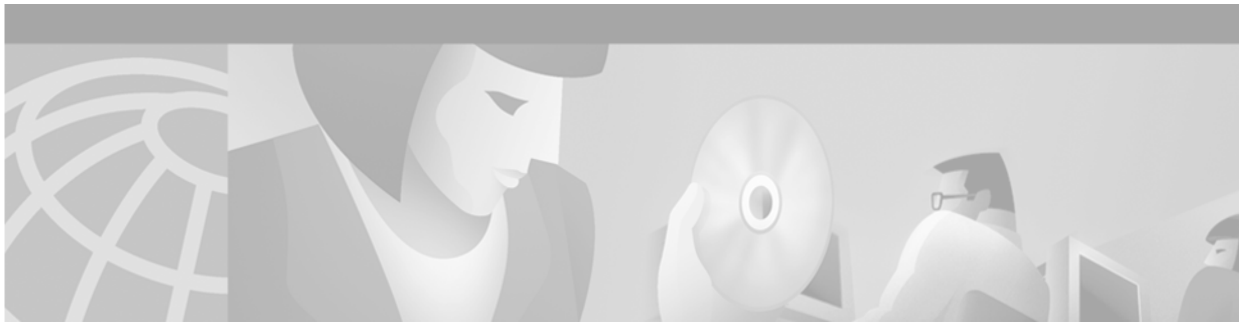


Configuring Routing Between VLANs with Inter-Switch Link Encapsulation

This chapter describes the Inter-Switch Link (ISL) protocol and provides guidelines for configuring ISL and Token Ring ISL (TRISL) features.

For a complete description of the commands in this chapter, refer to the the *Cisco IOS Switching Services Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

To identify the hardware platform or software image information associated with a feature, use the Feature Navigator on Cisco.com to search for information about the feature or refer to the software release notes for a specific release. For more information, see the section "Identifying Supported Platforms" in the chapter "Using Cisco IOS Software."

Overview of the ISL Protocol

ISL is a Cisco protocol for interconnecting multiple switches and maintaining VLAN information as traffic goes between switches. ISL provides VLAN capabilities while maintaining full wire speed performance on Fast Ethernet links in full- or half-duplex mode. ISL operates in a point-to-point environment and will support up to 1000 VLANs. You can define virtually as many logical networks as are necessary for your environment.

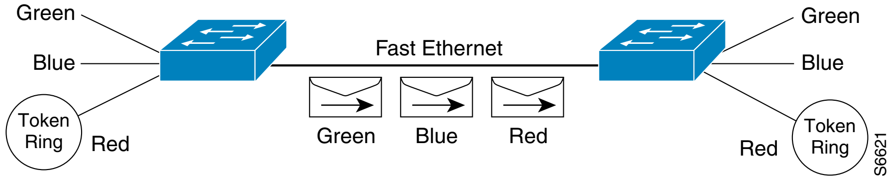
This chapter describes how to configure routing between VLANs using ISL encapsulation.

Frame Tagging in ISL

With ISL, an Ethernet frame is encapsulated with a header that transports VLAN IDs between switches and routers. A 26-byte header that contains a 10-bit VLAN ID is prepended to the Ethernet frame.

A VLAN ID is added to the frame only when the frame is destined for a nonlocal network. Figure 77 shows VLAN packets traversing the shared backbone. Each VLAN packet carries the VLAN ID within the packet header.

Figure 77 VLAN Packets Traversing the Shared Backbone



ISL Encapsulation Configuration Task List

You can configure routing between any number of VLANs in your network. This section documents the configuration tasks for each protocol supported with ISL encapsulation. The basic process is the same, regardless of the protocol being routed. It involves the following tasks:

- Enabling the protocol on the router
- Enabling the protocol on the interface
- Defining the encapsulation format as ISL or TRISL
- Customizing the protocol according to the requirements for your environment

To configure routing between any number of VLANs in your network, perform the tasks described in the following sections particular to your network:

- Configuring AppleTalk Routing over ISL
- Configuring Banyan VINES Routing over ISL
- Configuring DECnet Routing over ISL
- Configuring the Hot Standby Router Protocol over ISL
- Configuring IP Routing over TRISL
- Configuring IPX Routing over TRISL
- Configuring VIP Distributed Switching over ISL
- Configuring XNS Routing over ISL
- Configuring CLNS Routing over ISL
- Configuring IS-IS Routing over ISL
- Monitoring and Maintaining VLAN Subinterfaces

Refer to the "ISL Encapsulation Configuration Examples" section at the end of this chapter for sample configurations.

Configuring AppleTalk Routing over ISL

AppleTalk can be routed over VLAN subinterfaces using the ISL and IEEE 802.10 VLAN encapsulation protocols. The AppleTalk Routing over ISL and IEEE 802.10 Virtual LANs feature provides full-feature Cisco IOS software AppleTalk support on a per-VLAN basis, allowing standard AppleTalk capabilities to be configured on VLANs.

To route AppleTalk over ISL or IEEE 802.10 between VLANs, you need to customize the subinterface to create the environment in which it will be used. Perform the tasks described in the following sections in the order in which they appear:

- Enabling AppleTalk Routing
- Defining the VLAN Encapsulation Format
- Configuring AppleTalk on the Subinterface

Enabling AppleTalk Routing

To enable AppleTalk routing on either ISL or 802.10 interfaces, use the following command in global configuration mode:

Command	Purpose
Router(config)# **appletalk routing** [**eigrp** *router-number*]	Enables AppleTalk routing globally.

Defining the VLAN Encapsulation Format

To define the VLAN encapsulation format as either ISL or 802.10, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# **interface** *type slot***/***port***.***subinterface-number*	Specifies the subinterface the VLAN will use.
Step 2	Router(config-if)# **encapsulation isl** *vlan-identifier* or Router(config-if)# **encapsulation sde** *said*	Defines the encapsulation format as either ISL (**isl**) or IEEE 802.10 (**sde**), and specifies the VLAN identifier or security association identifier, respectively.

Configuring AppleTalk on the Subinterface

After you enable AppleTalk globally and define the encapsulation format, you need to enable it on the subinterface by specifying the cable range and naming the AppleTalk zone for each interface. To enable the AppleTalk protocol on the subinterface, use the following commands in interface configuration mode:

Command	Purpose
Router(config-if)# **appletalk cable-range** *cable-range* [*network***.***node*]	Assigns the AppleTalk cable range and zone for the subinterface.
Router(config-if)# **appletalk zone** *zone-name*	Assigns the AppleTalk zone for the subinterface.

Configuring Banyan VINES Routing over ISL

Banyan VINES can be routed over VLAN subinterfaces using the ISL encapsulation protocol. The Banyan VINES Routing over ISL Virtual LANs feature provides full-feature Cisco IOS software Banyan VINES support on a per-VLAN basis, allowing standard Banyan VINES capabilities to be configured on VLANs.

To route Banyan VINES over ISL between VLANs, you need to configure ISL encapsulation on the subinterface. Perform the tasks described in the following sections in the order in which they appear:

- Enabling Banyan VINES Routing
- Defining the VLAN Encapsulation Format
- Configuring Banyan VINES on the Subinterface

Enabling Banyan VINES Routing

To begin the VINES routing configuration, use the following command in global configuration mode:

Command	Purpose
`Router(config)#` **`vines routing`** [*`address`*]	Enables Banyan VINES routing globally.

