Configuring Protocol Translation

This chapter describes how to configure protocol translation connections. It assumes you understand how to use the configuration software. It provides procedures for specifying system-wide facilities, as well as application examples. Before continuing with this chapter, be sure that you are familiar with the information provided in the LAT, TN3270, and XRemote configuration chapters in this publication. For information on X.25 and TCP/IP (including TCP and Telnet), refer to the *Router Products Configuration Guide* and the *Router Products Command Reference* publications.

For more information about making connections and establishing translation sessions, see the *Cisco Access Connection Guide*.

Note   Telnet is a remote terminal protocol that is part of the Transmission Control Protocol/Internet Protocol (TCP/IP) suite. The descriptions and examples in the following sections use the term TCP as a reference to Telnet functionality.

Cisco's Implementation of Protocol Translation

The protocol translation software attempts to provide transparent protocol translation between systems running disparate protocols. The software fully supports two-way virtual terminal protocol translation between nodes running X.25, Local Area Transport (LAT), and Telnet.

The software supports limited translation for XRemote (to X.25 PAD environments) and TN3270 (to LAT, X.25, and Telnet environments). However, there is no translation between other services such as file-transfer protocols. The protocol translator allows terminal users on one network to access hosts on another network, despite differences in the native protocol stacks associated with the originating device and the target host. You can use either of two methods to accomplish this:

- One-step method—This method provides full transparent protocol translation by means of the **translate** global configuration command.This command defines a connection and the translation protocols in the configuration file. Then you can make single-step connections to remote resources.

- Two-step method—Actual connections between devices are made by first establishing a connection to the protocol translator, then by establishing the outgoing connection to the remote destination.

Number of Translation Sessions Supported

The number of translation sessions supported depends on the hardware model and software version you are running. The following router models support protocol translation in IOS Release 10:

- Cisco 2500

- Cisco 3102

- Cisco 3104

- Cisco 4500

Each of these systems supports a default of 64 sessions when the system is routing, 100 sessions when the system is not routing, and a maximum of 180 sessions with or without routing.

Change the Number of Supported Translation Sessions

To increase the number of protocol translation sessions, perform the following task in line configuration mode:

Task	Command
Increase the number of protocol translation sessions.	**line vty** *number*
Decrease the number of protocol translation sessions.	**no line vty** *number*

Support for Terminal Connections and Protocol Translation

In the context of this discussion, a router set up to run protocol translation software is referred to as a protocol translator.

The protocol translator uses the International Telecommunication Union Telecommunication Standardization Sector (ITU-T) Recommendation X.25 for transferring raw data over X.25 networks. The X.25 software supports both commercial and DDN versions. Protocol translators also support X.25 as a transport mechanism for IP packets, and X.3- and X.29-based PAD connections. This allows the protocol translators to connect to an X.25 public data network (PDN). This X.25 connection allows transport of TCP/IP packets across the X.25 packet-switching network in the same way as would occur on a protocol translator.

Note   The ITU-T carries out the functions of the former Consultative Committee for International Telegraph and Telephone (CCITT).

Routers without protocol translation do not support an X.25 PAD function, so they cannot communicate with hosts directly connected to the X.25 PDN. Routers encapsulate TCP/IP packets in X.25 packets for transfer over a packet-switching network. These packets can be received by routers, communication servers, and routers set up as protocol translators. Only protocol translators include PAD capabilities, but other servers can communicate with protocol translators using the TCP/IP or LAT protocols. Protocol translators can translate TCP/IP and LAT into X.25, and then communicate with X.25 hosts. Protocol translators support all PAD standards (X.28, X.29, and X.3). The connection to the packet-switching network is through a synchronous line.

Connections to a PAD are made using EXEC commands. You can configure PAD parameter profiles that can be used to set PAD parameters with other commands, and you can configure access lists to control X.25 network access. Both these features use the message fields defined in Recommendation X.29, which describes procedures for exchanging data between PADs or between a PAD and a DTE.

Protocol Translation Configuration Task List

Perform the tasks in one of the following sections when setting up the use of protocol translation:

- Configure One-Step Protocol Translation
- Configure Two-Step Protocol Translation

Perform the tasks in the following sections to apply X.29 PAD connections under X.25:

- Create X.29 Access Lists
- Create an X.29 Profile Script

See the end of this chapter for protocol translation application examples.

Configure One-Step Protocol Translation

The one-step method of protocol translation invokes a *translation session* process to provide fully transparent protocol conversion. In this method, the protocol translator masquerades as two or more hosts on the same network. When a connection is made to the protocol translator, the protocol translator determines which host the connection is for and what protocol that host is using. The protocol translator then establishes a new network connection using the networking protocol required by that host. This network connection is more efficient and allows the protocol translator to act upon greater knowledge of the protocols in use because the protocol translator acts as a network connection rather than a terminal.

One disadvantage of using the one-step method is that the initiating computer or user does not know that two networking protocols are being used. This means that parameters of the foreign network protocols cannot be changed once the connections are established. The exception to this limitation is the set of parameters common to both networking protocols. Any parameter in this set can be changed from the first host to the final destination.

To configure the one-step method of protocol translation, set up the following protocols and connection options in the configuration file:

- The incoming connection

  The configuration includes the protocol to be used—LAT, X.25, or TCP/IP—the address, and any options such as reverse charging or binary mode that are supported for the incoming connection.

- The outgoing connection

  The outgoing connection is defined in the same way as the incoming connection.

- The connection features and global options

  You can specify additional features for the connection that allows, for example, incoming call addresses to match access-list conditions or that limit the number of users that can make the connection.

Note To use the protocol translator to convert TN3270 or XRemote, you must use the two-step method.

To create one-step protocol translation connection specifications, perform the following global configuration task:

Task	Command
Create the connection specifications for one-step protocol translation.	**translate** *protocol incoming-address* [*in-options*] *protocol outgoing-address* [*out-options*] [*global-options*]

For examples of using the one-step method, see the section "Protocol Translation Application Configuration Examples" later in this chapter.

Configure Two-Step Protocol Translation

In the two-step method, perform the following two steps:

Step 1 Establish the incoming connection directly to the protocol translator.

Step 2 Establish the outgoing connection from the protocol translator to another network host.

Specifically, once the transmission protocols—LAT, X.25, TCP/IP—are configured, all that is required for the two-step protocol translation method is to enter the EXEC connection commands—**lat**, **pad**, **telnet**—first at one server, then onto another.

The protocol translator supports the two-step method in both directions (for example, from Telnet to PAD, and vice versa).

In general, you use the two-step method when you want to use a protocol translator as a general-purpose gateway between two types of networks (for example, X.25 PDN and TCP/IP). Instead of configuring every possible connection via embedded **translate** commands, the two-step method allows you greater flexibility in terms of connecting to network resources accessible via the protocol translator.

The two-step method also allows additional security when TACACS and password protection is enabled. These security features are described in *Router Products Configuration Guide*, in the chapter "Configuring the System." Additionally, if you use the two-step method to connect, you can change X.25 PAD and local terminal parameters dynamically during a session.

With the two-step connection method, you can individually modify the parameters of either network connection, even while a session is in process. This process is similar to connecting a group of terminal lines from a PAD to a group of terminal lines from a TCP protocol translator. The difference is that you do not encounter the wiring complexity, unreliability, management problems, and performance bottlenecks that occur when two separate devices are connected via asynchronous serial lines.

Note   You must use the two-step method for translations of TN3270 and XRemote.

To configure the two-step method of protocol translation, see the configuration task list in the *Cisco Access Connection Guide*.

Create X.29 Access Lists

The protocol translation software provides access lists, which make it possible to limit access to the protocol translator from certain X.25 hosts. Access lists take advantage of the message field defined by Recommendation X.29, which describes procedures for exchanging data between two PADs or between a PAD and a DTE device.

To define X.29 access lists, perform the following tasks:

Step 1   Create an access list.

Step 2   Apply an access list to a virtual line or to a translate command.

These tasks are described in the following sections.

When configuring protocol translation, you can specify an access list number with each **translate** command. In the case of translation sessions that result from incoming PAD connections, the corresponding X.29 access list is used.

