

Chapter 15

Routing ISO CLNS

15

This chapter describes Cisco Systems' implementation of the International Organization for Standardization (ISO) Connectionless Network Services (CLNS) protocol, which is a standard for the Network Layer of the Open Systems Interconnection (OSI) model. This chapter includes a section reviewing ISO terminology, as well as sections focusing on these tasks and topics:

- An overview of CLNS addresses and the basics of the configuration process.
- How to configure dynamic and static routing.

For both static and dynamic routing, you have choices of intra-domain and inter-domain routing, locally-sourced packets and various lower-layer protocols.

This chapter also explains how Cisco's implementation differs from the standards (where they do), and how to optimize and fine-tune your CLNS-based internet. Configuration examples are grouped in a separate section, with configuration command lists and illustrations.

Cisco's Implementation of ISO CLNS

The Cisco routing software supports packet forwarding and routing for ISO CLNS for networks using a variety of lower layer protocols: Ethernet, Token Ring, FDDI, and serial.

You can use CLNS routers on serial interfaces with either HDLC, LAPB, or X.25 encapsulation. To use HDLC encapsulation, you must have a Cisco router at both ends of the link. If you use X.25 encapsulation, you must manually enter the NSAP-to-X.121 mapping. The LAPB and X.25 encapsulations will interoperate with other vendors.

In addition, the Cisco CLNS implementation is compliant with the Government Open Systems Interconnection Profile (GOSIP) Version 2.

- As part of its CLNS support, Cisco routers fully support these ISO and ANSI standards:
- ISO 9542—Documents the End System-Intermediate System (ES-IS) routing exchange protocol.
- ISO 8473—Documents the ISO Connectionless Network Protocol (CLNP).
- ISO 8348/Ad2—Documents Network Service Access Points (NSAPs).

Cisco supports the Interior Gateway Routing Protocol (IGRP) for dynamic routing of ISO CLNS. IGRP discovers routes to all End Systems (hosts) in the routing domain, provides dynamic recovery from failures, determines if there are alternate paths available, and supports load balancing across equivalent routes.

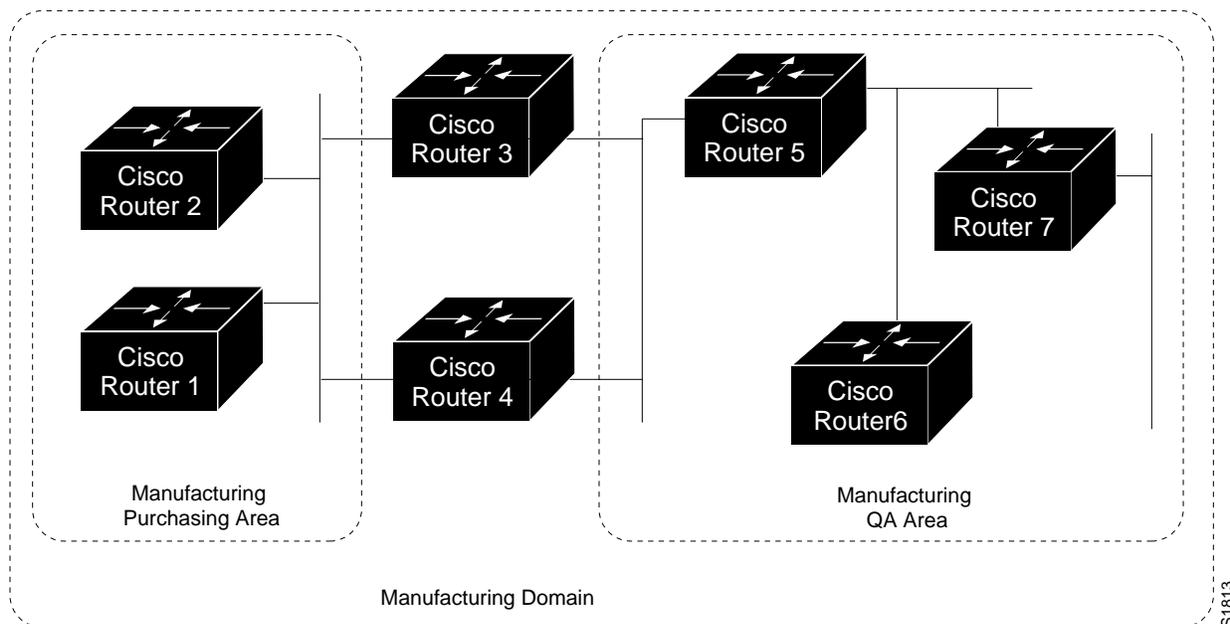
In addition, Cisco supports static routing for addresses (NSAPs) that do not conform to the addressing constraints. You must also choose static routing if you want to mix Cisco and nonCisco routers.

ISO Routing Terminology

The ISO network architecture uses terminology and concepts that differ from IP networks. In other network architectures, CLNS would be known as a *datagram service*; in the TCP/IP architecture, for example, datagram service is provided by IP.

The lowest level of the routing hierarchy is the *area*—a set of networks connected by routers (see Figure 1-1). The network itself has no separate identity, no network number; instead, the next addressable entity above the End System or Intermediate System (router) is the area.

Figure 1-1 ISO CLNS Areas



Areas are connected to other areas to form *routing domains*. Each domain is a separately administered region, similar in concept to an autonomous system in IP networks.

Some intermediate systems keep track of how to communicate with all of the End Systems in their area, and thereby function as *Level 1 routers*. Other intermediate systems keep track of how to communicate with other areas in the domain, functioning as *Level 2 routers*. Cisco routers are always Level 1 and Level 2 routers.

End Systems communicate with Intermediate Systems using the ES-IS protocol. Level 1 and Level 2 Intermediate Systems communicate with each other using the Cisco ISO IGRP protocol.

Routing across domains (inter-domain routing) may be done either statically or dynamically.

Configuring CLNS Routing

Follow these steps to configure your router for CLNS routing:

- Step 1:** Enable CLNS routing with the **clns routing** global configuration command.
- Step 2:** Create dynamic or static routing processes with the **clns router** global configuration command and either the **igrp** or **static** keyword.
- Step 3:** For each interface, specify which CLNS routers should be active with the **clns router** interface subcommand.

These steps enable CLNS routing. Optional commands are also available for customizing the routing environment, and you may follow these steps to complete configuration:

- Step 4:** Redistribute routing information.
- Step 5:** Map NSAP addresses to media addresses.
- Step 6:** Adjust ES-IS parameters, as necessary.

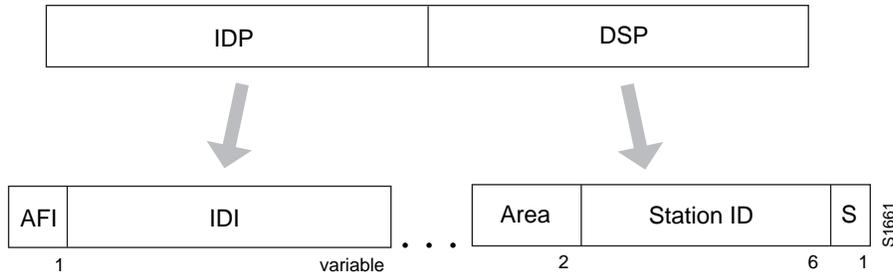
Each task is described in the following sections, and are followed by descriptions of the EXEC commands to maintain, monitor and debug the ISO CLNS network. Summaries of the global configuration commands and interface subcommands described in these sections appear at the end of this chapter.

CLNS Addresses

Addresses in the ISO network architecture are referred to as Network Entity Titles (NETs) and Network Service Access Points (NSAPs). Each node in an ISO network has one or more NETs. In addition, each node has many NSAPs. Each NSAP differs from one of the NETs for that node in only the last byte. This byte is called the *selector byte*, see . Its function is similar to the port number in other protocol suites.

Cisco's implementation supports all NSAPs that are defined by ISO 8348/Ad2; however, Cisco provides dynamic routing only for those NSAPs that conform to the address constraints defined in the draft proposal of ISO IS-IS (ANSI DP 10589). illustrates conforming NSAP.

Figure 1-2 Conforming NSAP Addressing Structure



The NSAP address is formed by the Initial Domain Part (IDP), followed by the Domain Specific Part (DSP).