Defining the VLAN Encapsulation Format

To define the VINES routing encapsulation format, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **`interface`** *`type slot/port.subinterface-number`*	Specifies the subinterface on which ISL will be used.
Step 2	`Router(config-if)#` **`encapsulation isl`** *`vlan-identifier`*	Defines the encapsulation format as ISL (**isl**), and specifies the VLAN identifier.

Configuring Banyan VINES on the Subinterface

After you enable Banyan VINES globally and define the encapsulation format, you need to enable VINES on the subinterface by specifying the VINES routing metric. To enable the Banyan VINES protocol on the subinterface, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **`vines metric`** [*`whole`* [*`fractional`*]]	Enables VINES routing on an interface.

Configuring DECnet Routing over ISL

DECnet can be routed over VLAN subinterfaces using the ISL VLAN encapsulation protocols. The DECnet Routing over ISL Virtual LANs feature provides full-feature Cisco IOS software DECnet support on a per-VLAN basis, allowing standard DECnet capabilities to be configured on VLANs.

To route DECnet over ISL VLANs, you need to configure ISL encapsulation on the subinterface. Perform the tasks described in the following sections in the order in which they appear.

- Enabling DECnet Routing
- Defining the VLAN Encapsulation Format
- Configuring DECnet on the Subinterface

Enabling DECnet Routing

To begin the DECnet routing configuration, use the following command in global configuration mode:

Command	Purpose
Router(config)# **decnet** [*network-number*] **routing** [*decnet-address*]	Enables DECnet on the router.

Defining the VLAN Encapsulation Format

To define the encapsulation format, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# **interface** *type slot/port.subinterface-number*	Specifies the subinterface on which ISL will be used.
Step 2	Router(config-if)# **encapsulation isl** *vlan-identifier*	Defines the encapsulation format as ISL (**isl**), and specifies the VLAN identifier.

Configuring DECnet on the Subinterface

To configure DECnet routing on the subinterface, use the following command in interface configuration mode:

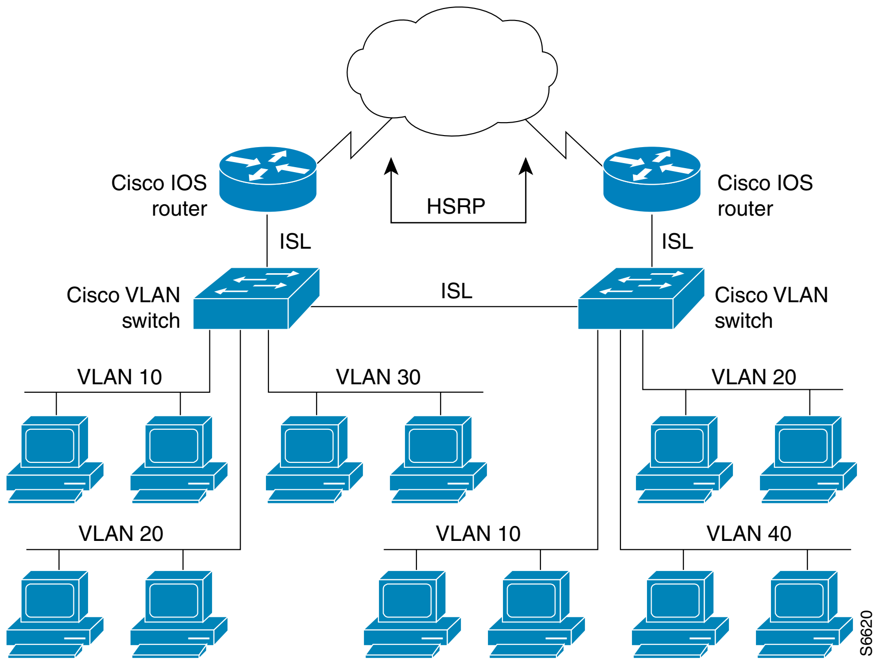
Command	Purpose
Router(config-if)# **decnet cost** [*cost-value*]	Enables DECnet routing on an interface.

Configuring the Hot Standby Router Protocol over ISL

The Hot Standby Router Protocol (HSRP) provides fault tolerance and enhanced routing performance for IP networks. HSRP allows Cisco IOS routers to monitor each other's operational status and very quickly assume packet forwarding responsibility in the event the current forwarding device in the HSRP group fails or is taken down for maintenance. The standby mechanism remains transparent to the attached hosts and can be deployed on any LAN type. With multiple Hot Standby groups, routers can simultaneously provide redundant backup and perform loadsharing across different IP subnets.

Figure 78 illustrates HSRP in use with ISL providing routing between several VLANs.

Figure 78 Hot Standby Router Protocol in VLAN Configurations



A separate HSRP group is configured for each VLAN subnet so that Cisco IOS router A can be the primary and forwarding router for VLANs 10 and 20. At the same time, it acts as backup for VLANs 30 and 40. Conversely, Router B acts as the primary and forwarding router for ISL VLANs 30 and 40, as well as the secondary and backup router for distributed VLAN subnets 10 and 20.

Running HSRP over ISL allows users to configure redundancy between multiple routers that are configured as front ends for VLAN IP subnets. By configuring HSRP over ISLs, users can eliminate situations in which a single point of failure causes traffic interruptions. This feature inherently provides some improvement in overall networking resilience by providing load balancing and redundancy capabilities between subnets and VLANs.

To configure HSRP over ISLs between VLANs, you need to create the environment in which it will be used. Perform the tasks described in the following sections in the order in which they appear.

- Defining the Encapsulation Format
- Defining the IP Address
- Enabling HSRP

Defining the Encapsulation Format

To define the encapsulation format as ISL, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **interface** *type slot/port.subinterface-number*	Specifies the subinterface on which ISL will be used.
Step 2	`Router(config-if)#` **encapsulation isl** *vlan-identifier*	Defines the encapsulation format, and specifies the VLAN identifier.

Defining the IP Address

After you have specified the encapsulation format, to define the IP address over which HSRP will be routed, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **ip address** *ip-address mask* [**secondary**]	Specifies the IP address for the subnet on which ISL will be used.

Enabling HSRP

To enable HSRP on an interface, enable the protocol, then customize it for the interface. Use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]	Enables HSRP.

Note For more information on HSRP, see the "Configuring IP Services" chapter in the *Cisco IOS IP Configuration Guide*.

To customize Hot Standby group attributes, use the following commands in interface configuration mode, as needed:

Command	Purpose
`Router(config-if)#` **standby** [*group-number*] **timers** *hellotime holdtime*	Configures the time between hello packets and the hold time before other routers declare the active router to be down.
`Router(config-if)#` **standby** [*group-number*] **priority** *priority*	Sets the Hot Standby priority used to choose the active router.
`Router(config-if)#` **standby** [*group-number*] **preempt**	Specifies that if the local router has priority over the current active router, the local router should attempt to take its place as the active router.

Command	Purpose
`Router(config-if)#` **`standby`** `[`*`group-number`*`]` **`track`** *`type-number`* `[`*`interface-priority`*`]`	Configures the interface to track other interfaces, so that if one of the other interfaces goes down, the Hot Standby priority for the device is lowered.
`Router(config-if)#` **`standby`** `[`*`group-number`*`]` **`authentication`** *`string`*	Selects an authentication string to be carried in all HSRP messages.