Create an Access List

To specify the access conditions, perform the following global configuration task:

Task	Command
Restrict incoming and outgoing connections between a particular virtual terminal line (into a Cisco device) and the addresses in an access list.	**x29 access-list** *access-list-number* {**permit**	**deny**} *regular-expression*

An access list can contain any number of lines. The lists are processed in the order in which you type the entries. The first match causes the permit or deny condition. If an X.121 address does not match any of the entries in the access list, access will be denied.

Apply an Access List to a Virtual Line

To apply an access list to a virtual line, perform the following tasks:

Task		Command[1]
Step 1	Specify a virtual line.	**line vty** *x*
Step 2	Restrict incoming and outgoing connections between a particular virtual terminal line (into a Cisco device) and the addresses in an access list.	**access-class** *access-list-number* **in**

1. These commands are documented in the *Router Products Command Reference* publication.

The access-list number is used both for incoming TCP access and for incoming PAD access. For TCP access, the protocol translator uses the defined IP access lists. For incoming PAD connections, the same X.29 access list is used. If you want to have access restrictions on only one of the protocols, you can create an access list that permits all addresses for the other protocol.

Note For an example of applying an access list to a translate command, see the section "X.29 Access List Example" at the end of this chapter.

Create an X.29 Profile Script

You can create an X.29 profile script for use by the **translate** command. When an X.25 connection is established, the protocol translator then acts as if an X.29 SET PARAMETER packet had been sent that contained the parameters and values set by this command.

To create an X.29 profile script, perform the following global configuration task:

Task	Command
Create an X.29 profile script.	**x29 profile** *name parameter***:***value* [*parameter***:***value*]

Protocol Translation Application Configuration Examples

The following sections describe examples of configuring protocol translation applications. The environment used in these examples is illustrated in Figure 5-1. The application environment described focuses on Cisco 3000 implementations. However, in applications featuring only protocol translation, any of the protocol translator products can be used.

- Basic Configuration Example
- Local LAT-to-TCP Example
- LAT-to-X.25 Host Example
- X.25 PAD-to-LAT Example
- X.25 PAD-to-TCP Example
- LAT-to-LAT via X.25 Translation Example
- LAT-to-TCP via X.25 Example
- LAT-to-LAT over Frame Relay or SMDS Example
- LAT-to-LAT over an IP WAN Example
- Central Site Protocol Translation Example
- Standalone LAT-to-TCP Translation Example
- X.29 Access List Example
- X.3 Profile Example
- Increase the Number of Translation Sessions Example
- Decrease the Number of Translation Sessions Example

In these examples, a LAT device can be a LAT terminal server or LAT host, and a TCP device can be either a TCP terminal server or TCP host.

Note In the application illustrations that follow throughout this chapter, source and destination device icons used to illustrate the flow of translated information are shown with black type in outlined shapes. Other elements in the environment are shown with reverse type on solid black shapes (as shown for all elements in Figure 5-1).

Basic Configuration Example

The following examples illustrate the basic global configuration commands and interface configuration commands for setting up 3000-A (connected to Network A) and 3000-B (connected to Network B), as illustrated in Figure 5-1. Refer to the chapter "Configuring LAT" in this publication, for more information on LAT. For information on configuring X.25, refer to the *Router Products Configuration Guide*.
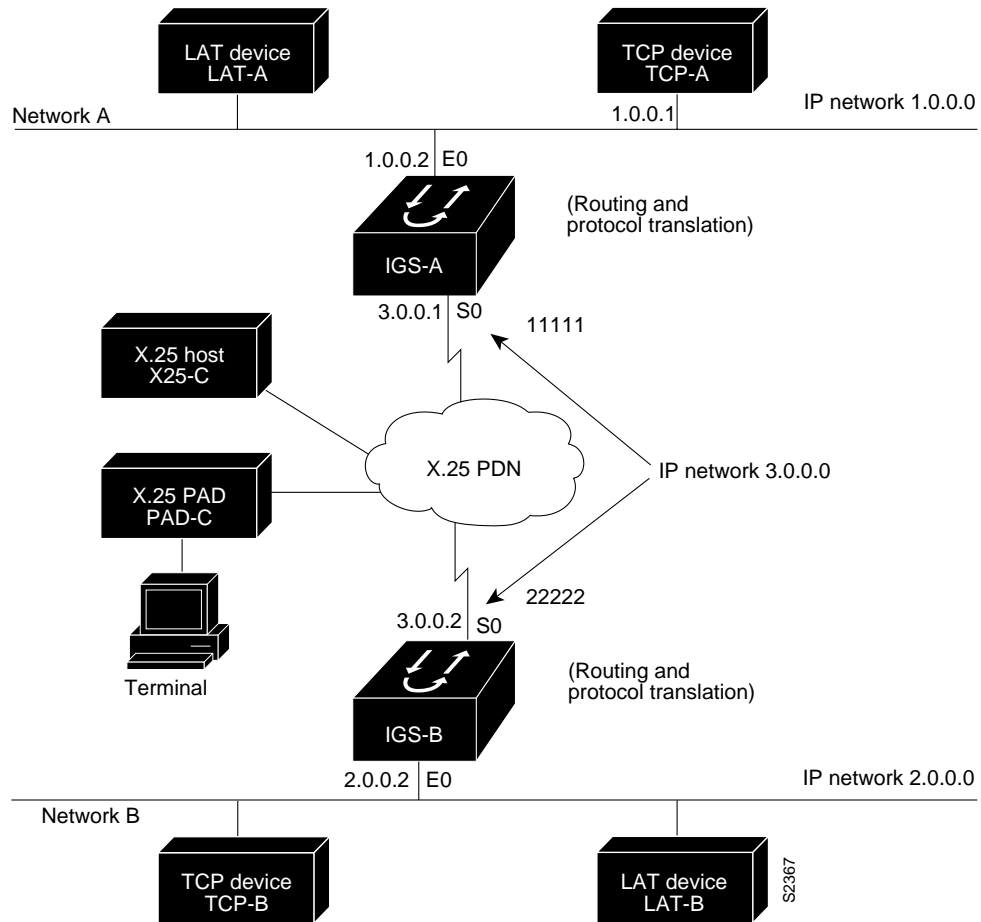
Figure 5-1   Summary Diagram Showing Cisco 3000 Systems as Protocol Translators

---

Note   The examples that follow focus on creating configurations that support one-step protocol translation. These connections can also be made using the two-step protocol translation method.

Configuration for Device 3000-A

The following partial configuration for device 3000-A outlines a baseline configuration for the device's Ethernet and serial interfaces and configures support for IP, LAT, and X.25:

```
interface ethernet 0
ip address 1.0.0.2 255.255.0.0
!
! Enable LAT on interface
lat enabled
!
interface serial 0
encapsulation X.25
x25 address 11111
!
! The following parameters may depend on your network
x25 facility packetsize 512 512
x25 facility windowsize 7 7
!
! IP address and MAP command needed only if routing IP
ip address 3.0.0.1 255.255.0.0
x25 map ip 3.0.0.2 22222 broadcast
!
! Set up IP routing
router igrp 100
network 1.0.0.0
network 3.0.0.0
!
! Advertise as available for connections via LAT
! Use this name (3000-A) if connecting via 2-step method
! (for connecting directly to a specific protocol translator)
lat service 3000-A enable
!
! Set up some IP host names/addresses
ip host 3000-A 1.0.0.2 3.0.0.1
ip host TCP-A 1.0.0.1
ip host TCP-B 2.0.0.1
ip host 3000-B 3.0.0.2 2.0.0.2
```

Configuration for Device 3000-B

The following partial configuration for device 3000-B outlines a baseline configuration for the device's Ethernet and serial interfaces and configures support for IP, LAT, and X.25:

```
Interface ethernet 0
ip address 2.0.0.2 255.255.0.0
!
! enable LAT on interface
lat enabled
!
interface serial 0
encapsulation X.25
x25 address 22222
! The following parameters may depend on your network
x25 facility packetsize 512 512
x25 facility windowsize 7 7
!
! IP address and MAP command needed only if routing IP
ip address 3.0.0.2 255.255.0.0
x25 map ip 3.0.0.2 11111 broadcast
!
! Set up IP routing
router igrp 100
network 2.0.0.0
network 3.0.0.0
!
! advertise as available for connections via LAT
! Use this name (3000-B) if connecting via 2-step method
! (for connecting directly to a specific Protocol Translator)
lat service 3000-B enable
!
! Set up some IP host names/addresses
ip host 3000-A 3.0.0.1 1.0.0.2
ip host TCP-A 1.0.0.1
ip host TCP-B 2.0.0.1
ip host 3000-B 2.0.0.2 3.0.0.2
```

Note You can specify IP host names used to identify specific hosts by explicitly using the **ip host** global configuration command or by using Domain Name System (DNS) facilities.

