



Adding System Components with MML

This chapter describes how to add system components, describes how to verify the addition of the components, and gives tips that can help you solve problems. It provides information about:

- [Adding SS7 Signaling Route Components, page 6-2](#)
- [Adding Signaling Link Components, page 6-12](#)
- [Adding MGW Control Links, page 6-17](#)
- [Adding Trunks, Trunk Groups, and Routing, page 6-26](#)

Before starting an actual configuration, refer to [Chapter 2, “Planning for Provisioning”](#) for instructions and worksheets for configuring your system. That chapter describes the system components that need to be configured on the MGC. Each component has a specified type, name, and description, and may have additional configuration parameters.

When adding components, add the components in the following order.

- External nodes
- Point codes (OPC, APC, and DPC)
- Adapter cards
- Adapter interfaces
- Signaling services
- Linksets
- Links
- SS7 routes
- SS7 subsystems
- Trunk groups
- Trunks
- Route groups

Adding SS7 Signaling Route Components

Your first task is to configure SS7 signaling routes that link the MGC to the SS7 network nodes (signaling points). You typically do this by:

- [Adding a Point Code, page 6-2](#)
- [Adding an Adjacent Point Code, page 6-5](#)
- [Adding a Linkset, page 6-6](#)
- [Adding an SS7 Subsystem, page 6-7](#)
- [Adding an SS7 Route, page 6-9](#)
- [Adding an SS7 Signaling Service, page 6-10](#)
- [Adding a TCAP over IP Signaling Service, page 6-10](#)
- [Adding a FAS Signaling Service, page 6-11](#)



Note

When provisioning, fully define all components (see [Figure 2-2](#)) before deploying a configuration.

To add a system component, do the following:

-
- Step 1** Start MML as described in the [“Working with MML”](#) section on page 5-1.
- Step 2** Start a provisioning session as described in the [“Starting a Provisioning Session”](#) section on page 5-8. The source configuration that you chose during startup determines the configuration to which you can add components.
- Step 3** Enter the following command:

```
prov-add:componentType:name="name", desc="description", paramName=value
```

Replace the *componentType* parameter with the type of component you want to create. To define more than one parameter, enter the additional *paramName=value* descriptions on the command line.

Provide a description. The description can be as many as 128 alphanumeric characters in length.

Replace the *name* parameter with the name you want to give to the component. The name can be as many as 20 characters in length and can only be number and letter symbols, and the dash (-) symbol.

SS7 System Component Hierarchy

When you are adding SS7 system components, create them in proper order. For example, before you add a point code, you must define the external nodes.

Adding a Point Code

A point code is an SS7 network address that identifies an SS7 network node, such as a switch, SCP, STP, or SSP. Its MML name is PTCODE. This can be the MGC’s originating point code (OPC) or the destination point code (DPC) of a remote node with which the MGC communicates.

**Note**

For information on point code parameters, refer to [Table 2-1 on page 2-4](#).

Procedure

To add a point code to the MGC configuration, use the **PROV-ADD** command.

Step	Command	Purpose
1	<code>mm1>prov-add:ptcode:name="opc",netaddr="1.1.5",netind=2,desc="id for vsc"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ptcode NAME: opc NETADDR: 1.1.5 NETIND: 2 DESCRIPTION: id for vsc
2	<code>mm1>prov-add:ptcode:name="dpc1",desc=dpc1-1.1.1",netaddr="1.1.1",netind=2,desc="id for dpc1"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ptcode NAME: dpc1 DESC: dpc1-1.1.1 NETADDR: 1.1.1 NETIND: 2 DESCRIPTION: id for dpc1

Verify

Use the **PROV-RTRV** command to verify.

**Tip**

Point codes provide the addressing scheme for the SS7 network. ITU point codes are 14 bits long, and ANSI point codes are 24 bits long.

Adding Multiple OPCs

Depending on your system configuration, you may have to assign more than one OPC to a single MGC. When adding multiple OPCs, keep the following information in mind.

- ITU point codes contain 14 bits and ANSI point codes contain 24 bits.

**Note**

Use care when supplying ITU point codes since they are not checked in the provisioning session.

- A maximum of 6 true OPCs can be supported per MGC.

- For each OPC added, you must specify a different local port number for each C7 IP link on the same interface.
- For each OPC added, you must create a duplicate DPC with a different name but with the same IP address.
- Each OPC must have a unique network address.
- Enter the OPC before creating the C7 IP link.
- When specifying a local port number, it must be greater than 1024 (for example, 7000).
- Each OPC requires its own linkset (a linkset cannot be shared by 2 OPCs).
- A maximum of 2 Session Manager sessions (1 active and 1 standby) can be supported per SLT (1 session per link).
- A maximum of 32 links can be supported per Control Channel.
- A maximum of 16 linksets can be included per Control Channel.
- A maximum of 4,096 DS0s (CICs) can be supported per OPC-DPC pair for ITU or a maximum of 16,384 DS0s (CICs) for ANSI.

Procedure

To add another point code to the MGC configuration, use the **PROV-ADD** command.

