

Protocol Translation

This chapter describes how to make and 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, Telnet, X.25, TN3270, and XRemote configuration chapters in this publication.

Note Telnet is a remote terminal protocol that is part of the Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite. In the configuration syntax of the protocol translation software, Telnet is called TCP.

For more information about making connections and establishing translation sessions, see the *Communication Server and Protocol Translator Connection Guide*.

Cisco's Implementation of Protocol Translation

The protocol translator 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, LAT (Local Area Transport), and Telnet (a remote terminal protocol that is part of the Transmission Control Protocol/Internet Protocol [TCP/IP] protocol suite).

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.

The number of translation sessions supported depends on the hardware model and software version you are running. Table 1-1 summarizes the number of translation sessions supported in Software Releases 9.0 and later. Table 1-2 summarizes the number of sessions supported in Software Releases 8.3 and earlier.

Table 1-1 Translation Sessions Supported in Software Release 9.0 and Later

Hardware Model	Number of Translation Sessions	
	With Routing	Without Routing
CPT	No routing	100
IGS	64	100
3102	64	100
3104	64	100

Table 1-2 Translation Sessions Supported in Software Release 8.3 and Earlier

Hardware Model	Number of Translation Sessions	
	With Routing	Without Routing
CPT	No routing	100
IGS	32	100

Protocol Translation Configuration Task List

The following choices are available for protocol translation:

- The one-step method
- The two-step method

Configure One-Step Protocol Translation

The one-step method 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, use the **translate** command to 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

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 <i>protocol incoming-address [in-options] protocol outgoing-address [out-options] [global-options]</i>

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, you 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 Public Data Network [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*, “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 *Communication Server and Protocol Translator Connection Guide*.

Protocol Translation Application Configuration Examples

This section provides several examples of configuring protocol translation applications. The environment used in these examples is illustrated in Figure 1-1. The application environment described focuses on IGS implementations. However, in applications featuring only protocol translation, any of the protocol translator products can be used.

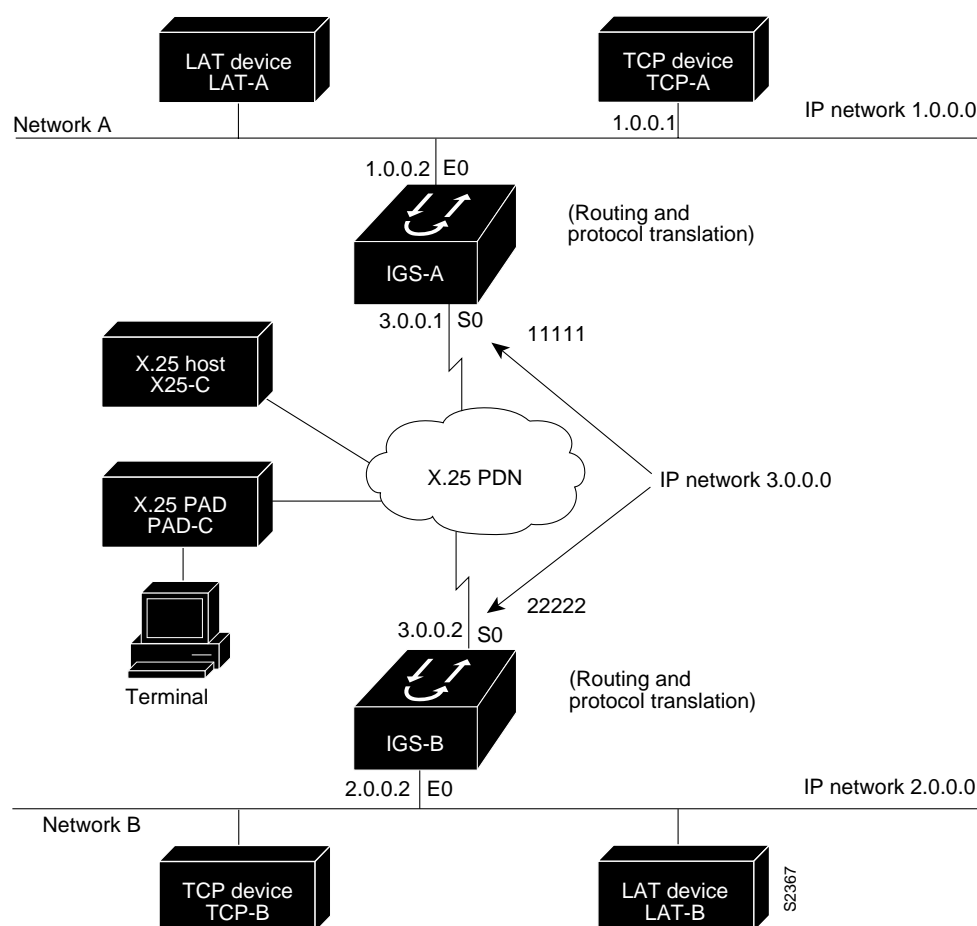
The examples are:

- Basic configuration example
- Local LAT-to-TCP example
- LAT-to-X.25 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

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 1-1).

Figure 1-1 Summary Diagram Showing IGS Systems as Protocol Translators



Basic Configuration Example

The following example illustrates the basic global configuration commands and interface configuration commands for setting up IGS-A (connected to Network A) and IGS-B (connected to Network B), as illustrated in Figure 1-1. Refer to the “Configuring X.25” and “Configuring LAT” chapters for more information configuring X.25 and LAT, respectively.