The IDP is made up of a one-byte Authority and Format Identifier (AFI), and a variable length Initial Domain Identifier (IDI).

The DSP is a nine-byte structure that contains a two-byte Area identifier, six-byte station ID, and one-byte selector (S) field.

Cisco's CLNS implementation interprets the bytes from the AFI up to the Area field in the DSP as a *domain identifier*. The Area field specifies the *area*, and the station ID specifies the *station*.

The domain address uniquely identifies the routing domain. Within each routing domain you can set up one or more areas. The area address uniquely identifies the area.

Addressing Rules

All NSAPs must obey the following constraints:

- No two nodes may have addresses with the same NET; that is, addresses that match all but the selector (S) field in the DSP.
- There must be at least nine bytes in the DSP portion of the address.
- No two nodes residing within the same area may have addresses in which the station ID fields are the same.

Examples:

Following are examples of OSInet and GOSIP network addresses.

OSInet:

```
47.0004.004D.0003.0000.0C00.62E6.00
|      Domain|Area|      Station ID| S|
```

GOSIP Network:

```
47.0005.80.FFFF.0000.00.FFFF.0004.0000.0C00.62E6.00
|Area|      Station ID| S|
```

Enabling CLNS Routing

Conceptually, each End System (ES) lives in one area. It discovers the nearest Level 1 IS (router) by listening to ES-IS packets. Each ES must be able to communicate directly with a Level 1 IS in its area.

When an ES wants to communicate with another ES, it sends the packet to the Level 1 IS for its area. The IS will look up the destination NSAP and forward the packet along the best route. If the destination NSAP is for an ES in another area, the Level 1 IS will send the packet to the nearest Level 2 IS. The Level 2 IS will forward the packet along the best path for the destination area until it gets to a Level 2 IS which is in the destination area. This IS will forward the packet along the best path inside the area until it is delivered to the destination ES.

The configuration process begins with enabling CLNS routing. You enable routing of CLNS packets using the **clns routing** global configuration command. The full syntax for this command follows.

```
clns routing  
no clns routing
```

Use the **no clns routing** command to disable CLNS routing.

You need to decide now if you want to use static or dynamic routing. You must use static routing if you are inter-operating with nonCisco routers. In addition, you may want to use static routing over an X.25 link.

You must also enable ISO CLNS for each interface. This is automatically done when configuring IS-IS or ISO-IGRP routing on an interface; however, if you do not intend to perform any dynamic routing on an interface, but intend to pass ISO CLNS packet traffic to end systems, use the **clns enable** interface subcommand. The syntax for this command is as follows:

```
clns enable  
no clns enable
```

Use the **no clns enable** command to disable ISO CLNS on a particular interface.

Configuring CLNS Static Routing

Static routing is used when it is not possible or desirable to use dynamic routing: when routing with both Cisco and nonCisco routers or when using an X.25 connection. Of

course, an interface that is configured for static routing cannot reroute around failed links or load share.

Routes are entered by specifying pairs (NSAP-prefix, next-hop-NET). NET is a Network Entity Title, similar in function to an NSAP. In the routing table, the best match means the longest NSAP-prefix entry that matches the beginning of the destination NSAP. In the following static routing table the next-hop-NETs are listed for completeness, but are not necessary to understand the routing algorithm. Table 1-2 offers examples of how the routing entries in Table 1-1 can be applied to various NSAPs.

Table 1-1 Sample Routing Table Entries

Entry #	NSAP Prefix	Next Hop NET
1	47.0005.000c.0001	47.0005.000c.0001.0000.1234.01
2	47.0004	47.0005.000c.0002.0000.0231.01
3	47.0005.0003	47.0005.000c.0001.0000.1234.01
4	47.0005.000c	47.0005.000c.0004.0000.0011.01
5	47.0005	47.0005.000c.0002.0000.0231.01

Table 1-2 Hierarchical Routing Examples

Datagram Destination NSAP	Table entry # used
47.0005.000c.0001.0000.3456.01	1
47.0005.000c.0001.6789.2345.01	1
47.0004.1234.1234.1234.1234.01	2
47.0005.0003.4321.4321.4321.01	3
47.0005.000c.0004.5678.5678.01	4
47.0005.000c.0005.3456.3456.01	4
47.0005.0023.9876.9876.9876.01	5

Octet boundaries must be used for the internal boundaries of NSAPs and NETs.

You enter a specific static route by using the **clns route** global configuration command. The full syntax of this command follows:

```
clns route nsap-prefix next-hop-net
no clns route nsap-prefix
```

NSAPs that start with *nsap-prefix* are forwarded to *next-hop-net*.

This variation of the **clns route** command uses the **discard** keyword to explicitly tell a router to discard packets with the specified *nsap-prefix* (full syntax listed).

```
clns route nsap-prefix discard
no clns route nsap-prefix
```

Example:

This example sets a static route for a router.

```
clns route 47.0004.000c 47.0005.0001.0000.0001.0000.00
```

Defining Areas

You must enter the **clns router** global configuration command for each area the router is in. The command has this syntax:

```
clns router static area-tag NET net  
no clns router static area-tag
```

The keyword **static** specifies that the router will use statically entered routes. The argument *area-tag* defines a meaningful name for an area. For example, you could define an area named *Finance* for the Finance department, and another area named *Marketing* for the Marketing department. Creating a name for an area means that you use names when configuring routing in the area, instead of having to enter the NET. Following the keyword **NET** you specify the Network Entity Title.

Use the **no clns router static** command with the appropriate arguments to remove the areas.

Example:

In the following example, you are configuring a router in the area *sales*:

```
clns router static sales NET 47.0005.0001.0000.0001.0000.00
```

Each area an interface is in must be specified with the **clns router** interface subcommand. The full syntax of this command is listed next:

```
clns router static area-tag  
no clns router static area-tag
```

The keyword **static** specifies static routing. The argument *area-tag* is the tag defined for the NET using the **clns router** global configuration command. Use the **no clns router static** command to remove the definition.

Example:

In this example, an interface is being configured for static routing in the marketing area:

```
clns router static marketing
```

Configuring CLNS Over X.25

X.25 is not a broadcast medium, and therefore ES-IS is not used to automatically advertise and record NSAP/NET (protocol address) to SNPA (media address) mappings. Operation of CLNS over X.25 requires that this mapping information be statically entered, by using the **clns is-neighbor** and/or the **clns es-neighbor** interface subcommands.

Configuring a serial line to use CLNS over X.25 requires configuring the general X.25 information and the CLNS-specific information. The general X.25 information must be configured first. Then, the **clns is-neighbor** and **clns es-neighbor** interface subcommands are issued to list all the Intermediate and End Systems that will be used. Full command syntax for these commands follows.

```
clns es-neighbor nsap snpa [X.25-facilities-info]  
no clns es-neighbor nsap
```

```
clns is-neighbor nsap snpa [X.25-facilities-info]  
no clns is-neighbor nsap
```

In this case, the Subnet Points of Attachment (SNPAs) are the X.25 network addresses (X.121 addresses). These are usually assigned by the X.25 network provider. Use the argument *X.25-facilities-info* to specify nondefault packet and window size, reverse charge information, and so on.

Example 1:

This command maps NSAP 47.0004.004d.3132.3334.3536.00 to X.121 address 310117.

```
clns es-neighbor 47.0004.004d.3132.3334.3536.00 310117
```

Only one **es-neighbor** or one **is-neighbor** entry can be made for each X.121 address.

The X.25 facilities information that can be specified is exactly the same as in the **x25 map** subcommand described in the section “Setting Address Mappings” in the chapter “Configuring Packet-Switched Software.”

You can specify the following information:

- Packet window size (send and receive)
- Packet size (send and receive)
- Reverse charges requested
- Reverse charges accepted
- Closed user group
- Maximum number of virtual circuits (VCs) to open
- Broadcasts

Example 2:

Following is a more complicated example that specifies nondefault packet and window size:

```
clns is-neighbor 47.0004.0021.0001.0000.0000.00 3101 windowsize 7 7 packetsize  
512 512
```

Note: All of this configuration command must be given on one line.

