reset the call (the default).

The parameters are:

DBIT\_DATA\_IN defines what happens when a data packet is received and there is no request for the D-bit from the local user. DBIT\_DATA\_OUT defines what happens when a data packet is sent and there is no indication of D-bit support from the remote party.

### Throughput Class Window/Packet Parameters

Some implementations of X.25 do not use the X.25 packet and window negotiation. These use instead the Throughput Class Window/Packet Map parameters. For PSDNs that use such implementations, the THCLASS\_TYPE parameter specifies the way in which the THCLASS\_WMAP and THCLASS\_PMAP parameters are used to map to the window and packet sizes.

### Throughput Class Negotiation - THCLASS\_NEG\_TO\_DEF

Specify a non-zero value for networks like TELENET, which require negotiation of the Throughput Class to be towards the defaults, held in LOC and REMDEFTHCLASS.

```
THCLASS_NEG_TO_DEF
```

```
The current value is N
```

```
Type: string
```

```
Or : RETURN = N
```

```
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

N indicates that negotiation towards the default is not used, but that the throughput class is negotiated according to the X.25 procedures instead.

### Throughput Class Type - THCLASS\_TYPE

This specifies the way in which the THCLASS\_WMAP and THCLASS\_PMAP parameters are used to map to the window and packet sizes. The prompt for THCLASS\_TYPE is displayed as follows:

```
THCLASS_TYPE
```

```
The current value is 0
```

```
Type: integer {0-3}
```

```
Or : RETURN = 0
```

```
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

```
New value:
```

For non-standard X.25 implementations, enter a value in the range 1-3. Possible values and their meanings are as follows:

- |   |  |
|---|--|
| 0 | No special action is to be taken on Throughput Class.  |
| 1 | Use only the low nibble of the Throughput Class map parameters to map window/packet size for both directions. Encode the high nibble as zero.    |
| 2 | Use only the high nibble of the Throughput Class map parameter to map window and packet size for both directions. Encode the low nibble as zero. |
| 3 | Use both nibbles of the Throughput Class map parameters to map window/packet sizes for the appropriate directions.                               |

Notes:

1. The values of LOC and REMMAXTHCLASS may limit the range of the mapped values.
2. For values 1 and 2, the following points apply:
  - Do not select these values when window and packet sizes can appear in call setup packets (that is, where there is window and packet size negotiation) because this algorithm is designed for PSDNs that support the mapping procedure only.
  - In call requests, the network service user should specify equal values for **lothroughput** and **remthroughput** in the qosformat, to ensure that it behaves correctly. See the high and low nibble usage above for these two values.
  - The user is stopped from negotiating window and packet size. The throughput class will not be indicated in a connect indication.
3. When using value 3, window and packet sizes can be negotiated by the network service user only through the throughput class parameter.

If you are subscribing to the extended facility option, negotiation using the flow negotiation parameters is overridden. Don't select value 3 when window and packet sizes can appear in call setup packets (that is, where there is window and packet size negotiation), because this algorithm is designed for PSDNs that support the mapping procedure only.

4. Because window and packet sizes can be mapped using these three values without window and packet negotiation, it is important that the THCLASS\_WMAP/THCLASS\_PMAP is correct for the PSDN. This ensures that both called and calling parties agree on the values associated with a particular Throughput Class.

#### **Throughput Class Window Mapping - THCLASS\_WMAP**

#### **Throughput Class Packet Mapping - THCLASS\_PMAP**

These two parameters map from the value of the throughput class for each direction of transmission (a number from 0 to 15) to a window and a packet parameter, respectively.

The first parameter is THCLASS\_WMAP:

```
THCLASS_WMAP
```

```
The current value is 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
```

```
Type: string
```

```
Or : RETURN = 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
```

```
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

Enter zero for any individual value to use the default value for the subnetwork. Press Return to use the default values for all elements. Entries should comprise 16 integers separated by commas or full stops (periods). The permitted range is between 1 and the value of LOC or REMMAXWSIZE that you have set, the maximum possible being 127. The maximum for the subnetwork is the smaller of the values, LOC and REMMAXWSIZE.

The second parameter is THCLASS\_PMAP:

```
THCLASS_PMAP
```

```
The current value is 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
```

```
Type: string
```

```
Or : RETURN = 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
```

```
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

Enter zero for any individual value to use the default value for the subnetwork. Press Return to use the default values for all elements. Entries should be of 16 integers, separated by commas or full stops (periods). The permitted range is between 4 and the value of LOC or REMMAXPKTSIZE that you have set, the maximum possible being 12. The maximum for the subnetwork is the smaller of the two values, LOC and REMMAXPKTSIZE.

These are the last X.25 parameters. You are asked if you want to confirm the changes you have made, and consistency checks are then performed.

### **X.25 Level II (LAPB) Parameters**

The **paramconfig** option of the Network Configuration menu lets you set the LAPB parameters for each link. LAPB(80/84/88) and LAP(80/84/88) are supported, as specified in the appropriate CCITT recommendations. LAPB supports both modulo 8 frame sequence numbering and modulo 128 frame sequence numbering.

**Note:** The terms LAP and LAPB are used interchangeably in this section.

From the Protocol Parameter Configuration menu, select one of the following to change LAPB parameters:

#### **changetemplate**

Change the values in the files you are using. You have to specify which template or subnetwork you want to edit from; refer to “Modifying Files” earlier in this chapter.

If you create a new file, you will have to implement the changes using the **x25config** option of the Network Daemon Configuration menu. To do this, reinitialize the network using the **netcreate** option of the Network Daemon Configuration menu, then start up the network using the **startnet** option of the Start/Stop menu.

#### **changecurrent**

Change the current configuration. You have to specify which subnetwork you want to reconfigure; refer to “Modifying Files” earlier in this chapter.

**changestack** Change the files held in the XLC and XLB stack files. Because you do not change individual parameters when you do this, it is not covered in detail here. Refer to “Changing a Stack File” (the next section) for more information.

Whether you select **changecurrent** or **changetemplate**, you are taken step-by-step through the parameters described here. If you do not want to change the value displayed, press Return to move to the next parameter. When you have changed all the parameters you need to, you can skip the remainder. The consistency of your changes is checked and you are asked to confirm that you want to implement them.

At each prompt, you can enter ? to display an explanation of its use and the range of values permitted. Enter ^ to return to the **changetemplate** menu or q to quit **paramconfig**.

All timers are measured in tenths of a second. You can configure the following parameters for each subnetwork:

**Maximum Number of Transmissions - N2**

This is the maximum number of times that a PDU is sent following the expiration of the Acknowledgment Timer, the P-bit timer, or the Reject Timer. It also limits the number of times Receive Ready with P-bit set is sent when remote busy is true and the Busy Timer expires.

The default value is 10, the permitted range 1 - 255.

**Acknowledgment Timer - T1**

This is the time, in tenths of a second, during which the LAPB expects to receive an acknowledgment to an outstanding IPDU or an expected response to a sent unnumbered PDU.

The default value is 10, the permitted range 1 - 3000.

**P-bit Timer**

This is the time, in tenths of a second, during which the LAPB expects to receive a PDU with the F-bit set to 1 in response to a command with the P-bit set to 1. This value should be less than that specified for the Acknowledgment Timer.

The default value is 7, the permitted range 1 - 3000.

**Reject Timer** This is the time, in tenths of a second, during which the LAPB expects to receive a reply to a sent REJ TPDU.  
The default value is 25, the permitted range 1 - 10000.

**Busy-state Timer**  
This is the time, in tenths of a second, during which the LAPB waits for an indication of the clearance of a busy condition at the other LAPB.  
The default value is 100, the permitted range 1 - 30000.

**Idle Timer** This is the time, in tenths of a second, during which the LAPB expects to receive a PDU from the other LAPB. If it expires, the P/F cycle, which may result in link disconnection, is initiated. A value of zero will disable this function.  
The default value is 250, the permitted range 0 - 32000.

**Maximum Delay before Sending RR**  
This is the maximum delay in tenths of a second before transmitting a delayed RR (Reset Request). This must be considerably less than the Acknowledgment Timer value.  
The default value is 4, the permitted range 0 - 3000.

**Maximum Number of Unacknowledged IPDUs**  
The maximum number of unacknowledged IPDUs that can be received before the RR acknowledging them all must be sent.  
The default value is 3, the permitted range 0 - 127.

**Transmit Window Size**  
The number of unacknowledged IPDUs that may be sent.  
In normal mode, when Modulo 8 sequence numbering is used, the default value is 7, the permitted range 1 - 7.  
In extended mode, when Modulo 128 sequence numbering is used, the default value is 7, the permitted range 1 - 127.

**Transmit Probe**  
The position before the window is closed at which an IPDU is sent with the P-bit set to ask for an acknowledgment from the receiver.

In normal mode, when Modulo 8 sequence numbering is used, the default value is 0, the permitted range 0 - 7.