Local LAT-to-TCP Example

A protocol translator such as the Cisco 3000 can translate between LAT and TCP (Telnet) traffic to allow communication among resources in these protocol environments. In Figure 5-2, the LAT device on Network A (LAT-A) is shown connecting to a TCP device (TCP-A).

This is only a partial example. The commands in this example would be only part of the complete configuration file for an individual protocol translator or router.
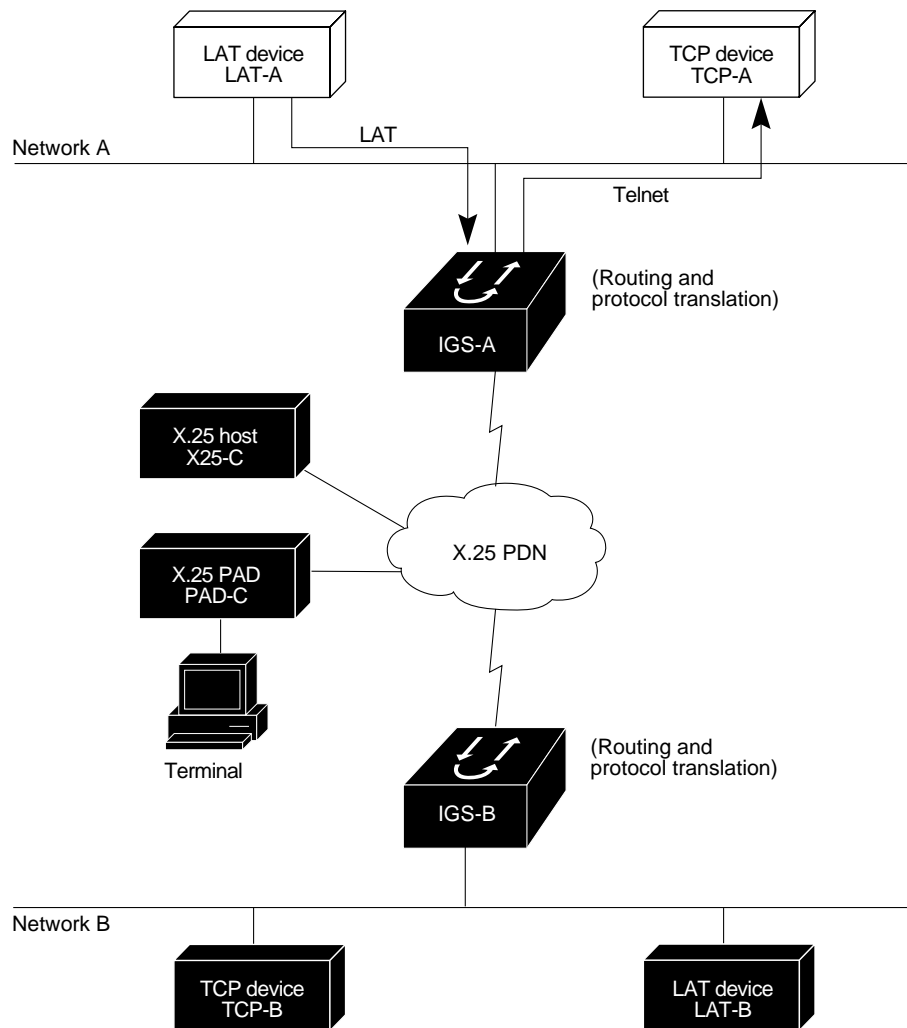
Figure 5-2   Local LAT-to-TCP Translation

The following example configures device 3000-A to translate from LAT to TCP:

```
! Translate LAT connections to TCP for connectivity to TCP-A
translate lat TCP-A tcp TCP-A
! Optional additional commands
lat service TCP-A ident Protocol Translation to TCP-A
```

In the last command, the text string "Protocol Translation to TCP-A" is an identification string for the LAT service named TCP-A. This string is sent to other servers on the local network.

LAT-to-X.25 Host Example

The Cisco 3000 protocol translation option allows LAT devices to communicate with X.25 hosts through an X.25 PDN. In the application illustrated in Figure 5-3, LAT-A is a LAT device that is communicating with X25-C, an X.25 host. The LAT traffic from LAT-A is translated to X.25.
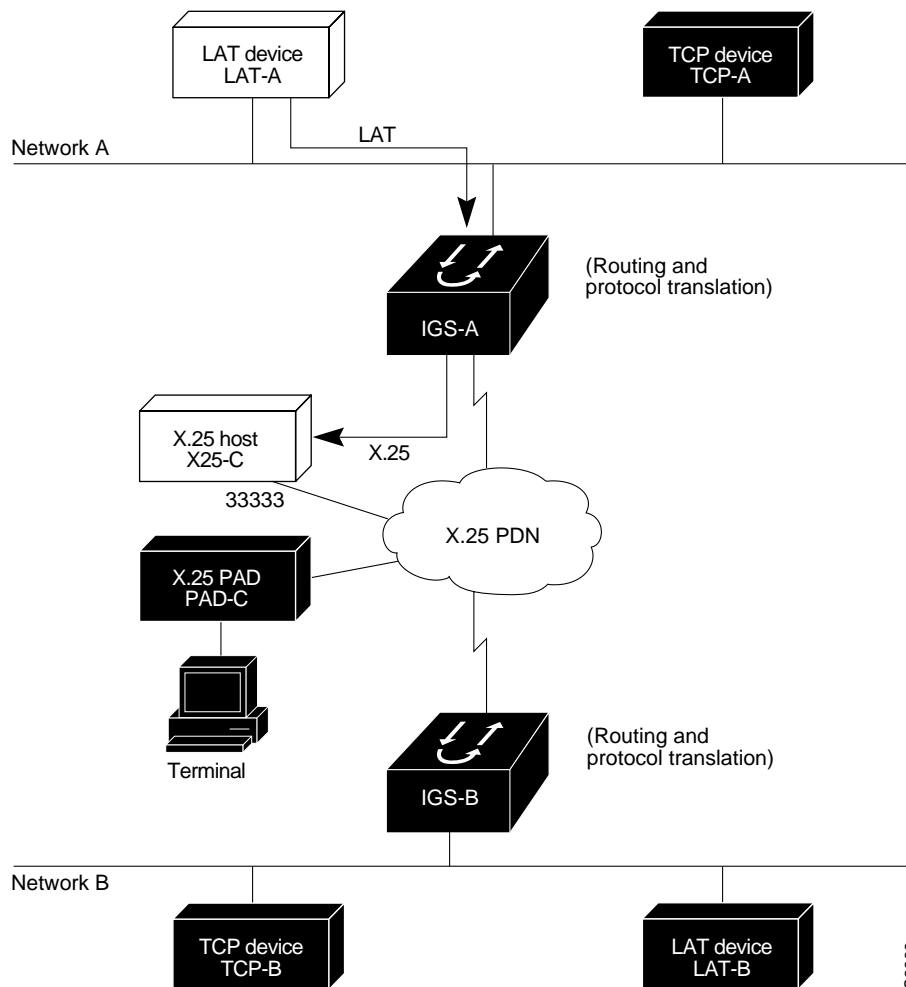

Figure 5-3   LAT-to-X.25 Host Translation

The following example illustrates how to use the **translate** global configuration command to translate from LAT to X.25. It is applied to 3000-A. This example sets up reverse charging for connections, which causes the protocol translator to instruct the PDN to charge the destination for the connection. It is essentially a collect call. The reversal of charges must be pre-arranged with the PDN and destination location (on an administrative basis), or the call will not be accepted.

```
! Translate LAT to X.25 host, with reverse charging
translate lat X25-C x25 33333 reverse
!
! Specify optional X.25 hostname
x25 host X25-C 33333
```