The system does not accept an X.25 call unless the source of the call has been configured with these commands. The system will not accept a call requesting reverse charges unless the keyword **accept-reverse** is used as follows.

```
clns is-neighbor 47.0005.0000.0001.0000.00 310120249 accept-reverse
```

All of the information that is entered using the **clns is-neighbor** and **clns es-neighbor** subcommands actually goes into two places in the system: the ES-IS table, and the X.25 map table. The ES-IS table stores only the NSAP/NET and X.121 address information.

The X.25 map table stores this information but, in addition, stores the facility information. If a virtual circuit (VC) has been established, the logical channel numbers (LCNs) in use are shown.

A configuration example using an X.25 link is included in the section “CLNS Configuration Examples.”

Configuring CLNS Dynamic Routing

Use the commands described in this section to configure CLNS dynamic routing. In the previous section, you turned on CLNS processing in the router. Now you need to identify the area the router will work in to create a routing process. You do this with the **clns router** global configuration command.

```
clns router igrp area-tag NET net
```

The keyword **igrp** specifies dynamic routing using the IGRP protocol. You can specify up to ten IGRP processes. The argument *area-tag* defines a meaningful name for an area. For example, you could define an area named *Finance* for the Finance department, and another area named *Marketing* for the Marketing department. Creating a name for an area means that you use names when configuring routing in the area, instead of having to enter the NET. Follow the keyword **NET** with the Network Entity Title.

Example:

In the following example, a router with its NET is specified in the *Marketing* area. (The command must be typed on one line.)

```
clns router igrp marketing NET 47.0004.0021.0001.0000.0000.00
```

Each interface in an area must also have a **clns router** interface subcommand entry. Following is the full command syntax for this command:

```
clns router igrp area-tag [level2]  
no clns router igrp area-tag
```

The argument *area-tag* is the tag defined for the NET using the **clns router** global configuration command above.

If you want this interface to advertise Level 2 information only, use the **level2** keyword. The purpose of this option is to reduce the amount of router to router traffic by specifying the

routers to only send out Level 2 routing updates on certain interfaces. The Level 1 information will not be passed between the routers with the Level 2 option set.

Use the **no clns router igrp** command with the appropriate area tag to disable the CLNS routing protocol on the interface.

Example:

In the following example, the interface will advertise Level 2 information only on this interface serial 0:

```
interface serial 0
  clns router igrp marketing level2
```

Inter-Domain Dynamic Routing

A router may be configured to do inter-domain dynamic routing by putting it into two domains and configuring the router to redistribute the routing information between the domains. Router configured this way are referred to as *border* routers.

When configuring inter-domain dynamic routing, keep in mind that routers that *are* border routers must have at least two ISO CLNS routing processes defined in two different domains.

If you have a router which is in two routing domains, you may want to redistribute routing information between the two domains. This is done with this variation of the **clns router igrp** global configuration command. The following lists the full command syntax:

```
clns router igrp tag1 redistribute domain tag2
no clns router igrp tag1 redistribute domain tag2
```

The keywords **redistribute domain** enable the redistribution. The arguments *tag2* and *tag1* are the defined tags for the areas in which the routing information is to be redistributed. When enabled, this command causes the area defined by *tag1* to redistribute information learned from domain *tag2*.

The **no clns router igrp** command with the appropriate arguments and keywords stops redistribution.

Example:

In the example below, information learned in the area named *MktDevelop* will be redistributed to the *CustAdv* areas.

```
clns router igrp custadv redistribute domain mktdevelop
```

The section “CLNS Configuration Examples” later in this chapter illustrates how to configure CLNS inter- and intra-domain dynamic routing.

Redistributing Static Routes

The following variation of the **clns router igrp** global configuration command causes the router to advertise static CLNS routes in the domain. Its full syntax is as follows:

clns router igrp *area-tag* redistribute static
no clns router igrp *area-tag* redistribute static

The argument *area-tag* is the defined tag for the area in which static routing information is to be advertised.

Use the **no clns router igrp** command with the appropriate arguments and keywords to remove the router from the list.

Example:

In the following example, the router will advertise any static routes it knows about in the *ship* domain.

```
clns router igrp ship redistribute static
```

The keyword **redistribute** injects the static routes into the routing domain. This eliminates the need for each router in a domain to have a static route configured for each destination. The argument *area-tag* is the tag defined for the NET using the **clns router** global configuration command.

Note: Only the router that advertises the static route needs to have a **redistribute** configuration command defined.

CLNS Configuration Examples

This section provides configuration examples of both intra- and inter-domain static and dynamic routing.

Basic Static Routing

Configuring FDDI, Ethernets, Token Rings, and serial lines using HDLC encapsulation for CLNS can be as simple as just enabling CLNS on the interfaces. This is all that is ever required on serial lines using HDLC encapsulation. If all systems on an Ethernet or Token Ring support ISO 9542 ES-IS, then nothing else is required as well. In this case, an Ethernet and a serial line can be configured as in the following example.

Example:

```
clns routing
clns router static Ydivision NET 47.0004.004D.0055.0000.0C00.BF3B.00
interface ethernet 0
clns router static Ydivision
interface serial 0
clns router static Ydivision
```

Systems Not Using ES-IS

If there are systems on the Ethernet which do not use ES-IS, or if X.25 is being used, NSAP/NET (protocol address) to SNPA (media address) mappings must be specified, using the **clns is-neighbor** and/or the **clns es-neighbor** interface subcommands.

Example:

An example of configuring an Ethernet interface with the Ethernet address (MAC address) of systems that do not use ES-IS follows.

```
interface ethernet 0
clns es-neighbor 47.0004.004D.0055.0000.00C0.A45B.00 0000.00C0.A45B
```

In this case, the End Systems with the NSAP (or NET) listed below is configured at an Ethernet MAC address of 0000.00C0.A45B.

```
47.0004.004D.0055.0000.00C0.A45B.00
```

Note: It is only necessary to use static mapping for those End Systems that do *not* support ES-IS. The router will continue to dynamically discover those End Systems that *do* support ES-IS.

Static Routing

The following is a more complete example of CLNS static routing on a system with two Ethernet interfaces. After configuring routing, you define an NET and enable CLNS on the Ethernet 0 and Ethernet 1 interfaces. You must then define an IS-neighbor and define a static route with the **clns route** command, as shown.

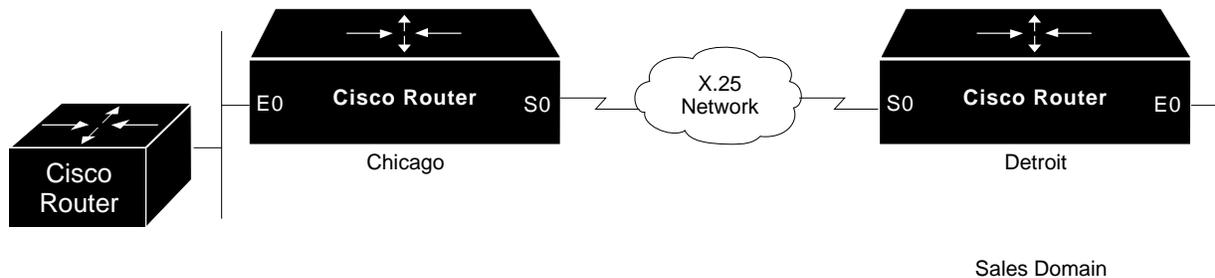
Example:

```
clns routing
clns router static Xdivision net 47.0004.004D.0055.0000.0C00.3F3B.00
interface Ethernet 0
clns router static Xdivision
interface Ethernet 1
clns router static Xdivision
clns is-neighbor 47.0005.0001.0000.0001.0000.00 0000.0C00.62E7
clns route 47.0004.000c 47.0005.0001.0000.0001.0000.00
```

Static Intra-Domain Routing

This example demonstrates how to use static routing inside of a domain. Imagine a company with two branch offices in Detroit and Chicago connected with an X.25 link. These offices are both in the domain named *Sales*.

Figure 1-3 CLNS Static Intra-Domain Routing



The following is one way to configure the router in Chicago.

