



Configuring Multichassis Multilink PPP

This chapter describes how to configure Multichassis Multilink PPP. It includes the following main sections:

- Multichassis Multilink PPP Overview
- Multichassis Multilink PPP Configuration Task List
- Monitoring and Maintaining MMP Virtual Interfaces
- MMP Configuration Examples

For a complete description of the MMP 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 master index or search online.

Multichassis Multilink PPP Overview

Prior to Release 11.2, Cisco IOS supported Multilink PPP (MLP). Beginning with Release 11.2, Cisco IOS software also supports Multichassis Multilink PPP (MMP).

Multilink PPP provides the capability of splitting and recombining packets to a single end-system across a logical pipe (also called a *bundle*) formed by multiple links. Multilink PPP provides bandwidth on demand and reduces transmission latency across WAN links.

MMP, however, provides the additional capability for links to terminate at multiple routers with different remote addresses. MMP can also handle both analog and digital traffic.

This feature is intended for situations with large pools of dial-in users, where a single chassis cannot provide enough dial ports. This feature allows companies to provide a single dialup number to its users and to apply the same solution to analog and digital calls. This feature allows internet service providers (ISPs), for example, to allocate a single ISDN rotary number to several ISDN PRIs across several routers. This capability allows for easy expansion and scalability, and for assured fault tolerance and redundancy.

MMP allows network access servers to be stacked together and appear as a single network access server chassis so that if one network access server fails, another network access server in the stack can accept calls.

With large-scale dial-out, these features are available for both outgoing and incoming calls.

Stack Groups

Routers or access servers are configured to belong to groups of peers, called *stack groups*. All members of the stack group are peers; stack groups do not need a permanent lead router. Any stack group member can answer calls coming from a single access number, which is usually an ISDN PRI hunt group. Calls can come in from remote user devices, such as routers, modems, ISDN terminal adapters, or PC cards.

Once a connection is established with one member of a stack group, that member owns the call. If a second call comes in from the same client and a different router answers the call, the router establishes a tunnel and forwards all packets belonging to the call to the router that owns the call. Establishing a tunnel and forwarding calls through it to the router that owns the call is sometimes called *projecting the PPP link to the call master*.

If a more powerful router is available, it can be configured as a member of the stack group and the other stack group members can establish tunnels and forward all calls to it. In such a case, the other stack group members are just answering calls and forwarding traffic to the more powerful *offload* router.

**Note**

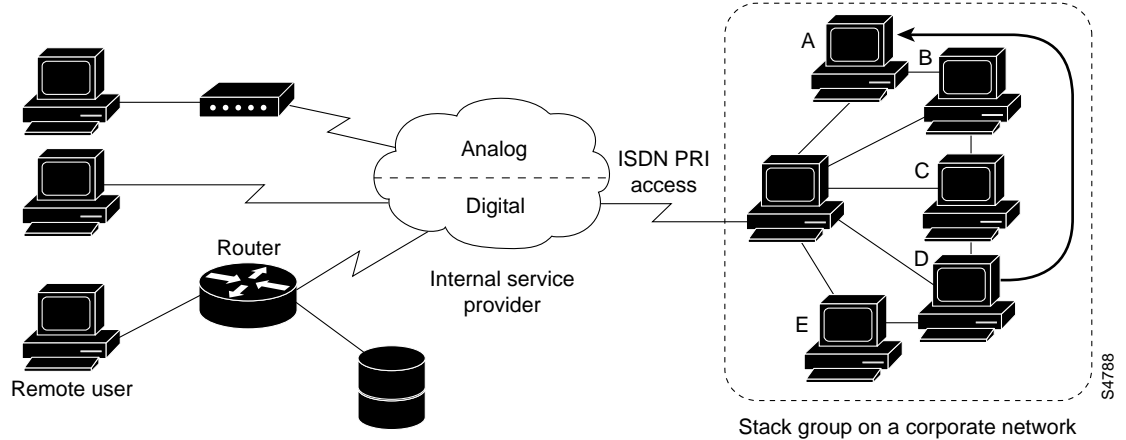
High-latency WAN lines between stack group members can make stack group operation inefficient.