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 IGS-A

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

```
!*****IGS ROUTER A - GENERAL CONFIGURATION*****
!
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 window size 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 (IGS-A) if connecting via 2-step method
! (for connecting directly to a specific protocol translator)
lat service IGS-A enable
!
! Set up some IP host names/addresses
ip host IGS-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 IGS-B 3.0.0.2 2.0.0.2
!
```

Configuration for Device IGS-B

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

```
!*****IGS ROUTER B - GENERAL CONFIGURATION*****
!
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 window size 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 1.0.0.2 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 (IGS-B) if connecting via 2-step method
! (for connecting directly to a specific Protocol Translator)
lat service IGS-B enable
!
! Set up some IP host names/addresses
ip host IGS-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 IGS-B 2.0.0.2 3.0.0.2
!
```

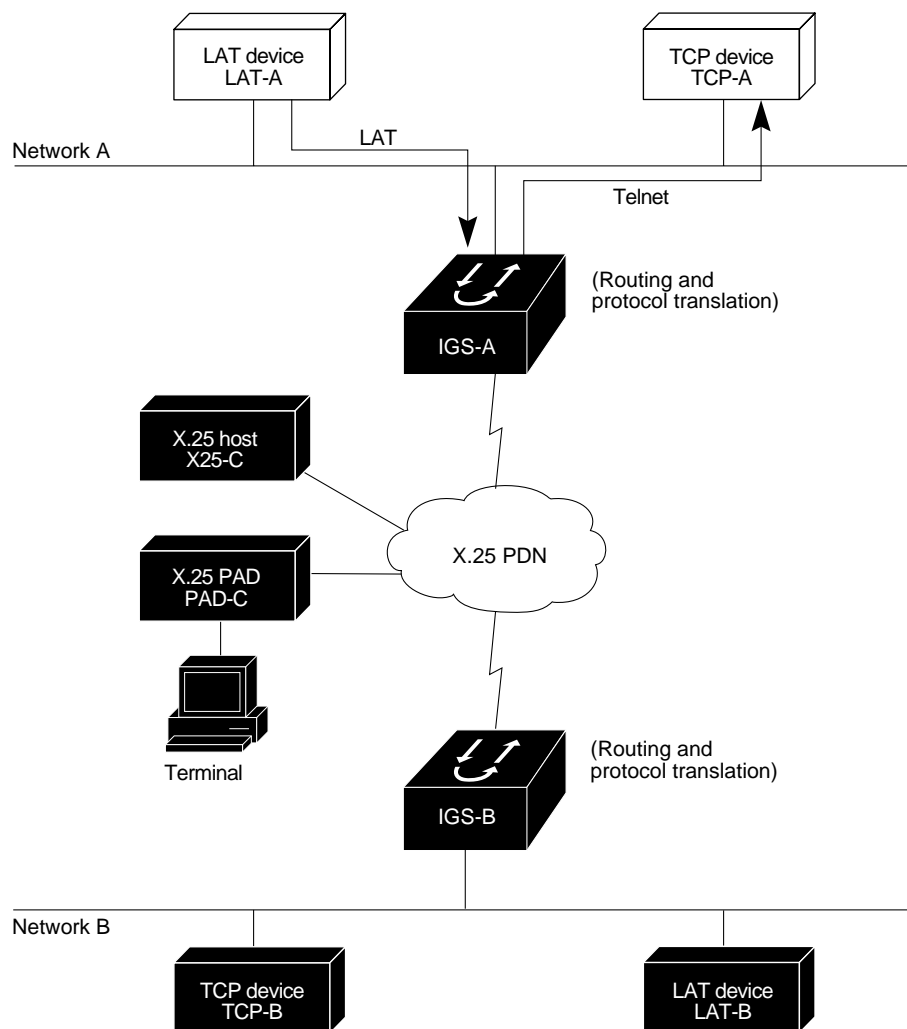
Note You can specify IP host names used to identify specific hosts by explicitly using the **ip host** global configuration command or by using Domain Name System (DNS) facilities.

Local LAT-to-TCP Example

The IGS can translate between LAT and TCP (Telnet) traffic to allow communication among resources in these protocol environments. In Figure 1-2, 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 1-2 Local LAT-to-TCP Translation



