

Configuring Legacy DDR Spokes

This chapter describes how to configure legacy dial-on-demand routing (DDR) on interfaces that function as a *spoke* in a hub-and-spoke network topology. It includes the following main sections:

- [DDR Spokes Configuration Task Flow](#)
- [How to Configure DDR](#)
- [Monitoring DDR Connections](#)
- [Configuration Examples for Legacy DDR Spoke](#)

This chapter considers a spoke interface to be any interface that calls or receives calls from exactly one other router, and considers a hub interface to be an interface that calls or receives calls from more than one router: all the spokes in the network.

This chapter also describes the DDR-independent tasks required to bridge protocols or to route protocols over DDR. Most of these tasks are global in scope and can be completed before you begin to configure DDR.

For configuration tasks for the central hub interface in a hub-and-spoke network topology, see the chapter “Configuring a Legacy DDR Hub” in this publication.

For information about the Dialer Profiles implementation of DDR, see the chapter “Configuring Peer-to-Peer DDR with Dialer Profiles” in this publication.

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 “Identifying Supported Platforms” section in the “Using Cisco IOS Software” chapter.

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

DDR Spokes Configuration Task Flow

Before you configure DDR, make sure you have completed the preparations for bridging or routing as described in the chapter “[Preparing to Configure DDR](#)” in this publication. That chapter provides information about the minimal requirements. For detailed information about bridging, routing, and wide-area networking configurations, refer to the appropriate chapters in other volumes of this documentation set.

When you configure DDR on a spoke interface in a hub-and-spoke topology, you perform the following general steps:

-
- Step 1** Specify the interface that will place calls to or receive calls from a single site. (See the chapter “Configuring Legacy DDR Hubs” in this publication for information about configuring an interface to place calls to or receive calls from multiple sites.)
 - Step 2** Enable DDR on the interface. This step is not required for some interfaces; for example, ISDN interfaces and passive interfaces that receive only from DTR-dialing interfaces.
 - Step 3** Configure the interface to receive calls only, if applicable. Receiving calls from multiple sites requires each inbound call to be authenticated.
 - Step 4** Configure the interface to place calls only, if applicable.
 - Step 5** Configure the interface to place and receive calls, if applicable.
 - Step 6** If the interface will place calls, specify access control for:
 - Transparent bridging—Assign the interface to a bridge group, and define dialer lists associated with the bridging access lists. The interface switches between members of the same bridge group, and dialer lists specify which packets can trigger calls.
 - or
 - Routed protocols—Define dialer lists associated with the protocol access lists to specify which packets can trigger calls.
 - Step 7** Customize the interface settings (timers, interface priority, hold queues, bandwidth on demand, and disabling fast switching) as needed.

When you have configured the interface and it is operational, you can monitor its performance and its connections as described in the “[Monitoring DDR Connections](#)” section later in this chapter.

You can also enhance DDR by configuring Multilink PPP and configuring PPP callback. The PPP configuration tasks are described in the chapter “Configuring Media-Independent PPP and Multilink PPP” in this publication.

See the section “[Configuration Examples for Legacy DDR Spoke](#)” later in this chapter for examples of how to configure DDR on your network.

How to Configure DDR

To configure DDR on an interface, perform the tasks in the following sections. The first five bulleted items are required. The remaining tasks are not required, but might be necessary in your networking environment.

- [Specifying the Interface](#) (Required)
- [Enabling DDR on the Interface](#) (Required)
- [Configuring the Interface to Place Calls](#) (Required)
- or
- [Configuring the Interface to Receive Calls](#) (Required)
- or
- [Configuring the Interface to Place and Receive Calls](#) (Required)
- [Defining the Traffic to Be Authenticated](#) (As required)

- [Configuring Access Control for Outgoing Calls](#) (As required)
- [Configuring Access Control for Bridging](#) (As required)
- [Configuring Access Control for Routing](#) (As required)
- [Customizing the Interface Settings](#) (As required)
- [Sending Traffic over Frame Relay, X.25, or LAPB Networks](#) (As required)

You can also monitor DDR connections. See the “[Monitoring DDR Connections](#)” section later in this chapter for commands and other information.

For examples of legacy DDR on a point-to-point connection, see the “[Configuration Examples for Legacy DDR Spoke](#)” section later in this chapter.

Specifying the Interface

This section assumes that you have completed any preparatory steps required for the relevant interface. For example, if you intend to use an asynchronous interface, it assumes that you have completed the modem support and line configuration steps and the chat script creation steps. If you intend to use an ISDN interface, it assumes that you have the ISDN line properly provisioned and running.

You can configure any asynchronous, synchronous serial, ISDN, or dialer interface for legacy DDR.



Note

When you specify an interface, make sure to use the interface numbering scheme supported on the network interface module or other port hardware on the router. On the Cisco 7200 series, for example, you specify an interface by indicating its type, slot number, and port number.

To specify an interface to configure for DDR, use one of the following commands in global configuration mode:

Command	Purpose
<pre>Router(config)# interface async <i>number</i> Router(config)# interface serial <i>number</i> Router(config)# interface bri <i>number</i> or Router(config)# interface serial <i>slot/port:23</i> Router(config)# interface serial <i>slot/port:15</i> or Router(config)# interface dialer <i>number</i></pre>	<p>Specifies an interface to configure for DDR.</p> <p>Specifies an ISDN PRI D channel (T1).</p> <p>Specifies an ISDN PRI D channel (E1).</p> <p>Specifies a logical interface to function as a dialer rotary group leader.</p>

Dialer interfaces are logical or virtual entities, but they use physical interfaces to place or receive calls.

Enabling DDR on the Interface

This task is required for asynchronous or synchronous serial interfaces but not for ISDN interfaces. The software automatically configures ISDN interfaces to be dialer type ISDN.

This step is not required for ISDN interfaces (BRI interfaces and ISDN PRI D channels) and for *purely passive* interfaces that will receive calls only from interfaces that use DTR dialing.

Enabling DDR on an interface usually requires you to specify the type of dialer to be used. This step is not required for ISDN interfaces because the software automatically configures ISDN interfaces to be dialer type ISDN.

To enable DDR and specify the dialer type, use one of the following commands in global configuration mode:

Command	Purpose
Router(config)# dialer dtr	Enables DDR and configures the specified serial interface to use DTR dialing—for interfaces with non-V.25bis modems using EIA Data Terminal Ready (DTR) signaling.
or	
Router(config)# dialer in-band [no-parity odd-parity]	Enables DDR and configures the specified serial interface to use in-band dialing—for asynchronous interfaces or interfaces using V.25bis modems.