Call Handling and Bidding

MMP call handling, bidding, and Layer 2 forwarding operations in the stack group proceed as follows:

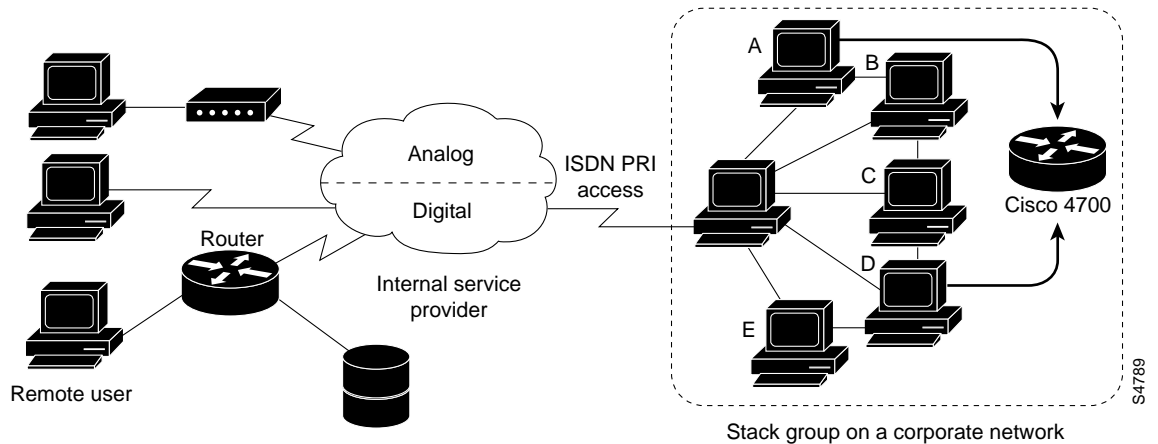
1. When the first call comes in to the stack group, router A answers.
2. In the bidding, router A wins because it already has the call. Router A becomes the *call-master* for that session with the remote device. (Router A might also be called the *host to the master bundle interface*.)
3. When the remote device that initiated the call needs more bandwidth, it makes a second MLP call to the group.
4. When the second call comes in, router D answers it and informs the stack group. Router A wins the bidding because it already is handling the session with that remote device.
5. Router D establishes a tunnel to router A, and forwards the raw PPP data to router A.
6. Router A reassembles and resequences the packets.
7. If more calls come in to router D and they too belong to router A, the tunnel between A and D enlarges to handle the added traffic. Router D does not establish an additional tunnel to A.
8. If more calls come in and are answered by any other router, that router also establishes a tunnel to A and forwards the raw PPP data.
9. The reassembled data is passed on the corporate network as if it had all come through one physical link.

Figure 4 Typical Multichassis Multilink PPP Scenario



In contrast to the previous figure, Figure 5 features an offload router. Access servers that belong to a stack group answer calls, establish tunnels, and forward calls to a Cisco 4700 router that wins the bidding and is the call master for all the calls. The Cisco 4700 reassembles and resequences all the packets coming in through the stack group.

Figure 5 Multichassis Multilink PPP with an Offload Router as a Stack Group Member



 **Note**

You can build stack groups using different access server, switching, and router platforms. However, universal access servers such as the Cisco AS5200 should not be combined with ISDN-only access servers such as the Cisco 4000-series platform. Because calls from the central office are allocated in an arbitrary way, this combination could result in an analog call being delivered to a digital-only access server, which would not be able to handle the call.

MMP support on a group of routers requires that each router be configured to support the following:

- Multilink PPP
- Stack Group Bidding Protocol (SGBP)
- Virtual template used for cloning interface configuration to support MMP

MMP is supported on the Cisco 7500, 4500, and 2500 series platforms and on synchronous serial, asynchronous serial, ISDN BRI, ISDN PRI, and dialer interfaces.

MMP does not require reconfiguration of telephone company switches.

Dialer profiles are not supported for SGBP (Stack Group Bidding Protocol).