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

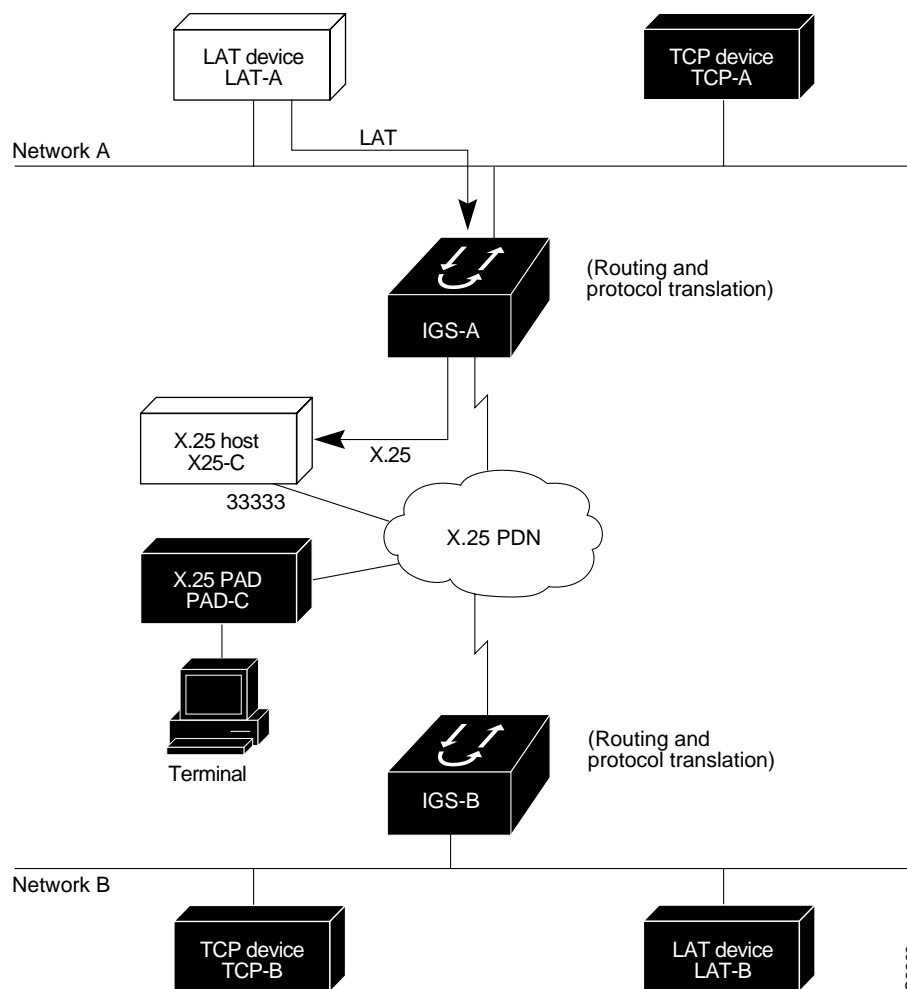
```
! Translate LAT connections to TCP for connectivity to TCP-A
translate lat TCP-A tcp TCP-A
! Optional additional commands
lat service TCP-A ident Protocol Translation to TCP-A
```

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 IGS protocol translation option allows LAT devices to communicate with X.25 hosts through an X.25 PDN. In the application illustrated in Figure 1-3, 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 1-3 LAT-to-X.25 Host Translation



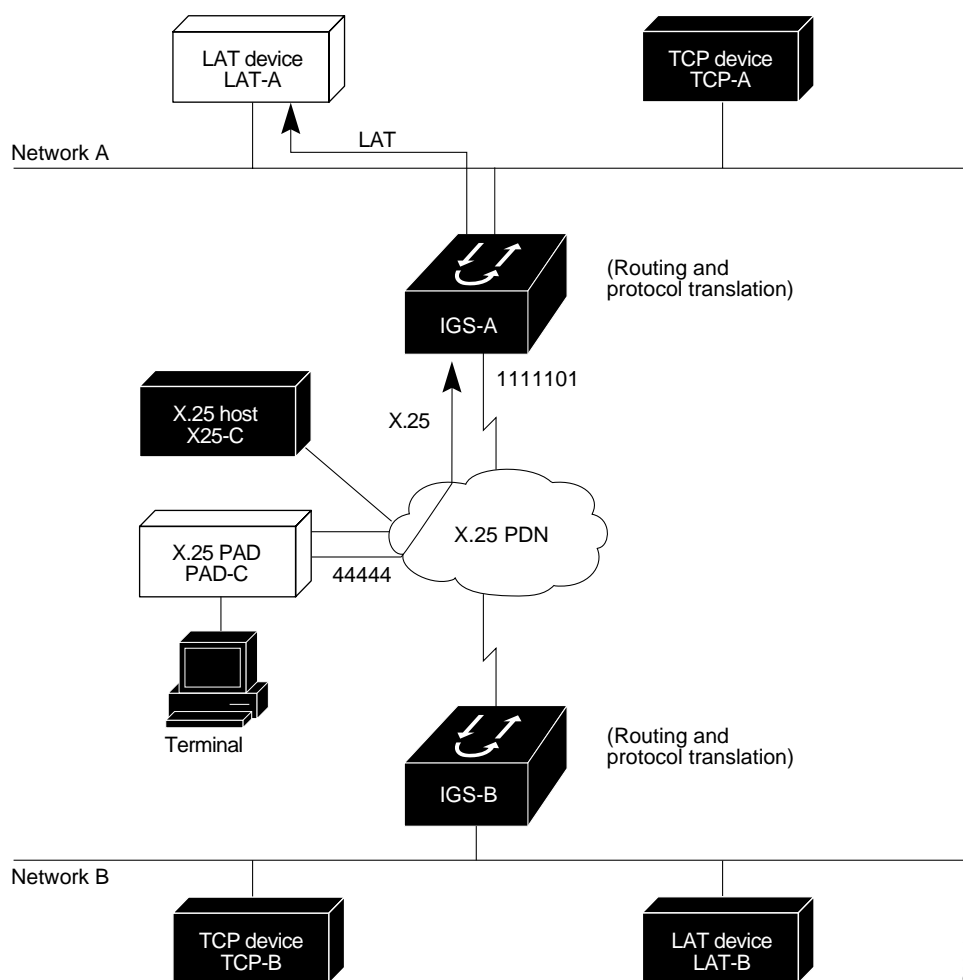
The following example illustrates how to use the **translate** global configuration command to translate from LAT to X.25. It is applied to IGS-A. This example sets up reverse charging for connections, which causes the protocol translator to instruct the PDN to charge the destination for the connection. 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

IGS protocol translation allows terminals connected to X.25 PADs to communicate with LAT devices on a remote LAN (Figure 1-4). X.25 PAD terminals make a call using an X.121 address, which is translated to a LAT address. To the PAD terminal user, the connection appears to be a direct connection to a host on the X.25 PDN. The IGS also supports X.29 access lists, which allow you to restrict LAN resources (LAT or TCP) available to the PAD user.