Configuring IP Routing over TRISL

The IP routing over TRISL VLANs feature extends IP routing capabilities to include support for routing IP frame types in VLAN configurations.

Enabling IP Routing

IP routing is automatically enabled in the Cisco IOS software for routers. To reenable IP routing if it has been disabled, use the following command in global configuration mode:

Command	Purpose
`Router(config)#` **`ip routing`**	Enables IP routing on the router.

Once you have IP routing enabled on the router, you can customize the characteristics to suit your environment. If necessary, refer to the IP configuration chapters in the *Cisco IOS IP Routing Configuration Guide* for guidelines on configuring IP.

Defining the VLAN Encapsulation Format

To define the encapsulation format as TRISL, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **`interface`** *`type slot`*`/`*`port.subinterface-number`*	Specifies the subinterface on which TRISL will be used.
Step 2	`Router(config-if)#` **`encapsulation tr-isl trbrf-vlan`** *`vlanid`* **`bridge-num`** *`bridge-number`*	Defines the encapsulation for TRISL.

The DRiP database is automatically enabled when TRISL encapsulation is configured, and at least one TrBRF is defined, and the interface is configured for SRB or for routing with RIF.

Assigning IP Address to Network Interface

An interface can have one primary IP address. To assign a primary IP address and a network mask to a network interface, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **`ip address`** *`ip-address mask`*	Sets a primary IP address for an interface.

A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is then referred to as a *subnet mask*.

Note TRISL encapsulation must be specified for a subinterface before an IP address can be assigned to that subinterface.

Configuring IPX Routing on 802.10 VLANs over ISL

The IPX Encapsulation for 802.10 VLAN feature provides configurable IPX (Novell-FDDI, SAP, SNAP) encapsulation over 802.10 VLAN on router FDDI interfaces to connect the Catalyst 5000 VLAN switch. This feature extends Novell NetWare routing capabilities to include support for routing all standard IPX encapsulations for Ethernet frame types in VLAN configurations. Users with Novell NetWare environments can now configure any one of the three IPX Ethernet encapsulations to be routed using Secure Data Exchange (SDE) encapsulation across VLAN boundaries. IPX encapsulation options now supported for VLAN traffic include the following:

- Novell-FDDI (IPX FDDI RAW to 802.10 on FDDI)
- SAP (IEEE 802.2 SAP to 802.10 on FDDI)
- SNAP (IEEE 802.2 SNAP to 802.10 on FDDI)

NetWare users can now configure consolidated VLAN routing over a single VLAN trunking FDDI interface. Not all IPX encapsulations are currently supported for SDE VLAN. The IPX interior encapsulation support can be achieved by messaging the IPX header before encapsulating in the SDE format. Fast switching will also support all IPX interior encapsulations on non-MCI platforms (for example non-AGS+ and non-7000). With configurable Ethernet encapsulation protocols, users have the flexibility of using VLANs regardless of their NetWare Ethernet encapsulation. Configuring Novell IPX encapsulations on a per-VLAN basis facilitates migration between versions of Netware. NetWare traffic can now be routed across VLAN boundaries with standard encapsulation options (*arpa*, *sap*, and *snap*) previously unavailable. Encapsulation types and corresponding framing types are described in the "Configuring Novell IPX" chapter of the *Cisco IOS AppleTalk and Novell IPX Configuration Guide*.

Note Only one type of IPX encapsulation can be configured per VLAN (subinterface). The IPX encapsulation used must be the same within any particular subnet; a single encapsulation must be used by all NetWare systems that belong to the same VLAN.

To configure Cisco IOS software on a router with connected VLANs to exchange different IPX framing protocols, perform the tasks described in the following sections in the order in which they are appear:

- Enabling NetWare Routing
- Defining the VLAN Encapsulation Format
- Configuring NetWare on the Subinterface

Enabling NetWare Routing

To enable IPX routing on SDE interfaces, use the following command in global configuration mode:

Command	Purpose
`Router(config)# ipx routing [node]`	Enables IPX routing globally.

Defining the VLAN Encapsulation Format

To define the encapsulation format as SDE, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface fddi slot/port.subinterface-number`	Specifies the subinterface on which SDE will be used.
Step 2	`Router(config-if)# encapsulation sde vlan-identifier`	Defines the encapsulation format and specifies the VLAN identifier.

Configuring NetWare on the Subinterface

After you enable NetWare globally and define the VLAN encapsulation format, to enable the subinterface by specifying the NetWare network number (if necessary) and the encapsulation type, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)# ipx network network encapsulation encapsulation-type`	Specifies the IPX encapsulation among Novell-FDDI, SAP, or SNAP.

Configuring IPX Routing over TRISL

The IPX Routing over ISL VLANs feature extends Novell NetWare routing capabilities to include support for routing all standard IPX encapsulations for Ethernet frame types in VLAN configurations. Users with Novell NetWare environments can configure either SAP or SNAP encapsulations to be routed using the TRISL encapsulation across VLAN boundaries. The SAP (Novell Ethernet_802.2) IPX encapsulation is supported for VLAN traffic.

NetWare users can now configure consolidated VLAN routing over a single VLAN trunking interface. With configurable Ethernet encapsulation protocols, users have the flexibility of using VLANs regardless of their NetWare Ethernet encapsulation. Configuring Novell IPX encapsulations on a per-VLAN basis facilitates migration between versions of Netware. NetWare traffic can now be routed across VLAN boundaries with standard encapsulation options (*sap* and *snap*) previously unavailable. Encapsulation types and corresponding framing types are described in the "Configuring Novell IPX" chapter of the *Cisco IOS AppleTalk and Novell IPX Configuration Guide*.

Note Only one type of IPX encapsulation can be configured per VLAN (subinterface). The IPX encapsulation used must be the same within any particular subnet: A single encapsulation must be used by all NetWare systems that belong to the same LANs.

To configure Cisco IOS software to exchange different IPX framing protocols on a router with connected VLANs, perform the tasks described in the following sections in the order in which they are appear:

- Enabling NetWare Routing
- Defining the VLAN Encapsulation Format
- Configuring NetWare on the Subinterface

Enabling NetWare Routing

To enable IPX routing on TRISL interfaces, use the following command in global configuration mode:

Command	Purpose
`Router(config)#` **`ipx routing`** `[`*`node`*`]`	Enables IPX routing globally.

Defining the VLAN Encapsulation Format

To define the encapsulation format as TRISL, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **`interface`** *`type slot/port.subinterface-number`*	Specifies the subinterface on which TRISL will be used.
Step 2	`Router(config-if)#` **`encapsulation tr-isl trbrf-vlan`** *`trbrf-vlan`* **`bridge-num`** *`bridge-num`*	Defines the encapsulation for TRISL.

Configuring NetWare on the Subinterface

After you enable NetWare globally and define the VLAN encapsulation format, to enable the subinterface by specifying the NetWare network number (if necessary) and the encapsulation type, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **`ipx network`** *`network`* **`encapsulation`** *`encapsulation-type`*	Specifies the IPX encapsulation.