Example 1:

```
clns host chicago 47.0004.0050.0002.0000.0c00.243b.00
clns host detroit 47.0004.0050.0001.0000.0c00.1e12.00
clns routing
clns router igrp sales net chicago
interface ethernet 0
clns router igrp sales
interface serial 0
encapsulation x25
x25 address 031342174523156
x25 nvc 4
clns router igrp sales
clns is-neighbor detroit 031343136931281 broadcast
```

This configuration will bring up an X.25 virtual circuit between the router in Chicago and the router in Detroit. Routing updates will be sent across this link. This implies that the virtual circuit could be up continuously.

If this is undesirable, use the following configuration instead.

Example 2:

```
clns host chicago net 47.0004.0050.0002.0000.0c00.243b.00
clns host detroit 47.0004.0050.0001.0000.0c00.1e12.00
clns router igrp sales net chicago
!
interface ethernet 0
clns router igrp sales
!
interface serial 0
encapsulation x25
x25 address 031342174523156
x25 nvc 4
clns enable
clns is-neighbor detroit 031343136931281
!
clns route 47.0004.0050.0001 detroit
```

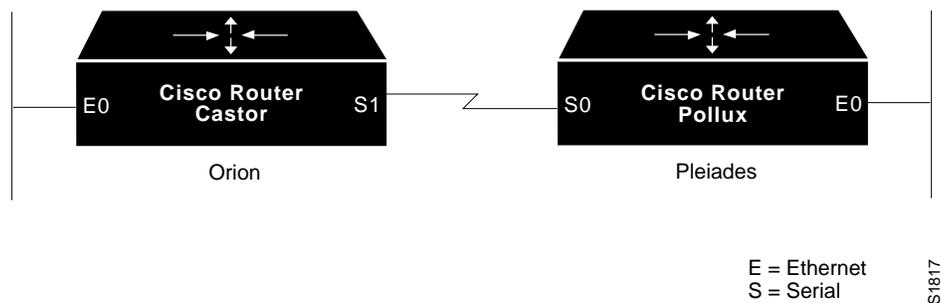
If the Chicago office should grow to contain multiple routers, it would be appropriate for each of those routers to know how to get to Detroit. Add the following command to redistribute information between the routers:

```
clns router igrp sales redistribute static
```

Static Inter-Domain Routing

The following sample illustrates how to configure two routers that distribute information across domains. In this example, Castor and Pollux communicate across a serial link.

Figure 1-4 CLNS Inter-Domain Static Routing



Example for Castor:

```
clns router igrp orion NET
47.0006.02.000000.0000.0100.0001.010203040506.00
!
clns host pollux 47.0006.02.000000.0000.0200.0003.111213141516.00
interface ethernet 0
clns router igrp orion
interface serial 1
clns enable
clns route 47.0006.02.000000.0000.0200 pollux
```

Example for Pollux:

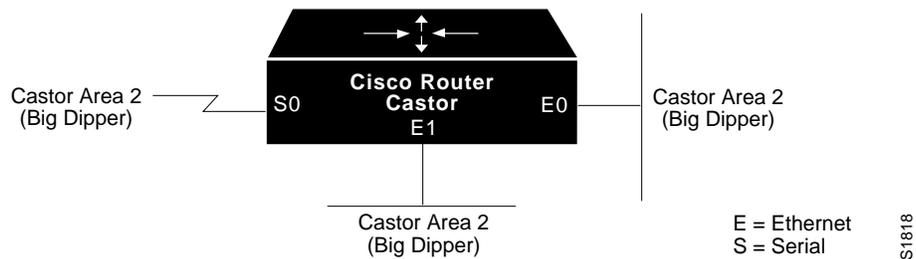
```
clns router igrp pleiades NET
47.0006.02.000000.0000.0200.0003.111213141516.00
!
clns host orion 47.0006.02.000000.0000.0100.0001.010203040506.00
interface ethernet 0
clns router igrp pleiades
interface serial 0
clns enable
clns route 47.0006.02.000000.0000.0100 orion
```

CLNS routing updates will not be sent on the serial link; however, CLNS packets will be sent and received over the serial link.

Routing Within the Same Area

This example illustrates how to configure dynamic routing **within** a routing domain. The router may exist in one or more areas within the domain. The router named *Castor* exists in a single area.

Figure 1-5 CLNS Dynamic Routing — Within a Single Area



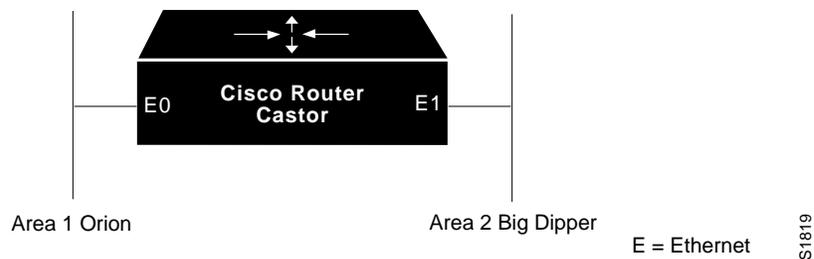
Example:

```
clns routing
clns router igrp bigdipper NET 47.0004.004D.0002.010203040506.00
interface Ethernet 0
clns router igrp bigdipper
interface Ethernet 1
clns router igrp bigdipper
interface Serial 0
clns router igrp bigdipper
```

Routing in More Than One Area

The following example illustrates how to configure a router named Castor that exists in two areas.

Figure 1-6 CLNS Dynamic Routing— Within Two Areas



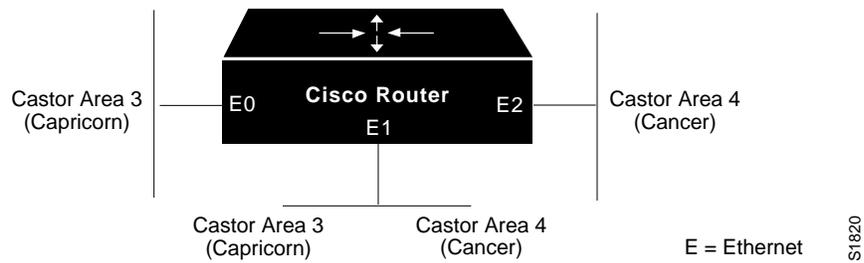
Example:

```
clns routing
clns router igrp orion NET 47.0004.004D.0001.212223242526.00
clns router igrp bigdipper NET 47.0004.004D.0002.212223242526.00
interface ethernet 0
clns router igrp orion
interface ethernet 1
clns router igrp bigdipper
```

Overlapping Areas

The following example illustrates how to configure a router with overlapping areas.

Figure 1-7 CLNS Dynamic Routing— Within Overlapping Areas



Example:

```

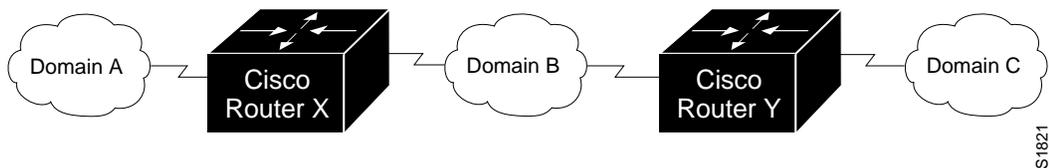
clns routing
clns router igrp capricorn NET 47.0004.004D.0003.010203040506.00
clns router igrp cancer NET 47.0004.004D.0004.010303040506.00
interface ethernet 0
clns router igrp capricorn
interface ethernet 1
clns router igrp capricorn
clns router igrp cancer
interface ethernet 2
clns router igrp cancer

```

Dynamic Inter-Domain Routing

The following example illustrates how to configure three domains that want to be transparently connected.

Figure 1-8 CLNS Inter-Domain Dynamic Routing



Example for Router X:

```

clns routing
clns router igrp A NET 47.0006.02.000000.0000.0100.0002.010201040506.00
clns router igrp B NET 47.0007.02.000000.0000.0100.0003.010201040506.00
clns router igrp A redistribute domain B
clns router igrp B redistribute domain A
interface serial 0
clns router igrp A
interface serial 1

```

```
clns router igrp B
```

Example for Router Y:

```
clns routing
clns router igrp B NET 47.0007.02.000000.0000.0100.0004.010201040506.00
clns router igrp C NET 47.0008.02.000000.0000.0100.0005.010201040506.00
clns router igrp B redistribute domain C
clns router IGRP C redistribute domain B
interface serial 0
clns router igrp B
interface serial 1
clns router igrp C
```

Router X will inject a prefix route for domain A into domain B. Domain B will inject this prefix route plus one for domain B into domain C.

