



## Interfaces, Controllers, and Lines Used for Dial Access Overview

---

This chapter describes the different types of calls, interfaces, software constructs, controllers, channels, and lines that are used for dial-up remote access. It includes the following main sections:

- Call Types
- Cisco IOS Dial Universe
- Logical Constructs
- Logical Interfaces
- T1 and E1 Controllers
- Non-ISDN Channelized T1 and Channelized E1 Lines
- ISDN Service
- Line Types

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

### Call Types

Three basic call types are used for dial access. These call types are discussed in the following sections:

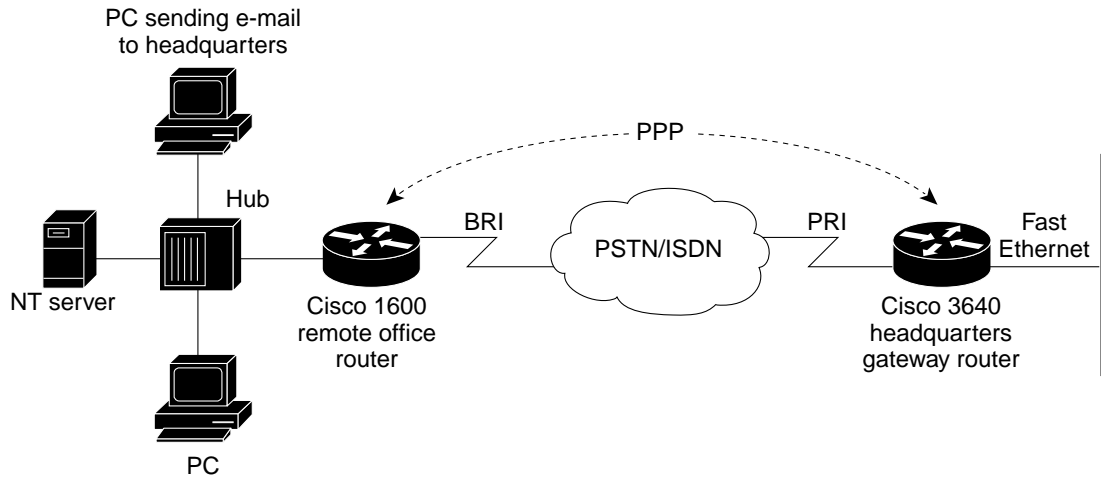
- Circuit-Switched Digital Calls
- Analog Modem Calls
- Asynchronous Character Stream Calls

### Circuit-Switched Digital Calls

Circuit-switched digital calls are usually ISDN 56-kbps or 64-kbps data calls that use PPP. These calls are initiated by an ISDN router, access server, or terminal adapter connected to a client workstation. Individual synchronous serial digital signal level 0 (DS0) bearer (B) channels are used to transport circuit-switched digital calls across WANs. These calls do not transmit across “old world” lines.

Figure 1 shows a Cisco 1600 series remote office router dialing in to a Cisco 3640 router positioned at a headquarters gateway.

**Figure 1 Remote Office LAN Dialing In to Headquarters**



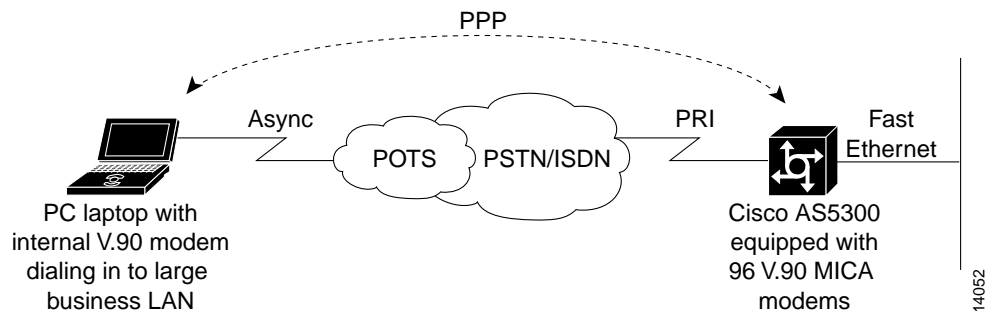
14053

## Analog Modem Calls

Analog modem calls travel through traditional telephone lines and ISDN lines. Regardless of the media used, these calls are initiated by a modem and terminate on another modem at the remote end.

Figure 2 shows a remote laptop using a V.90 internal modem to dial in to a Cisco AS5300 access server, which is loaded with 96 internal V.90 MICA technologies modems.

**Figure 2 Remote Node Dialing In to a Cisco AS5300**



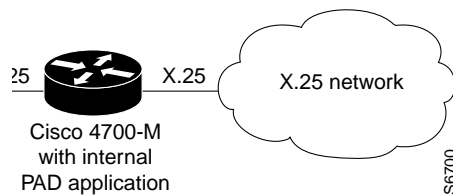
14052

## Asynchronous Character Stream Calls

Asynchronous character stream calls enter the router or access server through virtual terminal (vty) lines and virtual asynchronous interfaces (vty-async). These virtual lines and interfaces terminate incoming character streams that have no physical connection to the access server or router (such as a physical serial interface). For example, if you begin a PPP session over an asynchronous character stream, a vty-async interface is created to support the call. The following types of calls are terminated on a virtual asynchronous interface: Telnet, local-area transport (LAT), V.120, TN3270, and Link Access Procedure, Balanced-terminal adapter (LAPB-TA) and packet assembler/disassembler (PAD) calls.

Figure 3 shows a dumb terminal using a modem and packet assembler/disassembler (PAD) to place a call in to an X.25 switched network. The Cisco 4700-M router is configured to support vty lines and vty-async interfaces.