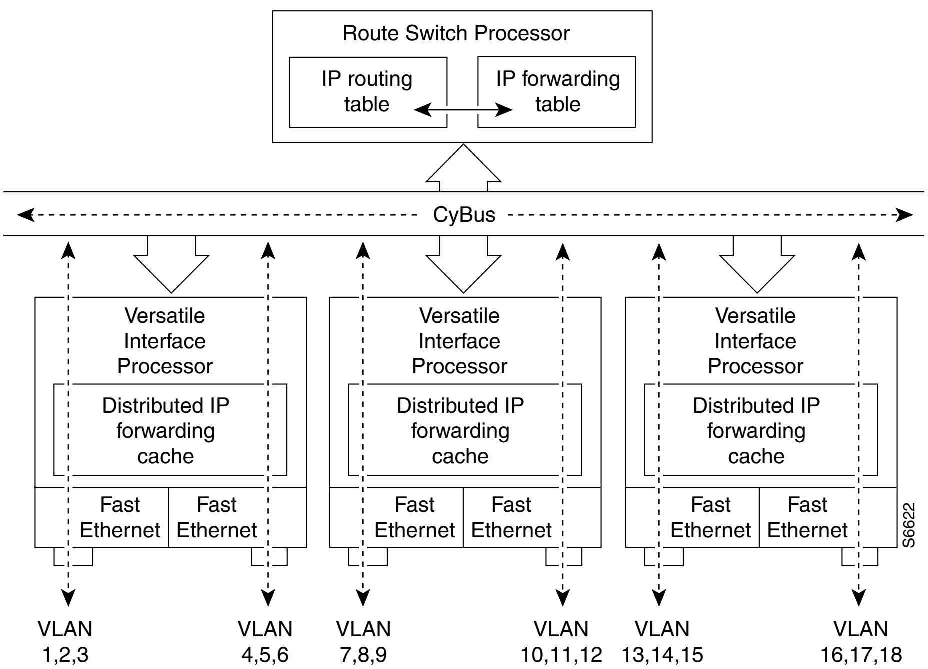
Note The default IPX encapsulation format for Cisco IOS routers is “novell-ether” (Novell Ethernet_802.3). If you are running Novell Netware 3.12 or 4.0, the new Novell default encapsulation format is Novell Ethernet_802.2 and you should configure the Cisco router with the IPX encapsulation format “sap.”

Configuring VIP Distributed Switching over ISL

With the introduction of the VIP distributed ISL feature, ISL encapsulated IP packets can be switched on Versatile Interface Processor (VIP) controllers installed on Cisco 7500 series routers.

The second generation VIP2 provides distributed switching of IP encapsulated in ISL in VLAN configurations. Where an aggregation route performs inter-VLAN routing for multiple VLANs, traffic can be switched autonomously on-card or between cards rather than through the central Route Switch Processor (RSP). Figure 79 shows the VIP distributed architecture of the Cisco 7500 series router.

Figure 79 Cisco 7500 Distributed Architecture



This distributed architecture allows incremental capacity increases by installation of additional VIP cards. Using VIP cards for switching the majority of IP VLAN traffic in multiprotocol environments substantially increases routing performance for the other protocols because the RSP offloads IP and can then be dedicated to switching the non-IP protocols.

VIP distributed switching offloads switching of ISL VLAN IP traffic to the VIP card, removing involvement from the main CPU. Offloading ISL traffic to the VIP card substantially improves networking performance. Because you can install multiple VIP cards in a router, VLAN routing capacity is increased linearly according to the number of VIP cards installed in the router.

To configure distributed switching on the VIP, you must first configure the router for IP routing. Perform the tasks described in the following sections in the order in which they appear:

- Enabling IP Routing
- Enabling VIP Distributed Switching
- Configuring ISL Encapsulation on the Subinterface

Enabling IP Routing

To enable IP routing, use the following command in global configuration mode:

Command	Purpose
`Router(config)# ip routing`	Enables IP routing on the router.

Once you have IP routing enabled on the router, you can customize the characteristics to suit your environment. Refer to the IP configuration chapters in the *Cisco IOS IP Routing Configuration Guide* for guidelines on configuring IP.

Enabling VIP Distributed Switching

To enable VIP distributed switching, use the following commands beginning in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# **interface** *type slot/port-adapter/port*	Specifies the interface and interface configuration mode.
Step 2	Router(config-if)# **ip route-cache distributed**	Enables VIP distributed switching of IP packets on the interface.

Configuring ISL Encapsulation on the Subinterface

To configure ISL encapsulation on the subinterface, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# **interface** *type slot/port-adapter/port*	Specifies the interface, and enters interface configuration mode.
Step 2	Router(config-if)# **encapsulation isl** *vlan-identifier*	Defines the encapsulation format as ISL, and specifies the VLAN identifier.

Configuring XNS Routing over ISL

XNS can be routed over VLAN subinterfaces using the ISL VLAN encapsulation protocol. The XNS Routing over ISL Virtual LANs feature provides full-feature Cisco IOS software XNS support on a per-VLAN basis, allowing standard XNS capabilities to be configured on VLANs.

To route XNS over ISL VLANs, you need to configure ISL encapsulation on the subinterface. Perform the tasks described in the following sections in the order in which they appear:

- Enabling XNS Routing
- Defining the VLAN Encapsulation Format
- Configuring XNS on the Subinterface

Enabling XNS Routing

To configure XNS routing, use the following command in global configuration mode:

Command	Purpose
`Router(config)#` **`xns routing`** [*`address`*]	Enables XNS routing globally.

Defining the VLAN Encapsulation Format

To define the VLAN encapsulation format, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **`interface`** *`type slot/port.subinterface-number`*	Specifies the subinterface on which ISL will be used.
Step 2	`Router(config-if)#` **`encapsulation isl`** *`vlan-identifier`*	Defines the encapsulation format as ISL (**isl**), and specifies the VLAN identifier.

Configuring XNS on the Subinterface

To enable XNS on the subinterface by specifying the XNS network number, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **`xns network`** [*`number`*]	Enables XNS routing on the subinterface.

Configuring CLNS Routing over ISL

CLNS can be routed over VLAN subinterfaces using the ISL VLAN encapsulation protocol. The CLNS Routing over ISL Virtual LANs feature provides full-feature Cisco IOS software CLNS support on a per-VLAN basis, allowing standard CLNS capabilities to be configured on VLANs.

To route CLNS over ISL VLANs, you need to configure ISL encapsulation on the subinterface. Perform the tasks described in the following sections in the order in which they appear:

- Enabling CLNS Routing
- Defining the VLAN Encapsulation Format
- Configuring CLNS on the Subinterface

Enabling CLNS Routing

To configure CLNS routing, use the following command in global configuration mode:

Command	Purpose
`Router(config)# clns routing`	Enables CLNS routing globally.

Defining the VLAN Encapsulation Format

To define the VLAN encapsulation format, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)# interface type slot/port.subinterface-number`	Specifies the subinterface on which ISL will be used.
Step 2	`Router(config-if)# encapsulation isl vlan-identifier`	Defines the encapsulation format as ISL (**isl**), and specifies the VLAN identifier.

Configuring CLNS on the Subinterface

To enable CLNS on the subinterface by specifying the CLNS network number, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)# clns enable`	Enables CLNS routing on the subinterface.