Note

An interface configured with the **dialer in-band** command can both place and receive calls. A serial interface configured for DTR dialing can place calls only; it cannot accept them.

You can optionally specify parity if the modem on this interface uses the V.25bis command set. The 1984 version of the V.25bis specification states that characters must have odd parity. However, the default for the **dialer in-band** command is no parity.

For an example of configuring an interface to support DTR dialing, see the section “[DTR Dialing Example](#)” later in this chapter.

To receive calls from an interface that is using DTR dialing, an interface can be configured for in-band dialing or not configured for anything but encapsulation, depending on the desired behavior. If you expect the receiving interface to terminate a call when no traffic is received for some time, you must configure in-band dialing (along with access lists and a dummy dialer string). If the receiving interface is purely passive, no additional configuration is necessary.



Note

You can configure an interface or dialer rotary group to both place and receive calls. If the interface is calling and being called by a single site, simply enable DDR and specify a dial string.

Configuring the Interface to Place Calls

To configure an interface to place calls to one site only, perform the tasks in one of the following sections:

- [Specifying the Dial String for Synchronous Serial Interfaces](#) (As required)
- [Specifying Chat Scripts and Dial Strings for Asynchronous Serial Interfaces](#) (As required)

Specifying the Dial String for Synchronous Serial Interfaces

If you want to call only one remote system per synchronous serial interface, use the **dialer string** command. Dialers pass the string you have defined to the external DCE device. ISDN devices call the number specified in the string.

To specify the telephone number call on a serial interface (asynchronous or synchronous), use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer string <i>dial-string[:isdn-subaddress]</i>	Specifies the number to dial.

Dialers pass the string (telephone number) to the external DCE device, which dials the number; ISDN devices themselves call the specified number.

Specifying Chat Scripts and Dial Strings for Asynchronous Serial Interfaces

The modem chat script becomes the default chat script for an interface, which means it becomes the default chat script for the **dialer string** and **dialer map** commands presented in this section.

To place a call to a single site on an asynchronous line for which either a modem dialing script has not been assigned or a system script login must be specified, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer map <i>protocol next-hop-address</i> [modem-script <i>modem-regexp</i>] [system-script <i>system-regexp</i>] <i>dial-string</i> [: <i>isdn-subaddress</i>]	Specifies chat scripts and a dial string.

Refer to the sections “[How To Configure Chat Scripts](#)” and “[Dialer Mapping Example](#)” in the chapter “[Creating and Using Modem Chat Scripts](#)” for more information about configuring chat scripts.

Configuring the Interface to Receive Calls

If you enable DDR on an interface by using the **dialer in-band** command, the interface can receive calls. No additional configuration steps are required simply to receive calls. Parity is not required for receiving calls only. An interface configured with the **dialer in-band** command can terminate calls when the line is idle for some configurable time.

You cannot set up an ISDN interface only to receive calls from a single site, but you can set it up to receive and place calls to a single site.

To receive calls from an interface that is using DTR dialing, an interface can be configured for in-band dialing or not configured for anything but encapsulation, depending on the desired behavior. If you expect the receiving interface to terminate a call when no traffic is received for some time, you must configure in-band dialing (along with access lists and a dummy dialer string). If the receiving interface is purely passive, no additional configuration is necessary.

Authentication is not required when traffic comes from only one site. However, you can configure authentication for security. See the “[Defining the Traffic to Be Authenticated](#)” section. If you want to receive calls *only*, do not provide a dial string in the **dialer map** command shown in that section.

Configuring the Interface to Place and Receive Calls

If you enable DDR on an interface by using the **dialer in-band** command, the interface can receive calls. To enable it to place calls to one site, you must define the dialing destination.

To define the dialing destination, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer string <i>dial-string[:isdn-subaddress]</i>	Specifies the number to dial one site.



Note

Use the **dialer map** command instead of the **dialer string** command if you want to authenticate calls received. See the section “[Defining the Traffic to Be Authenticated](#)” later in this chapter for more information.

When a dialer string is configured but PPP Challenge Handshake Authentication Protocol (CHAP) is not configured on the interface, the Cisco IOS software recognizes each incoming call as coming from the configured dialer string. That is, if your outgoing calls go to only one number and you do not authenticate incoming calls, it is assumed that all incoming calls come from that number. (If you received calls from multiple sites, you would need to authenticate the calls.)

Authentication is not required when traffic comes from only one site. However, you can configure authentication for an extra measure of security. See the following section, “[Defining the Traffic to Be Authenticated](#),” for more information. If you want to receive and place calls, use the **dialer map** command.

Defining the Traffic to Be Authenticated

Authentication can be done through CHAP or Password Authentication Protocol (PAP). In addition, the interface must be configured to map the protocol address of the host to the name to use for authenticating the remote host.

To enable CHAP or PAP on an interface and authenticate sites that are calling in, use the following commands in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# encapsulation ppp	Configures an interface for PPP encapsulation.
Step 2	Router(config-if)# ppp authentication chap [if-needed] or	Enables CHAP.
	Router(config-if)# ppp authentication pap [if-needed]	Enables PAP.
Step 3	Router(config-if)# dialer map protocol <i>next-hop-address name hostname</i> [modem-script <i>modem-regex</i>] [system-script <i>system-regex</i>] [<i>dial-string[:isdn-subaddress]</i>]	Maps the protocol address to a host name.

If the dial string is not provided in the chat script, the interface will be able to receive calls from the host but will not be able to place calls to the host.

Configuring Access Control for Outgoing Calls

Protocol access lists and dialer access lists are central to the operation of DDR. In general, access lists are used as the screening criteria for determining when to initiate DDR calls. All packets are tested against the dialer access list. Packets that match a permit entry are deemed *interesting*. Packets that do not match a permit entry or that do match a deny entry are deemed *uninteresting*. When a packet is found to be interesting, either the dialer idle timer is reset (if the line is active) or a connection is attempted (if the line is available but not active). If a tested packet is deemed *uninteresting*, it will be forwarded if it is intended for a destination known to be on a specific interface and the link is active. However, such a packet will not initiate a DDR call and will not reset the idle timer.

Configuring Access Control for Bridging

You can control access by defining any transparent bridge packet as *interesting*, or you can use the finer granularity of controlling access by Ethernet type codes. To control access for DDR bridging, perform one of the following tasks in global configuration mode:

- [Controlling Bridging Access by Ethernet Type Codes](#) (As required)
- [Permitting All Bridge Packets to Trigger Calls](#) (As required)
- [Assigning the Interface to a Bridge Group](#) (As required)



Note

Spanning-tree bridge protocol data units (BPDUs) are always treated as *uninteresting*.