Figure 1-4 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 IGS-A. The configuration example includes an access list that limits remote LAT access through IGS-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 Appendix C, “Regular Expressions,” in the *Protocol Translator Command Reference* 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 IGS-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.

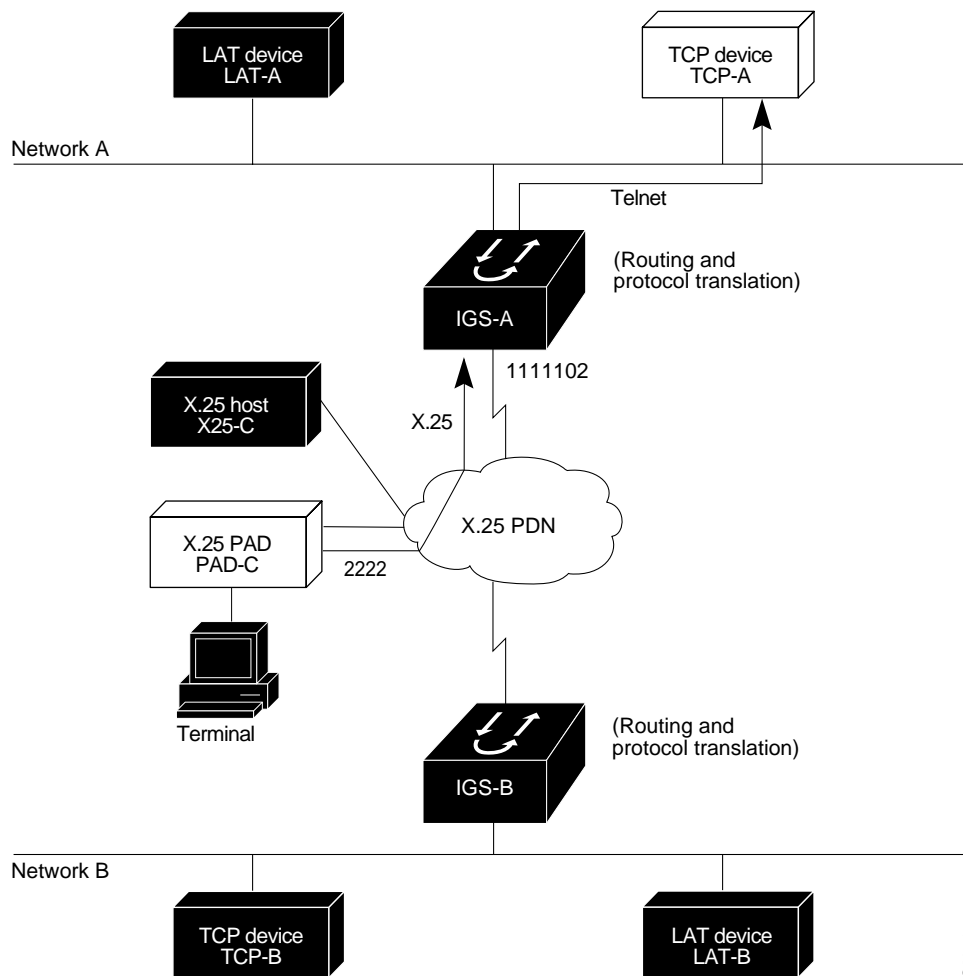
If you define multiple X.25 translate commands, each must have a unique X.121 address. It is also important that the Consultative Committee for International Telegraph and Telephone (CCITT) protocol used in transferring packets match among X.121 addresses. This is specified in the protocol identification 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 IGS-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 must match the number to which your PDN is sending data. 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 1-5) is analogous to the preceding X.25 PAD to LAT example. Instead of translating to LAT, the IGS-A configuration includes a statement to translate to Telnet (called TCP in the configuration syntax of protocol translation software). 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 1-5 X.25 PAD-to-TCP Translation



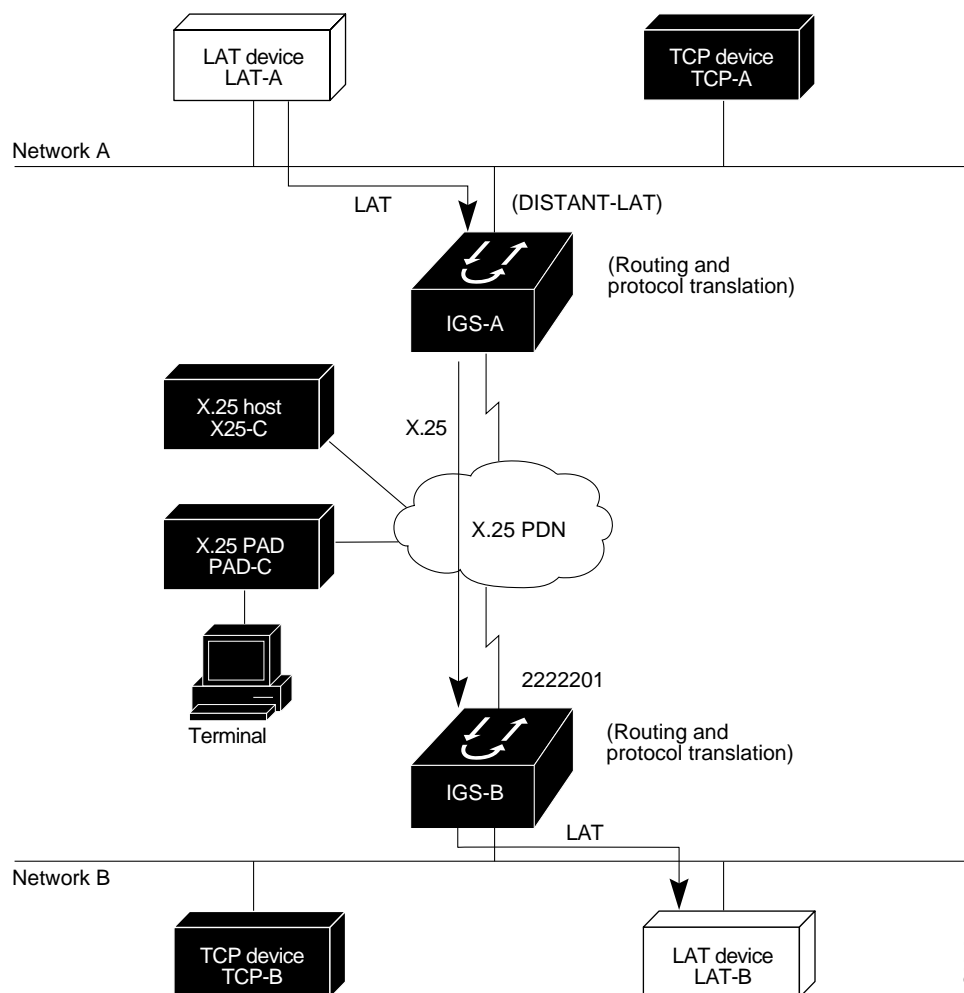
The following example illustrates how to use the **translate** global configuration command to translate from an X.25 PAD to a TCP device on Network A. It is applied to IGS-A.

```
! Set up translation
translate x25 1111102 tcp TCP-A
```