**Figure 3** Standard X.25 Dial-Up Connection

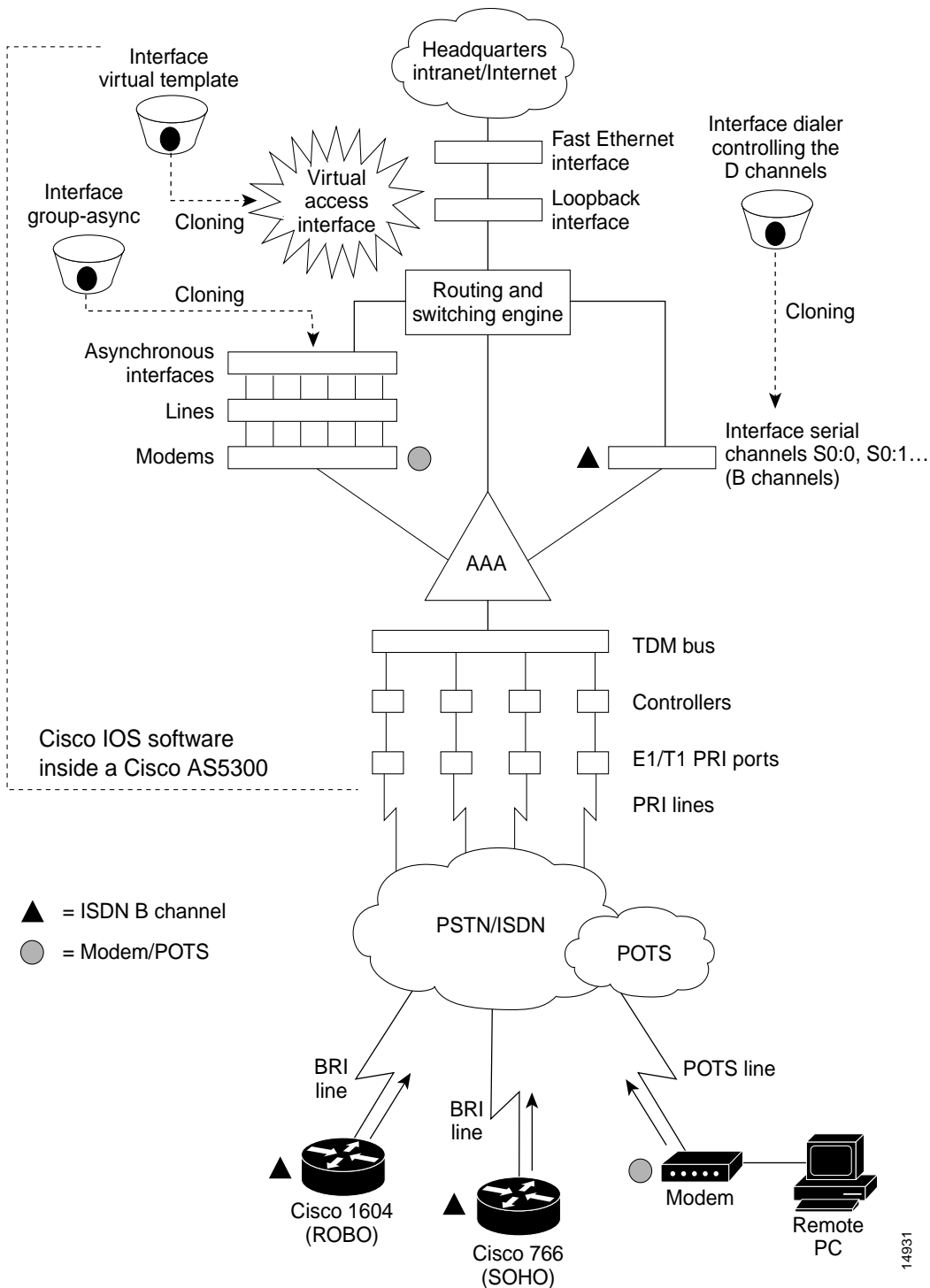


## Cisco IOS Dial Universe

Different components inside the Cisco IOS software work together to enable remote clients to dial in and send packets. Figure 4 shows one Cisco AS5300 access server receiving calls from a remote office, branch office (ROBO); small office, home office (SOHO); and modem client.

Depending on your network scenario, you may encounter all of the components in Figure 4. For example, you might decide to create a virtual IP subnet by using a loopback interface. This step saves address space. Virtual subnets can exist inside devices that you advertise to your backbone. In turn, IP packets get relayed to remote PCs, which route back to the central site.

Figure 4 Cisco IOS Dial Universe



14931

For more information about the components in Figure 4, see the following sections:

- Logical Constructs
- Logical Interfaces
- T1 and E1 Controllers
- Non-ISDN Channelized T1 and Channelized E1 Lines
- ISDN Service
- Line Types

## Logical Constructs

A logical construct stores core protocol characteristics to assign to physical interfaces. No data packets are forwarded to a logical construct. Cisco uses three types of logical constructs in its access servers and routers. These constructions are described in the following sections:

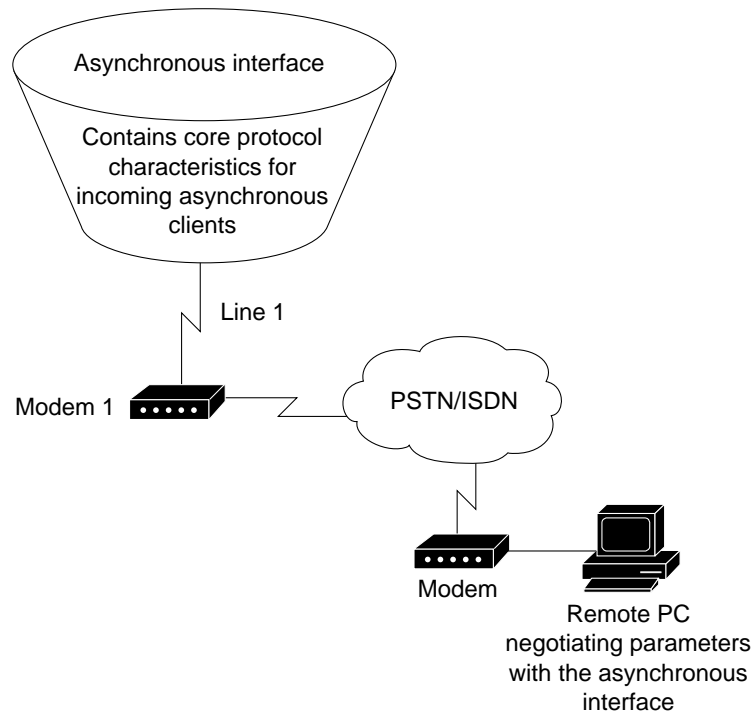
- Asynchronous Interfaces
- Group Asynchronous Interfaces
- Virtual Template Interfaces