Controlling Bridging Access by Ethernet Type Codes

To control access by Ethernet type codes, use the following command in global configuration mode:

Command	Purpose
Router(config)# access-list <i>access-list-number</i> { permit deny } <i>type-code</i> [<i>mask</i>]	Identifies <i>interesting</i> packets by Ethernet type codes (access list numbers must be in the range 200 to 299).

To enable packets with a specified Ethernet type code to trigger outgoing calls, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer-list <i>dialer-group</i> protocol bridge list <i>access-list-number</i>	Defines a dialer list for the specified access list.

For a table of some common Ethernet types codes, see the “Ethernet Types Codes” appendix in the *Cisco IOS Bridging and IBM Networking Command Reference*.

Permitting All Bridge Packets to Trigger Calls

To identify all transparent bridge packets as interesting, use the following command in interface configuration mode when you are configuring DDR:

Command	Purpose
Router(config-if)# dialer-list <i>dialer-group</i> protocol bridge permit	Defines a dialer list that treats all transparent bridge packets as <i>interesting</i> .

Assigning the Interface to a Bridge Group

Packets are bridged only among interfaces that belong to the same bridge group. To assign an interface to a bridge group, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# bridge-group <i>bridge-group</i>	Assigns the specified interface to a bridge group.

Configuring Access Control for Routing

Before you perform the tasks outlined in this section, configure access lists for the protocols you intend to route over DDR as described briefly in the chapter “[Preparing to Configure DDR](#)” in this publication, and as described in greater detail in the appropriate network protocol configuration guide (for example, the *Cisco IOS AppleTalk and Novell IPX Configuration Guide*).

An interface can be associated only with a single dialer access group; multiple dialer access group assignments are not allowed. To specify the dialer access group to which you want to assign an access list, use the following command in interface configuration mode:

Command	Purpose
Router (config-if) # dialer-group <i>group-number</i>	Specifies the number of the dialer access group to which the specific interface belongs.

Customizing the Interface Settings

To customize DDR in your network, perform the tasks in the following sections as needed:

- [Configuring Timers on the DDR Interface](#) (As required)
- [Setting Dialer Interface Priority](#) (As required)
- [Configuring a Dialer Hold Queue](#) (As required)
- [Configuring Bandwidth on Demand](#) (As required)
- [Disabling and Reenabling DDR Fast Switching](#) (As required)
- [Configuring Dialer Redial Options](#) (As required)

Configuring Timers on the DDR Interface

To set the timers, perform the tasks in the following sections as needed:

- [Setting Line-Idle Time](#) (As required)
- [Setting Idle Time for Busy Interfaces](#) (As required)
- [Setting Line-Down Time](#) (As required)
- [Setting Carrier-Wait Time](#) (As required)

Setting Line-Idle Time

To specify the amount of time for which a line will stay idle before it is disconnected, use the following command in interface configuration mode:

Command	Purpose
Router (config-if) # dialer idle-timeout <i>seconds</i> [inbound either]	Specifies the duration of idle time in seconds after which a line will be disconnected. By default, outbound traffic will reset the dialer idle timer. Adding the either keyword causes both inbound and outbound traffic to reset the timer; adding the inbound keyword causes only inbound traffic to reset the timer.



Note

The **dialer idle-timeout** interface configuration command specifies the duration of time before an idle connection is disconnected. Previously, both inbound and outbound traffic would reset the dialer idle timer; now you can specify that only inbound traffic will reset the dialer idle timer.

Setting Idle Time for Busy Interfaces

The dialer fast idle timer is activated if there is contention for a line. Contention occurs when a line is in use, a packet for a different next hop address is received, and the busy line is required to send the competing packet.

If the line has been idle for the configured amount of time, the current call is disconnected immediately and the new call is placed. If the line has not yet been idle as long as the fast idle timeout period, the packet is dropped because there is no way to get through to the destination. (After the packet is dropped, the fast idle timer remains active and the current call is disconnected as soon as it has been idle for as long as the fast idle timeout.) If, in the meantime, another packet is sent to the currently connected destination, and it is classified as interesting, the fast-idle timer is restarted.

To specify the amount of time for which a line for which there is contention will stay idle before the line is disconnected and the competing call is placed, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer fast-idle <i>seconds</i>	Sets idle time for high traffic lines.

This command applies to both inbound and outbound calls.

Setting Line-Down Time

To set the length of time for which the interface stays down before it is available to dial again after a line is disconnected or fails, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer enable-timeout <i>seconds</i>	Sets the interface downtime.

This command applies to both inbound and outbound calls.

Setting Carrier-Wait Time

To set the length of time for which an interface waits for the telephone service (carrier), use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer wait-for-carrier-time <i>seconds</i>	Sets the length of time for which the interface waits for the carrier to come up when a call is placed.

For asynchronous interfaces, this command sets the total time to wait for a call to connect. This time is set to allow for running the chat script.

Setting Dialer Interface Priority

Interface priority indicates which interface in a dialer rotary group will get used first for outgoing calls. You might give one interface a higher priority if it is attached to a faster, more reliable modem. In this way, the higher-priority interface will be used as often as possible.

To assign priority to an interface in a dialer rotary group, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer priority <i>number</i>	Sets the interface priority in the dialer rotary group.

The range of values for *number* is 0 through 255. Zero is the default value and lowest priority; 255 is the highest priority. This command applies to outgoing calls only.

Configuring a Dialer Hold Queue

Sometimes packets destined for a remote router are discarded because no connection exists. Establishing a connection using an analog modem can take time, during which packets are discarded. However, configuring a dialer hold queue will allow *interesting* outgoing packets to be queued and sent as soon as the modem connection is established.

A dialer hold queue can be configured on any type of dialer, including in-band synchronous, asynchronous, DTR, and ISDN dialers. Also, *hunt group leaders* can be configured with a dialer hold queue. If a hunt group leader (of a rotary dialing group) is configured with a hold queue, all members of the group will be configured with a dialer hold queue and no hold queue of an individual member can be altered.

To establish a dialer hold queue, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer hold-queue <i>packets</i>	Creates a dialer hold queue and specifies the number of packets to be held in it.

As many as 100 packets can be held in an outgoing dialer hold queue.

Configuring Bandwidth on Demand

You can configure a dialer rotary group to use additional bandwidth by placing additional calls to a single destination if the load for the interface exceeds a specified weighted value. Parallel communication links are established based on traffic load. The number of parallel links that can be established to one location is not limited.