LAT-to-LAT (via X.25 Translation) Example

Protocol translators can be used to provide transparent connectivity between LAT devices on different networks via an X.25 PDN. In Figure 1-6, which illustrates this application, the LAT device on Network A (LAT-A) first makes a virtual connection to the IGS on Network A using the LAT protocol. The IGS translates then the LAT packets into X.25 packets and sends them through the X.25 network to the IGS on Network B. This second IGS 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 1-6 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 IGS-A, which is on Network A
translate lat DISTANT-LAT x25 2222201
!

! Translate X.25 to LAT on IGS-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 IGS-A. When a user on device LAT-A attempts to connect to LAT-B, the target specified in the **connect** command is DISTANT-LAT. (The **connect** command is described in the *Communication Server and Protocol Translator Connection Guide*.)

In the **translate** command for IGS-B, the name of the LAT service on the target host (LAT-B) is LAT-B. IGS-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 LAT-B on Network B is attempted.

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

To configure IGS-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 1-6):

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

! Translate X.25 to LAT on IGS-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 IGS-A and IGS-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.

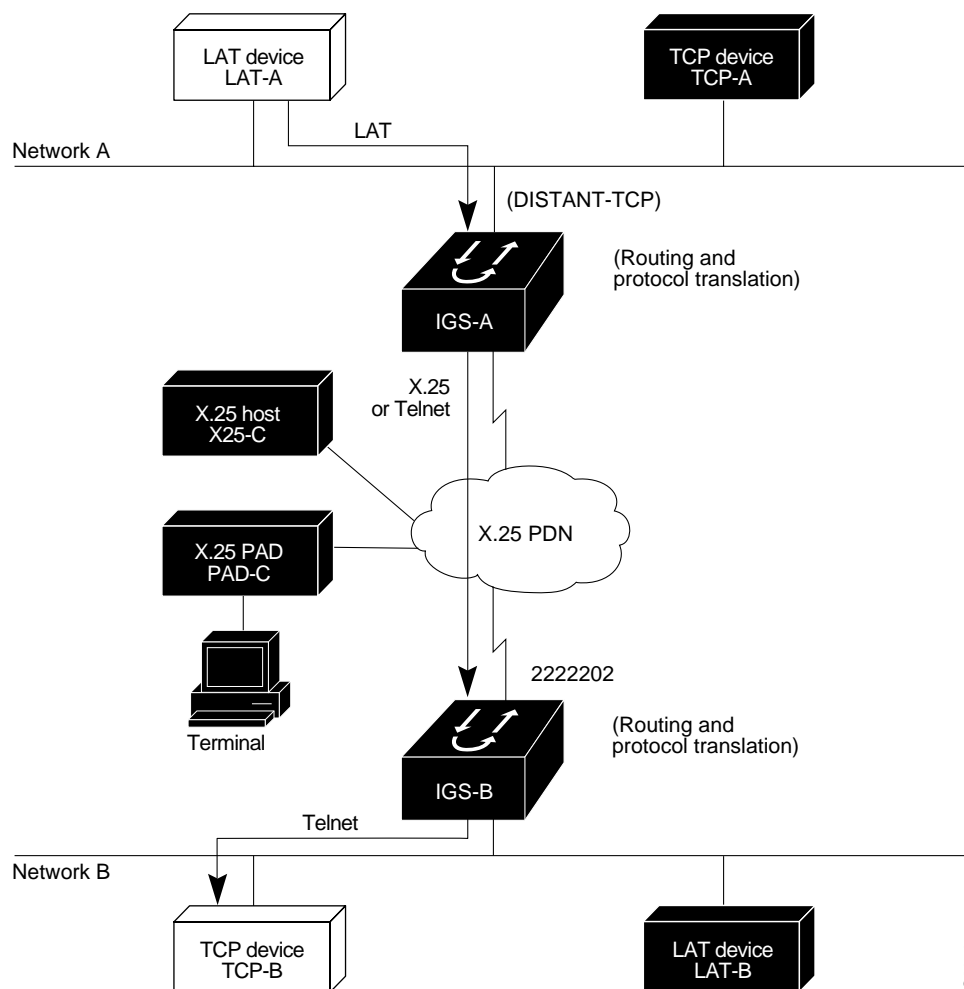
LAT-to-TCP (via X.25) Example

Protocol translators can be used to provide transparent connectivity between LAT and TCP devices on different networks via an X.25 PDN. In the application illustrated in Figure 1-7, 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 Telnet packets to be sent out on the X.25 PDN.