X.25 PAD-to-LAT Example

The Cisco 3000 protocol translation option allows terminals connected to X.25 PADs to communicate with LAT devices on a remote LAN (Figure 5-4). X.25 PAD terminals make a call using an X.121 address, which is translated to a LAT node. To the PAD terminal user, the connection appears to be a direct connection to a host on the X.25 PDN. The Cisco 3000 also supports X.29 access lists, which allow you to restrict LAN resources (LAT or TCP) available to the PAD user.
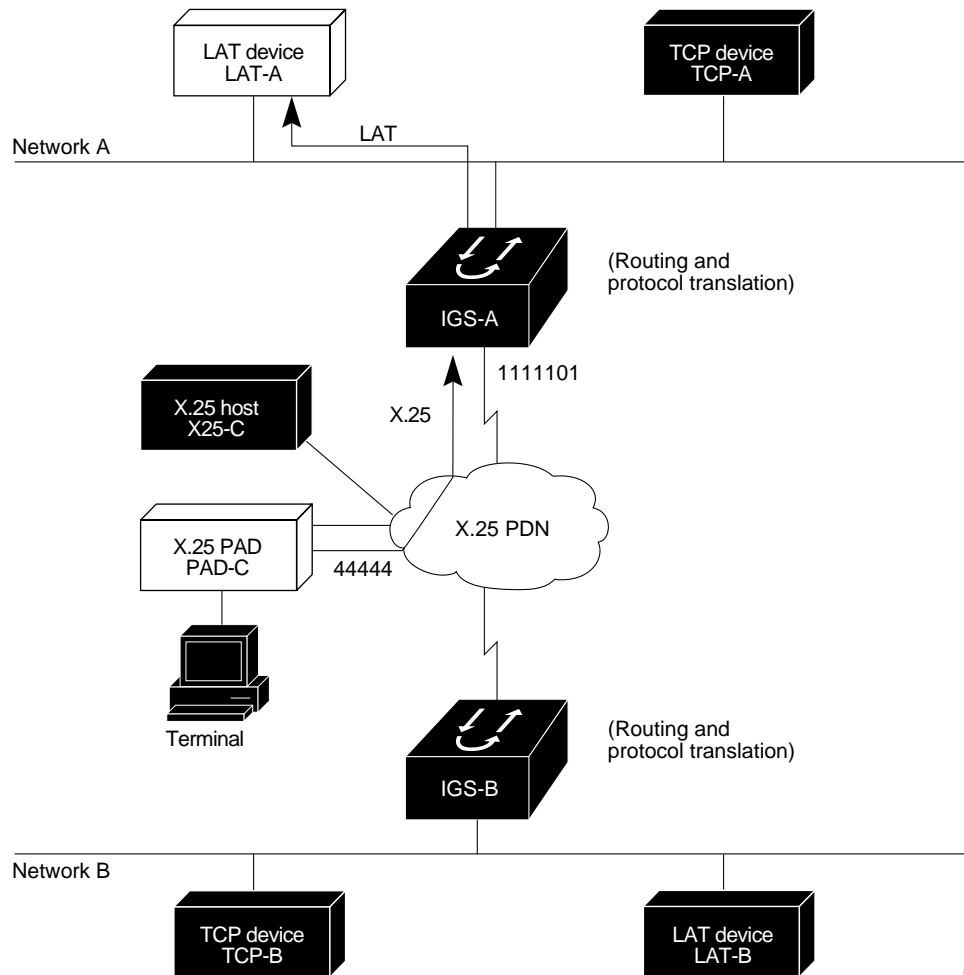

Figure 5-4   X.25 PAD-to-LAT Translation

The following example illustrates how to use the **translate** global configuration command to translate from an X.25 PAD to a LAT device on Network A. It is applied to 3000-A. The configuration example includes an access list that limits remote LAT access through 3000-A to connections from PAD-C.

```
! Define X25 access list to only allow pad-c
x29 access-list 1 permit ^44444
x29 access-list 1 deny .*
!
! Set up translation
translate x25 1111101 lat LAT-A access-class 1
```

This configuration example typifies the use of access lists in Cisco internetworking systems. The first two lines define the scope of *access-list 1*. The first line specifies that access list 1 will permit all calls from X.121 address 44444. The caret symbol (^) specifies that the first number 4 is the beginning of the address number. Refer to the appendix "Regular Expressions" later in this publication for details concerning the use of special characters in defining X.121 addresses. The second line of the definition explicitly denies calls from any other number.

This access list is then applied to all incoming traffic on the serial port for 3000-A (X.121 address 1111101) with the third configuration line in the example. However, it applies only to the **translate** command at the end of this example. This **translate** command specifies that incoming X.25 packets on the serial line (with address 1111101) are translated to LAT and sent to LAT-A if they pass the restrictions of the access list.

If you define multiple X.25 translate commands, each must contain a unique X.121 address. It is also important that the ITU-T protocol used in transferring packets match among X.121 addresses. This is specified in the protocol identification field of call-user data. This field specifies whether a packet is routed, translated, or handled as a virtual terminal connection.

Note The X.121 address 1111101 used in this example can be a subaddress of the address 11111 originally assigned to this serial port on 3000-A at the beginning of the configuration example section. However, that is not a requirement. The number to use in the **translate** command is negotiated (administratively) between your network management personnel and the PDN service provider. The X.121 address in the **translate** command represents the X.121 address of the calling device. That number might or might not be the number (or a subaddress of the number) administratively assigned to the protocol translator. It is up to you and the PDN to agree on a number to be used, because it is possible that the PDN can be configured to place calls on a given line that are intended for a destination that does not match the number assigned by you in the configuration file. Refer to the *1984 CCITT Red Book* specifications for more information concerning X.121 addresses.

X.25 PAD-to-TCP Example

Making a translated connection from an X.25 PAD to a TCP device (Figure 5-5) is analogous to the preceding X.25 PAD-to-LAT example. Instead of translating to LAT, the 3000-A configuration includes a statement to translate to TCP (Telnet). Note that a protocol translator can include statements supporting both translations (X.25 PAD-to-LAT and X.25 PAD-to-TCP). Different users on the same PAD can talk to X.25, LAT, or TCP devices.
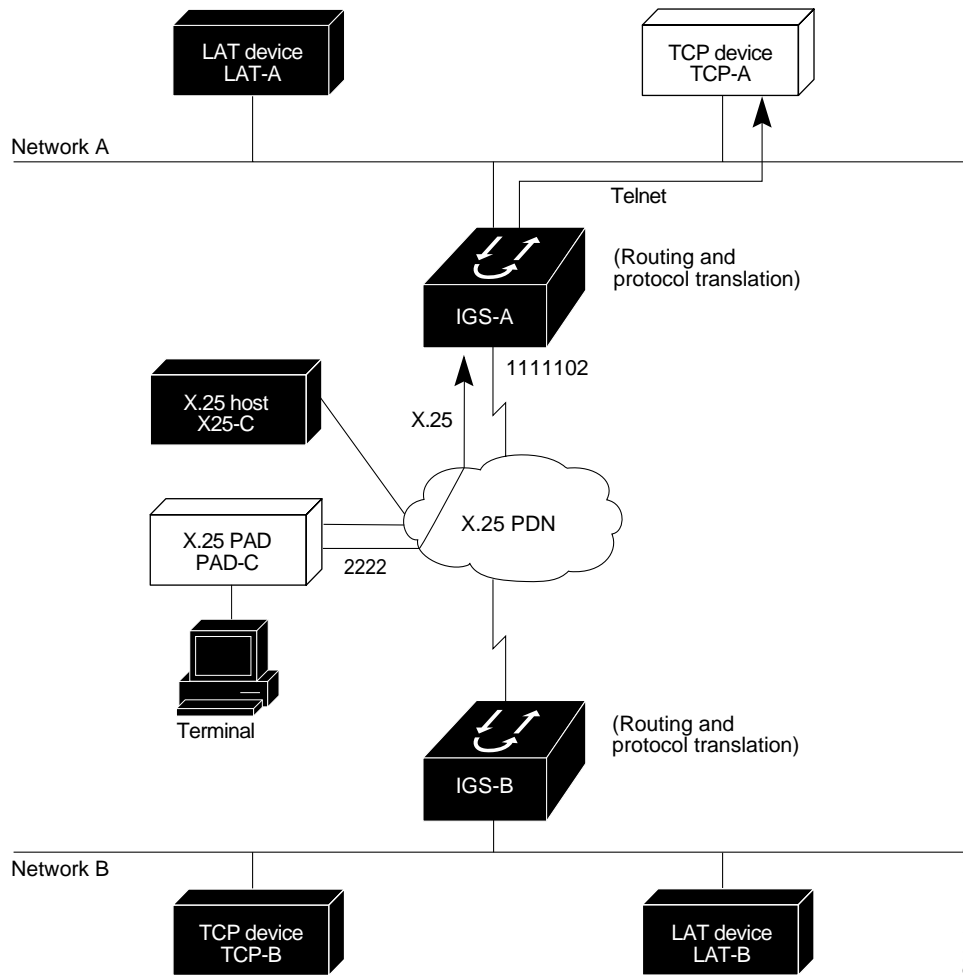
Figure 5-5   X.25 PAD-to-TCP Translation