In extended mode, when Modulo 128 sequence numbering is used, the default value is 0, the permitted range 0 - 127.

#### **Maximum Size of LAPB I frame**

LAPB requires all incoming I frames above a certain size to be rejected by a FRMR. This parameter specifies the maximum size. It comprises:

maximum X.25 data size + X.25 protocol length + LAPB protocol length

The default value is 261. Permitted values are: 261-263, 517-519, 1029-1031, 2053-2055, 4101-4103.

#### **Link Level Conformance**

Several parameters control LAPB conformance tuning. These parameters define the operation of LAPB in certain states. Each takes a yes or no entry. The descriptions show what happens if the parameter is set on (Y):

##### **IGN\_UA\_ERROR**

When the connection is in an ERROR state, ignore any UA (Unnumbered Acknowledgment) frame received.

##### **FRMR\_FRMR\_ERROR**

When the connection is in an ERROR state, retransmit a frame reject if one is received.

##### **FRMR\_INVRSP\_ERROR**

When the connection is in an ERROR state, transmit a frame reject if an invalid frame response is received.

##### **SFRAME\_PBIT**

If an S-Frame is received without the P-bit set, send a frame reject.

##### **NO\_DM\_ADM**

Do not transmit a DM on entry to ADM state.

This is the last LAPB parameter; you are asked to confirm the changes you have made:

Do you wish to save existing changes? [y/n]

TELENET requires FRMR responses to be sent in response to a received FRMR, and also to received Invalid Response frames.

DDN requires UA frames while in ERROR state to be ignored, otherwise the FRMR is retransmitted.

PSS requires supervisory command frames to have the P-bit set.

### WAN Parameters

The **paramconfig** option of the Network Configuration menu lets you set the WAN parameters for each link. From the Protocol Parameter Configuration menu, select one of the following to change LAPB parameters:

#### **changetemplate**

Change the values in the files you are using. You have to specify from which template or subnetwork you want to edit; refer to “Modifying Files” earlier in this chapter.

If you create a new file, you will have to implement the changes using the **x25config** option of the Network Daemon Configuration menu. To do this, reinitialize the network using the **netcreate** option of the Network Daemon Configuration menu, then start up the network using the **startnet** option of the Start/Stop menu.

#### **changecurrent**

Change the current configuration. You have to specify which subnetwork you want to reconfigure; refer to “Modifying Files” earlier in this chapter.

#### **changestack**

Change the files held in the XLC and XLB stack files. Because you do not change individual parameters when you do this, it is not covered in detail here. Refer to “Changing a Stack File” (the next section) for more information.

Select **changecurrent** or **changetemplate** to change the two parameters described here. These parameters are configurable on a per-link basis.

**WAN maximum frame size**

This is the maximum WAN frame size allowed.

The default value is 262.

Permitted values are: 262-264, 518-520, 1030-1032, 2054-2056, 4102-4104.

**WAN baud rate**

This is the baud rate for the WAN line.

The default value is 9600.

Permitted values are: 0, 110, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 48000, 56000, 64000.

A baud rate of zero means that external timing is used.

**WAN translate**

This specifies whether the supplied address should be translated into a format suitable for the class of interface.

The default value is N.

Permitted values are: Y, N.

**WAN phys\_int**

This is used to describe the physical interface of the WAN line.

The default value is 1.

Permitted values are:

0	X21
1	V28
2	V35

**WAN connect\_proc**

This specifies to the WAN driver which specified calling procedures to use.

The default value is 0.

Permitted values are:

- 0 No calling procedures used.
- 1 X21 calling procedures are used.
- 2 V25 calling procedures are used.

If X21 calling procedures are selected, then the following timers are configurable.

**WAN X21\_T1**

This is the time, in tenths of a second, that the DTE will wait for the DCE to respond with “proceed-to-select” from a “call request.”

The default value is 3.

Permitted values are: 3-30.

**WAN X21\_T2**

This is the time, in tenths of a second, that the DTE will wait for the DCE to enter data transfer state after an end-of-selection command has been issued to the DCE.

The default value is 20.

Permitted values are: 20-200.

**WAN X21\_T3A**

This is the time, in tenths of a second, that the DTE will wait for the DCE to respond with “call progress signals” or “DCE-provided information.”

The default value is 6.

Permitted values are: 6-60.

**WAN X21\_T4B**

This is the time, in tenths of a second, that the DTE will wait for the DCE to signal “ready for data” after receiving a “call accept.”

The default value is 6.

Permitted values are: 6-60.

**WAN X21\_T5**

This is the time, in tenths of a second, that the DTE will wait for the DCE to respond “DCE ready” to a “DTE clear request.”

The default value is 2.

Permitted values are: 2-20.

**WAN X21\_T6**

This is the time, in tenths of a second, that the DTE will wait for the DCE to respond “DCE ready” to a “DTE clear confirmation.”

The default value is 2.

Permitted values are: 2-20.

If V25 calling procedures are selected, then the following timer is configurable.

**WAN V25\_callreg**

This is the time, in tenths of a second, that the DTE will wait for the DCE to enter data transfer state after a “call request” command has been issued to the DCE. This time out should only be used if the network does not support the optional “call fail” indications.

The default value is 60.

Permitted values are: 60-300.

## Changing a Stack File

Stack files associate two files to provide parameters for an interface. There are two types:

**XLB stack files**

associate X.25, LAPB and WAN template files

**XLC stack files**

associate X.25 and LLC-2 template files

There are several files of parameters available, plus any you have created. You can choose to change the stack in order to implement the values held in a different file. Use the **changestack** option of the Parameter Configuration menu to change the X.25 stack files. You can also access the Change Stack File menu from the command line:

### **x25adm changestack**

The following menu is displayed:

```
CHANGE STACK FILE
1 XLB   change X.25 to LAPB stack file
2 XLC   change X.25 to LLC2 stack file
Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

The procedure for changing the XLB and XLC files is similar, except that for XLB you have to specify the WAN file too. If, for example, you want to change the XLB stack file, select the **XLB** option. The following screen is displayed:

```
Changing an X.25 to LAPB stack file.
-----
(You may type "quit" to exit, or "?" for help at any stage)
Choose which stack you wish to edit from.

Stack files:
dcex84      def.dce80      def.dce84      def.dce88
def.dte80   def.dte84      def.dte88

From:
```

If you need to display any file before you make a selection here, return to the Protocol Parameter Configuration menu and select the **displayconfig** option.

**Note:** If you want to quit this part of the menu tree, you have to type `quit` rather than just `q`, to avoid confusion with file names.

Enter a valid stack file name, say `dcex84` in this case, and press Return. You are prompted for the name of the file you want to write to:

```
Please enter the name of the file to write to
(Filename should have between two and nine characters from
[A-Za-z0-9._]).
```

```
To:[dcex84]
```

Enter the name of a new file or press Return to confirm the default, which is the file you are editing. The available X.25, LAPB, and WAN files are displayed and you are prompted for the name of the X.25 file you want to use in this stack:

```
X.25 files:
```

```
Datapac   dcex84   def.dce80   def.dce84   def.dce88
def.dte80  def.dte84  def.dte88   def.lan     dtex84
```

```
lan84
```

```
LAPB files:
```

```
def       lapbx64
```

```
WAN files:
```

```
def
```

```
WANMAP files:
```

```
null
```

```
You are prompted for the X.25 file:
```

```
X25_File
```

```
The current value is def.dte88
```

```
Type: string
```

```
Or : '<CR>' = def.dte84
```

```
      '?' = HELP, '^' = GO UP A LEVEL, 'q' = QUIT
```

```
New value:
```

Enter the name of one of the listed X.25 files. This will be entered in your stack file. You are prompted for the name of the LAPB file that holds the parameters for the LAPB interface:

```
LAPB_File
The current value is def
Type: string
Or : '<CR>' = def
      '?' = HELP, '^' = GO UP A LEVEL, 'q' = QUIT
New value:
```

Enter the name of one of the listed LAPB files. You are prompted for the name of the WAN file that holds the parameters for the WAN interface:

```
WAN_File
The current value is def
Type: string
Or : RETURN = def
      '?' = HELP, '^' = GO UP A LEVEL, 'q' = QUIT
New value:
```

Enter the name of one of the listed WAN files. You are prompted for the name of the WANMAP file that holds the address mappings for the WAN interface:

```
WANMAP_File
The current value is null
Type: string
Or : RETURN = null
      '?' = HELP, '^' = GO UP A LEVEL, 'q' = QUIT
New value:
```

The special file name 'null' indicates no mappings are required.

If you made any changes to the stack file you are asked to confirm them:

```
Do you wish to save existing changes? [y/n]:
```

To abandon any changes you have entered, type **n**. The following message is displayed:

```
Configuration abandoned
```

```
Press the RETURN key to see the changestack menu [?, ^, q]:
```

If you choose to confirm the changes, the name of the file that has been updated is confirmed:

```
Update of dcex84 has been made.
```



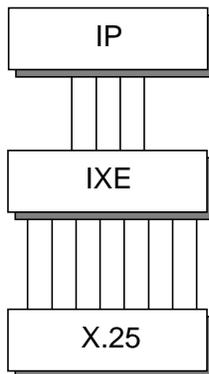
---

## Running SX.25 with TCP/IP

This chapter contains the following sections:

- “IP to X.25 Configuration”
- “IXE Kernel Configuration”
- “IXE Parameter Configuration”
- “IXE Network Configuration”
- “IXE Database Configuration”

If you plan to run SX.25 with TCP/IP, there are important additional considerations for configuring your network. The module that links IP networks with the X.25 stack is the IP to X.25 encapsulation (IXE) driver. The IXE driver sits below IP and above X.25. Figure 4-1 shows four connections from IP to IXE and eight virtual circuits from IXE to X.25.



**Figure 4-1** Module Relationships

The IXE driver lets you use one or more X.25 networks as Internet subnets. This is done by using X.25 virtual circuits to send IP datagrams from one node connected to an X.25 network to another node on the same or another network.

## IP to X.25 Configuration

The IXE driver needs information about the IP networks and their associated X.25 subnetworks. You supply this information using the *x25adm* utility. Select the **ixeconfig** option of the Kernel Configuration menu, or access **ixeconfig** directly from the command line:

```
x25adm ixeconfig
```

There are four steps in IXE configuration:

1. Kernel Configuration: to configure in-kernel STREAMS parameters and table sizes. Use the **ixeconfig** option of the Kernel Configuration menu.
2. Parameter Configuration: to set values for each parameter relevant to IP and X.25. Use the **ipconfig** option of the Network Configuration menu.
3. Network Configuration: to configure each IP network to run over X.25. Use the **ixe** option under the appropriate option of the Network Configuration menu, **changetemplate**, **deletetemplate**, or **display**.
4. Database Tables: to create an entry in the IXEmap for each IP network running over X.25. Use the **ixemap** option of the Database Tables menu.

Kernel configuration must be done before the kernel is created and the network started. Parameter configuration and network configuration must be done before starting the network. Changes to the IXEmap database can be made at any time and take effect immediately.

## IXE Kernel Configuration

The numbers you set up take effect at network initialization. IP subnets are linked above IXE, and X.25 linked below IXE, the required number of times. The default maximum number of IP subnets for IXE is 4.

The **ixconfig** option of the Kernel Configuration menu lets you set up the streams used by IXE. Select the **ixconfig** option at the Kernel Configuration menu, or directly from the command line:

```
x25adm ixconfig
```

You have to configure four parameters:

### Maximum Upper Streams - IXE\_UPPER

This is the maximum number of upper kernel Streams to be used by the IXE driver. The prompt for this parameter is:

```
IXE Kernel Parameter Configuration
-----

IXE_UPPER
The current value is 4
Type: integer {0-8}
Or : RETURN = 4
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

Configure an upper stream for each IP network running over X.25. Enter a new number, or press Return to confirm the current value. The value you set is displayed, for example:

```
IXE_UPPER remains as 4
```

or

```
IXE_UPPER is now 8
```

### Maximum Lower Streams - IXE\_LOWER

This is the maximum number of lower kernel streams to be used by the IXE driver:

```
IXE_LOWER
The current value is 8
Type: integer {0-32}
Or : RETURN = 8
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

This is the number of X.25 virtual circuits that can be used for virtual calls and permanent virtual circuits. Configure at least one lower stream for each connection to X.25, plus one for a listening stream. Enter a new number within the permitted range or press Return to confirm the current value.

### Maximum Address Table Entries - IXE\_ADDRSIZE

This is the maximum number of entries in the IXE Address map table.

```
The default value is 100
The range of values is 20-1000
```

Enter a number within the permitted range, or press Return to confirm the current value. The value you set is displayed.

### Maximum IXE Hash Table Entries- IXE\_HASHSIZE

You are now prompted to enter the maximum number of entries for the IXE hash table. The range of values is 41-3001.

Enter a number that is at least twice the value of IXE\_ADDRSIZE and preferably a prime number. You can press Return to confirm the current value. The value you set is displayed.

When kernel configuration is complete, access the **kernelcreate** option of the Kernel Configuration menu to rebuild the kernel according to the changes you have made.

## IXE Parameter Configuration

The IXE parameters are held in template files that you can use to create files with appropriate parameter values. Under IXE, there are two default template files according to whether you want to run IP over X.25 on the WAN or on the LAN:

*def.lan def.wan*

For a full explanation of the use of template files, refer to “X.25 Configuration Files” in Chapter 3.

The **paramconfig** option of the Network Configuration menu lets you set the configurable parameters for the IP links over X.25. To access the Protocol Parameter Configuration menu, select the **paramconfig** option directly from the command line:

```
x25adm paramconfig
```

The following menu is displayed:

```
PROTOCOL PARAMETER CONFIGURATION
```

```
1 changecurrent  change running configuration
2 changestack    change stack file
3 changetemplate change template files
4 deletestack    delete stack file
5 deletetemplate delete template files
6 display        display configuration
```

```
Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, q to QUIT:
```

The **display** option shows existing parameter settings. To modify or set up IXE parameters, select **changetemplate**. From the **changetemplate** menu, select **ixe**. Configure the following parameters for each network linked below, and registered with, IP. These control virtual circuit management.

**Note:** IP does not guarantee reliable delivery of datagrams, and has no flow control mechanisms. When there is a need to restrict the flow of IP packets across an X.25 connection, packets that cannot be forwarded are simply discarded.

### **Maximum NSDU Length - NSDU\_HWM**

SX.25 tries to concatenate data packets marked with the “More Data” mark (M-bit) into a single network service data unit (NSDU). You can override this by specifying a maximum length. When this maximum is exceeded, concatenation stops and the data currently held is passed to the Network Service user (with the More parameter set). For IP to send or receive packets of over 576 octets, which is the default, increase this value. The prompt is as follows:

```
NSDU_HWM
The current value is 576
Type: integer {128-4096}
Or : RETURN = 576
      ? = HELP, ^ = GO UP A LEVEL, q = QUIT
New value:
```

### **Disconnection Timer - IDLE\_TICKS**

The time, in tenths of a second, a connection on this network can remain idle before being closed. A timer is configurable for each X.25 network. When this timer expires, if there is no data on a connection, the virtual circuit to that network is closed. If data arrives from IP after this, the circuit is reopened. The default is 30.

### **Pre-emption Timer - CONN\_TICKS**

If all X.25 circuits available to IXE are in use, and an IP packet arrives for transmission to a new destination, then one of the existing connections is closed. This is done according to which has been idle for longest. You can configure the minimum time that a connection must stay open before it can be preempted in this way. The default value is 20.

### **X.25 Mode - X25\_VERSION**

This specifies whether the network operates in X.25(80), X.25(84), or X.25(88) mode.

**CCITT/DDN**

This specifies whether the network uses CCITT or DDN X.25. This is the way in which IP addresses are mapped onto X.25 addresses, as described in “Address Mapping” later in this chapter.

CCITT/DDN

The current value is 0

Type: integer {0, 1}

Or : RETURN = 0

? = HELP, ^ = GO UP A LEVEL, q = QUIT

New value: Enter 0 for CCITT or 1 for DDN.

**Exception Timers - RESET\_ACTION, EXP\_ACTION**

IXE does not use the X.25 Expedited Data or Reset features. You can configure how IXE responds if either is ever received on a connection. The possible actions are:

ACK	acknowledge Reset or Expedited Data but discard it
DISC	disconnect connection in response to Reset or Expedited Data
RESET	reset connection in response to Expedited Data (no reset)

**Logical Channel Range - LCI\_RANGE**

This specifies a range of Logical Channel Identifiers (LCIs) for the PVCs to be used by IXE. Specify the range as:

Lower PVC Channel (LPC) - Higher PVC Channel (HPC)

LPC and HPC comprise three hexadecimal digits and must be separated by a dash (“-”). A dash on its own configures no channels. “009-00b” for example, gives three IXE PVC channels.

### Listen for X.25 info. - LISTEN\_INFO

This sets the application to listen for incoming calls with the specified CUD fields and called addresses. There are two possible formats:

```
{ < CUD field > , < called addresses field >; }
```

or

```
{ < [ (I|S) [bytes] ] | X > , < [ (I|S) (D|N) [semi-octets] ] | X >; }
```

where:

I means “match this field identically”

S means “match this field starting with”

X means “ignore this field”

D means “this field refers to an X.25 DTE”

N means “this field refers to an X.25 NSAP”

The prompt is:

```
LISTEN_INFO
```

```
The current value is ICC,SD2342;
```

```
Type: string
```

```
Or : RETURN = ICC,SD2342;
```

```
    ? = HELP, ^ = GO UP A LEVEL, q = QUIT
```

```
New value:
```

This is the last parameter. You are asked to confirm the changes you have made:

```
Do you wish to save existing changes? [y/n]:
```

## IXE Network Configuration

The **ipconfig** option of the Network Daemon Configuration menu lets you initialize the IP-IXE subnets and the IXE-X.25 connections. To access the Network Daemon Configuration menu, select the **netconfig** option of the Network Management menu, or enter the option directly at the command line:

```
x25adm netconfig
```

The following menu is displayed:

```
NETWORK DAEMON CONFIGURATION
```

```
1 ipconfig      IP network configuration
2 netcreate     netconf creation
3 x25config     X25 network configuration
```

```
Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

Select the **ipconfig** option. The following message is displayed:

```
CONFIGURATION OF LINES BELOW IP
```

```
-----
```

The following commands may be used throughout:

Command	Action
-----	-----
?	Give specific help information
??	Give this general help information
RETURN	Use current value (in square brackets)
q	Quit

```
Please wait..... checking configuration files
```

To access the IXE driver configuration, enter **n** in response to each of these prompts. A message similar to the following is displayed:

```
IP-to-X25 (IXE) driver
-----
No  IP Address  IP Mask      Config File
--  -
0   14.0.7.1    255.255.0.0  def.wan
X25_address: A.1234567890123
1   14.1.7.1    255.255.0.0  ppl.wan
X25_address: B.1234567890124
2   14.2.7.1    255.255.0.0  def.lan
X25_address: C.0123456789AB7e
3   not set     not set      not set
X25 Address: not set

Do you want to (re)configure an IP network
through the IP-to-X25 (IXE) driver
(there are 4 such networks, 1 yet to be configured)?
```

Enter **y** to configure an IP network over X.25. The following message is displayed:

```
Enter the number of the network (0-3) over the IP-to-X25
(IXE) driver you want to configure (or '=' to list current
configurations):
```

Enter the number of the network you want to configure, for example, 3. You can add a new network or change an existing one. The current value for each parameter is the default; press Return at each prompt to retain the current value.

**Note:** The maximum number of networks you may configure is itself configurable.

There are several parameters to configure:

### **Internet Address**

The address you enter for a new network should not duplicate any address that has already been used. The Internet addresses that you have already assigned to networks are displayed with the prompt:

Enter the Internet address you wish to give this network (or '-' to clear whole entry).

IP network addresses in use:

```
89.0.7.1
14.0.7.1
14.1.7.1
14.2.7.1
```

Internet address [89.0.7.1] :

Enter the Internet address for the network in dotted notation.

### **Subnet Mask**

This is the subnet mask for this network. The mask indicates the network part of the Internet address. For example, the mask 255.254.0.0 shows that the network part is 15 bits and the host part is the remaining 17 bits. The default is 255.0.0.0:

Enter the IP mask you wish to give this network [255.0.0.0] :

Enter in dotted notation the subnet mask you want to use for this network.

### **X.25 address**

The default displayed is the current X.25 address of this network:

Enter the full X25 address of this network (the format is 'snid' .['LSAP' | 'DTE'] .[N|X] .['NSAP' | 'EXT'])[C.0123456789AB7e]

### Template file

This is the name of the template file that associates the IP and X.25 parameter files holding the parameter settings you want to use for this network. Refer to "X.25 Configuration Files" in Chapter 3 for more information on creating these files. The following prompt is displayed:

```
Enter the name of the IP-to-X25 (IXE) driver configuration
file you want this network to use.
```

```
Files Available
-----
def.lan      def.wan ppl.wan    test.wan
File [def.lan]
```

Enter the name of the template file you wish to use, or press Return to retain the default. This is the last parameter. The following message is displayed:

```
You have created the following entry:
```

```
IP-to-X25 (IXE) driver
-----
No  IP Address  IP Mask      Config File
--  -
1   14.1.7.1    255.255.0.0  def.lan
X25 Address: C.0123456789AB7e
```

```
Are you sure you want to install these values?
```

If the values are correct, confirm the configuration by entering **y**. The values are reconfirmed and you are prompted to configure another IP network over X.25:

```
The current configurations are:
```

```
IP-to-X25 (IXE) driver
-----
No  IP Address  IP Mask      Config File
--  -
0   14.0.7.1    255.255.0.0  def.wan
X25 Address: A.23423130014265
1   14.1.7.1    255.255.0.0  pvc.wan
X25 Address: B.23423130014266
2   14.2.7.1    255.255.0.0  def.lan
X25 Address: C.02cf1f7006537e
3   not set     not set      not set
```

X25 Address: not set

Do you want to (re)configure an IP network through the IP-to-X25 (IXE) driver (there are 4 such networks, 1 yet to be configured)?

Enter **n** to end IXE configuration. The following message is displayed:

You have finished configuring IP.

Press the RETURN key to see the netconfig menu [?, ^, q]:

## IXE Database Configuration

Hosts linked to the IXE driver have two associated addresses: an Internet address and an X.25 address. To configure a line, you supply these addresses, then set values for packet and window sizes and the number of virtual circuits. This information is used to create a dynamic table in the IXE driver.

This section gives background information on the address mapping done by the IXE driver. Then the process of adding, altering, or deleting an IXEmap entry is described.

### Address Mapping

The address mapping depends on the mode of operation, CCITT or DDN.

#### CCITT Mode

An internal address table, the IXEmap Database, holds the address pairs that you set up between the IP and X.25 addresses of specified nodes. The default maximum number of entries in this database is 100.

The X.25 addresses are X.121 DTE strings, with optional NSAPs or address extensions, depending on the mode of X.25 operation (80/84/88). Refer to "X.25 Addresses" in Chapter 1.

An Internet address has 32 bits. It can be thought of as containing four fields, each ranging from 0 to (decimal) 255. The four fields are sometimes

represented with “.” field delimiters. An Internet Address can be represented as decimal, octal, or hexadecimal:

- Decimal addresses have no prefixes and may not begin with zero. An example is “83.0.0.127”.
- Octal addresses begin with a zero. For example, “83.0.0.127” in decimal is “0123.0.0.177” in octal.
- Hexadecimal addresses are preceded by the 0x or 0X prefix. For example, “83.0.0.127” in decimal is “0x53.0x0x0x7F”.

The IXE database is consulted every time an outgoing IP packet requires a new connection, and also for incoming X.25 connections carrying IP packets as described in “Adding an IXEmap Database Entry” later in this chapter. Attempts to send IP packets to X.25 nodes for which there is no entry in the database result in a “Destination Unreachable” message being returned to the IP driver.

When an incoming connection arrives at the X.25 layer, destined for the IXE driver, the calling X.25 address is checked to see if it is in the address table. If not, the connection is refused. In general, if an incoming connection has been established, it is used for outgoing datagrams to the source of that connection.

For each name and address pair, you can specify parameters for each destination. These are the packet and window sizes to be requested on connections, and the number of virtual circuits allowed to be open concurrently.

**Note:** Sending IP broadcast packets over an X.25 network is impractical and is not supported.

#### **DDN Mode**

When operating in DDN mode, IP to X.25 address mappings are not normally obtained from the internal address table. This means that in normal operation, only the default X.25 facilities are available. It is possible, however, to use different facilities, packet and window sizes, and so on. To do this, you have to enter the information into the internal address table yourself. This means entering the correct X.25 address, derived from the

DDN address; refer to the *Defense Data Network X.25 Host Interface Specification*.

**Note:** Addresses you enter in the table in this way are not checked for correctness.

### Making Changes

The **ixemap** option of the Database Table menu lets you create an entry in the IXEmap database for each subnet or line running X.25 under IP. The changes you make to the IXEmap Database using the *x25adm* utilities take effect straight away. To access the Database Tables menu, select the **tables** option of the Network Management menu. Alternatively, you can access the IXEmap Database menu directly from the command line:

```
adm ixemap
```

A message is displayed, followed by the IXEmap Database menu:

```
Running subcommand 'ixemap' from menu 'tables',
DATABASE TABLES

IXEMAP DATABASE MENU

1 addentry      add ixemap entry
2 alterentry    alter ixemap entry
3 delentry      delete ixemap entry
4 listentries   list ixemap entries

Enter a number, a name, the initial part of a name, or ? or
<number>? for HELP, ^ to GO BACK, q to QUIT:
```

These options are described below.

### Adding an IXEmap Database Entry

To configure SX.25 for the first time, select option 1, **addentry**, or an abbreviation, for example, **ad**. You are prompted for the name or address of the line you want to configure:

```
ADD IXEMAP ENTRY

Enter Internet_addr
[Internet Name or Address, r, \x, ?, ^, q]:
```

Enter one of the following, then press Return:

- the Internet address in dotted notation as a 32 bit number, for example 89.0.1.3
- a hostname that has an entry in the */etc/hosts* file

If you enter a hostname that does not exist in the */etc/hosts* file, this is discovered when you exit the IXEMap Configuration menu and attempt to confirm the changes you have made. Remove or alter the entry by using the IXEMap Configuration menu.