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

If the traffic is translated to Telnet by IGS-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 Telnet over X.25 has 45 bytes of header information.

Figure 1-7 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 IGS-A) and from X.25 to Telnet (on IGS-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 IGS-A and IGS-B separately.

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

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

In the **translate** command for IGS-A, DISTANT-TCP defines a LAT service name for IGS-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 IGS-B, the Telnet service on the target host (TCP-B) is TCP-B. IGS-B translates the incoming X.25 packets from 2222202 to Telnet 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
```

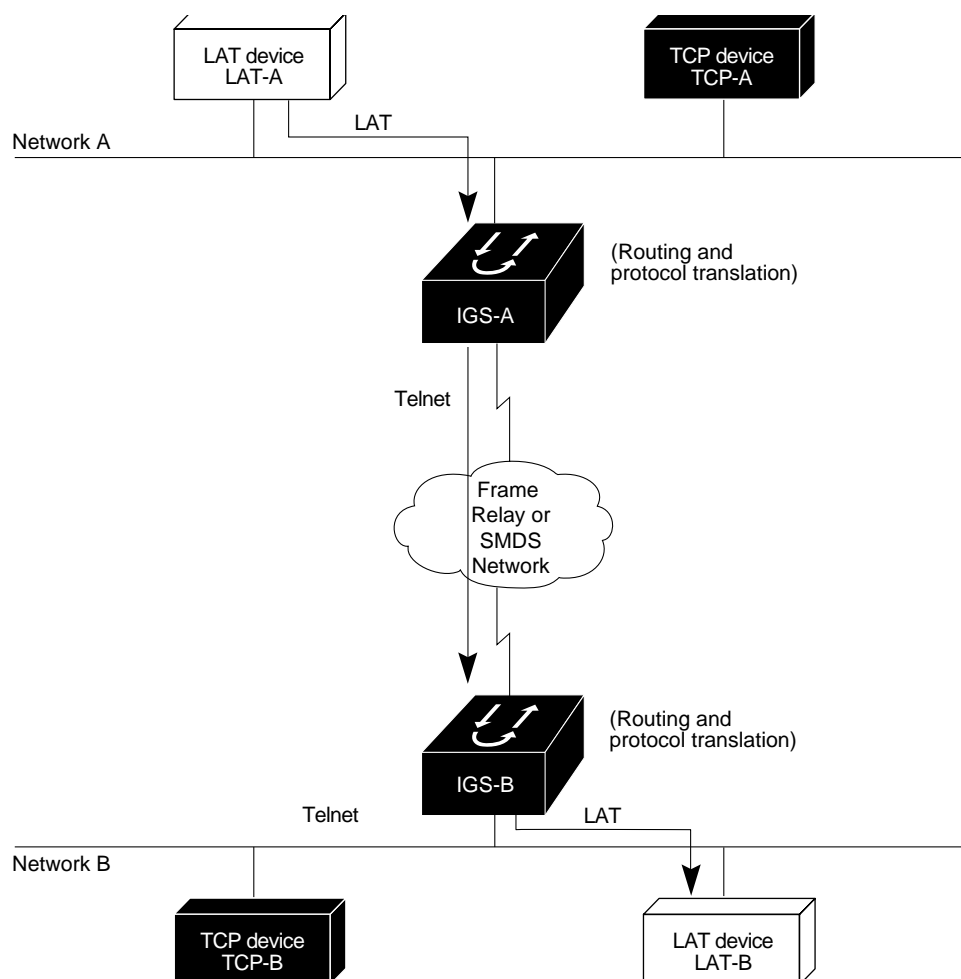
Note You can use the same name (for example, “TCP-B”) in the **translate** command for both IGS-A and IGS-B, because each protocol translator operates independently. However, this symmetry is not required. The key is the common X.121 address used in both **translate** commands. If you prefer to have 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 (Telnet). The TCP traffic is routed over the Frame Relay network and then translated back to LAT on the IGS on Network B. See Figure 1-8.

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 1-8 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 IGS is routing encapsulated packets translated from LAT to TCP (Telnet) 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 IGS-A, which is on Network A
translate lat DISTANT-LAT tcp IGS-B
!

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

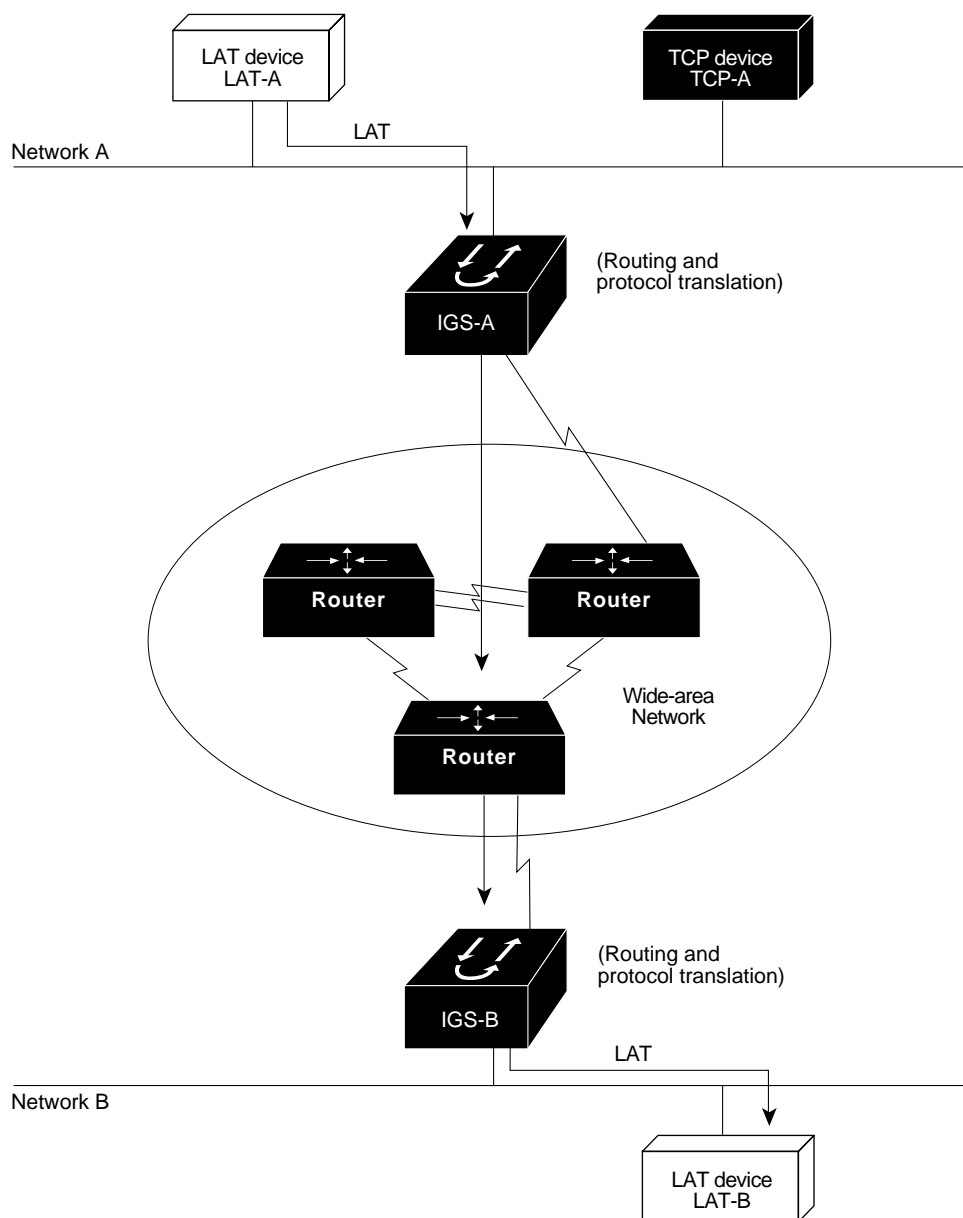
Note You can use the same name (for example, “LAT-B”) in the **translate** command for both IGS-A and IGS-B, because each protocol translator operates independently. However, this symmetry is not required. The key is the common IP name used in both **translate** commands.

LAT-to-LAT (over an IP WAN) Example