Multichassis Multilink PPP Configuration Task List

To configure MMP, perform the tasks in the following sections, in the order listed:

- Configuring the Stack Group and Identifying Members (Required)
- Configuring a Virtual Template and Creating a Virtual Template Interface (Required)

See the section “Monitoring and Maintaining MMP Virtual Interfaces” later in this chapter for tips on maintaining MMP. See the examples in the section “MMP Configuration Examples” later in this chapter for ideas on how to configure MMP in your network.

Configuring the Stack Group and Identifying Members

To configure the stack group on the router, use the following commands beginning in configuration mode:

	Command	Purpose
Step 1	<code>username name password password</code>	Creates authentication credentials for the stack group.
Step 2	<code>sgbp group name</code>	Creates the stack group and assign this router to it.
Step 3	<code>sgbp member peer-name [peer-ip-address]</code>	Specifies a peer member of the stack group.

Repeat these steps for each additional stack group peer.



Note

Only one stack group can be configured per access server or router.

Configuring a Virtual Template and Creating a Virtual Template Interface

You need to configure a virtual template for MMP when asynchronous or synchronous serial interfaces are used, but dialers are not defined. When dialers are configured on the physical interfaces, do not specify a virtual template interface.

To configure a virtual template for any nondialer interfaces, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	<code>multilink virtual-template number</code>	Defines a virtual template for the stack group. This step is not required if ISDN interfaces or other dialers are configured and used by the physical interfaces.
Step 2	<code>ip local pool default ip-address</code>	Specifies an IP address pool by using any pooling mechanism—for example, IP local pooling or Dynamic Host Configuration Protocol (DHCP) pooling.
Step 3	<code>interface virtual-template number</code>	Creates a virtual template interface, and enter interface configuration mode. This step is not required if ISDN interfaces or other dialers are configured and used by the physical interfaces.
Step 4	<code>ip unnumbered ethernet 0</code>	Specifies unnumbered IP.
Step 5	<code>no ip route-cache</code>	Disables fast switching, which enables per-packet load sharing and enhances performance on slower serial links.
Step 6	<code>encapsulation ppp</code>	Enables PPP encapsulation on the virtual template interface.
Step 7	<code>ppp multilink</code>	Enables Multilink PPP on the virtual template interface.
Step 8	<code>ppp authentication chap</code>	Enables PPP authentication on the virtual template interface.

If dialers are or will be configured on the physical interfaces, the **ip unnumbered** command, mentioned in Step 4, will be used in configuring the dialer interface. For examples that show MMP configured with and without dialers, see the “MMP Configuration Examples” at the end of this chapter.



Note

Never define a specific IP address on the virtual template because projected virtual access interfaces are always cloned from the virtual template interface. If a subsequent PPP link also gets projected to a stack member with a virtual access interface already cloned and active, we will have identical IP addresses will be on the two virtual interfaces. IP will erroneously route between them.

For more information about address pooling, see the “Configuring Media-Independent PPP and Multilink PPP” chapter in the *Cisco IOS Dial Services Configuration Guide: Terminal Services* publication.

Monitoring and Maintaining MMP Virtual Interfaces

To monitor and maintain virtual interfaces, use any of the following commands in EXEC mode:

Command	Purpose
<code>show ppp multilink</code>	Displays MLP and MMP bundle information.

Command	Purpose
<code>show sgbp</code>	Displays the status of the stack group members.
<code>show sgbp queries</code>	Displays the current seed bid value.

MMP Configuration Examples

The following examples show MMP configuration without and with dialers:

- Multichassis Multilink PPP Using PRI but No Dialers
- Multichassis Multilink PPP with Dialers
- Multichassis Multilink PPP with Offload Server