To set the dialer load threshold for bandwidth on demand, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# dialer load-threshold <i>load</i>	Configures the dialer rotary group to place additional calls to a single destination, as indicated by interface load.

Once multiple links are established, they are still governed by the load threshold. If the total load on all the links falls below the threshold, an idle link will be torn down.

Disabling and Reenabling DDR Fast Switching

Fast switching is enabled by default on all DDR interfaces. When fast switching is enabled or disabled on an ISDN D channel, it is enabled or disabled on all B channels. When fast switching is enabled or disabled on a dialer interface, it is enabled or disabled on all rotary group members but cannot be enabled or disabled on the serial interfaces individually.

Fast switching can be disabled and re-enabled on a protocol-by-protocol basis. To disable fast switching and re-enable it, use one of the following protocol-specific commands in interface configuration mode:

Command	Purpose
Router(config-if)# no ip route-cache	Disables IP fast switching over a DDR interface.
Router(config-if)# ip route cache	Reenables IP fast switching over a DDR interface.
Router(config-if)# no ip route-cache distributed	Disables distributed IP fast switching over a DDR interface. This feature works in Cisco 7500 routers with a Versatile Interface Processor (VIP) card.
Router(config-if)# ip route-cache distributed	Enables distributed IP fast switching over a DDR interface. This feature works in Cisco 7500 routers with a VIP card.
Router(config-if)# no ipx route-cache	Disables IPX fast switching over a DDR interface.
Router(config-if)# ipx route-cache	Reenables IPX fast switching over a DDR interface.

Configuring Dialer Redial Options

By default, the Cisco IOS software generates a single dial attempt for each interesting packet. Dialer redial allows the dialer to be configured to make a maximum number of redial attempts if the first dial-out attempt fails, wait a specific interval between redial attempts, and disable the interface for a specified duration if all redial attempts fail. New dialout attempts will not be initiated if a redial is pending to the same destination.

To configure redial options, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	Router(config)# interface dialer	Enters interface configuration mode.
Step 2	Router(config-if)# dialer redial interval time attempts number re-enable disable-time	Configures redial options on the router.

Sending Traffic over Frame Relay, X.25, or LAPB Networks

An interface configured for DDR can send traffic over networks that require Link Access Procedure, Balanced (LAPB), X.25, or Frame Relay encapsulation.

Before Cisco IOS software Release 12.0(6)T, encapsulation techniques such as Frame Relay, HDLC, LAPB-TA, and X.25 could support only one ISDN B-channel connection over the entire link. HDLC and PPP could support multiple B channels, but the entire ISDN link needed to use the same encapsulation. The Dynamic Multiple Encapsulations feature allows incoming calls over ISDN to be assigned encapsulation type based on calling line identification (CLID) or DNIS. With the Dynamic Multiple Encapsulations feature, once CLID binding is completed, the topmost interface is always used for all

configuration and data structures. The ISDN B channel becomes a forwarding device, and the configuration on the D channel is ignored, thereby allowing the different encapsulation types and per-user configurations.

To configure an interface for those networks, perform the tasks in the following sections:

- [Configuring the Interface for Sending Traffic over a Frame Relay Network](#) (As required)
- [Configuring the Interface for Sending Traffic over an X.25 Network](#) (As required)
- [Configuring the Interface for Sending Traffic over a LAPB Network](#) (As required)

Configuring the Interface for Sending Traffic over a Frame Relay Network

Access to Frame Relay networks is now available through dialup connections as well as leased lines. Dialup connectivity allows Frame Relay networks to be extended to sites that do not generate enough traffic to justify leased lines, and also allows a Frame Relay network to back up another network or point-to-point line.

DDR over Frame Relay is supported for synchronous serial and ISDN interfaces and for rotary groups, and is available for in-band, DTR, and ISDN dialers.

Frame Relay supports multiple permanent virtual circuit (PVC) connections over the same serial interface or ISDN B channel, but only one *physical* interface can be used (dialed, connected, and active) in a rotary group or with ISDN.

The Dynamic Multiple Encapsulations feature supports the following Frame Relay features:

- Frame Relay RTP Header Compression (RFC 1889)
- Frame Relay TCP/IP Header Compression
- Legacy DDR over Frame Relay
- Frame Relay Interface/Subinterface Backup

Dynamic multiple encapsulations support at least four Frame Relay PVCs on either dialer interfaces or dialer subinterfaces.



Note

Frame Relay encapsulations in the Dynamic Multiple Encapsulations feature do not support IETF or Cisco Encapsulation for IBM Systems Network Architecture (SNA). Frame Relay for SNA support is not applicable.

Configuration Restrictions

The following restrictions apply to DDR used over Frame Relay:

- Frame Relay is not available for asynchronous dialers.
- The Frame Relay Dynamic Multiple Encapsulations feature does *not* provide bidirectional support.
- With the Dynamic Multiple Encapsulations feature, there is no process switching for Frame Relay packets; these packets are always fast switched.
- Like HDLC, LAPB, and X.25, Frame Relay does not provide authentication. However, ISDN dialers can offer some authentication through the caller ID feature.
- Only one ISDN B channel can be dialed at any one time. When configuring a rotary group, you can use only one serial interface.

Frame Relay subinterfaces work the same on dialup connections as they do on leased lines.

Configuration Overview

No new commands are required to support DDR over Frame Relay. In general, you configure Frame Relay and configure DDR. In general, complete the following tasks to configure an interface for DDR over Frame Relay:

- Specify the interface.
- Specify the protocol identifiers for the interface.

For example, enter the IP address and mask, the IPX network number, and the AppleTalk cable range and zone.

- Configure Frame Relay.

As a minimum, you must enable Frame Relay encapsulation and decide whether you need to do static or dynamic address mapping. If you decide to do dynamic mapping, you need not enter a command because Inverse Address Resolution Protocol is enabled by default. If you decide to do static mapping, you must enter Frame Relay mapping commands.

You can then configure various options as needed for your Frame Relay network topology.

- Configure DDR.

At a minimum, you must decide and configure the interface for outgoing calls only, incoming calls only, or both outgoing and incoming calls.

You can also configure DDR for your routed protocols (as specified in the section “Preparations for Routing or Bridging over DDR” in the chapter “[Preparing to Configure DDR](#)” in this publication) and for snapshot routing (as specified in the chapter “Configuring Snapshot Routing” later in this publication). You can also customize DDR interfaces on your router or access server (as described in the section “[Customizing the Interface Settings](#)” in this chapter).

For examples of configuring various interfaces for DDR over Frame Relay, see the section “[Frame Relay Support Example](#)” later in this chapter.