The IGS with protocol translation can be used to connect LAT devices over a wide-area network (WAN) backbone that only allows routable protocols (see Figure 1-9). This configuration exists when LAT networks either are isolated or are on their own internetwork.

With the IGS, LAT traffic can be translated to TCP and then routed on the WAN as Telnet traffic. The LAT connections stay local between the LAT device and the IGS. 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 1-9 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 IGS is routing encapsulated packets translated from LAT to TCP (Telnet) over the WAN. Packets are then translated back to LAT on the other side of the WAN. Example translation configurations for both IGS-A and IGS-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 IGS-A, which is on Network A
translate lat DISTANT-LAT tcp IGS-A
!

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

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

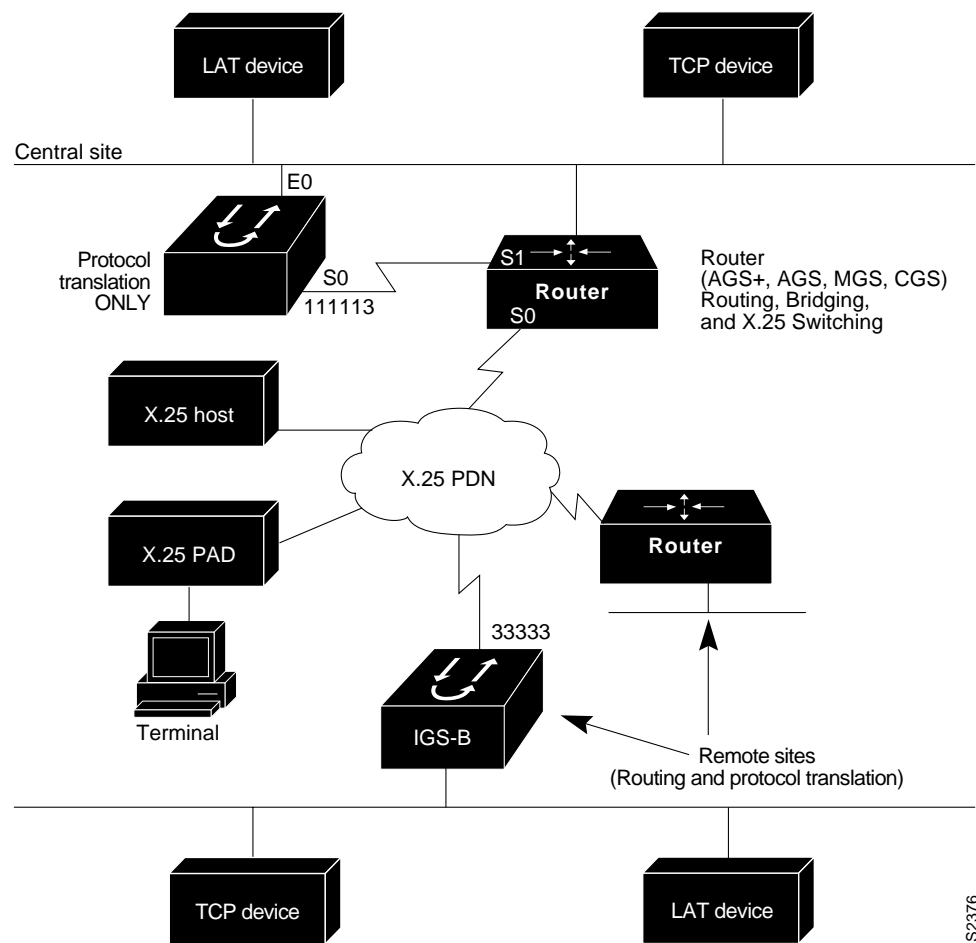
Central Site Protocol Translation Example

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

To support this application, an IGS is directly connected back-to-back (Figure 1-10) to another Cisco router (for example, AGS+, AGS, MGS, or CGS). This second router acts as an X.25 switch, by sending X.25 packets to the IGS while concurrently routing and bridging other protocols.

Because the IGS is operating only as a protocol translator, it can support 100 concurrent protocol translation sessions compared to the 64 that are supported when the IGS is routing, bridging, and protocol translating.

Figure 1-10 Central Site Protocol Translation Example



The following example illustrates how to configure IGS and AGS+ systems 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 IGS while concurrently routing and bridging other protocols.

The following example also illustrates how to use the **translate** global configuration command to translate LAT, TCP, and X.25 over X.25 WAN media. In this configuration, IGS-A can translate LAT or TCP (Telnet) 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.