The following example illustrates how to use the **translate** global configuration command to translate from an X.25 PAD to a TCP device on Network A. It is applied to 3000-A.

```
! Set up translation
translate x25 2222 tcp TCP-A
```

LAT-to-LAT via X.25 Translation Example

You can use protocol translators to provide transparent connectivity between LAT devices on different networks via an X.25 PDN. In Figure 5-6, which illustrates this application, the LAT device on Network A (LAT-A) first makes a virtual connection to the Cisco 3000 on Network A using the LAT protocol. The Cisco 3000 then translates the LAT packets into X.25 packets and sends them through the X.25 network to the Cisco 3000 on Network B. This second Cisco 3000 translates the X.25 packets back to LAT packets and establishes a virtual connection to the LAT device on Network B (LAT-B). These handoffs are handled transparently when the protocol translator is configured for a one-step translation.
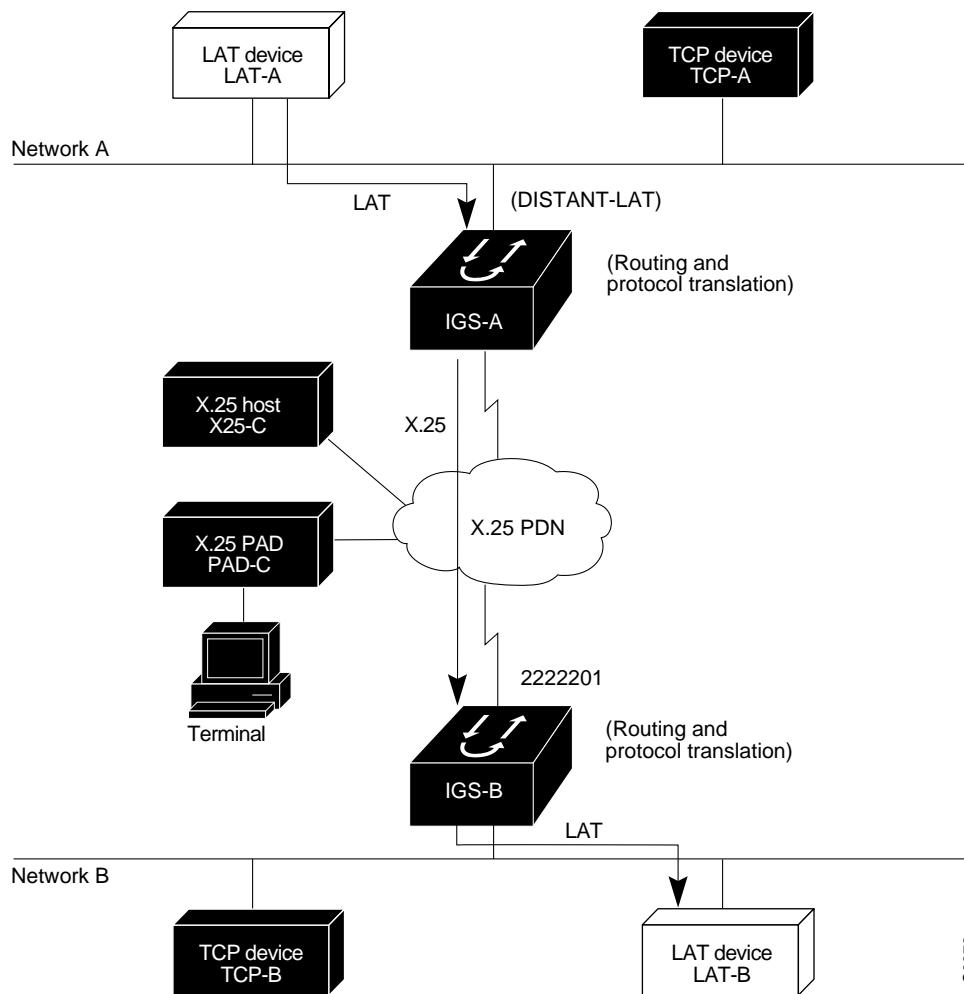
Figure 5-6   LAT-to-LAT via an X.25 PDN

The following two examples illustrate how to use the **translate** global configuration command to translate from LAT to X.25 and from X.25 back to LAT to allow connection service to a LAT device on Network B from a LAT device on Network A. This requires two separate configurations, one for each LAT device.

```
! Translate LAT to X.25 on 3000-A, which is on Network A
translate lat DISTANT-LAT x25 2222201

! Translate X.25 to LAT on 3000-B, which is on Network B
translate x25 2222201 lat LAT-B
```

In the first **translate** command, DISTANT-LAT defines a LAT service name for 3000-A. When a user on device LAT-A attempts to connect to TCP-B, the target specified in the **connect** command is DISTANT-LAT. (The **connect** command is described in the *Cisco Access Connection Guide*.)

In the **translate** command for 3000-B, the name of the LAT service on the target host (LAT-B) is LAT-B. 3000-B translates the incoming X.25 packets from 2222201 to LAT and then transparently relays these packets to LAT-B.

The following is an example of a connection request. In this configuration example, when the user enters this command, a connection attempt from LAT-A on Network A to TCP-B on Network B is attempted.

```
local> connect DISTANT-LAT
```

To configure 3000-B to send information back from LAT-B to LAT-A, use commands symmetrical to the prior configuration (this path is not shown in Figure 5-6):

```
! Translate LAT to X.25 on 3000-B, which is on Network B
translate lat FAR-LAT x25 1111103

! Translate X.25 to LAT on 3000-A, which is on Network A
translate x25 1111103 lat LAT-A
```

Note You can use the same name (for example, "LAT-B") in the **translate** command for both 3000-A and 3000-B because each protocol translator operates independently. However, this symmetry is not required. The key is the common X.121 address used in both **translate** commands. If you prefer to have unique service names, set the names in each protocol translator to be the same.

LAT-to-TCP via X.25 Example

You can use protocol translators to provide transparent connectivity between LAT and TCP devices on different networks via an X.25 PDN. In the application illustrated in Figure 5-7, the LAT device on Network A is communicating with the TCP device on Network B. There are two ways to provide this connectivity: the LAT traffic from Network A can be translated into either X.25 packets or TCP/IP packets to be sent out on the X.25 PDN.

If the traffic is translated from LAT directly into X.25 frames by 3000-A, then 3000-B on Network B translates incoming packets intended for device TCP-B into TCP. If 3000-A converts LAT to TCP, then the TCP traffic is being encapsulated in X.25 and sent on the X.25 network; the Cisco 3000 on Network B strips off the encapsulation and routes the TCP packet. In this case, protocol translation is not needed on 3000-B.