Configuring the Interface for Sending Traffic over an X.25 Network

X.25 interfaces can now be configured to support DDR. Synchronous serial and ISDN interfaces on Cisco routers and access servers can be configured for X.25 addresses, X.25 encapsulation, and mapping of protocol addresses to the X.25 address of a remote host. In-band, DTR, and ISDN dialers can be configured to support X.25 encapsulation, but rotary groups cannot.

Remember that for ISDN interfaces, once CLID binding is completed, the topmost interface is always used for all configuration and data structures. The ISDN B channel becomes a forwarding device, and the configuration on the D channel is ignored, thereby allowing the different encapsulation types and per-user configurations. For X.25 encapsulations, the configurations reside on the dialer profile. The Dynamic Multiple Encapsulations feature provides support for packet assembler/disassembler (PAD) traffic and X.25 encapsulated and switched packets.

To configure an interface to support X.25 and DDR, use the following X.25-specific commands in interface configuration mode:

	Command	Purpose
Step 1	Router(config-if)# encapsulation x25 [dte dce] [ietf]	Configures the interface to use X.25 encapsulation.

	Command	Purpose
Step 2	Router(config-if)# x25 address <i>x.121-address</i>	Assigns an X.25 address to the interface.
Step 3	Router(config-if)# x25 map <i>protocol address</i> [<i>protocol2 address2</i> [... [<i>protocol9 address9</i>]]] <i>x.121-address</i> [<i>option</i>]	Sets up the LAN protocols-to-remote host address mapping.

The order of DDR and X.25 configuration tasks is not critical; you can configure DDR before or after X.25, and you can even mix the DDR and X.25 commands.

For an example of configuring an interface for X.25 encapsulation and then completing the DDR configuration, see the section “[X.25 Support Example](#)” later in this chapter.

Configuring the Interface for Sending Traffic over a LAPB Network

DDR over serial lines now supports LAPB encapsulation, in addition to the previously supported PPP, HDLC, and X.25 encapsulations.

LAPB encapsulation is supported on synchronous serial, ISDN, and dialer rotary group interfaces, but not on asynchronous dialers.

Because the default encapsulation is HDLC, you must explicitly configure LAPB encapsulation. To configure an interface to support LAPB encapsulation and DDR, use the following command in interface configuration mode:

Command	Purpose
Router(config-if)# encapsulation lapb [<i>dte</i> <i>dce</i>] [<i>multi</i> <i>protocol</i>]	Specifies LAPB encapsulation.

For more information about the serial connections on which LAPB encapsulation is appropriate, refer to the **encapsulation lapb** command in the chapter “X.25 and LAPB Commands” in the *Cisco IOS Wide-Area Networking Command Reference*.

For an example of configuring an interface for DDR over LAPB, see the section “[LAPB Support Example](#)” later in this chapter.

Monitoring DDR Connections

To monitor DDR connections, use any of the following commands in privileged EXEC mode:

Command	Purpose
Router# show dialer [<i>interface type number</i>]	Displays general diagnostics about the DDR interface.
Router# show dialer map	Displays current dialer maps, next-hop protocol addresses, user names, and the interfaces on which they are configured.
Router# show interfaces bri 0	Displays information about the ISDN interface.
Router# show ipx interface [<i>type number</i>]	Displays status about the IPX interface.
Router# show ipx traffic	Displays information about the IPX packets sent by the router or access server, including watchdog counters.

Command	Purpose
Router# <code>show appletalk traffic</code>	Displays information about the AppleTalk packets sent by the router or access server.
Router# <code>show vines traffic</code>	Displays information about the Banyan VINES packets sent by the router or access server.
Router# <code>show decnet traffic</code>	Displays information about the DECnet packets sent by the router or access server.
Router# <code>show xns traffic</code>	Displays information about the XNS packets sent by the router or access server.
Router# <code>clear dialer</code>	Clears the values of the general diagnostic statistics.

Configuration Examples for Legacy DDR Spoke

The following section provides various DDR configurations examples:

- [Legacy Dial-on-Demand Routing Example](#)
- [Transparent Bridging over DDR Examples](#)
- [DDR Configuration in an IP Environment Example](#)
- [Two-Way DDR for Novell IPX Example](#)
- [AppleTalk Configuration Example](#)
- [DECnet Configuration Example](#)
- [ISO CLNS Configuration Example](#)
- [XNS Configuration Example](#)
- [Single Site Dialing Example](#)
- [DTR Dialing Example](#)
- [Hub-and-Spoke DDR for Asynchronous Interfaces and Authentication Example](#)
- [Two-Way Reciprocal Client/Server DDR Without Authentication Example](#)
- [Frame Relay Support Example](#)
- [X.25 Support Example](#)
- [LAPB Support Example](#)

Legacy Dial-on-Demand Routing Example

The following example shows a Cisco 2600 series router that has enabled the **dialer idle-timeout** command with the **inbound** keyword. This command allows only inbound traffic that conforms to the dialer list to establish a connection and reset the dialer idle timer.

```
interface BRI0/0
 ip address 10.1.1.1 255.255.255.0
 no ip directed-broadcast
 encapsulation ppp
 dialer idle-timeout 120 inbound
 dialer map ip 10.1.1.2 name 2611-7 0201
 dialer-group 1
```

```

    isdn switch-type basic-5ess
    no cdp enable
    ppp authentication chap
!
    ip classless
    ip route 10.2.1.1 255.255.255.255 10.1.1.2
!
access-list 101 permit icmp any any
access-list 101 deny ip any any
dialer-list 1 protocol ip list 101
tftp-server flash c2600-i-mz.jtong-CSCdm88145-120

```

Transparent Bridging over DDR Examples

The following two examples differ only in the packets that cause calls to be placed. The first example specifies by protocol (any bridge packet is permitted to cause a call to be made); the second example allows a finer granularity by specifying the Ethernet type codes of bridge packets.