Configuring IS-IS Routing over ISL

IS-IS routing can be enabled over VLAN subinterfaces using the ISL VLAN encapsulation protocol. The IS-IS Routing over ISL Virtual LANs feature provides full-feature Cisco IOS software IS-IS support on a per-VLAN basis, allowing standard IS-IS capabilities to be configured on VLANs.

To enable IS-IS over ISL VLANs, you need to configure ISL encapsulation on the subinterface. Perform the tasks described in the following sections in the order in which they appear:

- Enabling IS-IS Routing
- Defining the VLAN Encapsulation Format
- Configuring IS-IS on the Subinterface

Enabling IS-IS Routing

To configure IS-IS routing, use the following command in global configuration mode:

	Command	Purpose
Step 1	`Router(config)#` **`router isis`** `[`*`tag`*`]`	Enables IS-IS routing, and enters router configuration mode.
Step 2	`Router(config)#` **`net network-entity-title`**	Configures the NET for the routing process.

Defining the VLAN Encapsulation Format

To define the VLAN encapsulation format, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	`Router(config-if)#` **`interface`** *`type slot/port.subinterface-number`*	Specifies the subinterface on which ISL will be used.
Step 2	`Router(config-if)#` **`encapsulation isl`** *`vlan-identifier`*	Defines the encapsulation format as ISL (**isl**), and specifies the VLAN identifier.

Configuring IS-IS on the Subinterface

To enable IS-IS on the subinterface by specifying the IS-IS network number, use the following command in interface configuration mode:

Command	Purpose
`Router(config-if)#` **`clns router isis network`** `[`*`tag`*`]`	Specifies the interfaces that should be actively routing IS-IS.

Monitoring and Maintaining VLAN Subinterfaces

To indicate whether a VLAN is a native VLAN, use the following command in privileged EXEC mode:

Command	Purpose
`Router#` **`show vlans`**	Displays VLAN subinterfaces.

ISL Encapsulation Configuration Examples

This section provides the following configuration examples for each of the protocols described in this chapter:

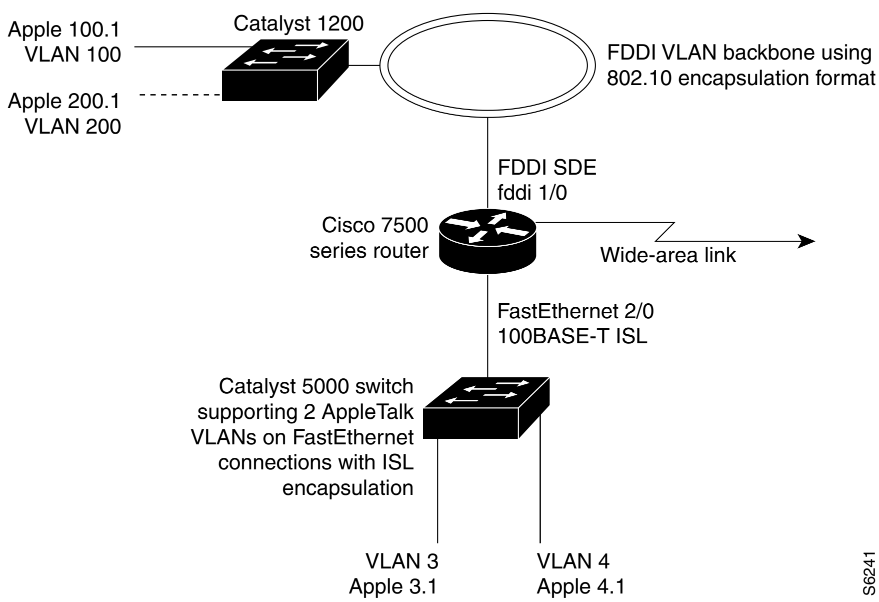
- AppleTalk Routing over ISL Configuration Examples
- Banyan VINES Routing over ISL Configuration Example
- DECnet Routing over ISL Configuration Example

- HSRP over ISL Configuration Example
- IP Routing with RIF Between TrBRF VLANs Example
- IP Routing Between a TRISL VLAN and an Ethernet ISL VLAN Example
- IPX Routing over ISL Configuration Example
- IPX Routing on FDDI Interfaces with SDE Example
- Routing with RIF Between a TRISL VLAN and a Token Ring Interface Example
- VIP Distributed Switching over ISL Configuration Example
- XNS Routing over ISL Configuration Example
- CLNS Routing over ISL Configuration Example
- IS-IS Routing over ISL Configuration Example

AppleTalk Routing over ISL Configuration Examples

The configuration example illustrated in Figure 80 shows AppleTalk being routed between different ISL and IEEE 802.10 VLAN encapsulating subinterfaces.

Figure 80 Routing AppleTalk over VLAN Encapsulations



As shown in Figure 80, AppleTalk traffic is routed to and from switched VLAN domains 3, 4, 100, and 200 to any other AppleTalk routing interface. This example shows a sample configuration file for the Cisco 7500 series router with the commands entered to configure the network shown in Figure 80.

Cisco 7500 Router Configuration

```
!
appletalk routing
interface Fddi 1/0.100
 encapsulation sde 100
```

```
 appletalk cable-range 100-100 100.2
 appletalk zone 100
!
interface Fddi 1/0.200
 encapsulation sde 200
 appletalk cable-range 200-200 200.2
 appletalk zone 200
!
interface FastEthernet 2/0.3
 encapsulation isl 3
 appletalk cable-range 3-3 3.2
 appletalk zone 3
!
interface FastEthernet 2/0.4
 encapsulation isl 4
 appletalk cable-range 4-4 4.2
 appletalk zone 4
!
```

Banyan VINES Routing over ISL Configuration Example

To configure routing of the Banyan VINES protocol over ISL trunks, you need to define ISL as the encapsulation type. This example shows Banyan VINES configured to be routed over an ISL trunk:

```
vines routing
interface fastethernet 0.1
 encapsulation isl 100
 vines metric 2
```