Multichassis Multilink PPP Using PRI but No Dialers

The following example shows the configuration of MMP when no dialers are involved. Comments in the configuration discuss the commands. Variations are shown for a Cisco AS5200 access server or Cisco 4000 series router, and for an E1 or T1 controller.

```
sgbp group stackq
sgbp member systemb 1.1.1.2
sgbp member systemc 1.1.1.3

username stackq password therock
! First make sure the multilink virtual template number is defined globally on
! each router that is a member of the stack group.
multilink virtual-template 1

! If you have not configured any dialer interfaces for the physical interfaces in
! question (PRI, BRI, async, sync serial), you can define a virtual template.

interface virtual-template 1
 ip unnumbered e0
 no ip route-cache
 ppp authentication chap
 ppp multilink

! Never define a specific IP address on the virtual template because projected
! virtual access interfaces are always cloned from the virtual template interface.
! If a subsequent PPP link also gets projected to a stack member with a virtual
! access interface already cloned and active, we will have identical IP addresses
! on the two virtual interfaces. IP will erroneously route between them.

! On an AS5200 or 4XXX platform:

! On a TI controller
!
```

```
controller T1 0
  framing esf
  linecode b8zs
  pri-group timeslots 1-24
!
interface Serial 0:23
  no ip address
  encapsulation ppp
  no ip route-cache
  ppp authentication chap
  ppp multilink
!
! On an E1 Controller
!
controller E1 0
  framing crc4
  linecode hdb3
  pri-group timeslots 1-31

interface Serial 0:15
  no ip address
  encapsulation ppp
  no ip route-cache
  ppp authentication chap
  ppp multilink
```

Multichassis Multilink PPP with Dialers

When dialers are configured on the physical interfaces and when the interface itself is a dialer, do not specify a virtual template interface. For dialers, you only need to define the stack group name, common password, and its members across all the stack members. No virtual template interface is defined at all.

Only the PPP commands in dialer interface configuration are applied to the bundle interface. Subsequent projected PPP links are also cloned with the PPP commands from the dialer interface.

Dialer profiles are not supported for SGBP (Stack Group Bidding Protocol).

This section includes the following examples:

- MMP with Explicitly Defined Dialer
- MMP with ISDN PRI but No Explicitly Defined Dialer

MMP with Explicitly Defined Dialer

The following example includes a dialer that is explicitly specified by the **interface dialer** command and configured by the commands that immediately follow:

```
sgbp group stackq
sgbp member systemb 1.1.1.2
sgbp member systemc 1.1.1.3

username stackq password therock

interface dialer 1
 ip unnumbered e0
 dialer map .....
 encapsulation ppp
 ppp authentication chap
 dialer-group 1
 ppp multilink
 !
 ! on a T1 controller
 !
controller T1 0
 framing esf
 linecode b8zs
 pri-group timeslots 1-24

interface Serial0:23
 no ip address
 encapsulation ppp
 dialer in-band
 dialer rotary 1
 dialer-group 1
 !
 ! or on an E1 Controller
 !
controller E1 0
 framing crc4
 linecode hdb3
 pri-group timeslots 1-31

interface Serial0:15
 no ip address
 encapsulation ppp
 no ip route-cache
 ppp authentication chap
 ppp multilink
```

MMP with ISDN PRI but No Explicitly Defined Dialer

ISDN PRIs and BRIs by default are dialer interfaces. That is, a PRI configured without an explicit **interface dialer** command is *still* a dialer interface. The following example configures ISDN PRI. The D-channel configuration on serial interface 0:23 is applied to all the B channels. MMP is enabled, but no virtual interface template needs to be defined.

```
sgbp group stackq
sgbp member systemb 1.1.1.2
sgbp member systemc 1.1.1.3

username stackq password therock

isdn switch-type primary-4ess
controller t1 0
 framing esf
 linecode b8zs
 pri-group timeslots 1-23

isdn switch-type basic-net3
interface Serial0:23
ip unnumbered e0
 dialer map .....
 encaps ppp
 ppp authentication chap
 dialer-group 1
 dialer rot 1
!
ppp multilink
```

Multichassis Multilink PPP with Offload Server

The following example shows a virtual template interface for a system being configured as an offload server (via the **sgbp seed-bid offload** command). All other stack group members must be defined with the **sgbp seed-bid default** command (or if you do not enter any **sgbp seed-bid** command, it defaults to this command).

```
multilink virtual-template 1
sgbp group stackq
sgbp member systemb 1.1.1.2
sgbp member systemc 1.1.1.3
sgbp seed-bid offload
username stackq password therock

interface virtual-template 1
ip unnumbered e0
no ip route-cache
ppp authentication chap
ppp multilink
```