The first example configures the serial 1 interface for DDR bridging. Any bridge packet is permitted to cause a call to be placed.

```

no ip routing
!
interface Serial1
    no ip address
    encapsulation ppp
    dialer in-band
    dialer enable-timeout 3
    dialer map bridge name urk broadcast 8985
    dialer hold-queue 10
    dialer-group 1
    ppp authentication chap
    bridge-group 1
    pulse-time 1
!
dialer-list 1 protocol bridge permit
bridge 1 protocol ieee
bridge 1 hello 10

```

The second example also configures the serial 1 interface for DDR bridging. However, this example includes an **access-list** command that specifies the Ethernet type codes that can cause calls to be placed and a **dialer list protocol list** command that refers to the specified access list.

```

no ip routing
!
interface Serial1
    no ip address
    encapsulation ppp
    dialer in-band
    dialer enable-timeout 3
    dialer map bridge name urk broadcast 8985
    dialer hold-queue 10
    dialer-group 1
    ppp authentication chap
    bridge-group 1
    pulse-time 1
!
access-list 200 permit 0x0800 0xFFFF8
!
dialer-list 1 protocol bridge list 200
bridge 1 protocol ieee
bridge 1 hello 10

```

DDR Configuration in an IP Environment Example

The following example illustrates how to use DDR on an synchronous interface in an IP environment. You could use the same configuration on an asynchronous serial interface by changing *interface serial 1* to specify an asynchronous interface (for example, *interface async 0*).

```
interface serial 1
ip address 172.18.126.1 255.255.255.0
dialer in-band
! The next command sets the dialer idle time-out to 10 minutes.
dialer idle-timeout 600
! The next command inserts the phone number.
dialer string 5551234
! The next command gives the modem enough time to recognize that
! DTR has dropped so the modem disconnects the call.
pulse-time 1
! The next command adds this interface to the dialer access group defined with
! the dialer-list command.
dialer-group 1
!
! The first access list statement, below, specifies that IGRP updates are not
! interesting packets. The second access-list statement specifies that all
! other IP traffic such as Ping, Telnet, or any other IP packet are interesting
! packets. The dialer-list command then creates dialer access group 1 and states
! that access list 101 is to be used to classify packets as interesting or
! uninteresting. The ip route commands specify that there is a route to network
! 172.18.29.0 and to network 172.18.1.0 via 131.108.126.2. This means that several
! destination networks are available through a router that is dialed from interface
! async 1.
!
access-list 101 deny igmp 0.0.0.0 255.255.255.255 255.255.255.255 0.0.0.0
access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
dialer-list 1 list 101
ip route 172.18.29.0 172.18.126.2
ip route 172.18.1.0 172.18.126.2
ip local pool dialin 10.102.126.2 10.102.126.254
```

With many modems, the **pulse-time** command must be used so that DTR is dropped for enough time to allow the modem to disconnect.

The **redistribute static** command can be used to advertise static route information for DDR applications. Refer to the **redistribute static ip** command, described in the chapter “IP Routing Commands” of the *Cisco IOS IP Command Reference*. Without this command, static routes to the hosts or network that the router can access with DDR will not be advertised to other routers with which the router is communicating. This behavior can block communication because some routes will not be known.

Two-Way DDR for Novell IPX Example

You can set DDR for Novell IPX so that both the client and server have dial-in access to each other. This configuration is demonstrated in the following two subsections.

Remote Configuration Example

The following example is performed on the remote side of the connection:

```
username local password secret
ipx routing
!
```

```

interface ethernet 0
 ipx network 40
!
interface async
 ip unnumbered e0
 encapsulation ppp
 async mode dedicated
 async dynamic routing
 ipx network 45
 ipx watchdog-spoof
 dialer in-band
 dialer map ipx 45.0000.0cff.d016 broadcast name local 1212
 dialer-group 1
 ppp authentication chap
!
access-list 901 deny 0 FFFFFFFF 452
access-list 901 deny 0 FFFFFFFF 453
access-list 901 deny 0 FFFFFFFF 457
access-list 901 deny 0 FFFFFFFF 0 FFFFFFFF 452
access-list 901 deny 0 FFFFFFFF 0 FFFFFFFF 453
access-list 901 deny 0 FFFFFFFF 0 FFFFFFFF 457
access-list 901 permit 0
ipx route 41 45.0000.0cff.d016
ipx route 50 45.0000.0cff.d016
ipx sap 4 SERVER 50.0000.0000.0001 451 2
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
!
dialer-list 1 list 901
!
line 7
 modem InOut
 speed 38400
 flowcontrol hardware
 modem chat-script generic

```

Local Configuration Example

The following example is performed on the local side of the connection:

```

username remote password secret
ipx routing
!
interface ethernet 0
 ipx network 41
!
interface async
 ip unnumbered e0
 encapsulation ppp
 async mode dedicated
 async dynamic routing
 ipx network 45
 ipx watchdog-spoof
 dialer in-band
 dialer map ipx 45.0000.0cff.d016 broadcast name remote 8888
 dialer-group 1
 ppp authentication chap
!
access-list 901 deny 0 FFFFFFFF 452
access-list 901 deny 0 FFFFFFFF 453
access-list 901 deny 0 FFFFFFFF 457
access-list 901 deny 0 FFFFFFFF 0 FFFFFFFF 452
access-list 901 deny 0 FFFFFFFF 0 FFFFFFFF 453
access-list 901 deny 0 FFFFFFFF 0 FFFFFFFF 457

```

```

access-list 901 permit 0
ipx route 40 45.0000.0cff.d016
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
!
dialer-list 1 list 901
!
line 7
modem InOut
speed 38400
flowcontrol hardware
modem chat-script generic

```

AppleTalk Configuration Example

The following example configures DDR for AppleTalk access using an ISDN BRI. Two access lists are defined: one for IP and Interior Gateway Routing Protocol (IGRP) and one for AppleTalk. AppleTalk packets from network 2141 only (except broadcast packets) can initiate calls.

```

interface BRI0
ip address 172.17.20.107 255.255.255.0
encapsulation ppp
appletalk cable-range 2141-2141 2141.65
appletalk zone SCruz-Eng
no appletalk send-rtmps
dialer map ip 172.17.20.106 broadcast 1879
dialer map appletalk 2141.66 broadcast 1879
dialer-group 1
!
access-list 101 deny igrp 0.0.0.0 255.255.255.255 255.255.255.255 0.0.0.0
access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
access-list 601 permit cable-range 2141-2141 broadcast-deny
access-list 601 deny other-access
!
dialer-list 1 list 101
dialer-list 1 list 601

```

DECnet Configuration Example

The following example configures DDR for DECnet:

```

decnet routing 10.19
username RouterB password 7 030752180531
interface serial 0
no ip address
decnet cost 10
encapsulation ppp
dialer in-band
dialer map decnet 10.151 name RouterB broadcast 4155551212
dialer-group 1
ppp authentication chap
pulse-time 1
access-list 301 permit 10.0 0.1023 0.0 63.1023
dialer-list 1 protocol decnet list 301