```

! Basic Configuration for IGS-A, which is on Network A
!
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 packetize 512 512
x25 facility windowize 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 IGS
interface serial 1
x25 route .* interface serial 0
no ip address
!

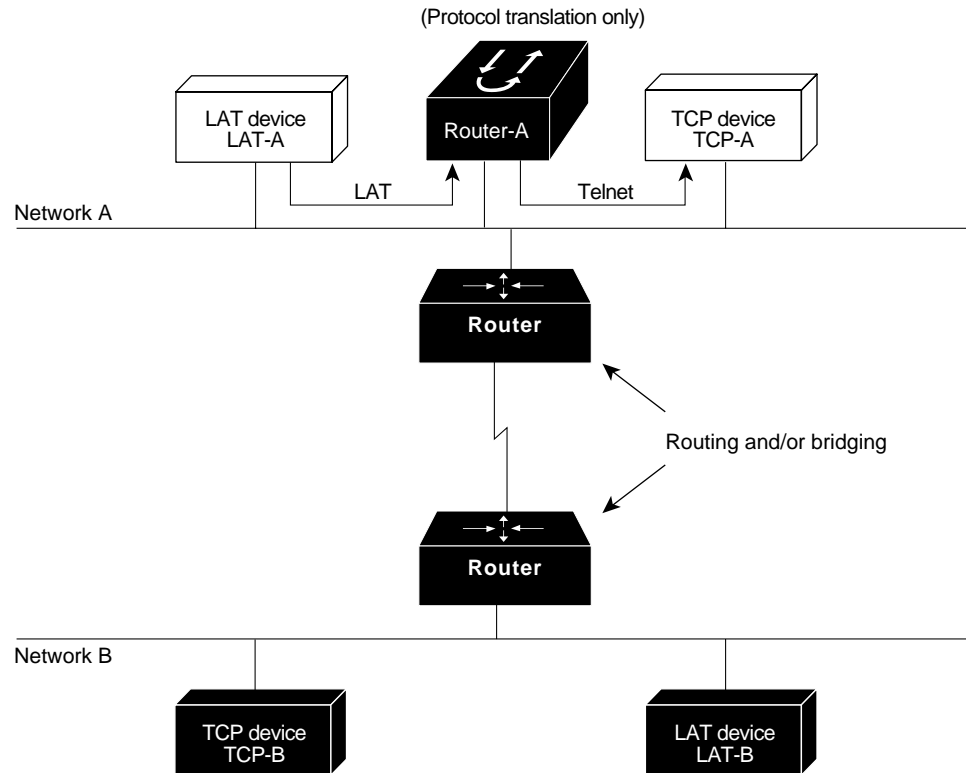
! Translate Configuration for IGS-A
!
no ip routing
! Note subaddress of subaddress 11111(3(3))
translate x25 1111133 tcp tcpdevice
translate lat TCP-B x25 3333301
translate lat tcp-device tcp tcp-device
! etc...any translate commands needed by application
!

```

Standalone LAT-to-TCP Translation Example

If you need a large number of local LAT-to-TCP translation sessions, you can set up the IGS to use only an Ethernet port. This application allows 100 concurrent translation sessions. In the applications illustrated in Figure 1-11, any other Cisco router, including an IGS, can be used to interconnect network segments performing bridging or routing.

Figure 1-11 IGS as Standalone Protocol Translator



```

! Translation Configuration for IGS-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
    
```

S2377