DECnet Routing over ISL Configuration Example

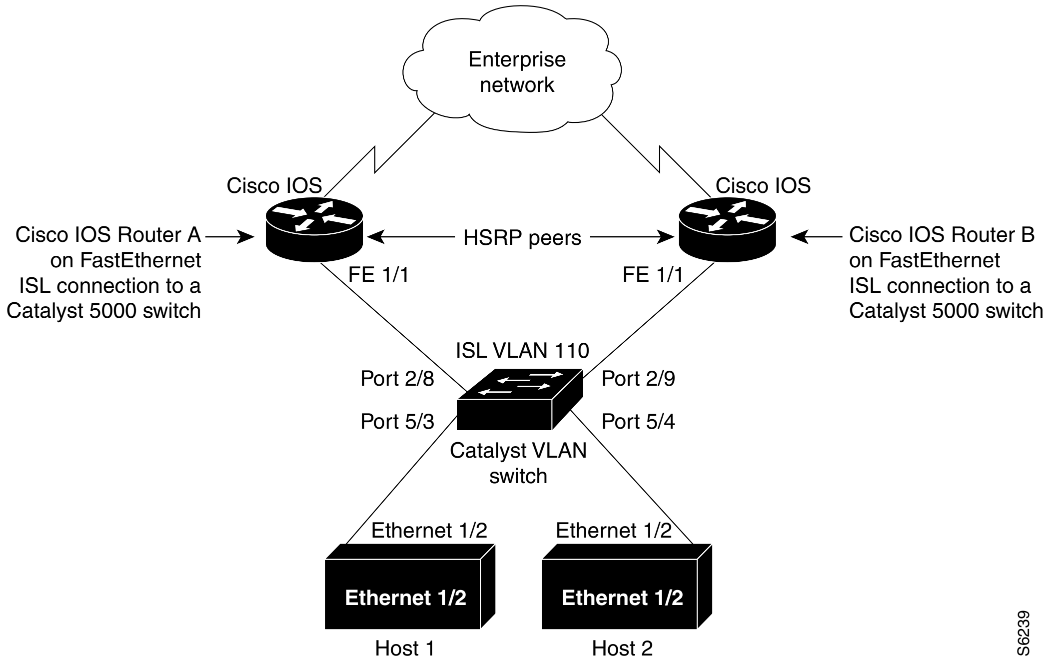
To configure routing the DECnet protocol over ISL trunks, you need to define ISL as the encapsulation type. This example shows DECnet configured to be routed over an ISL trunk:

```
decnet routing 2.1
interface fastethernet 1/0.1
 encapsulation isl 200
 decnet cost 4
```

HSRP over ISL Configuration Example

The configuration example shown in Figure 81 shows HSRP being used on two VLAN routers sending traffic to and from ISL VLANs through a Catalyst 5000 switch. Each router forwards its own traffic and acts as a standby for the other.

Figure 81 Hot Standby Router Protocol Sample Configuration



The topology shown in Figure 81 shows a Catalyst VLAN switch supporting Fast Ethernet connections to two routers running HSRP. Both routers are configured to route HSRP over ISLs.

The standby conditions are determined by the standby commands used in the configuration. Traffic from Host 1 is forwarded through Router A. Because the priority for the group is higher, Router A is the active router for Host 1. Because the priority for the group serviced by Host 2 is higher in Router B, traffic from Host 2 is forwarded through Router B, making Router B its active router.

In the configuration shown in Figure 81, if the active router becomes unavailable, the standby router assumes active status for the additional traffic and automatically routes the traffic normally handled by the router that has become unavailable.

Host 1 Configuration

```
interface Ethernet 1/2
 ip address 10.1.1.25 255.255.255.0
 ip route 0.0.0.0 0.0.0.0 10.1.1.101
```

Host 2 Configuration

```
interface Ethernet 1/2
 ip address 10.1.1.27 255.255.255.0
 ip route 0.0.0.0 0.0.0.0 10.1.1.102
!
```

Router A Configuration

```
interface FastEthernet 1/1.110
 encapsulation isl 110
 ip address 10.1.1.2 255.255.255.0
 standby 1 ip 10.1.1.101
 standby 1 preempt
 standby 1 priority 105
 standby 2 ip 10.1.1.102
 standby 2 preempt
```

```
!
end

!
```

Router B Configuration

```
interface FastEthernet 1/1.110
 encapsulation isl 110
 ip address 10.1.1.3 255.255.255.0
 standby 1 ip 10.1.1.101
 standby 1 preempt
 standby 2 ip 10.1.1.102
 standby 2 preempt
 standby 2 priority 105
router igrp 1
!
network 10.1.0.0
network 10.2.0.0
!
```

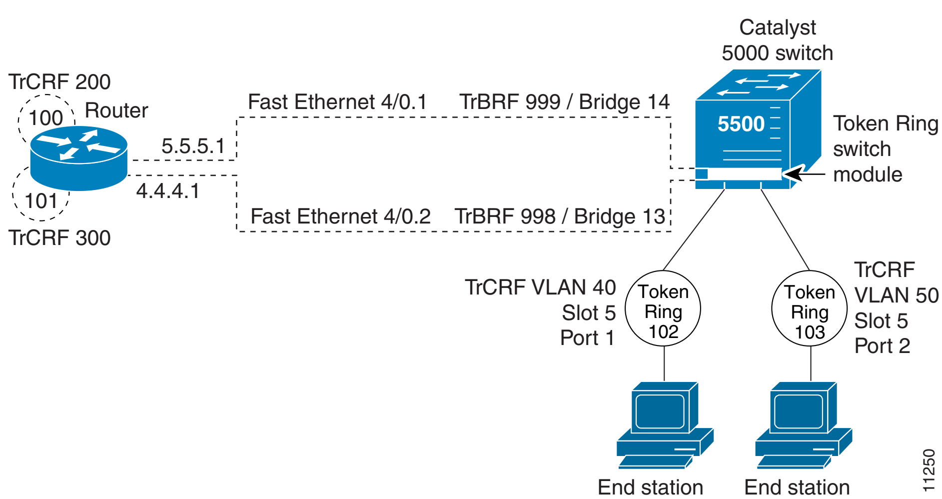
VLAN Switch Configuration

```
set vlan 110 5/4
set vlan 110 5/3
set trunk 2/8 110
set trunk 2/9 110
```

IP Routing with RIF Between TrBRF VLANs Example

Figure 82 shows IP routing with RIF between two TrBRF VLANs.

Figure 82 IP Routing with RIF Between TrBRF VLANs



The following is the configuration for the router:

```
interface FastEthernet4/0.1
  ip address 10.5.5.1 255.255.255.0
  encapsulation tr-isl trbrf-vlan 999 bridge-num 14
  multiring trcrf-vlan 200 ring 100
  multiring all
```

```
!
  interface FastEthernet4/0.2
  ip address 10.4.4.1 255.255.255.0
  encapsulation tr-isl trbrf-vlan 998 bridge-num 13
  multiring trcrf-vlan 300 ring 101
  multiring all
```

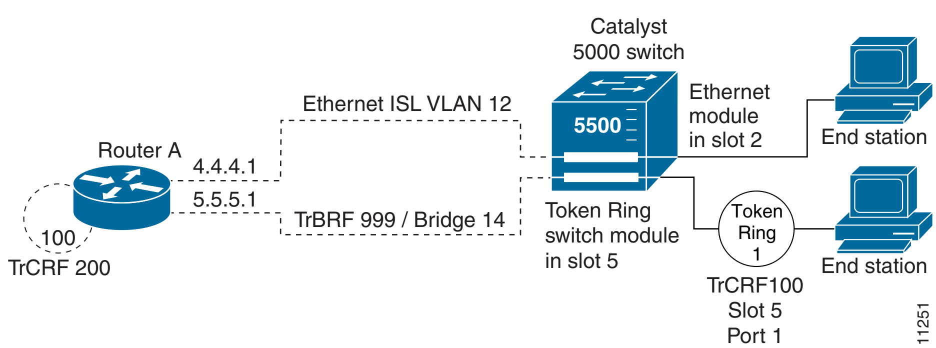
The following is the configuration for the Catalyst 5000 switch with the Token Ring switch module in slot 5. In this configuration, the Token Ring port 102 is assigned with TrCRF VLAN 40 and the Token Ring port 103 is assigned with TrCRF VLAN 50:

```
#vtp
set vtp domain trisl
set vtp mode server
set vtp v2 enable
#drip
set set tokenring reduction enable
set tokenring distrib-crf disable
#vlans
set vlan 999 name trbrf type trbrf bridge 0xe stp ieee
set vlan 200 name trcrf200 type trcrf parent 999 ring 0x64 mode srb
set vlan 40 name trcrf40 type trcrf parent 999 ring 0x66 mode srb
set vlan 998 name trbrf type trbrf bridge 0xd stp ieee
set vlan 300 name trcrf300 type trcrf parent 998 ring 0x65 mode srb
set vlan 50 name trcrf50 type trcrf parent 998 ring 0x67 mode srb
#add token port to trcrf 40
set vlan 40   5/1
#add token port to trcrf 50
set vlan 50   5/2
set trunk 1/2 on
```

IP Routing Between a TRISL VLAN and an Ethernet ISL VLAN Example

Figure 83 shows IP routing between a TRISL VLAN and an Ethernet ISL VLAN.