You can also configure a border router between domain A and domain C.

Configuring ES-IS Parameters

This section describes the commands used to configure the ES-IS parameters for device-router communication. In general, however, these should be left at their default values.

When configuring an ES-IS router, be aware that:

- ES-IS does not run over X.25 links.
- ES HELLO (ESH) packets and IS HELLO (ISH) packets are sent without options. Options in received ESH and ISH packets are ignored.

Specifying HELLO Packets

The **clns configuration-time** global configuration command specifies the rate at which ESH and ISH packets are sent.

```
clns configuration-time seconds
no clns configuration-time
```

The default value for *seconds* is 60, and this value is restored by the **no clns configuration-time** command.

The **clns holding-time** subcommand allows the sender of an ESH or ISH to specify the length of time during which the information in the HELLO packets will be believed.

```
clns holding-time seconds
no clns holding-time
```

The argument *seconds* specifies the time in seconds. The default value is 300 seconds (five minutes), which is restored by the **no clns holding-time command**.

Configuring Static Configuration of ESs

End Systems need to know how to get to a Level 1 IS for their area and Level 1 ISs need to know all of the ESs that are directly reachable through each of their interfaces. To provide this information, Cisco routers support the ES-IS protocol. A Cisco router dynamically discovers all ESs running the ES-IS protocol. ESs that are not running the ISO ES-IS protocol must be statically configured. This is done using the **clns es-neighbor** interface subcommand:

```
clns es-neighbor nsap snpa  
no clns es-neighbor nsap
```

The argument *nsap* specifies the CLNS address. The argument *snpa* specifies the data link address. The **no** keyword deletes the ES neighbor. There must be a static **clns es-neighbor** subcommand entry for each End System that does not support ES-IS. If you have configured any ISO IGRP routers using the **clns router igrp** subcommand, the Cisco ES-IS routing software automatically turns ES-IS on.

It is sometimes desirable for a router to have a neighbor entry statically configured rather than learning it through ES-IS. The **clns is-neighbor** interface subcommand enters an IS neighbor.

```
clns is-neighbor nsap snpa  
no clns is-neighbor nsap
```

The argument *nsap* specifies the NSAP address. The argument *snpa* specifies the data link Subnet Point of Attachment (SNPA) MAC-layer address. The **no clns is-neighbor** command deletes the specified IS neighbor.

Configuring Performance Parameters

Generally, you do not need to change the default settings for CLNS packet switching, but there are some modifications you can make when you decide that it is advantageous.

Specifying the MTU Size

All interfaces have a default maximum packet size. You can set the maximum transmission unit (MTU) size of the packets sent on the interface using the **clns mtu** interface subcommand. The full syntax of this command follows.

```
clns mtu size  
no clns mtu
```

The minimum value for the *size* argument is 512; the default and maximum packet size depends on the interface type. The default value is restored by the **no clns mtu** command.

Configuring Checksums

When the ISO CLNS routing software sources a CLNS packet, by default it generates checksums. The **clns checksum** interface subcommand specifies this function. Use the **no clns checksum** command to disable checksum generation.

clns checksum
no clns checksum

Enabling Fast Switching

The **clns route-cache** interface subcommand allows fast switching through the cache, and by default, is enabled. To disable fast switching, use the **no clns route-cache** command.

clns route-cache
no clns route-cache

Note: The cache still exists and is used after the **no clns route-cache** command is used; the software just does not do fast switching through the cache.

Setting the Congestion Threshold

If a router configured for CLNS experiences congestion, it sets the congestion experienced bit. The congestion threshold is a per-interface parameter set by the **clns congestion-threshold** interface subcommand. The full syntax for this command follows.

clns congestion-threshold *number*
no clns congestion-threshold *number*

This subcommand causes the system to set the congestion experience bit if the output queue has more than the specified number of packets in it. A *number* value of zero or the **no** keyword prevents this bit from being set. The default value for *number* is 4.

Use the **no clns congestion-threshold** command with the appropriate value to remove the parameter setting.

Transmitting Error PDUs

When a CLNS packet comes in, the routing software looks in the routing table for the next hop. If it does not find the next hop, the packet is discarded and an error Protocol Data Unit (ERPDU) may be sent.

Sending an Error PDU

The **clns send-erpdu** interface subcommand allows CLNS to send an error PDU when it detects an error in a data PDU, and by default, is enabled. To disable this function, use the **no clns send-erpdu** command. The syntax for both commands follows.

clns send-erpdu
no clns send-erpdu

Determining the Interval Between ERPDU

The **clns erpdu-interval** interface subcommand determines the minimum interval time, in milliseconds, between ERPDU. The full syntax of this command follows.

clns erpdu-interval *milliseconds*
no clns erpdu-interval *milliseconds*

A *milliseconds* value of zero or the **no clns erpdu-interval** command turns off the interval rate and effectively sets no limit to the ERPDU rate. The default rate is once every ten milliseconds.

The **clns erpdu-interval** subcommand will not send ERPDU more frequently than one per interface per ten milliseconds. If a packet is sent out the same interface it came in on, a redirect PDU (RDPDU) may also be sent to the sender of the packet.

Redirecting PDUs

The **clns send-rdpdu** interface subcommand allows CLNS to send redirect PDUs when a better route for a given host is known, and this is the default behavior. The full syntax of the command follows.

clns send-rdpdu
no clns send-rdpdu

To disable this function, use the **no clns send-rdpdu** command.

Disabling RDPDU

An RDPDU may be disabled on a per-interface basis using the **clns rdpdu-interval** interface subcommand. The full syntax of the command follows.

clns rdpdu-interval *milliseconds*
no clns rdpdu-interval *milliseconds*

The command determines the minimum interval time, in milliseconds, between RDPDU. A *milliseconds* value of zero or the **no clns rdpdu-interval** command turns off the interval rate and effectively sets no limit to the RDPDU rate. The default rate is once every 100 milliseconds. An RDPDU is rated-limited, and is not sent more frequently than one per interface, per 100 milliseconds.

Disabling the RDPDU Mask

The address mask is normally present on all RDPDU, but can be disabled with the **no clns rdpdu-mask** interface subcommand. The full syntax of the **clns rdpdu-mask** command follows.

clns rdpdu-mask
no clns rdpdu-mask

Note: SNPA masks are never sent, and RDPDUs are ignored by Cisco routers when the router is acting as an IS.

Configuring Parameters for Locally Sourced Packets

Use these commands to configure parameters for packets sourced by this router. Full command syntax for is command is listed with the descriptions. The **no** forms of the commands remove the parameters settings.

The **clns packet-lifetime** global configuration command specifies the initial lifetime for locally generated packets.

clns packet-lifetime *number*
no clns packet-lifetime *number*

The default value for *number* is 64.

The **clns want-erpdu** interface subcommand specifies whether to request error PDUs on packets sourced by the router.

clns want-erpdu
no clns want-erpdu

The default is to request error PDUs.

Header Options

The ISO CLNS routing software ignores the Record Route option, the Source Route option, and the QOS (quality of service) option other than congestion experienced. The security option causes a packet to be rejected with a bad option indication.

NSAP Shortcut Command

The **clns host** global configuration command can be useful for creating a name for an NSAP. This name can then be used in place of typing the long set of numbers associated with an NSAP. The command syntax follows.

clns host *name nsap*

The argument *name* is the desired name for the NSAP, which is specified by the *nsap* argument.

Maintaining a CLNS Network

Use the EXEC commands described in this section to maintain the ISO CLNS caches, tables, and databases.

clear clns cache

The EXEC command **clear clns cache** clears and re-initializes the CLNS routing cache.

clear clns route

The command **clear clns route** removes all of the dynamically derived CLNS routing information.

Monitoring a CLNS Network

Use the EXEC commands described in this section to obtain displays of activity on the ISO CLNS network.