The address is checked to see if it is in a valid format. If it is, you are prompted for the X.25 address:

```
Enter X25_address  
[X25 Address, r, \x, ?, ^, q]:
```

Enter the X.25 address of the line and press Return. Your entry is checked to see if it is a valid address; refer to “X.25 Addresses” in Chapter 1. When you have set the addresses, you can set a number of other parameters:

### Packet Sizes

In accordance with RFC 877, when IP datagrams are longer than the X.25 packet size, the X.25 M-bit is used to send the datagrams as complete packet sequences. IP fragmentation takes place only if a packet exceeds the maximum subnet fragment size configured for that network (default 576 octets). The maximum packet length supported by IXE is 4096 octets.

You are prompted for the packet size:

```
Enter Packet_size  
[integer(16,32,64,128,256,512,1024,2048,4096), l, r, w, \x,  
?, ^, q]:
```

Enter one of the decimal values shown and press Return. This packet size applies to calls in both directions and should not be bigger than the configured maximum for packet sizes.

## Window Sizes

The window size is for calls in both directions and should not be bigger than the configured maximum for window sizes. The following prompt is displayed:

```
Enter Window_size
[integer(1- 127), l, r, w, \x, ?, ^, q]:
```

Enter the window size as a decimal within the range shown and press Return.

## Virtual Circuits

This is the maximum number of virtual circuits for this line. The only permitted value for this release is **1**, because only one virtual circuit is supported per physical link under IP. The parameter is configurable to provide for future enhancements.

```
Enter Num_VCs
[integer(1), l, r, w, \x, ?, ^, q]:
```

This is the last parameter. When you enter a valid number, the complete set of information for this link is displayed for you to confirm, as shown in the following example:

```
Internet_addr = 89.0.0.5
X25_address = A.1234567801234.X.12789
Packet_size = 128
Window_size = 5
Num_VCs = 1
Ok? [y/n]:
```

To edit the details for this link, enter **n**. To confirm your entry press **y**; the following message is displayed, and you are prompted to enter details for the next link to set up:

```
Added ixemap entry
Enter Internet_addr
[Internet Address, r, \x, ?, ^, q]:
```

## Altering an IXEmap Entry

To change the details associated with a link, select the **alterentry** option at the IXEmap Database menu, **2**, or an abbreviation, for example, **al**. The following prompt is displayed:

```
ALTER IXEMAP
Enter Internet_addr to be altered
[Internet Name or Address, ?, ^, q]:
```

Enter one of the following:

- the Internet address, in dotted notation, of the link entry that you wish to alter
- the name entered in the */etc/hosts* file, or an abbreviation of at least two characters

The IXEmap database is searched to find a match for the address you entered. If no match is found, a message similar to the following is displayed and you are prompted to re-enter the Internet address:

```
Couldn't find a match for 14.0.0.7
Enter Internet_addr to be altered
[Internet Name or Address, ?, ^, q]:
```

If a match is found, the details for that entry in the database are displayed. If you do not want to alter a line, just press Return. If you change an entry, enter **1** to leave the rest of the details for that link unchanged. When you finish altering link entries, press **q** to return to the IXEmap Database menu.

## Deleting an IXEmap Entry

To delete the details held about a link under IP, select the **delentry** option from the IXEmap Database menu, **3**, or an abbreviation, for example, **ad**. The following prompt is displayed:

```
DELETE IXEMAP ENTRY
Enter Internet_addr to be deleted
[Internet Name or Address, ?, ^, q]:
```

To delete details of a link, enter the Internet address or hostname. Use dotted notation, for example: 89.0.0.3. The database is searched for this link. If no match is found, a message similar to the following is displayed, prompting you to re-enter an Internet address:

```
Couldn't find a match for "89.0.0.3"
Enter Internet_addr to be deleted
[Internet Name or Address, ?, ^, q]:
```

If a match for the Internet address is found, the details for the link are displayed:

```
Internet_addr = 89.0.0.5
X25_address = A.1234567801234.X.12789
Packet_size = 128
Window_size = 5
Num_VCs = 1
Delete? [y/n]:
```

Enter **y** to delete this entry. The following message is displayed and you are prompted to enter the Internet address of another link to be deleted:

```
Deleted ixemap entry
Enter Internet_addr to be deleted
[Internet Name or Address, ?, ^, q]:
```

If you enter **n** to stop the entry being deleted, you are prompted for the Internet address of the link you want to delete.

When you have finished deleting details of links, enter **^** to return to the IXEmap Database menu.

### Listing IXEmap Database Entries

To list the entries in the IXEmap Database, select the **listentries** option, **4**, or an abbreviation, for example, **li**. The IXEmap database entries are displayed, for example:

```
LIST IXEMAP

Internet_addr = 14.0.7.1
X25_address = A.1234567890123
Packet_size = 256
Window_size = 7
Num_VCs = 1

Internet_addr = 14.1.7.1
X25_address = C.0123456789AB7E
Packet_size = 1024
Window_size = 7
Num_VCs = 1

Internet_addr = 14.0.7.2
X25_address = A.1234567890124
Packet_size = 256
Window_size = 7
Num_VCs = 1
<CR> for more [^,q]:
```

**Note:** The display pauses after each screenful of IXEmap entries. Press Return to display the next screen. When you finish looking at the IXEmap entries, press ^ to return to the IXEmap Database menu, or q to exit *x25adm*.

---

## UUCP Configuration

This chapter contains these sections:

- “Files Associated with UUCP”
- “Files Associated with UUCP”

SX.25 provides a mechanism for using the standard UNIX *uucp*, *cu*, and associated commands over the *pad* program by using the *xyt* device driver. The *xyt* device behaves like a *tty* device that supports both blocking and non-blocking opens of the same logical device. It can support *gettys* for incoming calls that make a blocking open on the *xyt* device.

### Configuring the XTY System

SX.25 supports a system, called the XYT\_OUT system, which uses the *xyt* device driver only for outgoing *uucp*/*cu* calls. This type of system does not need *getty* processes. Incoming calls are handled directly by the *pad* daemon that executes the login process.

The XYT\_OUT system is configured by a parameter XYT\_OUT, which specifies the maximum number of simultaneous outgoing *uucp*-type calls that can be supported.

To configure this parameter, enter:

```
x25adm kernelconfig/xytconfig
```

For more information on using *x25adm*, see “XTY Kernel Parameters” in Chapter 2.

After saving any changes, enter:

```
x25adm kernelcreate
```

This brings the changes into effect and rebuild the kernel. It creates the appropriate devices for outgoing *xy* calls. You have to reboot after running *kernelcreate*.

## Files Associated with UUCP

A number of files need modification before *uucp/cu* type calls can be made over the *xy* driver. These files and their uses are described below. All these files can be found in the directory */usr/lib/uucp*. For more details, refer to the *IRIS Advanced Site and Server Administration Guide*.

<i>Devices</i>	Specifies which <i>xy</i> devices are available for outgoing calls.
<i>Dialcodes</i>	Specifies a number of host-to-SX.25 address mappings.
<i>Permissions</i>	Specifies permissions that remote users have when sending files, whether local users can request files, and which commands users are allowed to use. You should read the section in the <i>IRIS Advanced Site and Server Administration Guide</i> that shows you how to configure this file. System security can be compromised if it is incorrectly configured.
<i>Dialers</i>	Specifies the command that <i>uucp</i> issues to the <i>pad</i> program to make a <i>uucp</i> call.
<i>Systems</i>	Specifies the hosts that <i>uucp</i> knows about. It includes login names and character sequences that <i>uucp</i> is to send and expects to receive as responses.

These files would have the format presented below for a typical *uucp-over-X.25* configuration.

## Devices

There will typically be one line per outgoing *xy* device. That is, the number of XTY\_OUT plus XTY\_BOTHWAYS (if applicable) configured earlier. The *xy* device numbers should match those generated by *kernelcreate*. Lines like the following should be added to the *Devices* file:

```
Pad xty80 - 9600 pad
Pad xty81 - 9600 pad
Pad xty82 - 9600 pad
Pad xty83 - 9600 pad
```

## Dialcodes

The new entries in the *Dialcodes* file specify a name to X.25 address mapping. This name is used in the *Permissions* and *Systems* files. Lines like the following should be added to the *Dialcodes* file:

```
hostname1 A.1234567891
hostname2 A.1234567892
```

## Permissions

Lines like the following should be added to the *Permissions* file:

```
MACHINE=hostname1:hostname2 \
REQUEST=yes \
SENDFILES=yes \
READ=/ WRITE=/ \
COMMANDS=ALL
LOGNAME=nuucp \
REQUEST=yes \
SENDFILES=yes \
READ=/ WRITE=/ \
COMMANDS=ALL
```

**Note:** These permissions let users do almost anything, so you may need to be more restrictive. For more detail, refer to the *IRIS Advanced Site and Server Administration Guide*.