Figure 83 IP Routing Between a TRISL VLAN and an Ethernet ISL VLAN



The following is the configuration for the router:

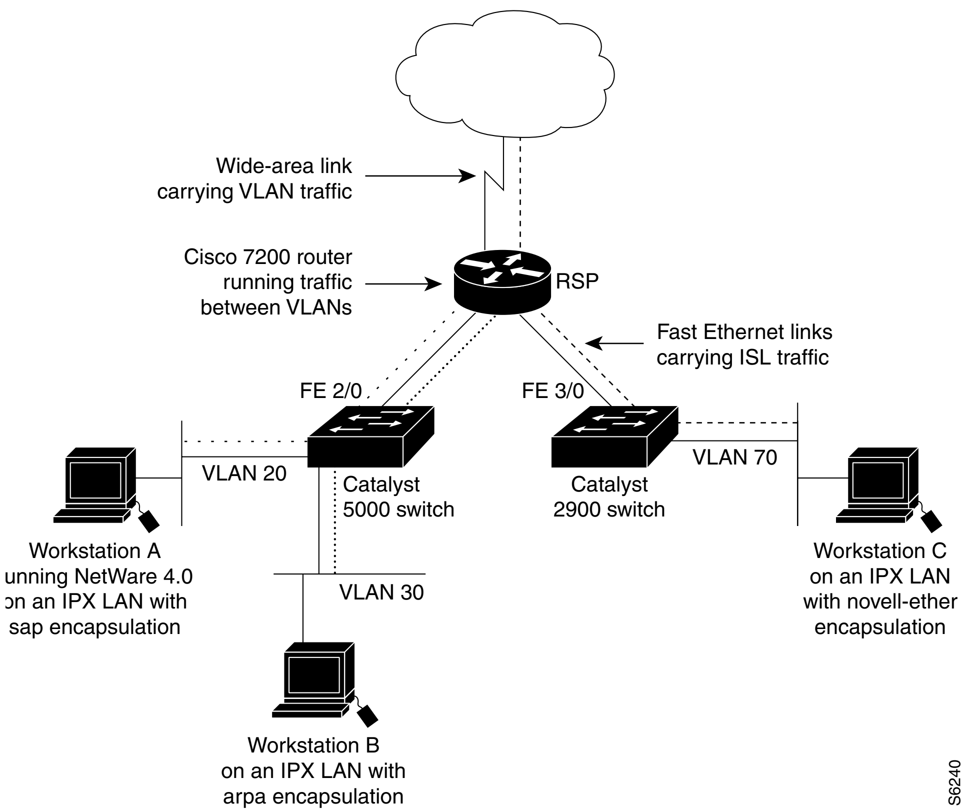
```
interface FastEthernet4/0.1
 ip address 10.5.5.1 255.255.255.0
 encapsulation tr-isl trbrf-vlan 999 bridge-num 14
 multiring trcrf-vlan 20 ring 100
 multiring all
!
```

```
interface FastEthernet4/0.2
 ip address 10.4.4.1 255.255.255.0
 encapsulation isl 12
```

IPX Routing over ISL Configuration Example

Figure 84 shows IPX interior encapsulations configured over ISL encapsulation in VLAN configurations. Note that three different IPX encapsulation formats are used. VLAN 20 uses SAP encapsulation, VLAN 30 uses ARPA, and VLAN 70 uses novell-ether encapsulation. Prior to the introduction of this feature, only the default encapsulation format, “novell-ether,” was available for routing IPX over ISL links in VLANs.

Figure 84 Configurable IPX Encapsulations Routed over ISL in VLAN Configurations



VLAN 20 Configuration

```
ipx routing
interface FastEthernet 2/0
 no shutdown
interface FastEthernet 2/0.20
 encapsulation isl 20
 ipx network 20 encapsulation sap
```

VLAN 30 Configuration

```
ipx routing
interface FastEthernet 2/0
 no shutdown
interface FastEthernet 2/0.30
 encapsulation isl 30
 ipx network 30 encapsulation arpa
```

VLAN 70 Configuration

```
ipx routing
interface FastEthernet 3/0
 no shutdown
interface Fast3/0.70
 encapsulation isl 70
 ipx network 70 encapsulation novell-ether
```

IPX Routing on FDDI Interfaces with SDE Example

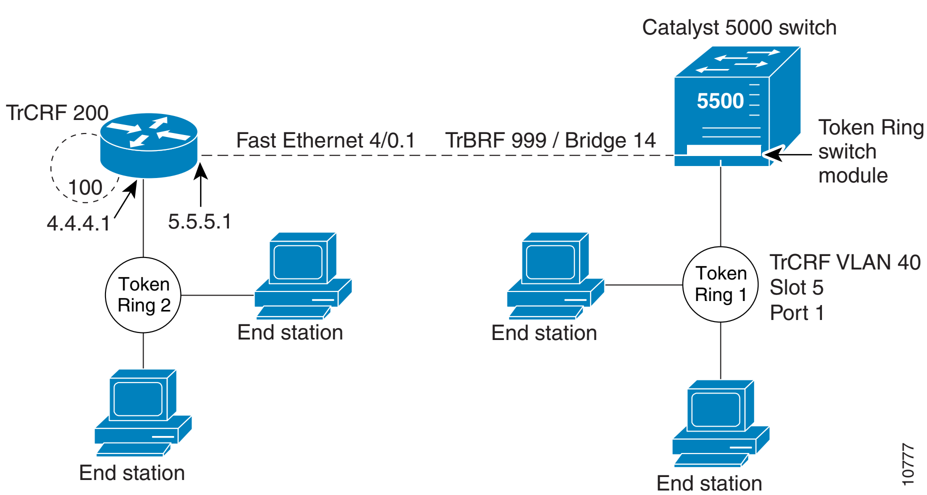
The following example enables IPX routing on FDDI interfaces 0.2 and 0.3 with SDE. On FDDI interface 0.2, the encapsulation type is SNAP. On FDDI interface 0.3, the encapsulation type is Novell's FDDI_RAW.

```
ipx routing

interface fddi 0.2 enc sde 2
 ipx network f02 encapsulation snap

interface fddi 0.3 enc sde 3
 ipx network f03 encapsulation novell-fddi
```

Routing with RIF Between a TRISL VLAN and a Token Ring Interface Example

Figure 85 shows routing with RIF between a TRISL VLAN and a Token Ring interface.