If the traffic is translated to TCP by 3000-A, the packets are encapsulated within X.25 frames. In general, translating the traffic directly to X.25 is more efficient in this application because no encapsulation is necessary. X.25 packets have only 5 bytes of header information, while TCP over X.25 has 45 bytes of header information.
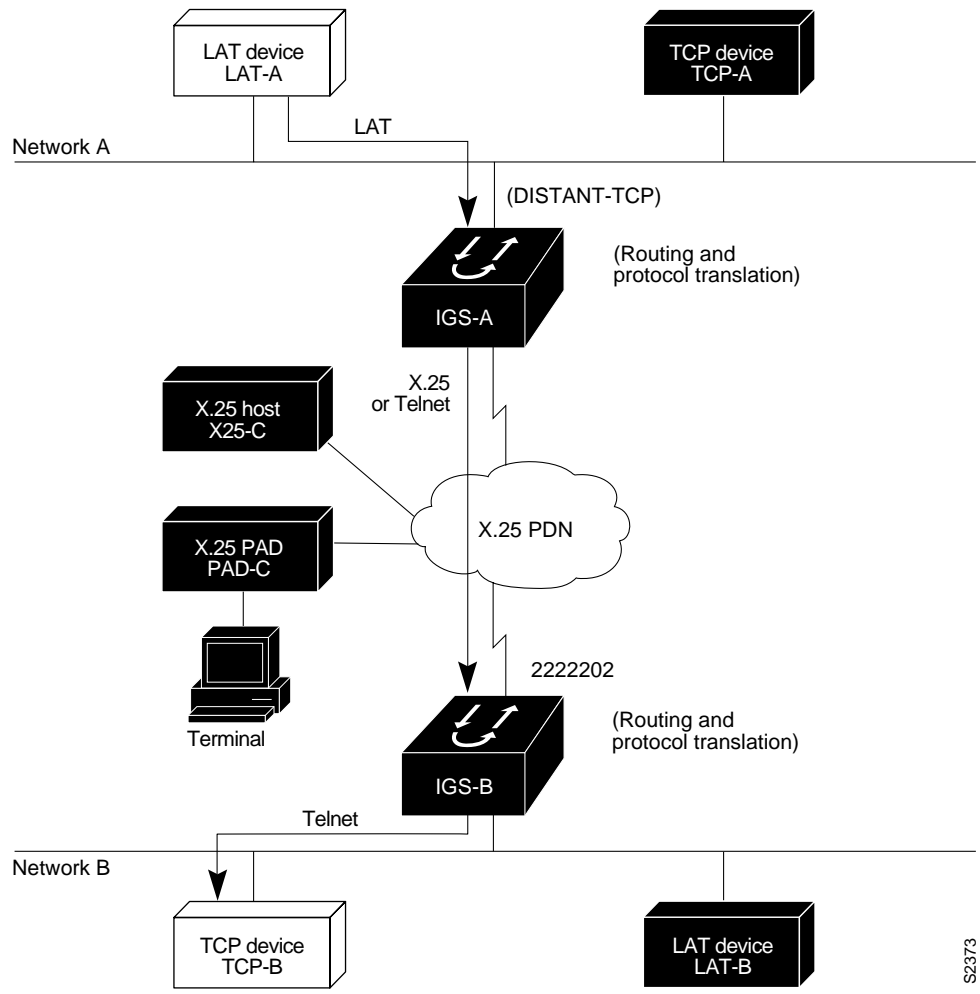
Figure 5-7   LAT-to-TCP via X.25

The following examples illustrate how to use the **translate** global configuration command to translate from LAT to X.25 (on 3000-A) and from X.25 to TCP (on 3000-B), thus allowing connection service to a TCP device on Network B (TCP-B) from a LAT device on Network A (LAT-A). You must configure 3000-A and 3000-B separately.

```
! Translate LAT to X.25 on 3000-A, which is on Network A
translate lat DISTANT-TCP x25 2222202

! Translate X.25 to TCP on 3000-B, which is on Network B
translate x25 2222202 tcp TCP-B
```

In the **translate** command for 3000-A, DISTANT-TCP defines a LAT service name for 3000-A. When a user on device LAT-A attempts to connect to LAT-B, the target specified in the **connect** command is DISTANT-TCP.

In the **translate** command for 3000-B, the TCP service on the target host (TCP-B) is TCP-B. 3000-B translates the incoming X.25 packets from 2222202 to TCP packets and transparently relays these packets to TCP-B.

The following is an example of a connection request. In this configuration example, when the user enters this command, a connection attempt from LAT-A on Network A to LAT-B on Network B is attempted.

```
local> connect DISTANT-TCP
```

Note You can use the same name (for example, "TCP-B") in the **translate** command for both 3000-A and 3000-B, because each protocol translator operates independently. However, this symmetry is not required. The key is the common X.121 address used in both **translate** commands. If you prefer to have unique service names, set the names in each protocol translator to be the same.

LAT-to-LAT over Frame Relay or SMDS Example

To transport LAT traffic over a Frame Relay or an SMDS network, LAT must first be translated to TCP. The TCP traffic is routed over the Frame Relay network and then translated back to LAT on the Cisco 3000 on Network B. See Figure 5-8.

Note The interface configurations for a Frame Relay or an SMDS implementation differ from the specifications at the beginning of this chapter. For more information about configuring Frame Relay and SMDS, see the *Router Products Configuration Guide*.
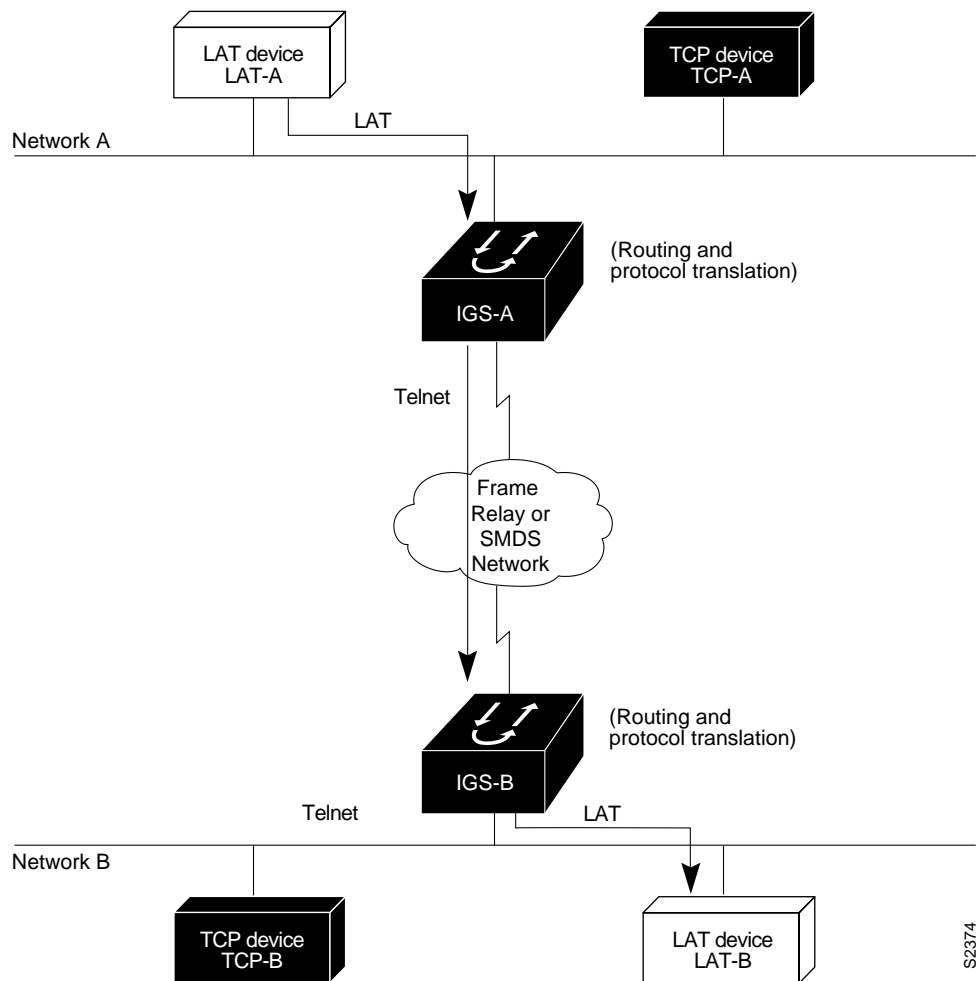
Figure 5-8   LAT-to-LAT over Frame Relay or SMDS

The following example illustrates how to use **translate** global configuration command to translate from LAT to LAT when the WAN uses Frame Relay or SMDS. In this configuration, the Cisco 3000 routes encapsulated packets translated from LAT to TCP over the Frame Relay or SMDS network. Packets are then translated back to LAT on the other side of the Frame Relay or SMDS network.

```
! Translate LAT to TCP/Telnet on 3000-A, which is on Network A
translate lat DISTANT-LAT tcp 3000-B

! Translate TCP to LAT on 3000-B, which is on Network B
translate tcp 3000-B lat LAT-B
```

Note   You can use the same name (for example, "LAT-B") in the **translate** command for both 3000-A and 3000-B because each protocol translator operates independently. However, this symmetry is not required. The key is the common IP name used in both **translate** commands.