## Dialers

The added entry in this file tells *uucp* the command to issue to the *pad* program to let outgoing calls be made. Add the following line to the *Dialers* file:

```
pad " " " CALL/s/T
```

## Systems

Add lines like the following to the *Systems* file:

```
hostname1 Any Pad 9600 hostname1 " " \\r\\d\\r in:--in: nuucp  
hostname2 Any Pad 9600 hostname2 " " \\r\\d\\r in:--in: nuucp
```

For more detail on this file, refer to the *IRIS Advanced Site and Server Administration Guide*.

---

## Troubleshooting

This chapter contains the following sections:

- “STREAMS Logging Facilities”
- “Error Messages”

### STREAMS Logging Facilities

STREAMS provides its own mechanism for logging trace and error information. You can capture and print this information using the two IRIX utilities described here, **strace** and **strerr**.

### STREAMS Trace Messages

STREAMS trace messages can be printed out by using **strace(1M)**. Without arguments, it writes all STREAMS event trace messages from all drivers and modules to its standard output, or a subset of these when arguments are given. Arguments define what the subset should be and they must be in triplets of the form *mid, sid, level*, where

- **mid** is the STREAMS module/driver ID number for the module or driver submitting the log message.
- **sid** is an internal sub\_id number.
- **level** is a tracing level that allows for selective filtering out of low priority messages from the tracer.

Each triplet indicates that tracing messages are to be received from the given module/driver, sub\_id, and priority level equal to or less than the given level. The token *all* may be used for any member to indicate no restriction for that attribute.

The following modules and drivers within SX.25 currently generate STREAMS trace messages.

**X.25 PLP Driver**

**Module Id**     200  
**Sid**            Set to identifier of subnetwork packet tx/rx on  
**Levels**        1 - 6

Six levels of tracing are available.

**Trace Level 1 - Call Activity**

Monitors incoming calls, outgoing calls, calls accepted, and calls connected. For each event, a description, the subnetwork ID, LCI, and GFI are logged.

```
CALL In  : '65' lcn 400 gfi 5  
CALL Out : '66' lcn 401 gfi 1  
CAA Out  : '65' lcn 400 gfi 1  
CAA In   : '66' lcn 401 gfi 1
```

**Trace Level 2 - Clear Activity**

Monitors clear requests, clear indications, and clear confirmations. For each event, a description, the subnetwork ID, and the LCI are logged. For clear requests and clear indications, the cause and diagnostic are also logged.

```
CLR In   : '65' lcn 400 C/D 0043  
CLR Out  : '66' lcn 401 C/D 80F2  
CLC In   : '66' lcn 401  
CLC Out  : '65' lcn 400
```

**Trace Level 3 - Reset Activity**

Monitors reset indications, reset requests, and reset confirmations. For each event, a description, the subnetwork ID, and the LCI are logged. For reset requests and reset indications, the cause and diagnostic are also logged.

```
RST In   : '65' lcn 400 C/D 0003  
RST Out  : '66' lcn 401 C/D 00FA  
RSC In   : '66' lcn 401  
RSC Out  : '67' lcn 400
```

**Trace Level 4 - Restart Activity**

Monitors restart indications, restart requests, and restart confirmations. For each event, a description and the subnetwork ID are logged. For restart requests and restart indications, the cause and diagnostic are also logged.

```
REST In   : '65' C/D 1300
REST Out  : '66' C/D 0000
RESC In   : '66'
RESC Out  : '65'
```

**Trace Level 5 - Interrupt Activity**

Monitors interrupt data indications, interrupt data requests, and interrupt confirmations. For each event, a description, the subnetwork ID, and the LCI are logged.

```
INT In    : '65' lcn 400
INT Out   : '66' lcn 401
INTC In   : '66' lcn 401
INTC Out  : '65' lcn 400
```

**Trace Level 6 - Data Activity**

Monitors data indications and data requests. For each event, a description, the subnetwork ID, the LCI, the send sequence number, and the receive sequence number are logged.

```
DATA In   : '65' lcn 400 ns/nr 0702
DATA Out  : '66' lcn 401 ns/nr 0500
```

**Example Trace Logging Output**

```
# strace 200 all all
000187 16:07:30 0001c15b 4 ... 200 67 REST In   : '67' C/D 0000
000188 16:07:30 0001c15b 4 ... 200 67 RESC Out  : '67'
000189 16:08:01 0001cf60 1 ... 200 67 CALL Out  : '67' lci 001 gfi 1
000190 16:08:03 0001cf62 1 ... 200 67 CLR In   : '67' lci 001 C/D 0900
000199 16:08:11 0001cfa8 6 ... 200 65 DATA In  : '65' lci 43f ns/nr 0000
000200 16:08:11 0001cfb0 6 ... 200 65 DATA Out : '65' lci 43f ns/nr 0001
000201 16:08:12 0001d5c8 3 ... 200 65 RST Out   : '65' lci 43f C/D 00fa
000202 16:08:12 0001d5cb 3 ... 200 65 RSC In    : '65' lci 43f
```

Notes:

1. Tracing of an incoming event does not imply that the packet has been accepted by the X.25 driver. This is because tracing of incoming events is triggered on receiving data from the link level. At this point, the packet has still to be verified and, if in error, may be discarded or cause some further protocol action.
2. Similarly, tracing of an outgoing event does not imply that the packet has been sent to the link level driver. This is because tracing of outgoing events is triggered on requests to send data to the link level. At this point, various consistency checks are performed on the required link-level database, and, if they fail, the packet will be discarded.

**LAPB Driver**

**Module Id**     201  
**Sid**             Set to identifier of subnetwork packet tx/rx on  
**Levels**         1 - 4

Four levels of tracing are available.

**Trace Level 1 - Set Up and Disconnection Activity**

Monitors all link set up and disconnection events. For each event, a description and the subnetwork ID are logged. For disconnection events, an indication of whether locally or remotely initiated is also logged.

```
LINK Up : '65'  
LINK Dwn : '66' Remote
```

**Trace Level 2 - Reset Activity**

Monitors all link reset events. For each event, a description, the subnetwork ID, and an indication of whether locally or remotely initiated are logged.

```
LINK Rst : '66' Local  
LINK Rst : '65' Remote
```

**Trace Level 3 - Error Activity**

Monitors sending and receiving of FRMR and REJ frames. For each event, a description and the subnetwork ID are logged. For FRMR frames, the error flag is also logged. For REJ frames, the send sequence number in error is also logged.

```
FRMR In : '65' Err 01
FRMR Out : '66' Err 08
REJ In : '66' N(007)
REJ Out : '68' N(006)
```

**Trace Level 4 - Busy Activity**

Monitors all link busy conditions. For each event, a description, the subnetwork ID, and an indication of whether locally or remotely initiated events are logged.

```
LINK Bsy : '65' Local
LINK Bsy : '66' Remote
```

**Example Trace Logging Output**

```
# strace 201 all all
000187 16:07:30 0001c15b 1 ... 201 65 LINK Up : '65'
000188 16:07:30 0001c15b 1 ... 201 65 LINK Dwn : '65' Local
000189 16:08:01 0001cf60 3 ... 201 65 FRMR In : '65' Err 01
000190 16:08:03 0001cf62 2 ... 201 65 LINK Rst : '65' Remote
000197 16:08:06 0001cf6c 3 ... 201 65 REJ Out : '65' N(006)
000198 16:08:06 0001cfa7 4 ... 201 65 LINK Bsy : '65' Local
```

**Notes:**

1. Tracing of an incoming event does not imply that the frame has been accepted by the LAPB driver. This is because tracing of incoming events is triggered on receiving data from the WAN. At this point, the frame has still to be verified and, if in error, may be discarded or cause some further protocol action.
2. Tracing of an outgoing event implies that the frame has been sent to the WAN driver. It does not imply that the frame will be transmitted on the line.

**LLC2 Driver**

**Module Id** 202  
**Sid** Set to identifier of subnetwork packet tx/rx on  
**Levels** 1 - 5

Five levels of tracing are available.