```

ISO CLNS Configuration Example

The following example configures a router for International Organization for Standardization Connectionless Network Service (ISO CLNS) DDR with in-band dialing:

```
username RouterB password 7 111C140B0E
clns net 47.0004.0001.0000.0c00.2222.00
clns routing
clns filter-set ddrline permit 47.0004.0001....
!
interface serial 0
 no ip address
 encapsulation ppp
 dialer in-band
 dialer map clns 47.0004.0001.0000.0c00.1111.00 name RouterB broadcast 1212
 dialer-group 1
 ppp authentication chap
 clns enable
 pulse-time 1
!
clns route default serial 0
dialer-list 1 protocol clns list ddrline
```

XNS Configuration Example

The following example configures DDR for XNS. The access lists deny broadcast traffic to any host on any network, but allow all other traffic.

```
xns routing 0000.0c01.d8dd

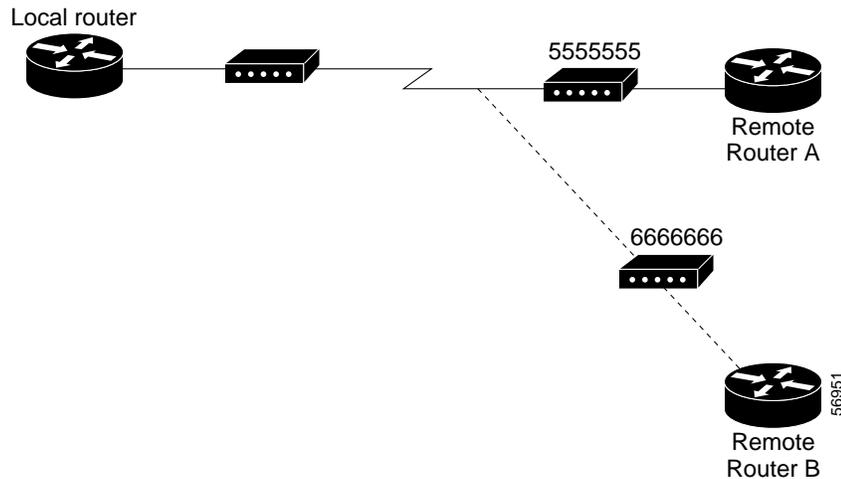
username RouterB password 7 111B210A0F

interface serial 0
 no ip address
 encapsulation ppp
 xns network 10
 dialer in-band
 dialer map xns 10.0000.0c01.d877 name RouterB broadcast 4155551212
 dialer-group 1
 ppp authentication chap
 pulse-time 1
!
access-list 400 deny -1 -1.ffff.ffff.ffff 0000.0000.0000
access-list 400 permit -1 10
!
dialer-list 1 protocol xns list 400
```

Single Site Dialing Example

The following example is based on the configuration shown in [Figure 49](#); the router receives a packet with a next hop address of 10.1.1.1.

Figure 49 Sample Dialer String or Dialer Map Configuration



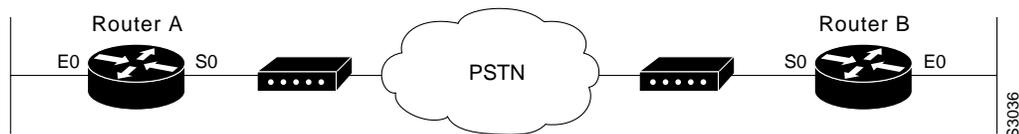
If the single site called by the DDR spoke interface on your router has the phone number 5555555, it will send the packet to that site, assuming that the next hop address 10.1.1.1 indicates the same remote device as phone number 5555555. The **dialer string** command is used to specify the string (telephone number) to be called.

```
interface serial 1
  dialer in-band
  dialer string 5555555
```

DTR Dialing Example

The following example shows Router A and Router B connected to a Public Switched Telephone Network (PSTN). Router A is configured for DTR dialing. Remote Router B is configured for in-band dialing so it can disconnect an idle call. (See [Figure 50](#).)

Figure 50 DTR Dialing Through a PSTN



Router A

```
interface serial 0
  ip address 172.18.170.19 255.255.255.0
  dialer dtr
  dialer-group 1
  !
  access-list 101 deny igmp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
  access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
  !
  dialer-list 1 list 101
```

Router B

```

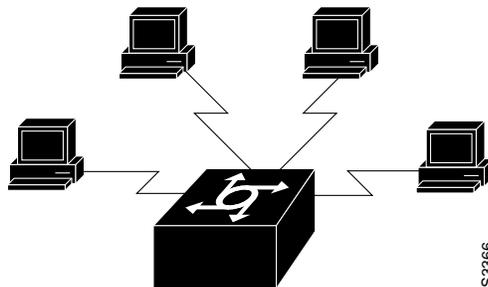
interface serial 0
 ip address 172.18.170.20 255.255.255.0
 dialer in-band
 dialer string 9876543
 pulse-time 1
 !
access-list 101 deny igmp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
 !
dialer-list 1 list 101

```

Hub-and-Spoke DDR for Asynchronous Interfaces and Authentication Example

The following example sets up DDR to provide service to multiple remote sites. In a hub-and-spoke configuration, you can use a generic configuration script to set up each remote connection. [Figure 51](#) illustrates a typical hub-and-spoke configuration.

Figure 51 Hub-and-Spoke DDR Configuration



Commands in the following sections are used to create this configuration.

Spoke Topology Configuration

The following commands are executed on the spoke side of the connection. (A different “spoke” password must be specified for each remote client.) The configuration provides authentication by identifying a password that must be provided on each end of the connection.

```

interface ethernet 0
 ip address 172.30.44.1 255.255.255.0
 !
interface async 7
 async mode dedicated
 async default ip address 172.30.45.1
 ip address 172.30.45.2 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 dialer in-band
 dialer map ip 172.30.45.1 name hub system-script hub 1234
 dialer map ip 172.30.45.255 name hub system-script hub 1234
 dialer-group 1
 !
ip route 172.30.43.0 255.255.255.0 172.30.45.1
ip default-network 172.30.0.0
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT

```

```

chat-script hub "" "" name: spokel word: <spokel-passwd> PPP
dialer-list 1 protocol ip permit
!
username hub password <spokel-passwd>
!
router igrp 109
 network 172.30.0.0
 passive-interface async 7
!
line 7
 modem InOut
 speed 38400
 flowcontrol hardware
 modem chat-script generic