## Asynchronous Interfaces

An asynchronous interface assigns network protocol characteristics to remote asynchronous clients that are dialing in through physical terminal (TTY) lines and modems. (Refer to Figure 5.)

Use the **interface async** command to create and configure an asynchronous interface.

**Figure 5** Logical Construct for an Asynchronous Interface



14054

To enable clients to dial in, you must configure two asynchronous components: async lines and async interfaces. Asynchronous interfaces correspond to physical terminal (TTY) lines. For example, asynchronous interface 1 corresponds to TTY line 1.

Commands entered in asynchronous interface mode enable you to configure protocol-specific parameters for asynchronous interfaces, whereas commands entered in line configuration mode let you configure the physical aspects for the same port.

Specifically, you configure asynchronous interfaces to support PPP connections. An asynchronous interface on an access server or router can be configured to support the following functions:

- Network protocol support such as IP, Internet Protocol Exchange (IPX), or AppleTalk
- Encapsulation support (such as PPP)
- IP client addressing options (default or dynamic)
- IPX network addressing options
- PPP authentication
- ISDN BRI and PRI configuration

For additional information about configuring asynchronous interfaces, see the chapter “Preparing Asynchronous Interfaces.”

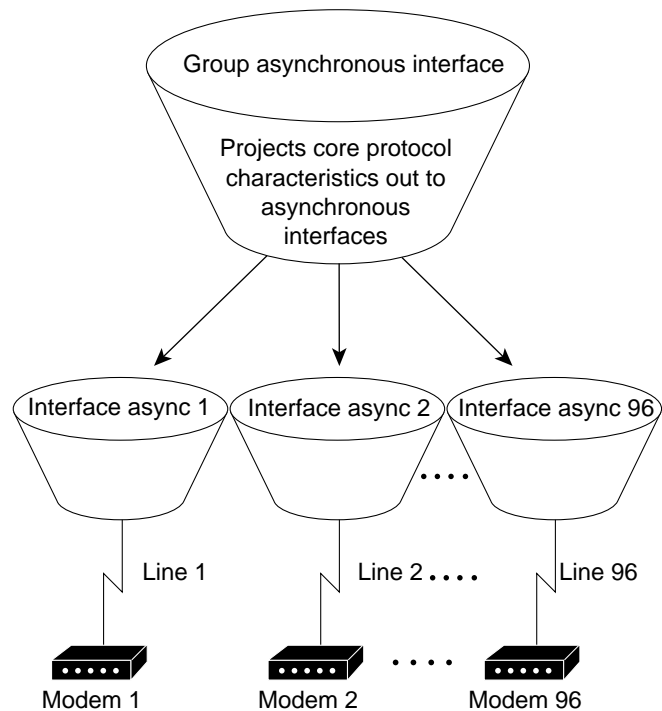
## Group Asynchronous Interfaces

A group asynchronous interface is a parent interface that stores and projects core protocol characteristics to a specified range of asynchronous interfaces. Asynchronous interfaces clone protocol information from group asynchronous interfaces. No data packets arrive in a group asynchronous interface.

By setting up a group asynchronous interface, you also eliminate the need to repeatedly configure identical configuration information across several asynchronous interfaces. For example, on a Cisco AS5300 one group asynchronous interface is used instead of 96 individual asynchronous interfaces. (Refer to Figure 6.)

Use the **interface group-async** command to create and configure a group asynchronous interface.

**Figure 6** Logical Construct for Group Asynchronous Interface



The following example shows a group asynchronous configuration for a Cisco AS5300 access server loaded with one 4-port ISDN PRI card and 96 MICA modems:

```
as5300(config)# interface group-async 1
as5300(config-if)# ip unnumbered loopback 0
as5300(config-if)# encapsulation ppp
as5300(config-if)# async mode interactive
as5300(config-if)# peer default ip address pool dialin_pool
as5300(config-if)# no cdp enable
as5300(config-if)# ppp authentication chap pap dialin
as5300(config-if)# group-range 1 96
```

Building configuration...

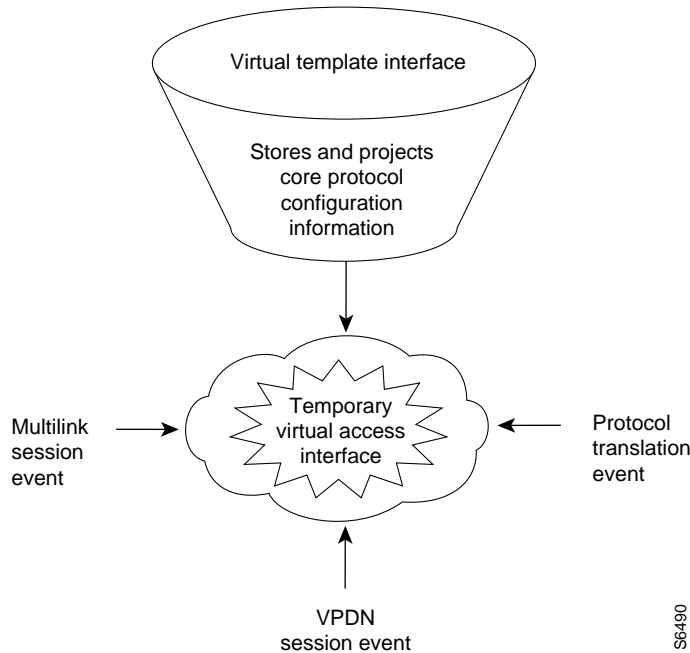
```
as5300(config-if)#
```

For more information, see the chapter “Configuring Modems and Chat Scripts.”