Displaying General CLNS Information

The show clns command displays information about the CLNS network. Enter this command at the EXEC prompt:

show clns

Sample output follows.

```
Global CLNS Information:
  1 interfaces enabled for clns
NET: 47.0004.0001.2122.2324.2526.00
Configuration Timer: 60, default holding timer: 300, packet lifetime 64
ERPDU's requested on locally generated packets
Intermediate system operation enabled (forwarding allowed)
Level 1 Router: AREATWO
  Routing for domain: 47.0004 area: 0001
Level 2 Router: DOMAIN_AREATWO
  Routing for domain: 47.0004
```

In the display:

- The first line indicates how many interfaces have the CLNS routing protocol turned on.
- The second line contains the NET for this router. Note that there may be more than one NET for one router.
- The Configuration Timer field displays the frequency with which the router will

send out IS HELLO packets. The number following the default holding timer field is the length of time the timer will remember ES HELLO packets. The packet lifetime displayed is the default value used in packets sourced by this router.

- The next line indicates whether error PDUs (ERPDUs) will be requested for packets sourced by the router.
- The last line of global information indicates whether or not this router is configured to be an End System or an Intermediate System. (It is not generally useful for a customer to configure a router to be an End System.)
- The last lines of this display list the areas and domains that this router is in.

Displaying CLNS Routes

Use the EXEC command **show clns route** to display all of the destinations to which this router knows how to route packets. Enter this command at the EXEC prompt:

```
show clns route [nsap]
```

Destinations are sorted by category. The optional argument *nsap* specifies the CLNS address.

Sample output follows:

```
AREATWO: <47.0004><0001>
I B1.B2B3.B4B5.B6
->47.0004.0001.3132.3334.3536.00, HT 128 metric: 44C
  DOMAIN_AREATWO: <47.0004><0001>
I 0003
->47.0004.0002.6162.6364.6566.00, HT 119 metric: 2180
  Prefix Routes
S 47.0004.5555
->47.0004.0001.A1A2.A3A4.A5A6.00, <permanent>metric: 0
```

In this display:

- The ES in the same area as the router are displayed first, followed by the paths to other areas in this domain. Finally, paths to other domains are displayed. All of these are listed in a similar format. The first field lists the way that the route was discovered: **I** stands for IGRP; **S** stands for Static.
- The second field lists the destination. Station IDs are used for ESes; Area numbers are used for other areas in the domain; and prefixes are used for destinations outside of this domain.
- The next field is the first hop that packets for this destination will take.
- Next is the HT (hold time) field which lists the number of seconds this route will be remembered. If the route was statically entered and will not be forgotten, the string **<permanent>** will be displayed instead.
- The final field is the metric that is used for this route.

Neighbors are not included in the **show clns route** display.

Displaying the CLNS Routing Cache

Use the EXEC command **show clns cache** to display the CLNS routing cache. Enter this command at the EXEC prompt:

show clns cache

The cache contains an entry for each destination that has packet switching enabled.

Following is sample output:

```
Cache version 4
47.0004.0001.2122.2324.2526.00
47.0004.0001.3132.3334.3536.00 ->
    47.0004.0001.3132.3334.3536.00@Ethernet0:0000.0C00.62E6
47.0004.5555.3232 ->
    47.0004.0001.A1A2.A3A4.A5A6.00@Ethernet0:0000.0C00.5153
```

In the display, there will be an entry in the cache for each destination that the Cisco router has switched a packet for in the recent past. This includes the Cisco router. Each entry has the following format:

```
destination → first hop address@interface:MAC-layer address
```

Displaying CLNS Traffic

Use the **show clns traffic** command to list the CLNS packets this router has seen. Enter this command at the EXEC prompt:

show clns traffic

Sample output follows:

```
CLNS & ESIS Output: 177, Input: 431
CLNS Local: 0, Forward: 1
CLNS Discards:
    Hdr Syntax: 0, Checksum: 0, Lifetime: 0, Output cngstn: 0
    No Route: 0, Dst Unreachable 0, Encaps. Failed: 0
    NLP Unknown: 0, Not an IS: 0
CLNS Options: Packets 7, total 7, bad 0, GQOS 0, cngstn exprncd 0
CLNS Segments: Segmented: 0, Failed: 0
CLNS Broadcasts: sent: 0, rcvd: 0
Echos: Rcvd 10 requests, 4 replies
    Sent 68 requests, 10 replies
ESIS(sent/rcvd): ESHs: 0/18, ISHs: 19/30, RDs: 1/0, QCF: 0/0
ISO IGRP:Querys (sent/rcvd): 0/0 Updates (sent/rcvd): 60/103
Router Hellos: (sent/rcvd): 68/127
```

In the display:

- The first line lists the total number of packets that this router has sent (the Output field) and received (the Input field).
- The CLNS Local field lists the number of packets that were generated by this router.
- The CLNS Forward field lists the number of packets that this router has forwarded.
- The CLNS Discards field lists the packets that CLNS has discarded, along with the reason

for the discard.

- The CLNS Options field lists the options that have been seen in CLNS packets.
- The CLNS Segments field lists the number of packets that have been segmented and the number of failures that occurred because a packet could not be segmented.
- The CLNS Broadcasts field lists the number of CLNS broadcasts that have been sent and received.
- The Echos field lists the number of echo request packets and echo reply packets that have been received. The line following this field lists the number of echo request packets and echo reply packets that have been sent.
- The ESIS(sent/rcvd) field lists the Number of ESH, ISH, and Redirects sent and received.
- The ISO IGRP field lists the number of IGRP queries, and updates sent and received.
- The Router Hellos field lists the number of IGRP router HELLO packets which have been sent and received.

Displaying CLNS Redirect Information

The **show clns redirects** command displays CLNS redirect information. Enter this command at the EXEC prompt:

```
show clns redirects
```

Only ESs maintain redirect information.

Displaying Status About Specific Interfaces

The **show clns interface** command lists the CLNS-specific information about each interface, and is entered at the EXEC prompt, as follows:

```
show clns interface [interface unit ]
```

Following is sample output:

```
Ethernet 0 is up, line protocol is up
  Checksums enabled, MTU 1500, Encapsulation ISO1
  Next ESH/ISH in 20 seconds
  ERPDUs enabled, min. interval 10 msec.
  RDPDUs enabled, min. interval 100 msec., Addr Mask enabled
  Congestion Experienced bit set at 4 packets
  Routing Domain/Area: <47.0004><0001>
  CLNS fast switching enabled
```

In the display:

- The Checksums enabled field may be enabled or disabled. The number following MTU is the maximum transmission size for a packet on this interface. The Encapsulation field will always be ISO1.
- The Next field displays when the next ESH or ISH will be sent on this interface.

- The next line displays information about the generation of error PDUs (ERPDUs). They may be either enabled or disabled. If they are enabled, they will be sent out no more frequently than the specified interval.
- The next line provides information about the generation of redirect PDUs (RDPDUs). They may be either enabled or disabled. If they are enabled, they will be sent out no more frequently than the specified interval. If the address mask is enabled, redirects will be sent out with an address mask.
- The next line tells when CLNS will turn on the congestion experienced bit. The default is to turn this bit on when there are more than four packets in a queue.
- The next line lists the areas that this interface is in. In most cases, an interface will be in only one area.
- The last line displays whether or not fast switching is supported for CLNS on this interface.

Displaying CLNS ES Neighbors

The **show clns es-neighbor** command lists the ES neighbors that this router knows about. Enter this command at the EXEC prompt:

```
show clns es-neighbor
```

Following is sample output:

```
AREATWO: <47.0004><0001>
E ES 41.4243.4445.46, Ethernet0:0000.0C00.40AF, HT 247
S ES A1.A2A3.A4A5.A6, Ethernet0:0000.0C00.5153, <permanent>
```

In the display, the neighbors are sorted by which area they are in. The area tag, domain, and area address are printed first. Below them, all of the neighbors are listed.

Each ES neighbor entry consists of the following fields:

- The first field lists the way that this neighbor was discovered. There are several possible values for End Systems: E (ES-IS), S (Static), D (DECnet Phase IV).
- The second field lists the type of neighbor; ES is an End System.
- The next field lists the station address for this neighbor.
- The next field lists the interface which is used to communicate with this neighbor and its MAC layer address.
- The HT (hold time) field in the last line lists the number of seconds the neighbor information will be held. If the neighbor was statically entered, the string <permanent> will be displayed instead.