```

Hub Router Configuration

The following commands are executed on the local side of the connection—the hub router. The commands configure the server for communication with three clients and provide authentication by identifying a unique password for each “spoke” in the hub-and-spoke configuration.

```

interface ethernet 0
 ip address 172.30.43.1 255.255.255.0
!
interface async 7
 async mode interactive
 async dynamic address
 dialer rotary-group 1
!
interface async 8
 async mode interactive
 async dynamic address
 dialer rotary-group 1
!
interface dialer 1
 ip address 172.30.45.2 255.255.255.0
 no ip split-horizon
 encapsulation ppp
 ppp authentication chap
 dialer in-band
 dialer map ip 172.30.45.2 name spokel 3333
 dialer map ip 172.30.45.2 name spoke2 4444
 dialer map ip 172.30.45.2 name spoke3 5555
 dialer map ip 172.30.45.255 name spokel 3333
 dialer map ip 172.30.45.255 name spoke2 4444
 dialer map ip 172.30.45.255 name spoke3 5555
 dialer-group 1
!
ip route 172.30.44.0 255.255.255.0 172.30.45.2
ip route 172.30.44.0 255.255.255.0 172.30.45.3
ip route 172.30.44.0 255.255.255.0 172.30.45.4
dialer-list 1 list 101
 access-list 101 deny igrp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
 access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
 chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
!
username spokel password <spokel-passwd>
username spoke2 password <spoke2-passwd>
username spoke3 password <spoke3-passwd>
username spokel autocommand ppp 172.30.45.2
username spoke2 autocommand ppp 172.30.45.3
username spoke3 autocommand ppp 172.30.45.4

```

```
!  
router igrp 109  
  network 172.30.0.0  
  redistribute static  
!  
line 7  
  login tacacs  
  modem InOut  
  speed 38400  
  flowcontrol hardware  
  modem chat-script generic
```

Two-Way Reciprocal Client/Server DDR Without Authentication Example

You can set up two-way reciprocal DDR without authentication in which both the client and server have dial-in access to each other. This configuration is demonstrated in the following two sections.

Remote Configuration

The following commands are executed on the remote side of the connection. This configuration provides authentication by identifying a password that must be provided on each end of the connection.

```
interface ethernet 0  
  ip address 172.30.44.1 255.255.255.0  
!  
interface async 7  
  ip address 172.30.45.2 255.255.255.0  
  async mode dedicated  
  async default ip address 172.30.45.1  
  encaps ppp  
  dialer in-band  
  dialer string 1234  
  dialer-group 1  
!  
ip route 172.30.43.0 255.255.255.0 async 7  
  ip default-network 172.30.0.0  
  chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT  
  dialer-list 1 protocol ip permit  
!  
line 7  
  no exec  
  modem InOut  
  speed 38400  
  flowcontrol hardware  
  modem chat-script generic
```

Local Configuration

The following commands are executed on the local side of the connection. As with the remote side configuration, this configuration provides authentication by identifying a password for each end of the connection.

```
interface ethernet 0  
  ip address 172.30.43.1 255.255.255.0  
!  
interface async 7  
  async mode dedicated  
  async default ip address 172.30.45.2  
  encapsulation ppp
```

```

dialer in-band
dialer string 1235
dialer rotary-group 1
!
interface async 8
async mode dedicated
async default ip address 172.30.45.2
dialer rotary-group 1
!
ip route 172.30.44.0 255.255.255.0 async 7
ip address 172.30.45.2 255.255.255.0
encapsulation ppp
ppp authentication chap
dialer in-band
dialer map ip 172.30.45.2 name remote 4321
dialer load-threshold 80
!
ip route 172.30.44.0 255.255.255.0 128.150.45.2
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
dialer-list 1 protocol ip permit
!
route igrp 109
network 172.30.0.0
redistribute static
passive-interface async 7
!
line 7
modem InOut
speed 38400
flowcontrol hardware
modem chat-script generic

```

Frame Relay Support Example

The examples in this section present various combinations of interfaces, Frame Relay features, and DDR features.

Frame Relay Access with In-Band Dialing (V.25bis) and Static Mapping Example

The following example shows how to configure a router for IP over Frame Relay using in-band dialing. A Frame Relay static map is used to associate the next hop protocol address to the data-link connection identifier (DLCI). The dialer string allows dialing to only one destination.

```

interface Serial0
ip address 10.1.1.1 255.255.255.0
encapsulation frame-relay
frame-relay map ip 10.1.1.2 100 broadcast
dialer in-band
dialer string 4155551212
dialer-group 1
!
access-list 101 deny igrp any host 255.255.255.255
access-list 101 permit ip any any
!
dialer-list 1 protocol ip list 101

```

Frame Relay Access with ISDN Dialing and DDR Dynamic Maps Example

The following example shows a BRI interface configured for Frame Relay and for IP, IPX, and AppleTalk routing. No static maps are defined because this setup relies on Frame Relay Local Management Interface (LMI) signaling and Inverse ARP to determine the network addresses-to-DLCI mappings dynamically. (Because Frame Relay Inverse ARP is enabled by default, no command is required.)

```
interface BRI0
 ip address 10.1.1.1 255.255.255.0
 ipx network 100
 appletalk cable-range 100-100 100.1
 appletalk zone ISDN
 no appletalk send-rtmps
 encapsulation frame-relay IETF
 dialer map ip 10.1.1.2 broadcast 4155551212
 dialer map apple 100.2 broadcast 4155551212
 dialer map ipx 100.0000.0c05.33ed broadcast 4085551234
 dialer-group 1
!
access-list 101 deny igrp any host 255.255.255.255
access-list 101 permit ip any any
access-list 901 deny -1 FFFFFFFF 452
access-list 901 deny -1 FFFFFFFF 453
access-list 901 deny -1 FFFFFFFF 457
access-list 901 deny -1 FFFFFFFF 0 FFFFFFFF 452
access-list 901 deny -1 FFFFFFFF 0 FFFFFFFF 453
access-list 901 deny -1 FFFFFFFF 0 FFFFFFFF 457
access-list 901 permit -1
access-list 601 permit cable-range 100-100 broadcast-deny
access-list 601 deny other-access
!
dialer-list 1 protocol ip list 101
dialer-list 1 protocol novell list 901
dialer-list 1 protocol apple list 601
```

X.25 Support Example

The following example configures a router to support X.25 and DTR dialing:

```
interface serial 0
 ip address 172.18.170.19 255.255.255.0
 encapsulation x25
 x25 address 12345
 x25 map ip 172.18.171.20 67890 broadcast
 dialer dtr
 dialer-group 1
!
access-list 101 deny igrp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
!
dialer-list 1 list 101
```

LAPB Support Example

The following example configures a router for LAPB encapsulation and in-band dialing:

```
interface serial 0
 ip address 172.18.170.19 255.255.255.0
 encapsulation lapb
 dialer in-band
 dialer string 4155551212
 dialer-group 1
!
 access-list 101 deny igmp 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
 access-list 101 permit ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
!
 dialer-list 1 protocol ip list 101
```