## Virtual Template Interfaces

A virtual template interface stores protocol configuration information for virtual access interfaces and protocol translation sessions. (Refer to Figure 7.)

**Figure 7** Logical Construct for a Virtual Template Interface



## Templates for Virtual Access Interfaces

Virtual templates project configuration information to temporary virtual access interfaces triggered by multilink or virtual private dialup network session events. When a virtual access interface is triggered, the configuration attributes in the virtual template are cloned and the negotiated parameters are applied to the connection.

The following example shows a virtual template interface on a Cisco 7206 router, which is used as a home gateway in a VPDN scenario:

```
c7206# configure terminal
c7206 (config)# interface virtual-template 1
c7206 (config-if)# ip unnumbered ethernet 2/1
c7206 (config-if)# peer default ip address pool cisco-pool
c7206 (config-if)# ppp authentication chap pap
c7206 (config-if)# exit
c7206 (config)# vpdn enable
c7206 (config)# vpdn incoming isp cisco.com virtual-template 1
c7206 (config)#
```

For more information, see the chapter “Configuring Virtual Template Interfaces” in the *Cisco IOS Dial Services Configuration Guide: Network Services* publication.

## Templates for Protocol Translation

Virtual templates are used to simplify the process of configuring protocol translation to tunnel PPP or Serial Line Internet Protocol (SLIP) across X.25, Transmission Control Protocol (TCP), and local-area transport (LAT) networks. You can create a virtual interface template with the **interface virtual-template** command and use it for one-step and two-step protocol translation. When a user dials in through a vty line and a tunnel connection is established, the router clones the attributes of the virtual interface template onto a *virtual access interface*. This virtual access interface is a temporary interface that supports the protocol configuration specified in the virtual interface template. This virtual access interface is created dynamically and lasts only as long as the tunnel session is active.

The virtual template in the following example explicitly specifies PPP encapsulation. The translation is from X.25 to PPP, which enables tunneling of PPP across an X.25 network.

```
c7206# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
c7206(config)# interface virtual-template 1
c7206(config-if)# ip unnumbered ethernet 0
c7206(config-if)# peer default ip address 162.18.2.131
c7206(config-if)# encapsulation ppp
c7206(config-if)# exit
c7206(config)# translate x25 5555678 virtual-template 1
c7206(config)#
```

For more information, see the chapter “Configuring Protocol Translation and Virtual Asynchronous Devices.”

## Logical Interfaces

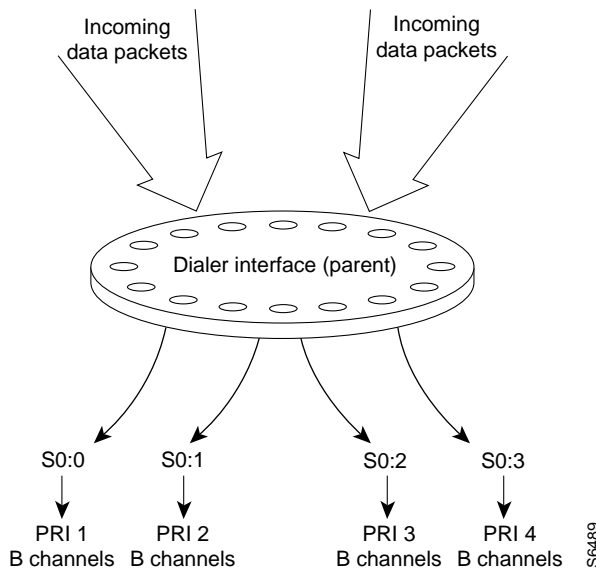
A logical interface receives and sends data packets and controls physical interfaces. The Cisco IOS software provides three logical interfaces used for dial access. These interfaces are described in the following sections:

- Dialer Interfaces
- Virtual Access Interfaces
- Virtual Asynchronous Interfaces

## Dialer Interfaces

A dialer interface is a parent interface that stores and projects protocol configuration information that is common to all data (D) channels that are members of a dialer rotary group. Data packets pass through dialer interfaces, which in turn initiate dialing for inbound calls. In most cases, D channels get their core protocol intelligence from dialer interfaces.

Figure 8 shows packets coming into a dialer interface, which contains the configuration parameters common to four D channels (shown as S0:0, S0:1, S0:3, and S0:4). All the D channels are members of the same rotary group. Without the dialer interface configuration, each D channel must be manually configured with identical properties. Dialer interfaces condense and streamline the configuration process.

**Figure 8** *Dialer Interface and Its Neighboring Components*

A dialer interface is user configurable and linked to individual B channels where it delivers data packets to their physical destinations. Dialer interfaces seize physical interfaces to cause packet delivery. If a dialer interface engages in a multilink session, a dialer interface is in control of a virtual access interface, which in turn controls S0:3 or chassis 2 S0:3, for example. A dialer interface is created with the **interface dialer** global configuration command.

The following example shows a fully configured dialer interface:

```
as5300# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
as5300(config)# interface dialer 0
as5300(config-if)# ip unnumbered loopback 0
as5300(config-if)# no ip mroute-cache
as5300(config-if)# encapsulation ppp
as5300(config-if)# peer default ip address pool dialin_pool
as5300(config-if)# dialer in-band
as5300(config-if)# dialer-group 1
as5300(config-if)# no fair-queue
as5300(config-if)# no cdp enable
as5300(config-if)# ppp authentication chap pap callin
as5300(config-if)# ppp multilink
as5300(config-if)#
```

All the D channels are members of rotary group 1.