**Trace Level 1 - Set Up and Disconnection Activity**

Monitors all link setup and disconnection events. For each event, a description, the subnetwork ID, and the connection identifier are logged. For disconnection events, an indication of whether locally or remotely initiated is also logged.

```
LINK Up : 'C' Id 001  
LINK Dwn : 'C' Id 002 Remote
```

**Trace Level 2 - Reset Activity**

Monitors all link reset events. For each event, a description, the subnetwork ID, the connection identifier, and an indication of whether locally or remotely initiated are logged.

```
LINK Rst : '68' Id 001 Local  
LINK Rst : '67' Id 021 Remote
```

**Trace Level 3 - Error Activity**

Monitors sending of FRMR and REJ frames. For each event, a description, the subnetwork ID, the connection identifier, and an indication of whether locally or remotely initiated are logged. For FRMR frames, the error flag is also logged. For REJ frames, the send sequence number in error is also logged.

```
FRMR In : '67' Id 010 Err 01  
FRMR Out : '68' Id 009 Err 08  
REJ In : '67' Id 010 N(007)  
REJ Out : '67' Id 023 N(006)
```

**Trace Level 4 - Busy Activity**

Monitors all link busy conditions. For each event, a description, the subnetwork ID, the connection identifier, and an indication of whether locally or remotely initiated are logged.

```
LINK Bsy : '67' Id 012 Local
LINK Bsy : '67' Id 013 Remote
```

**Trace Level 5 - Type 1 Activity**

The following type 1 events are monitored: sending and receiving of XID and TEST frames. For each event, a description, the subnetwork ID, and the connection identifier are logged.

```
TEST Out : '67' Id 000
XID In : '67' Id 013
XID Out : '67' Id 013
```

**Example Trace Logging Output**

```
# strace 202 67 all
000186 16:07:27 0001c150 5 ... 202 67 XID Out : '67' Id 001
000187 16:07:30 0001c152 1 ... 202 67 LINK Up : '67' Id 001
000188 16:07:30 0001c15b 1 ... 202 67 LINK Dwn : '67' Id 010 Local
000189 16:08:01 0001cf60 3 ... 202 67 FRMR In : '67' Id 038 Err 01
000190 16:08:03 0001cf62 2 ... 202 67 LINK Rst : '67' Id 231 Remote
000197 16:08:06 0001cf6c 3 ... 202 67 REJ Out : '67' Id 128 N(006)
000198 16:08:06 0001cfa7 4 ... 202 67 LINK Bsy : '67' Id 007 Local
```

**Notes:**

1. Tracing of an incoming event does not imply that the frame has been accepted by the LLC2 driver. This is because tracing of incoming events is triggered on receiving data from the Ethernet level. At this point, the frame has still to be verified and, if in error, may be discarded or cause some further protocol action.
2. Tracing of an outgoing event implies that the frame has been sent to the Ethernet driver. It does not imply that the frame will be transmitted on the Ethernet.

### Tracing Summary

Table 6-1 shows the available tracing levels for the three types of drivers:

**Table 6-1** Tracing Levels

Level	X.25 Driver ID 200	LAPB Driver ID 201	LLC2 Driver ID 202
1	Call Setup	Link Up/Down	Link Up/Down
2	Call Clearing	Link Reset	Link Reset
3	Call Reset	Error Activity	Error Activity
4	Restart Activity	Link Busy	Link Busy
5	Interrupt Packets	--	Type 1 Activity
6	Data Packets	--	--

### STREAMS Error Messages

STREAMS error messages can be captured using **strerr(1M)**. This receives error messages and appends them to an error log file. This utility is normally run when the system goes multi-user.

The following types of errors are logged:

**System errors**

Errors in the internal workings of the module/driver.

**Reference errors**

Errors in the references used between upper and lower modules/drivers

**Interface errors**

Errors in messages passed over the interfaces.

**Allocation errors**

Errors in the allocation of memory for database elements.

The X25 PLP driver, the LAPB driver, and the LLC2 driver of SX.25 currently generate STREAMS error log messages.

## Error Messages

### Network Board-Loading (nload) Error Messages

Error *<reason>* in GETDET command

The specified board could not be opened for loading, for the reason stated in *<reason>*. For example, a board has been configured but is not present. Reconfigure the boards using *x25adm ipconfig* and *x25adm x25config*. Refer to “Network Configuration” in Chapter 2.

Error *<reason>* in EXECUTE command - line *<lineno>* of */var/opt/snet/brdconf*

The program downloaded to the board could not be executed, for the reason stated in *<reason>*. The board can be identified by referring to line number *<lineno>* in */etc/brdconf*. This error may be due to a board crash. Reboot the machine.

nload: Invalid address for X.25 (WAN) board on line *<lineno>* of */var/opt/snet/brdconf*

An address has been configured for use on an X.25 (WAN) board. This field should be null. Reconfigure the board using *x25adm x25config*.

### Network Daemon (snetd) Error Messages

attempt to multiplex module *<module>*

bad format

The */etc/config/snetd.options* file is not in the expected format. Configure the network parameters using *x25adm*.

ECHO\_TYPE ioctl failed:*<reason>*

The Echo/Ethernet registration failed. This may be due to an incorrectly configured Ethernet card.

Reconfigure the Ethernet card. Refer to “Network Configuration” in Chapter 2.

failed to open *<file>*:*<reason>*

The configuration file *<file>* (default:*/etc/config/snetd.options*) could not be opened, for the reason stated in *<reason>*. The configuration of the Internet address and other network parameters has not been successfully completed.

Configure the network parameters using *x25adm*. Refer to “Network Configuration” in Chapter 2.

failed to open driver *<driver>*:*<reason>*

The lines referring to the driver *<driver>* in the */etc/config/snetd.options* file are wrong. The operation failed for the reason stated in *<reason>*. Configure the network parameters using *x25adm*. Refer to “Network Configuration” in Chapter 2.

failed to open Ethernet Driver:*<reason>*

The Ethernet driver could not be opened, for the reason stated.

failed to open stream *<stream>*:*<reason>*

The lines referring to the stream *<stream>* in the */etc/config/snetd.options* file are wrong. The operation failed for the reason stated in *<reason>*. Configure the network parameters using *x25adm*. Refer to “Network Configuration” in Chapter 2.

failed to link stream *<stream>* under driver *<driver>*:*<reason>*

The lines referring to the stream *<stream>* in the */etc/config/snetd.options* file are wrong. The operation failed for the reason stated in *<reason>*. Configure the network parameters using *x25adm*. Refer to “Network Configuration” in Chapter 2.

failed to push module *<module>*:*<reason>*

The lines referring to the module *<module>* in the */etc/config/snetd.options* file are wrong. The operation failed for the reason stated in *<reason>*. Configure the network parameters using *x25adm*. Refer to “Network Configuration” in Chapter 2.

I\_PUSH (*<module>*):*<reason>*

The kernel module *<module>* could not be pushed onto a stream, for the reason stated in *<reason>*.

IP\_NET\_ADDR ioctl failed

The Internet address, subnet mask, and other parameters on the IP\_NET line in the */etc/config/snetd.options* file have not been correctly configured. This error may also be given if more IP-IXE lines have been configured in the */etc/config/snetd.options* file than are configured in the IXE kernel. This error can only occur if you also have SGI TCP in your network.

Reconfigure the network parameters for this IP network. Refer to “Network Configuration” in Chapter 2.

IXE\_SNREG ioctl failed

This is due to IXE being incorrectly configured. For example, an IP address referring to an IP-IXE link in the */etc/config/snetd.options* does not match the address in the */etc/ixetuneconf* file.

Reconfigure IXE correctly using *x25adm ipconfig*.

must specify 'd' or 'm' flag

One of the flags in a line at the top of the */etc/config/snetd.options* file declaring the name of a driver, cloneable driver, or a module is not valid.

no driver below module <module>

unknown module <module>

The lines referring to the module <module> in the */etc/config/snetd.options* file are wrong.

Configure the network parameters using *x25adm*. Refer to “Network Configuration” in Chapter 2.

unexpected delimiter

The */etc/config/snetd.options* file is not in the expected format.

Configure the network parameters using *x25adm*.

non numeric instantiation

The explicit association of multiple instances of a module in the */etc/config/snetd.options* file has a non-numeric label.

out of memory

There is insufficient memory for the data structures required to start the network.

```
unknown control action: <reason>
```

The file */etc/config/snetd.options* contains a line with the unknown registration message *<reason>*

```
<parameter> arg count (number) wrong!
```

The number of arguments to the parameter *<parameter>* in the */etc/config/snetd.options* file is incorrect.

Configure the network parameters using *adm*.

```
warning: ignoring <flag> flag
```

One of the flags in a line at the top of the */etc/config/snetd.options* file declaring the name of a driver, cloneable driver, or a module is not valid.

```
warning: assuming 'c' flags means driver
```

One of the flags in a line at the top of the */etc/config/snetd.options* file declaring the name of a driver, cloneable driver, or a module is not valid.

```
warning: 'm' flag ignored for driver
```

One of the flags in a line at the top of the */etc/config/snetd.options* file declaring the name of a driver, cloneable driver, or a module is not valid.

```
warning: 'c' flag ignored for
```

One of the flags in a line at the top of the */etc/config/snetd.options* file declaring the name of a driver, cloneable driver, or a module is not valid.

---

## Glossary

### **CCITT**

International Consultative Committee for Telegraphy and Telephony.

### **CONS**

Connection-Oriented Network Service.

### **DCE**

Data Circuit-Terminating Equipment.

### **DDN**

Defense Data Network.

### **DNIC**

Data Network Identification Code.