LAT-to-LAT over an IP WAN Example

The Cisco 3000 with protocol translation can be used to connect LAT devices over a WAN backbone that only allows routable protocols (see Figure 5-9). This configuration exists when LAT networks are either isolated or on their own internetwork.

With the Cisco 3000, LAT traffic can be translated to TCP and then routed on the WAN as TCP traffic. The LAT connections stay local between the LAT device and the Cisco 3000. Thus, connections are not susceptible to delays on the WAN. This reduces the amount of traffic on the WAN because only the data from specific LAT sessions is forwarded on the WAN rather than all the LAT protocol status information packets.
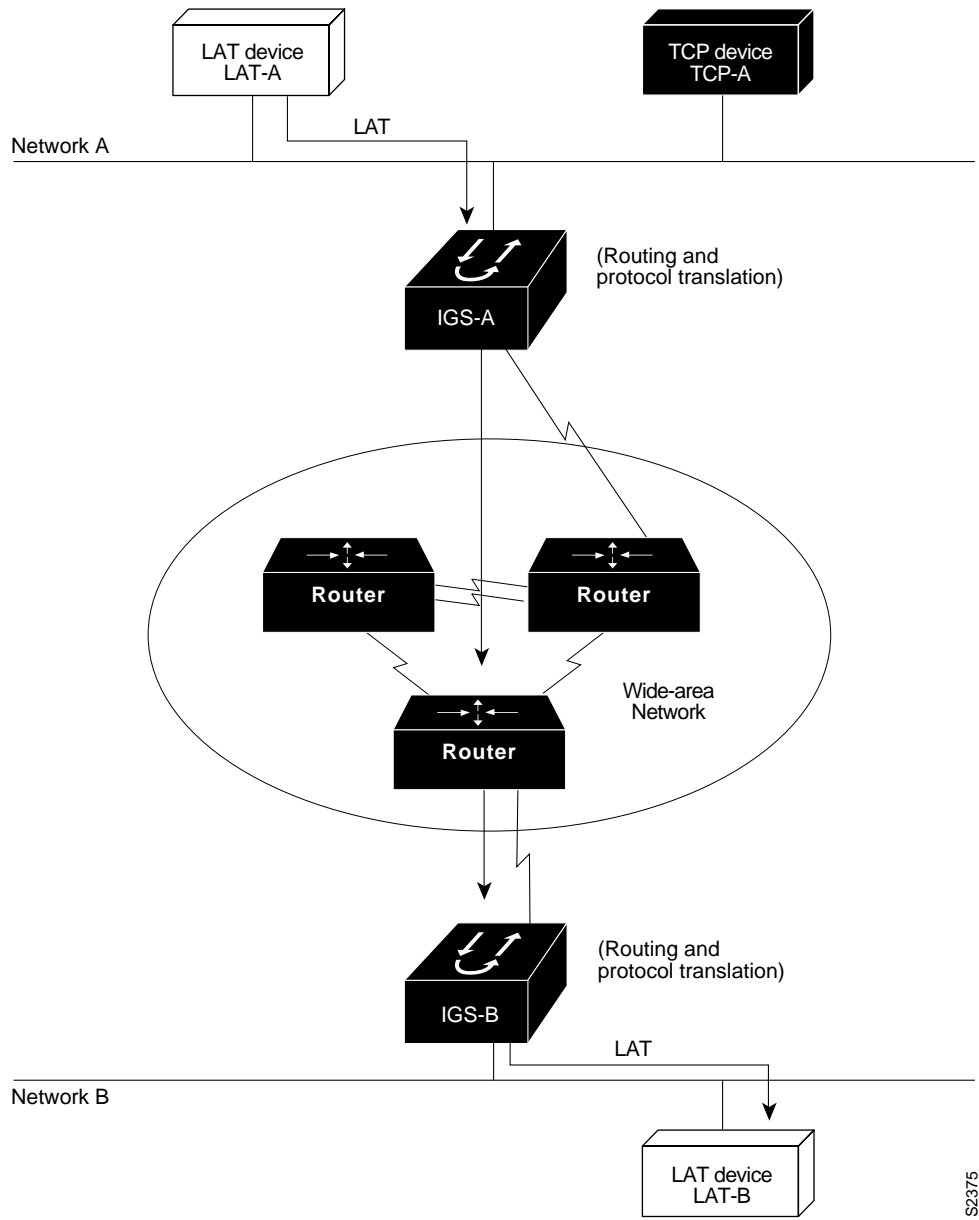

Figure 5-9   LAT-to-LAT over an IP WAN

The following example illustrates how to use the **translate** global configuration command to translate from LAT to LAT when an IP WAN is used. In this configuration, the Cisco 3000 routes encapsulated packets translated from LAT to TCP over the WAN. Packets are then translated back to LAT on the other side of the WAN. Example translation configurations for both 3000-A and 3000-B are shown, but these examples do not include specifics of configuration for routers in the WAN. These examples are essentially the same configurations for protocol translation as those in the earlier Frame Relay example.

```
! Translate LAT to TCP/Telnet for 3000-A, which is on Network A
translate lat DISTANT-LAT tcp 3000-B

! Translate TCP to LAT for 3000-B, which is on Network B
translate tcp 3000-B lat LAT-B
```

Note You can use the same name (for example, "LAT-B") in the **translate** command for both 3000-A and 3000-B, because each protocol translator operates independently. However, this symmetry is not required. The key is the common IP name in both **translate** commands.

Central Site Protocol Translation Example

If you require the translation of protocols among multiple remote sites over an X.25 network, with retranslation at a central site, a Cisco 3000 might be required. The Cisco 3000 in this configuration allows more than 64 concurrent translation sessions.

To support this application, a Cisco 3000 is directly connected back-to-back (Figure 5-10) to another Cisco router. This second router acts as an X.25 switch, by sending X.25 packets to the Cisco 3000 while concurrently routing and bridging other protocols.