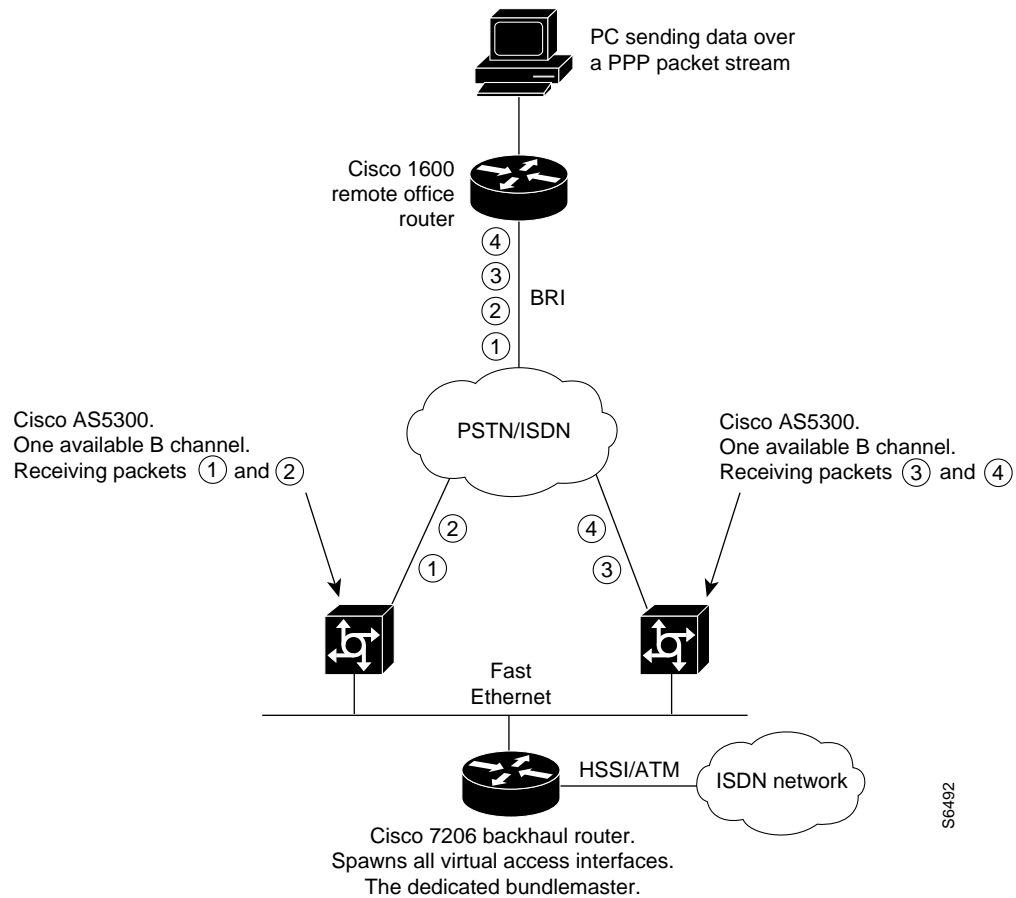
## Virtual Access Interfaces

A virtual access interface is a temporary interface that is spawned to terminate incoming PPP streams that have no physical connections. PPP streams, Layer 2 Forwarding Protocol (L2F), and Layer 2 Tunnel Protocol (L2TP) frames coming in on multiple B channels are reassembled on virtual access interfaces. These access interfaces are constructs used to terminate packets.

Virtual access interfaces obtain their set of instructions from virtual interface templates. The attributes configured in virtual templates are projected or cloned to a virtual access interfaces. Virtual access interfaces are not directly user configurable. These interfaces are created dynamically and last only as long as the tunnels or multilink sessions are active. After the sessions are ended, the virtual access interfaces disappear.

Figure 9 shows how a virtual access interface functions to accommodate a multilink session event. Two physical interfaces on two different access servers are participating in one multilink call from a remote PC. However, each Cisco AS5300 access server only has one B channel available to receive a call. All other channels are busy. Therefore all four packets are equally dispersed across two separate B channels and two access servers. Each Cisco AS5300 access server receives only half the total packets. A virtual access interface is dynamically spawned upstream on a Cisco 7206 backhaul router to receive the multilink protocol, track the multilink frames, and reassemble the packets. The Cisco 7206 router is configured to be the bundle master, which performs all packet assembly and reassembly for both Cisco AS5300 access servers.

**Figure 9 Virtual Access Interfaces Used for Multichassis Multilink Session Events**



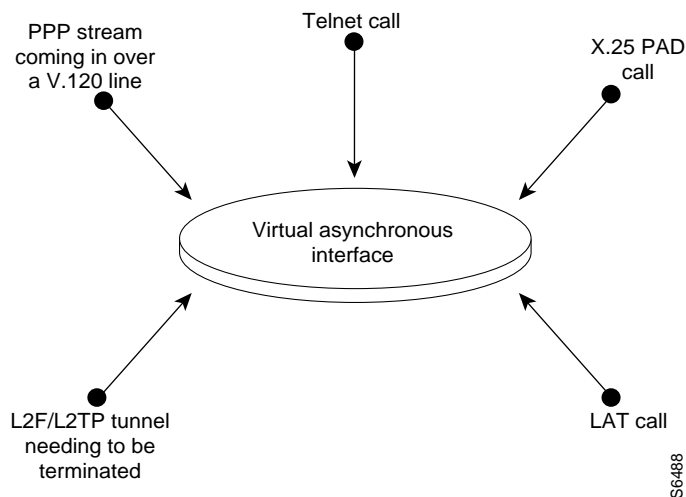
S6492

## Virtual Asynchronous Interfaces

A virtual asynchronous interface is created on demand to support calls that enter the router through a nonphysical interface. For example, asynchronous character stream calls terminate or land on nonphysical interfaces. These types of calls include inbound Telnet, LAT, PPP over character-oriented protocols (such as V.120 or X.25), and LAPB-TA and PAD calls. A virtual asynchronous interface is also used to terminate L2F/L2TP tunnels, which are often traveling companions with Multilink protocol sessions. Virtual asynchronous interfaces are not user configurable; rather, they are dynamically created and torn down on demand. A virtual asynchronous line is used to access a virtual asynchronous interface.