Figure 85 Routing with RIF Between a TRISL VLAN and a Token Ring Interface



The following is the configuration for the router:

```
source-bridge ring-group 100
!
interface TokenRing 3/1
  ip address 10.4.4.1 255.255.255.0
!
interface FastEthernet4/0.1
  ip address 10.5.5.1 255.255.255.0
  encapsulation tr-isl trbrf 999 bridge-num 14
  multiring trcrf-vlan 200 ring-group 100
  multiring all
```

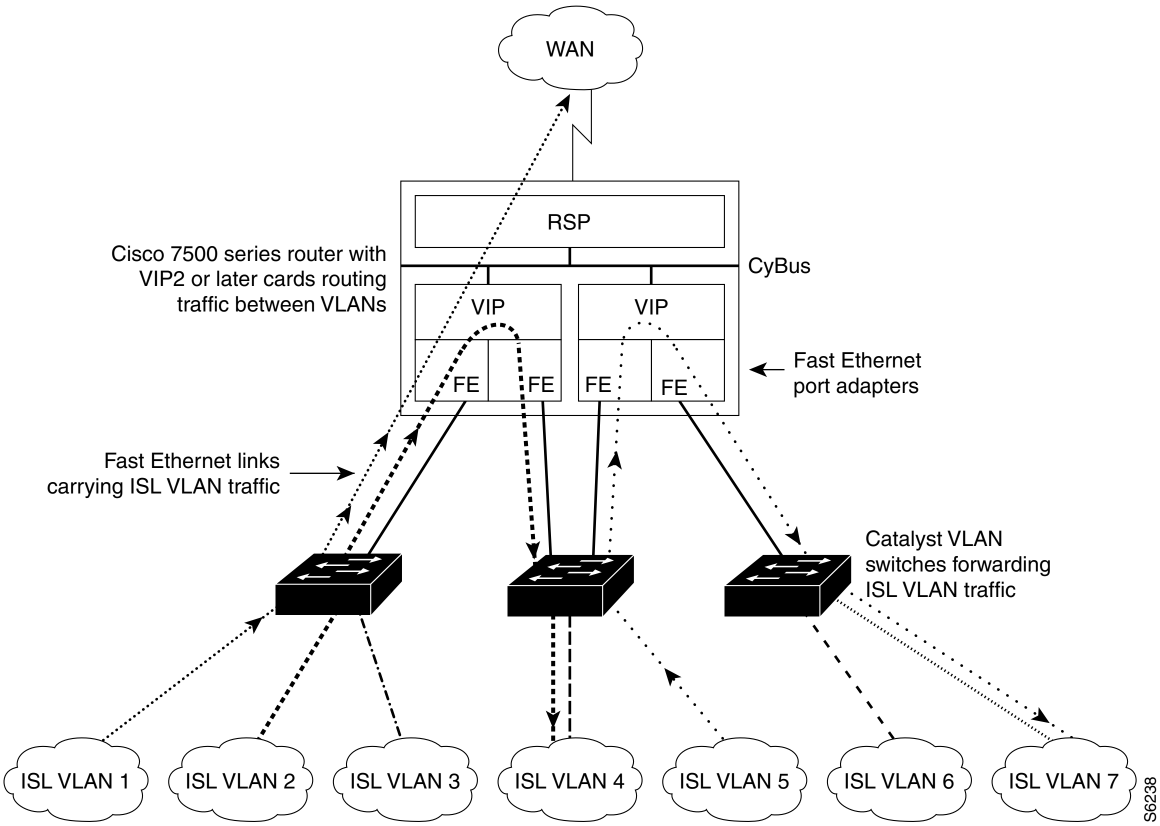
The following is the configuration for the Catalyst 5000 switch with the Token Ring switch module in slot 5. In this configuration, the Token Ring port 1 is assigned to the TrCRF VLAN 40:

```
#vtp
set vtp domain trisl
set vtp mode server
set vtp v2 enable
#drip
set set tokenring reduction enable
set tokenring distrib-crf disable
#vlans
set vlan 999 name trbrf type trbrf bridge 0xe stp ieee
set vlan 200 name trcrf200 type trcrf parent 999 ring 0x64 mode srt
set vlan 40 name trcrf40 type trcrf parent 999 ring 0x1 mode srt
#add token port to trcrf 40
set vlan 40   5/1
set trunk 1/2 on
```

VIP Distributed Switching over ISL Configuration Example

Figure 86 shows a topology in which Catalyst VLAN switches are connected to routers forwarding traffic from a number of ISL VLANs. With the VIP distributed ISL capability in the Cisco 7500 series router, each VIP card can route ISL-encapsulated VLAN IP traffic. The inter-VLAN routing capacity is increased linearly by the packet-forwarding capability of each VIP card.

Figure 86 VIP Distributed ISL VLAN Traffic



In Figure 86, the VIP cards forward the traffic between ISL VLANs or any other routing interface. Traffic from any VLAN can be routed to any of the other VLANs, regardless of which VIP card receives the traffic.

These commands show the configuration for each of the VLANs shown in Figure 86:

```
interface FastEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
 ip route-cache distributed
 full-duplex

interface FastEthernet1/0/0.1
 ip address 10.1.1.1 255.255.255.0
 encapsulation isl 1

interface FastEthernet1/0/0.2
 ip address 10.1.2.1 255.255.255.0
 encapsulation isl 2

interface FastEthernet1/0/0.3
 ip address 10.1.3.1 255.255.255.0
 encapsulation isl 3

interface FastEthernet1/1/0
 ip route-cache distributed
 full-duplex

interface FastEthernet1/1/0.1
 ip address 172.16.1.1 255.255.255.0
 encapsulation isl 4
```

```
interface Fast Ethernet 2/0/0
 ip address 10.1.1.1 255.255.255.0
 ip route-cache distributed
 full-duplex

interface FastEthernet2/0/0.5
 ip address 10.2.1.1 255.255.255.0
 encapsulation isl 5

interface FastEthernet2/1/0
 ip address 10.3.1.1 255.255.255.0
 ip route-cache distributed
 full-duplex

interface FastEthernet2/1/0.6
 ip address 10.4.6.1 255.255.255.0
 encapsulation isl 6

interface FastEthernet2/1/0.7
 ip address 10.4.7.1 255.255.255.0
 encapsulation isl 7
```

XNS Routing over ISL Configuration Example

To configure routing of the XNS protocol over ISL trunks, you need to define ISL as the encapsulation type. This example shows XNS configured to be routed over an ISL trunk:

```
xns routing 0123.4567.adcb
interface fastethernet 1/0.1
 encapsulation isl 100
 xns network 20
```

CLNS Routing over ISL Configuration Example

To configure routing of the CLNS protocol over ISL trunks, you need to define ISL as the encapsulation type. This example shows CLNS configured to be routed over an ISL trunk:

```
clns routing
interface fastethernet 1/0.1
 encapsulation isl 100
 clns enable
```

IS-IS Routing over ISL Configuration Example

To configure IS-IS routing over ISL trunks, you need to define ISL as the encapsulation type. This example shows IS-IS configured over an ISL trunk:

```
isis routing test-proc2
net 49.0001.0002.aaaa.aaaa.aaaa.00
interface fastethernet 2.0
 encapsulation isl 101
 clns router is-is test-proc2
```