### **DSAP**

Destination Service Access Point.

### **DTE**

Data Terminal Equipment. The user equipment (computer, terminal, etc.) connected to a network.

### **Ethernet**

Ethernet, or IEEE 802.3, is a network based on an access method called CSMA/CD. Ethernet was originally developed by the Xerox Corporation in 1972. A specification was published in 1980 for other implementors to use.

### **Ethernet Address**

A 48-bit station identifier. Each network node has a unique address. The high-order bytes are usually manufacturer-specific. Certain reserved addresses have special meaning.

**getty**

A process that handles incoming calls.

**IEEE**

Institute of Electrical and Electronic Engineers.

**Internet Address**

A 32-bit address used to identify nodes on the network that use the Internet Protocol.

**ISO**

International Organization for Standardization.

**IXE**

IP to X.25 Encapsulation.

**LAN**

A Local Area Network, where several hosts and devices are located near each other and connected physically by cables. Typically, these cables are Ethernet coaxial cables.

**LAP**

Link Access Procedure.

**LAPB**

Link Access Procedure - Balanced.

**LCI**

Logical Channel Identifier.

**LLC**

Logical Link Control.

**LLC-2**

Logical Link Control - Class II.

**LSAP**

Link Service Access Point.

**NLI**

Network Layer Interface.

**NSAP**

Network Service Access Point.

**OSI**

Open Systems Interconnection.

**Packet**

Units of information that contain a header with addresses and handling information and a message. Longer messages can be split up into several different packets and sent separately.

**PAD**

Packet Assembler/Disassembler. A means of interfacing a character terminal to an X.25-based network.

**PDU**

Protocol Data Unit.

**PVC**

Permanent Virtual Circuit.

**RFC**

A "Request for Comment" discussion document, produced by interested parties, which may be adopted as a standard by the Internet Activities Board.

**RPOA**

Recognized Private Operating Agency.

**SNPA**

Subnetwork Point of Attachment.

**SSAP**

Source Service Access Point.

**STREAMS**

A set of system calls, kernel resources, and utility routines that create, use, and dismantle a stream.

**Subnetwork**

In this document, this term refers to a single physical DTE/DCE link or a single physical Ethernet link.

**TOA/NPI**

Type Of Address / Numbering Plan Identification. An address format that provides increased addressing capacity suitable for communication with ISDNs.

**VC**

Virtual Call.

**WAN**

Wide Area Network. Computers with long distances between them connected by, for example, telephone lines.

**X.25**

The CCITT recommendations covering network access protocols for ISO layers 1, 2, and 3 made in 1980, 1984, and 1988.

**X.3, X.28, X.29**

The CCITT recommendations covering PAD operation made in 1980, 1984, and 1988.

---

# Index

## A

Acknowledgment Delay, 59  
Acknowledgment Timer, 75

## B

broadcast packets, 100  
Busy-state Timer, 76

## C

CCITT, 1, 74, 93, 99  
Closed User Groups, 64, 65  
configuration using x25adm, 9  
counter values, 61  
CUD fields, 94

## D

Database Tables Configuration menu, 10  
Data Network Identification Code, 68  
D-Bit Control, 69  
DCE, 22  
DDN, 65, 93, 100  
Destination Service Access Point, 7  
Devices file, 109

diagnostic packets, 67  
Dialcodes file, 109  
Dialers file, 110  
Disconnection Timer, 92  
DTE, 7, 20, 22

## E

error messages, 119  
/etc/config/snetd.options file, 119  
/etc/hosts file, 102, 104  
/etc/netconf file, 120  
Ethernet driver, 117, 120  
Exception Timers, 93

## F

FRMR frames, 116

## I

installation of SX.25, 2  
international call priority, 68  
international call recognition, 67  
Internet, 88  
Internet address, 97, 100, 102, 104  
IP address, 121

ipconfig option, 95  
IP parameters, 9  
IRIS Advanced Site and Server Administration Guide, 109  
ISO 8208, 62  
IXE address map table, 90  
ixeconfig option, 88  
IXE configuration, 88  
IXE Database Configuration, 99  
IXE driver, 87, 99, 100  
IXE hash table, 90  
IXEmap Configuration menu, 102  
IXEmap Database menu, 104  
IXE Network Configuration, 95  
IXE template files, 49

## K

kernelconfig, 10, 14  
kernel configuration, 3, 4  
Kernel Configuration menu, 10, 88

## L

LAP, 74  
LAPB, 21, 47, 74  
LAPB driver, 4, 14, 114, 115  
LAPB parameters, 74  
LAPD template file, 48  
Link Level Conformance, 77  
LLC-2 driver, 4, 116, 117  
LLC-2 template file, 48  
localization, 67  
logical channel identifier, 7, 93, 112  
logical channels, 55

loopback driver, 21  
lower kernel streams, 90  
LSAP, 7, 18

## M

maintenance of X.25 network, 3  
M-bit, 58, 92, 102  
Modulo 128, 56, 58, 76  
Modulo 8, 56, 58, 77

## N

netconf file, 2  
netconfig, 10, 18  
netcreate option, 52, 53  
Network Configuration menu, 10, 74, 78, 91  
Network Daemon Configuration menu, 53  
network hardware, 2  
network management, 10  
Network Management menu, 50, 52, 95, 101  
Network Protocol Mode, 54  
Network Service Access Point, 7  
Network Start, Stop, and Status menu, 10  
NSAP address, 7  
NSDU length, 58, 92  
Number of Transmissions, 75

## P

Packet Level Protocol Mode, 55  
packet size, 57, 62, 65, 72, 102  
PAD kernel parameters, 15  
PAD program configuration, 6

paramconfig, 10, 20, 24  
paramconfig option, 50  
Parameter Configuration menu, 10  
P-bit, 75  
permanent virtual circuits, 1, 6, 55, 90, 93  
Permissions file, 109  
Pre-emption Timer, 92  
Priority Packet Forced Value, 69  
Protocol Parameter Configuration menu, 53, 74

## R

Reject Timer, 76

## S

Sequence Numbering Option, 56  
SGI line discipline, 16  
SLD kernel parameters, 16  
snetd program, 2, 119  
snetmgmt program, 10  
Source Address Control, 69  
stack files, 47, 82  
startstop, 10  
strace utility, 111  
STREAMS logging facilities, 111  
STREAMS parameters, 88  
strerr utility, 111, 118  
subnet mask, 97  
subnetwork ID, 7, 20, 52, 112  
subnetwork modes, 67  
Subnetwork Point of Attachment, 7  
Subscription Options, 65  
Systems file, 110

## T

tables menu, 10  
TCP/IP, 9, 47, 87, 121  
TELENET, 63, 71  
template files, 47, 48, 51  
TEST frames, 117  
throughput class, 62, 71, 72  
timer parameters, 59  
Transit Delay Parameters, 62  
Transmit Probe, 76  
troubleshooting, 111

## U

upper stream, 89  
UUCP configuration, 107

## V

virtual calls, 1, 90  
virtual circuits, 17, 87, 88, 103

## W

WAN configuration parameters, 78  
WAN driver, 115  
WAN kernel parameters, 16  
WANMAP file, 84  
WAN subnetwork, 20  
WAN template file, 48  
window size, 57, 58, 65, 72, 76, 99, 103

**X**

- X.121 string, 99
- X.25 address, 6, 97
- x25adm, 10
- x25adm menu tree, 12
- x25adm utility, 2, 3
- x25config, 19
- X.25 configuration files, 47
- X.25 configuration parameters, 53
- X.25 driver, 4
- X.25 kernel parameters, 17
- X.25 PLP driver, 112
- X.25 template files, 48
- X.25 Version, 54
- XID frames, 117
- XLB stack files, 49
- XLC stack files, 49
- xty device driver, 107
- XTY kernel parameters, 18



---

## We'd Like to Hear From You

As a user of Silicon Graphics documentation, your comments are important to us. They help us to better understand your needs and to improve the quality of our documentation.

Any information that you provide will be useful. Here is a list of suggested topics to comment on:

- General impression of the document
- Omission of material that you expected to find
- Technical errors
- Relevance of the material to the job you had to do
- Quality of the printing and binding

Please include the title and part number of the document you are commenting on. The part number for this document is 007-2179-002.

Thank you!

### Three Ways to Reach Us



The **postcard** opposite this page has space for your comments. Write your comments on the postage-paid card for your country, then detach and mail it. If your country is not listed, either use the international card and apply the necessary postage or use electronic mail or FAX for your reply.



If **electronic mail** is available to you, write your comments in an e-mail message and mail it to either of these addresses:

- If you are on the Internet, use this address: [techpubs@sgi.com](mailto:techpubs@sgi.com)
- For UUCP mail, use this address through any backbone site:  
*[your\_site]!sgi!techpubs*



You can forward your comments (or annotated copies of manual pages) to Technical Publications at this **FAX** number:

415 965-0964