Figure 10 shows a variety of calls terminating on a virtual asynchronous interface. After the calls end, the interface is torn down.

**Figure 10** Async Character Stream Calls Terminating on a Virtual Asynchronous Interface



## T1 and E1 Controllers

Cisco controllers negotiate the following parameters between an access server and a central office: line coding, framing, clocking, DS0/time slot provisioning, and signalling.

Time slots are provisioned to meet the needs of particular network scenarios. T1 controllers have 24 time slots, and E1 controllers have 30 time slots. To support traffic flow for one ISDN PRI line in a T1 configuration, use the **pri-group** command. To support traffic flow for analog calls over a channelized E1 line with receive and transmit (E&M -- also ear and mouth) signalling, use the **cas-group 1 timeslots 1-30 type e&m-fgb** command. Most telephone companies do not support provisioning one trunk for different combinations of time slot services, though this provisioning is supported on Cisco controllers. On a T1 controller for example, time slots 1 to 10 could run PRI, time slots 11 to 20 could run channel-associated signalling (CAS), and time slots 21 to 24 could support leased-line grouping.

The following example configures one of four T1 controllers on a Cisco AS5300 access server:

```
as5300# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
as5300(config)# controller t1 ?
<0-3> Controller unit number
as5300(config)# controller t1 0
as5300(config-controller)# framing esf
as5300(config-controller)# linecode b8zs
as5300(config-controller)# clock source line primary
as5300(config-controller)# pri-group timeslots 1-24
as5300(config-controller)#
```

This example supports modem calls and circuit-switched digital calls over ISDN PRI.

## Non-ISDN Channelized T1 and Channelized E1 Lines

A channelized T1 or channelized E1 line is an analog line that was originally intended to support analog voice calls, but has evolved to support analog data calls. ISDN is not sent across channelized T1 or E1 lines. Channelized T1 and channelized E1 lines are often referred to as CT1 and CE1. These channelized lines are found in “old world,” non-ISDN telephone networks.

The difference between traditional channelized lines (analog) and nonchannelized lines (ISDN) is that channelized lines have no built-in D channel. That is, all 24 channels on a T1 line only carry data. The signalling is in-band or associated to the data channels. Traditional channelized lines do not support digitized data calls (for example, BRI with 2B + D). Channelized lines support a variety of in-band signal types, such as ground start, loop start, wink start, immediate start, E&M, and R2.

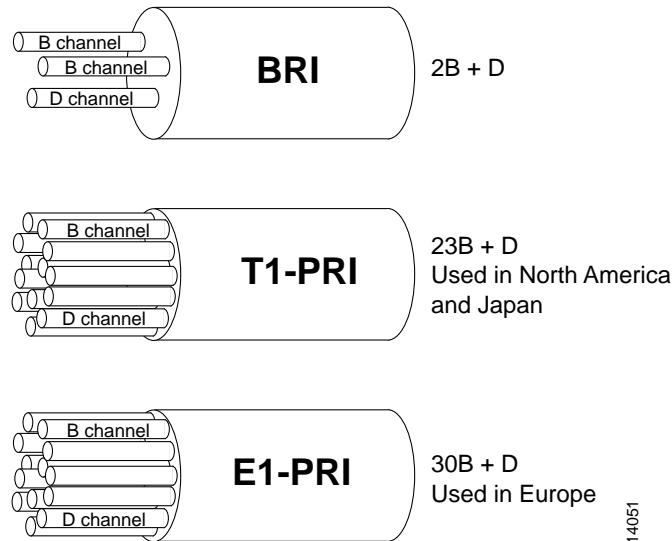
Signalling for channelized lines is configured with the **cas-group** controller configuration command. The following example configures E&M group B signalling on a T1 controller:

```
as5300# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
as5300(config)# controller t1 0
as5300(config-controller)# cas-group 1 timeslots 1-24 type ?
e&m-fgb          E & M Type II FGB
e&m-fgd          E & M Type II FGD
e&m-immediate-start E & M Immediate Start
fxs-ground-start FXS Ground Start
fxs-loop-start   FXS Loop Start
r1-modified      R1 Modified
sas-ground-start SAS Ground Start
sas-loop-start   SAS Loop Start
as5300(config-controller)# cas-group 1 timeslots 1-24 type e&m-fgb
as5300(config-controller)# framing esf
as5300(config-controller)# clock source line primary
as5300(config-controller)#
```

# ISDN Service

Cisco routing devices support ISDN BRI and ISDN PRI both media types use B channels and D channels. Figure 11 shows how many B channels and D channels are assigned to each media type.

**Figure 11** Logical Relationship of B Channels and D Channels



## ISDN BRI

ISDN BRI operates over most of the copper twisted-pair telephone wiring in place. ISDN BRI delivers a total bandwidth of a 144 kbps via three separate channels. Two of the B channels operate at 64 kbps and are used to carry voice, video, or data traffic. The third channel, the D channel, is a 16-kbps signalling channel used to tell the Public Switched Telephone Network (PSTN) how to handle each of the B channels. ISDN BRI is often referred to as “2 B + D.”

Enter the **interface bri** command to bring up and configure a single BRI interface, which is the overseer of the 2 B + D channels. The D channel is not user configurable.