Because the Cisco 3000 is operating only as a protocol translator, it can support 100 concurrent protocol translation sessions compared to the 64 that are supported when the Cisco 3000 is routing, bridging, and translating protocols.
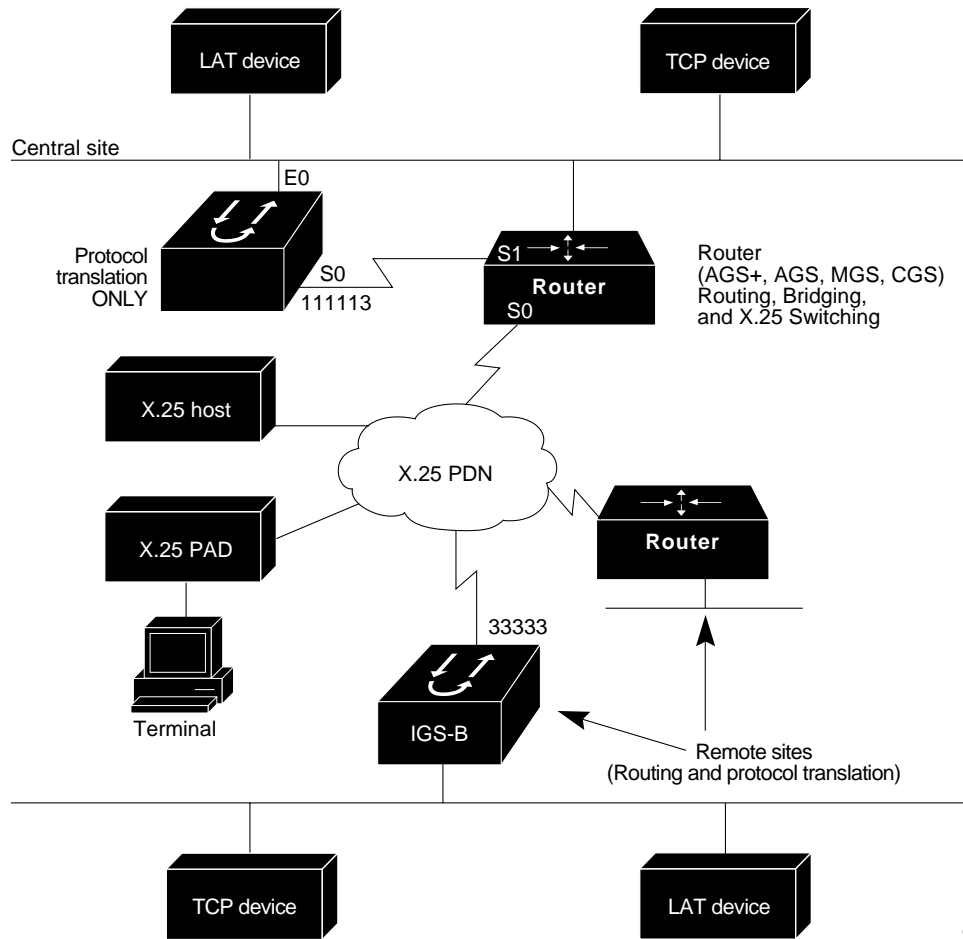
Figure 5-10   Central Site Protocol Translation Example

The following example illustrates how to configure routers to support translating protocols over an X.25 network among multiple sites.

The second Cisco router is configured to act as an X.25 switch to send X.25 packets to the Cisco 3000 while concurrently routing and bridging other protocols.

The following example also illustrates how to use the **translate** global configuration command to translate LAT and TCP over X.25 WAN media. In this configuration, 3000-A can translate LAT or TCP traffic into X.25 packets for transmission over an X.25 PDN network. Packets are then translated back to LAT or TCP on the other side of the WAN.

```
Interface ethernet 0
ip address 1.0.0.2 255.255.0.0
!
! enable LAT on interface if concurrently routing (8.3 feature)
lat enable
!
interface serial 0
encapsulation X.25
! note that this is subaddress 3 of 11111
x25 address 111113
! The following parameters may depend on your network
x25 facility packetsize 512 512
x25 facility windowsize 7 7
no ip address

! "Other" Central Site Cisco Router Configuration
!
! Interface to WAN
interface serial 0
x25 address 11111
x25 route ^111113 interface serial 1
ip address ....
! Interface to 3000
interface serial 1
x25 route .* interface serial 0
no ip address

! Translate Configuration for 3000-A
!
no ip routing
! Note subaddress of subaddress 11111(3(3))
translate x25 1111133 tcp tcpdevice
translate lat TCP-B x25 3333301
translate lat tcp-device tcp tcp-device
! etc...any translate commands needed by application
```

Standalone LAT-to-TCP Translation Example

If you need a large number of local LAT-to-TCP translation sessions, you can set up the Cisco 3000 to use only an Ethernet port. This application allows 100 concurrent translation sessions. In the applications illustrated in Figure 5-11, any other Cisco router, including a Cisco 3000, can be used to interconnect network segments performing bridging or routing.
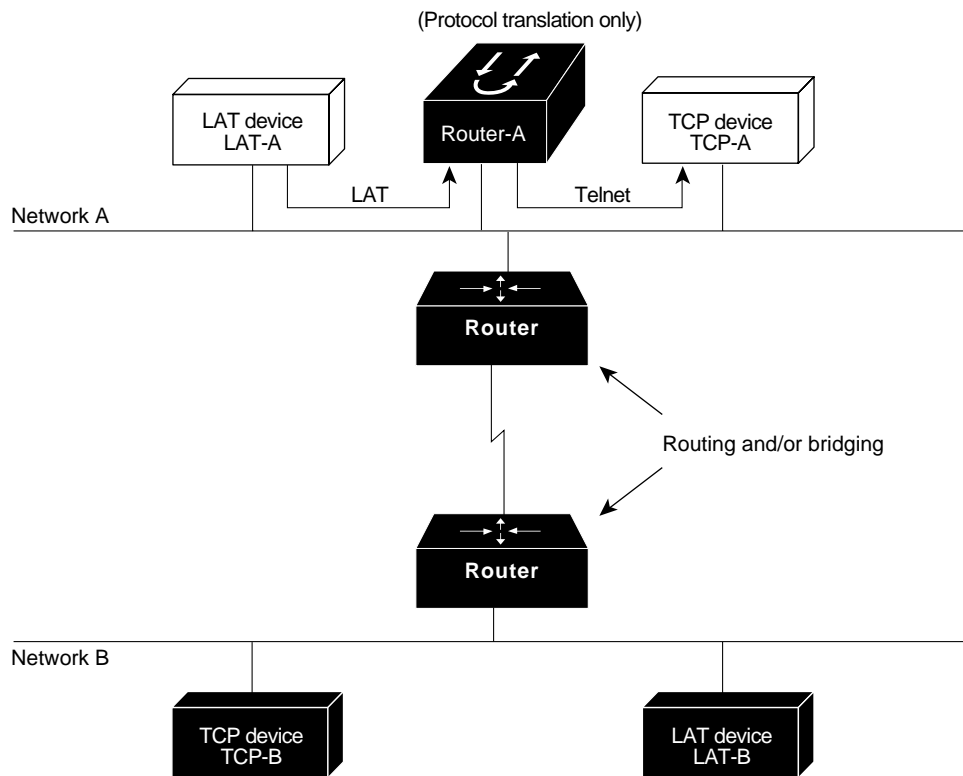
Figure 5-11 Cisco 3000 as Standalone Protocol Translator

```
! Translation Configuration for 3000-A only
!
interface ethernet 0
ip address 1.0.0.2 255.255.0.0
!
! enable LAT on this interface
lat enabled
!
interface serial 0
shutdown
no ip routing
default-gateway 1.0.0.100
!
translate lat TCP-A tcp TCP-A
translate lat TCP-B tcp TCP-B
translate tcp LAT-A lat lat-z
! etc...translate commands as required
```

X.29 Access List Example

The following example illustrates an X.29 access list. Incoming permit conditions are set for all IP hosts and LAT nodes that have specific characters in their names. All X.25 connections to a printer are denied. Outgoing connections are restricted.

```
!Permit all IP hosts and LAT nodes beginning with "VMS".
!Deny X.25 connections to the printer on line 5.
!
access-list 1 permit 0.0.0.0 255.255.255.255
lat access-list 1 permit ^VMS.*
x29 access-list 1 deny .*
!
line vty 5
access-class 1 in
!
!Permit outgoing connections for other lines.
!
!Permit IP access with the network 131.108
access-list 2 permit 131.108.0.0 0.0.255.255
!
!Permit LAT access to the boojum/snark complexes.
lat access-list 2 permit ^boojum$
lat access-list 2 permit ^snark$
!
!Permit X.25 connections to Infonet hosts only.
x29 access-list 2 permit ^31370
!
line vty 0 16
access-class 2 out
!
translate tcp 131.108.1.26 x25 5551234 access-class 2
```

X.3 Profile Example

The following profile script turns local edit mode on when the connection is made and establishes local echo and line termination upon receipt of a Return. The name "linemode" is used with the **translate** command to effect use of this script.

```
x29 profile linemode 2:1 3:2 15:1
translate tcp 131.108.1.26 x25 55551234 profile linemode
```

The X.3 PAD parameters are described in the appendix "X.3 PAD Parameters" later in this publication, and in the *Cisco Access Connection Guide*.

Increase the Number of Translation Sessions Example

The following example sets the number of protocol translation sessions to 120, whether routing is turned on or off:

```
line vty 120
```

Decrease the Number of Translation Sessions Example

The following example sets the number of protocol translation sessions to 10, whether routing is turned on or off:

```
no line vty 10
```