Displaying CLNS IS Neighbors

The **show clns is-neighbor** command displays neighbor entries sorted by which area they are in. Enter this command at the EXEC prompt:

show clns is-neighbors

Following is sample output:

```
AREATWO: <47.0004><0001>
I IS 01.0203.0405.06, Ethernet0:0000.0C00.3E29, HT 0
I IS 31.3233.3435.36, Ethernet0:0000.0C00.62E6, HT 42
DOMAIN_AREATWO: <47.0004>
I IS 0002:61.6263.6465.66, Ethernet0:0000.0C00.3E29, HT 48
nonconforming neighbors
E IS 47.0005.0001.6162.6364.6566.00, Ethernet0:0000.0C00.3E29, HT 288
```

In the display:

- The first field lists the way that this neighbor was discovered. There are several possible values for this field: E (ESIS), S (Static), I (IGRP), D (DECnet Phase IV).
- The second field lists the type of neighbor; IS is Intermediate System.
- If the neighbor is in one of our areas, the next field lists the station address. If the neighbor is in our domain, the next field lists the area and station address. If the neighbor is a nonconforming neighbor, its entire NET is listed.
- The next field lists the interface and MAC-layer address used to communicate with that neighbor.
- The HT (hold time) field in the last line lists the number of seconds the neighbor information will be held. If the neighbor was statically entered, the string <permanent> will be displayed instead.

Some neighbors are in our domain, but not in one of our areas and these are listed under the domain. Finally, some neighbors are not in our domain at all. These are listed under the `nonconforming neighbors` field.

Displaying ES and IS Neighbors

The **show clns neighbors** command displays both ES and IS neighbors. Enter this command at the EXEC prompt:

```
show clns neighbors
```

This display is a composite of the **show clns es-neighbor** and **show clns is-neighbor** commands.

Displaying Protocol-Specific Information

The **show clns protocol** command lists the protocol-specific information for each IGRP routing process in this router. Enter this command at the EXEC prompt:

```
show clns protocol [domain]
```

There will always be at least two routing processes, a Level 1 and a Level 2, and there may be more. The optional argument *domain* specifies a particular routing domain.

Following is sample output:

```

Level 1 Router: AREATWO
  Routing for domain: 47.0004 area: 0001
  Sending Updates every 45 seconds. Next due in 42 seconds
  Invalid after 135 seconds,
  Hold down for 145 seconds
  Sending Router Hellos every 17 seconds. Next due in 10 seconds
  Invalid after 51 seconds,
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  Interfaces in domain/area:
    Ethernet0
  -More-
Level 2 Router: DOMAIN_AREATWO
  Routing for domain: 47.0004
  Sending Updates every 45 seconds. Next due in 26 seconds
  Invalid after 135 seconds,
  Hold down for 145 seconds
  Sending Router Hellos every 17 seconds. Next due in 11 seconds
  Invalid after 51 seconds,
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  Interfaces in domain/area:
    Ethernet0

```

In the display:

- The first line provides the domain address and area number for Level 1 routing processes. For Level 2 routing processes, this command lists the domain address.
- The next set of fields indicate some of the protocol timers. The field labeled Sending updates displays when the next routing updates will be sent.
- The Invalid field indicates how long routing updates are to be believed.
- The Hold Down field indicates how long a route will be held down before new information is to be believed.
- The Sending Router Hellos field indicates how often the routers will send hellos to each other and when the next is due.
- The field labeled Invalid indicates how long a neighbor entry will be remembered.
- The IGRP metric weight displays lists the weights applied to the various components of the metric. These fields are followed by the list of interfaces that are in this area.

The ISO CLNS Ping Command

The OSI Connectionless Network Protocol (ISO 8473) does not specify a network-level echo protocol. The Internet Engineering Task Force (IETF) has specified and proposed such a protocol in RFC 1139. Cisco has implemented this specification using the proposed new PDU types, Echo Request Selector (1E), and Echo Reply Selector (1F). noncisco routers may or may not forward these packets, depending on whether they are specific about the packet types they will forward. End Systems will not recognize these packets, but will typically generate an error packet (ERPDU) as a response. This ERPDU is useful, as it confirms the reachability of the end system. Table 1-3 lists the characters displayed during the **ping** test and their meaning.

Table 1-3 Ping Test Characters

Char	Meaning
!	Each exclamation point indicates receipt of a reply.
.	Each period indicates the network server timed out while waiting for a reply.
U	A destination unreachable, error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

The host name that you specify in response to the **ping** prompt “Target CLNS address:” cannot begin with a number. The output concludes with the success rate and minimum, average, and maximum round-trip times. To abort a **ping** session, type the escape sequence (by default, Ctrl-^, X).

Example 1:

The following example uses a name to specify the source.

```
Protocol [ip]: clns
Target CLNS address: thoth
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]:
Type escape sequence to abort.
Sending 5, 100-byte CLNS Echos to
55.0006.0100.0000.0000.0001.8888.1112.1314.151
6, timeout is 2 seconds:
!!!!
Success rate is 100 percent, round-trip min/avg/max = 112/113/116 ms
```

Example 2:

In this example, an NET address is specified.

```
Protocol [ip]: clns
Target CLNS address: 47.0004.0050.0002.0000.0c00.243b.00
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]:
Type escape sequence to abort.
Sending 5, 100-byte CLNS Echos to 47.0004.0050.0002.0000.0C00.243B.00,
timeout is 2 seconds:
!!!!
Success rate is 100 percent, round-trip min/avg/max = 1/4/8 ms
```

The ISO CLNS Trace Command

The ISO CLNS **trace** command allows you to discover the path packets are taking through your network. It sends Probe packets and takes advantage of the ERPDUs that are generated when a packet exceeds its time-to-live (TTL) value. The **trace** command offers default and settable parameters for specifying a simple or extended trace mode. Enter the following command at the EXEC prompt:

```
trace [destination]
```

To invoke a simple **trace** test, enter the destination address or host name on the command line. The default parameters for the appropriate protocol are assumed and the tracing action begins.

To use nondefault parameters and invoke an extended **trace** test, enter the command without a destination argument. You will be stepped through a dialog to select the desired parameters.

Typing the escape sequence (by default, Ctrl^, X) terminates a **trace** command.

How Trace Works

The **trace** command works by taking advantage of the error messages generated by routers when a datagram exceeds its time-to-live (TTL) value.

The **trace** command starts by sending probe datagrams with a TTL value of one. This causes the first router to discard the probe datagram and send back an error message. The **trace** command sends several probes at each TTL level and displays the round trip time for each.

The **trace** command sends out one probe at a time. Each outgoing packet may result in one of two error messages. A *time exceeded* error message indicates that an intermediate router has seen and discarded the probe. A *destination unreachable* error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, **trace** prints an asterisk (*).

The **trace** command terminates when the destination responds, when the maximum TTL was exceeded, or when the user interrupts the trace with the escape sequence. The information is encoded as follows:

```
hop-count name(nsap) result-of-probe
```

Tracing CLNS Routes

You may use the **trace** command to trace routes on a Cisco router configured with the ISO CLNS protocol. When stepping through the **trace** dialog for CLNS, the following parameters may be specified:

- Protocol [ip]. The default protocol for trace is IP. You must specify CLNS to begin tracing a router on a CLNS router.
- Target CLNS address. You may specify either an NSAP or host name.

- Timeout in seconds. You may specify the length of time to wait after sending each probe before giving up on getting a response.
- Probe count. You may specify the number of probes to be sent at each TTL level. The default is three.
- Minimum Time to Live [1]: You may set the TTL value for the first probes. The default is 1. Set to a higher value to suppress the display of known hops.
- Maximum Time to Live [30]: You may set the largest TTL value which may be used. The default is 30. The **trace** command terminates when the destination is reached or when this value is reached.

The following table describes the output.

Table 1-4 Trace Test Characters

Char	Meaning
<i>nn</i> msec	The probe was successfully returned in <i>nn</i> milliseconds.
&	A time-to-live-exceeded error PDU was received.
U	A destination unreachable error PDU was received.
I	The user interrupted the test.
*	The probe timed out.
C	A congestion experienced packet was received.