The following example configures an ISDN BRI interface on a Cisco 1600 series router. The **isdn spid** command defines the service profile identifier (SPID) number, which is assigned by the ISDN service provider, to both B channels. Not all ISDN lines have SPIDs.

```
1600# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
1600(config)# interface bri 0
1600(config-if)# isdn spid1 55598760101
1600(config-if)# isdn spid2 55598770101
1600(config-if)# isdn switch-type basic-ni
1600(config-if)# ip unnumbered ethernet 0
1600(config-if)# dialer map ip 192.168.37.40 name hq 5552053
1600(config-if)# dialer load-threshold 70
1600(config-if)# dialer-group 1
1600(config-if)# encapsulation ppp
1600(config-if)# ppp authentication chap pap callin
1600(config-if)# ppp multilink
1600(config-if)# no shutdown
```

## ISDN PRI

ISDN PRI is designed to carry large numbers of incoming ISDN calls at point of presences (POPs) and other large central site locations. All the reliability and performance of ISDN BRI applies to ISDN PRI, but ISDN PRI has 23 B channels running at 64 kbps each and a shared 64 kbps D channel that carries signalling traffic. ISDN PRI is often referred to as “23 B + D” (North America and Japan) or “30 B + D” (rest of the world).

The D channel notifies the central office switch to send the incoming call to particular timeslots on the Cisco access server or router. Each one of the B channels carries data or voice. The D channel carries signalling for the B channels. The D channel identifies if the call is a circuit-switched digital call or an analog modem call. Analog modem calls are decoded and then sent to the onboard modems. Circuit-switched digital calls are directly relayed to the ISDN processor in the router. Enter the **interface serial** command to bring up and configure the D channel, which is user configurable.

Figure 12 shows the logical contents of an ISDN PRI interface used in a T1 network configuration. The logical contents include 23 B channels, 1 D channel, 24 time slots, and 24 virtual serial interfaces (total number of B + D channels).

Figure 12 Logical Relationship of ISDN PRI Components for T1

Channel Type	Time Slot Number	Virtual Serial Interface Number
B (data channel)	1	S0:0
B (data channel)	2	S0:1
B (data channel)	3	S0:2
B (data channel)	4	S0:3
•	•	•
•	•	•
•	•	•
•	•	•
•	•	•
B (data channel)	21	S0:20
B (data channel)	22	S0:21
B (data channel)	23	S0:22
D (signaling channel)	24	S0:23

Logical contents of a PRI interface

S6487

The following example is for a Cisco AS5300 access server. It configures one T1 controller for ISDN PRI, then configures the neighboring D channel (interface serial 0:23). Controller T1 0 and interface serial 0:23 are both assigned to the first PRI port. The second PRI port is assigned to controller T1 1 and interface serial 1:23, and so on. The second PRI port configuration is not shown in this example. This Cisco AS5300 access server is used as part of a stack group dial-in solution for an Internet service provider.

```
as5300# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.
as5300(config)# controller t1 0
as5300(config-controller)# framing esf
as5300(config-controller)# linecode b8zs
as5300(config-controller)# clock source line primary
as5300(config-controller)# pri-group timeslots 1-24
as5300(config-controller)# exit
as5300(config)# interface serial 0:23
as5300(config-if)# ip unnumbered Loopback 0
as5300(config-if)# ip accounting output-packets
as5300(config-if)# no ip mroute-cache
as5300(config-if)# encapsulation ppp
as5300(config-if)# isdn incoming-voice modem
as5300(config-if)# dialer-group 1
as5300(config-if)# no fair-queue
as5300(config-if)# compress stac
as5300(config-if)# no cdp enable
as5300(config-if)# ppp authentication chap
as5300(config-if)# ppp multilink
as5300(config-if)# netbios nbf
```

# Line Types

This section describes the different line types used for dial access. It also describes the relationship between lines and interfaces.



Note

Cisco devices have four types of lines: console, auxiliary, asynchronous, and virtual terminal. Different routers have different numbers of these line types. Refer to the hardware and software configuration guides that shipped with your device for exact configurations.

Table 3 shows the types of lines that can be configured.

**Table 3 Available Line Types**

Line Type	Interface	Description	Numbering Rules
CON or CTY	Console	Typically used to log in to the router for configuration purposes.	Line 0.
AUX	Auxiliary	EIA/TIA-232 data terminal equipment (DTE) port used as a backup (TTY) asynchronous port. Cannot be used as a second console port.	Last TTY line number plus 1.
TTY	Asynchronous	Same as asynchronous interface. Used typically for remote-node dial-in sessions that use such protocols as SLIP, PPP, AppleTalk Remote Access (ARA), and XRemote.	The numbering widely varies between platforms. This number is equivalent to the maximum number of modems or asynchronous interfaces supported by your access server or router. <sup>1</sup>
VTY	Virtual asynchronous	Used for incoming Telnet, LAT, X.25 PAD, and protocol translation connections into synchronous ports (such as Ethernet and serial interfaces) on the router.	Last TTY line number plus 2 through the maximum number of vty lines specified. <sup>2</sup>

1. Enter the **interface line tty ?** command to view the maximum number of TTY lines supported.
2. Increase the number of vty lines on a router using the **line vty** global configuration command. Delete vty lines with the **no line vty line-number** command. The **line vty** command accepts any line number larger than 5 up to the maximum number of lines supported by your router with its current configuration. Enter the **interface line vty ?** command to view the maximum number of vty lines supported.

Use the **show line** command to see the status of each of the lines available on a router. (Refer to Figure 13.)