Following is simple **trace** output.

```

Protocol [ip]: clns
Target CLNS address: thoth
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Type escape sequence to abort.
Tracing the route to THOTH (55.0006.0100.0000.0000.0001.8888.1112.1314.1516)
  1 HORUS(55.0006.0100.0000.0000.0001.6666.3132.3334.3536) 32 msec ! 28 msec !
28 msec !
  2 ISIS(55.0006.0100.0000.0000.0001.7777.2122.2324.2526) 56 msec ! 80 msec ! 56
msec !
  3 THOTH(55.0006.0100.0000.0000.0001.8888.1112.1314.1516) 80 msec ! 80 msec ! 8

```

This example traces a more complex route.

```

Protocol [ip]: clns
Target CLNS address: cerdiwen
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Type escape sequence to abort.
Tracing the route to CERDIWEN (49.BEAD.0000.0C00.40A7.00)
  1 DEMETER(49.BEAD.0000.0C00.40AF.00) 72 msec !
    ARTEMIS(49.BEAD.0000.0C00.2D57.00) 72 msec !
    DEMETER(49.BEAD.0000.0C00.40AF.00) 76 msec !
  2 CERDIWEN(49.BEAD.0000.0C00.40A7.00) 148 msec ! 148 msec ! 144 msec !

```

The output from the **trace** command displays information for each probe that it sends out.

As you can see, there were two equal cost paths to the destination. The first packet went via Demeter, the second went via Artemis, and the third went via Demeter again.

Debugging a CLNS Network

Use the EXEC commands described in this section to troubleshoot and monitor the ISO CLNS network transactions. For each **debug** command, there is a corresponding **undebug** command that turns the message logging off. Generally, you will enter these commands during troubleshooting sessions with Cisco engineers.

debug clns-esis-events

The **debug clns-esis-events** command traces the more unusual ES-IS events, including previously unknown neighbors, neighbors which have aged out, and neighbors which have changed roles (ES to IS, and so on).

debug clns-esis-packets

The **debug clns-esis-packets** command traces ES-IS activity, including sending and receiving of ESHes and ISHes, receiving Redirects (RDs), and aging out of ESH/ISH/RD entries.

debug clns-events

The **debug clns-events** command traces the more unusual CLNS events, including packet discards, sending of redirects, and so forth.

debug clns-igrp-packets

The **debug clns-igrp-packets** command causes all ISO IGRP routing activity to be displayed.

debug clns-packets

The **debug clns-packets** command causes all CLNS activity to be traced, including forwarding of packets. You can use this to circumvent a potential problem with the **ping** command in mixed Cisco and nonCisco installations; see the Ping command for more information.

debug clns-routing

The **debug clns-routing** command causes all CLNS routing table activity to be traced.

ISO CLNS Global Configuration Command Summary

This section provides an alphabetical summary of the ISO CLNS global configuration commands.

[no] clns configuration-time *seconds*

Specifies the rate at which ESHs and ISHs are sent. The default value for *seconds* is 60.

[no] clns holding-time *seconds*

Allows the sender of an ESH or ISH to specify the length of time during which the information in the HELLO packets will be believed. The argument *seconds* specifies the time in seconds. The default value is 300 seconds (five minutes).

clns host *name nsap*

Defines a name for an NSAP that can then be used in commands requiring NSAPs.

[no] clns packet-lifetime *number*

Specifies the initial lifetime for locally generated packets. The default value for *number* is 64.

[no] clns route *nsap-prefix next-hop-net*

Enters a specific static route. NSAPs that start with *nsap-prefix* are forwarded to *next-hop-nsap*.

[no] clns route *nsap-prefix discard*

A variation of the **clns route** command that uses the **discard** keyword to explicitly tell a router to discard packets with the specified *nsap-prefix*.

[no] clns router igrp *area-tag NET net*

Identifies the area the router will work in and lets the router know that it will be routing dynamically rather than statically. The keyword **igrp** specifies dynamic routing using the IGRP protocol. The argument *area-tag* defines a meaningful name for an area. Following the keyword **NET** you specify the Network Entity Title. The **no** form of the command turns off this IGRP router.

[no] clns router igrp *area-tag2 redistribute domain area-tag1*

Redistributes routing information throughout a routing domain. The keywords **redistribute domain** enable the redistribution. The arguments *area-tag2* and *area-tag1* are the defined tags for the areas in which the routing information is to be redistributed. The **no** form disables this CLNS routing protocol on this interface.

[no] clns router igrp *area-tag redistribute static*

Causes the router to inject any static CLNS routes into the domain. The **no** form stops redistribution.

[no] clns router static *area-tag NET net*

Identifies the area the router will work in and lets the router know that it will be routing statically. The keyword **static** specifies that the router will use statically entered routes only. The argument *area-tag* defines a meaningful name for an area. The keyword **NET** specifies the Network Entity Title.

[no] clns routing

Enables or disables routing of CLNS packets.

ISO CLNS Interface Subcommand Summary

This section provides an alphabetical list of the ISO CLNS interface subcommands.

[no] clns checksum

Enables or disables checksum generation when ISO CLNS routing software sources a CLNS packet. By default, this function is on. Use the **no** form of the command to disable checksum generation.

[no] clns congestion-threshold *number*

Sets the congestion experience bit if the output queue has more than the specified number of packets in it. A *number* value of zero or the **no** form of the command prevents this bit from being set. The default value for *number* is four.

[no] clns erpdu-interval *milliseconds*

Determines the minimum interval time, in milliseconds, between ERPDUs. A *milliseconds* value of zero or the **no** form of the command turns off the interval rate and effectively sets no limit to the ERPDU rate. The default rate is once every ten milliseconds.

[no] clns es-neighbor *nsap snpa [X.25-facilities-info]*

Lists all End Systems that will be used when mapping information is statically entered. The SNPAs are the X.25 network addresses (X.121 addresses). These are usually assigned by the X.25 network provider. Use the argument *X.25-facilities-info* to specify nondefault packet and window size, reverse charge information, and so on.

[no] clns is-neighbor *nsap snpa [X.25-facilities-info]*

Lists all Intermediate Systems that will be used when mapping information is statically entered. The SNPAs are the X.25 network addresses (X.121 addresses). These are usually assigned by the X.25 network provider. Use the argument *X.25-facilities-info* to specify nondefault packet and window size, reverse charge information, and so on.

[no] clns mtu *size*

Sets the MTU packet size for the interface. The minimum value for *size* is 512. The **no** form of the command restores the default and maximum packet size.

[no] clns rdpdu-interval *milliseconds*

Determines the minimum interval time, in milliseconds, between RDPDUs. A *milliseconds* value of zero or the **no** keyword turns off the interval rate and effectively sets no limit to the RDPDU rate. The default rate is once every 100 milliseconds.

[no] clns rdpdu-mask

Enables or disables the address mask on RDPDUs. The address mask is normally present on all RDPDUs, but may be disabled with the **no clns rdpdu-mask** command.

Note: SNPA masks are never sent, and RDPDUs are ignored by Cisco routers when the router is acting as an IS.

[no] clns route-cache

Allows fast switching through the cache, and by default, is enabled. To disable fast switching, use the **no** keyword.

Note: The cache still exists and is used after the **no clns route-cache** command is used; the software just does not do fast switching through the cache.

clns router igrp *area-tag* [**level2**]

This command specifies IGRP routing. The argument *area-tag* is the tag defined for the NET using the **clns router** global configuration command. The optional **level2** keyword allows the interface to advertise Level 2 information.

clns router static *area-tag*

Specifies static routing. The argument *area-tag* is the tag defined for the NET using the **clns router** global configuration command.

[**no**] **clns send-erpdu**

Allows or prevents CLNS to send an error PDU when it detects an error in a data PDU, and by default, is enabled. To disable this function, use the **no** keyword.

[**no**] **clns send-rdpdu**

Allows or prevents CLNS to send redirect PDUs when a better route for a given host is known, and this is the default behavior. To disable this function, use the **no** keyword.

[**no**] **clns want-erpdu**

Specifies whether to request error PDUs on packets sourced by the router. The default is to request error PDUs.