Figure 13 Sample Show Line Output Showing CTY, TTY, AUX, and VTY Line Statistics

Autoselect state	Tty	Typ	Tx/Rx	Rotary group #	Modem	Roty	ACCO	ACCT	Uses	Noise	Overruns
	* 0	CTY	-	-	-	-	-	-	0	0	0/0
	* 1	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	* 2	TTY	115200/115200	-	inout	-	21630	-	37	23	0/0
Absolute line number	A 3	TTY	115200/115200	-	inout	-	25	-	10	24	1/0
	* 4	TTY	115200/115200	-	inout	-	4	-	20	63	1/0
	* 5	TTY	115200/115200	-	inout	-	32445	-	18	325	22/0
	A 6	TTY	115200/115200	-	inout	-	25	-	7	0	0/0
	I 7	TTY	115200/115200	-	inout	-	6	-	6	36	1/0
Line speed	I 8	TTY	115200/115200	-	inout	-	-	-	3	25	3/0
	* 9	TTY	115200/115200	-	inout	-	4	-	2	0	0/0
	A 10	TTY	115200/115200	-	inout	-	56	-	2	470	216/0
	I 11	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 12	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 13	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 14	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 15	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	I 16	TTY	115200/115200	-	inout	-	4	-	31	26	0/0
	17	AUX	9600/9600	-	-	-	-	-	2	1	2/104800
	* 18	VTY	9600/9600	-	-	-	-	-	103	0	0/0
	19	VTY	9600/9600	-	-	-	-	-	6	0	0/0
This is VTY2 and VTY) line 20	20	VTY	9600/9600	-	-	-	-	-	1	0	0/0
	21	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	22	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	23	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	24	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	25	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	26	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	27	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	28	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	29	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	30	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	31	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	32	VTY	9600/9600	-	-	-	-	-	0	0	0/0
	33	VTY	9600/9600	-	-	-	-	-	0	0	0/0

## Relationship Between Lines and Interfaces

The following sections describe the relationship between lines and interfaces:

- Asynchronous Interfaces and TTY Lines
- Synchronous Interfaces and Virtual Terminal Lines

### Asynchronous Interfaces and TTY Lines

Asynchronous interfaces correspond to physical terminal (TTY) lines. Commands entered in asynchronous interface mode enable you to configure protocol-specific parameters for asynchronous interfaces; commands entered in line configuration mode let you configure the physical aspects of the line port.

For example, to enable IP resources to dial in to a network through a Cisco 2500 series access server, configure the lines and asynchronous interfaces as follows.

- Configure the physical aspect of a line that leads to a port. You might enter the following commands to configure lines 1 through 16 (asynchronous TTY lines on a Cisco 2511 access server):

```
line 1 16
 login local
 modem inout
 speed 115200
 flowcontrol hardware
 ! configures the line to autosense PPP; physical line attribute
 autoselect ppp
```

- On asynchronous interface 1, you configure your protocol-specific commands. You might enter the following commands:

```
interface async 1
 encapsulation ppp
 async mode interactive
 async dynamic address
 async dynamic routing
 async default ip address 198.192.16.132
 ppp authentication chap
```

The remote node services SLIP, PPP, and XRemote are configured in asynchronous interface mode. ARA is configured in line configuration mode on virtual terminal lines or TTY lines.

## Synchronous Interfaces and Virtual Terminal Lines

Virtual terminal lines provide access to the router through a synchronous interface. Virtual terminal lines do not correspond to synchronous interfaces in the same way that TTY lines correspond to asynchronous interfaces because vty lines are created dynamically on the router, whereas TTY lines are static physical ports. When a user connects to the router on a vty line, that user is connecting into a *virtual* port on an interface. You can have multiple virtual ports for each synchronous interface.

For example, several Telnet connections can be made to an interface (such as an Ethernet or serial interface).

The number of virtual terminal lines available on a router is defined using the **line vty number-of-lines** global configuration command.

## Asynchronous Modem Lines

Asynchronous line configuration commands configure ports for the following options:

- Physical layer options (such as modem configuration)
- Security for EXEC mode
- ARA protocol configuration (PPP is configured in interface configuration mode)
- Autoselect to detect incoming protocols (ARA and PPP)

To enter line configuration mode, first connect to the console port of the access server and enter privileged EXEC mode. Then enter global configuration mode and finally enter line configuration mode for the asynchronous lines that you want to configure. The following example shows how you enter line configuration mode for lines 1 through 16:

```
c2500> enable
c2500# configure terminal
c2500(config)# line 1 16
c2500(config-line)#
```

For additional information about configuring asynchronous modem lines, see the chapter “Configuring Modems and Chat Scripts.”

## Absolute Versus Relative Line Numbers

When you enter line configuration mode, you can specify an *absolute line number* or a *relative line number*. For example, in Figure 13, absolute line number 20 is vty 2 (line 18 is vty 0). Referring to lines in a relative format is often easier than attempting to recall the absolute number of a line on a large system. Internally, the router uses absolute line numbers.

You can view all of the absolute and relative line numbers with the **show users all** EXEC command. In the following sample display, absolute line numbers are listed at the far left. Relative line numbers are in the third column, after the line type. In this example, the second virtual terminal line, vty 1, is absolute line number 3.

Line	User	Host(s)	Idle Location
0	con	0	
1	aux	0	
2	vtty	0	incoming
3	vtty	1	0 SERVER.COMPANY.COM
4	vtty	2	
5	vtty	3	
6	vtty	4	

Compare the line numbers in this sample display to the output from the **show line** command, as shown in Figure 13.

## Encapsulation Types

Synchronous serial interfaces default to High-Level Data Link Control (HDLC) encapsulation, and asynchronous serial interfaces default to SLIP encapsulation. Cisco IOS software provides a long list of encapsulation methods that can be set on the interface to change the default encapsulation method. See the *Cisco IOS Interface Configuration Command Reference* publication for a complete list and description of these encapsulation methods.

The following list summarizes the encapsulation commands available for serial interfaces used in dial configurations:

- **encapsulation frame-relay**—Frame Relay
- **encapsulation hdlc**—HDLC protocol
- **encapsulation lapb**—X.25 LAPB DTE operation
- **encapsulation ppp**—PPP
- **encapsulation slip**—SLIP

To use SLIP or PPP encapsulation, the router or access server must be configured with an IP routing protocol or with the **ip host-routing** command.

For more information and examples about serial interface encapsulations, see the chapters “Preparing Asynchronous Interfaces” and “Preparing Synchronous Serial Ports.”