Step	Command	Purpose
1	<code>mm1>prov-add:ptcode:name="opc1",netaddr="1.1.1.6",netind=2,desc="id for vsc"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ptcode NAME: opc1 NETADDR:1.1.6 NETIND: 2 DESCRIPTION: id for vsc
2	<code>mm1>prov-add:ptcode:name="dpc2",desc="dpc1-1.1.2",netaddr="1.1.2",netind=2,desc="id for dpc1"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ptcode NAME: dpc2 DESC: dpc1-1.1.2 NETADDR: 214.110.92 NETIND: 2 DESCRIPTION: id for dpc1

Due to the number of commands involved to add an additional OPC, the commands have been included in the following series of commands.

```
prov-sta::srcver="new",dstver="est6"
prov-add:card:name="holy0",type="EN",slot=0,desc="Motherboard ethernet card 0 in va-holyfield"
prov-add:card:name="holy1",type="EN",slot=1,desc="Motherboard ethernet card 1 in va-holyfield"
prov-add:enetif:name="en0",desc="Ethernet IF 1",card="holy0"
```

```

prov-add:enetif:name="en1",desc="Ethernet IF 1",card="holy1"
prov-add:extnode:name="nas1",desc="NAS 5300-1"
prov-add:extnode:name="nas2",desc="NAS 5300-47"
prov-add:extnode:name="nas3",desc="NAS 5300-32"
prov-add:extnode:name="nas4",desc="NAS 5300-34"
prov-add:ptcode:name="opc",desc="opc-1.1.5",netaddr="1.1.5",netind=2,desc="id for vsc"
prov-add:ptcode:name="opc1",desc="opc-1.1.5",netaddr="1.1.6",netind=2,desc="id for vsc"
prov-add:ptcode:name="dpc1",desc="dpc1-1.1.1",netaddr="1.1.1",netind=2,desc="id for dpc1"
prov-add:ptcode:name="dpc2",desc="dpc1-1.1.2",netaddr="1.1.2",netind=2,desc="id for dpc1"
prov-add:ptcode:name="dpc3",desc="dpc1-1.1.1",netaddr="1.1.1",netind=2,desc="id for dpc1"
prov-add:ptcode:name="dpc4",desc="dpc1-1.1.1",netaddr="1.1.2",netind=2,desc="id for dpc1"
prov-add:ss7path:name="ss7-dpc1",side="network",mdo="ANSISS7_STANDARD",dpc="dpc1",desc="SS7 signalling
service for DPC1"
prov-add:ss7path:name="ss7-dpc2",side="network",mdo="ANSISS7_STANDARD",dpc="dpc2",desc="SS7 signalling
service for DPC1"
prov-add:ss7path:name="ss7-dpc3",side="network",mdo="ANSISS7_STANDARD",dpc="dpc3",desc="SS7 signalling
service for DPC1"
prov-add:ss7path:name="ss7-dpc4",side="network",mdo="ANSISS7_STANDARD",dpc="dpc4",desc="SS7 signalling
service for DPC1"
prov-add:naspath:name="nassrv1",extnode="nas1",mdo="BELL_1268_C3",desc="srv to nas1"
prov-add:naspath:name="nassrv2",extnode="nas2",mdo="BELL_1268_C3",desc="nas serice to nas2"
prov-add:naspath:name="nassrv3",extnode="nas3",mdo="BELL_1268_C3",desc="nas serice to nas3"
prov-add:naspath:name="nassrv4",extnode="nas4",mdo="BELL_1268_C3",desc="nas serice to nas4"
prov-add:lnkset:name="ls1",apc="dpc1",type="IP",proto="SS7-ANSI",desc="Linkset 1"
prov-add:lnkset:name="ls2",apc="dpc2",type="IP",proto="SS7-ANSI",desc="Linkset 1"
prov-add:lnkset:name="ls3",apc="dpc3",type="IP",proto="SS7-ANSI",desc="Linkset 1"
prov-add:lnkset:name="ls4",apc="dpc4",type="IP",proto="SS7-ANSI",desc="Linkset 1"
prov-add:ss7route:name="rte1",desc="route to dpc1",dpc="dpc1",lnkset="ls1",opc="opc",pri=1
prov-add:ss7route:name="rte2",desc="route to dpc1",dpc="dpc2",lnkset="ls2",opc="opc",pri=1
prov-add:ss7route:name="rte3",desc="route to dpc1",dpc="dpc3",lnkset="ls3",opc="opc1",pri=1
prov-add:ss7route:name="rte4",desc="route to dpc1",dpc="dpc4",lnkset="ls4",opc="opc1",pri=1
prov-add:c7iplnk:name="link1",if="en0",ipaddr="IP_Addr1",port=7000,peeraddr="172.24.233.193",
peerport=7000,lnkset="ls1",timeslot=0,slc=0,desc="ls1"
prov-add:c7iplnk:name="link2",if="en0",ipaddr="IP_Addr1",port=7000,peeraddr="172.24.233.193",
peerport=7000,lnkset="ls2",timeslot=1,slc=0,desc="ls1"
prov-add:c7iplnk:name="link1-1",if="en0",ipaddr="IP_Addr1",port=7002,peeraddr="172.24.233.194",
peerport=7000,lnkset="ls3",timeslot=0,slc=0,desc="ls2"
prov-add:c7iplnk:name="link2-1",if="en0",ipaddr="IP_Addr1",port=7002,peeraddr="172.24.233.194",
peerport=7000,lnkset="ls4",timeslot=1,slc=0,desc="ls2"
prov-add:iplnk:name="nlink2",if="en0",ipaddr="IP_Addr1",port=3001,peeraddr="172.24.233.75",
peerport=3001,svc="nassrv2",desc="link 1 to NAS2"
prov-add:iplnk:name="nlink3",if="en0",ipaddr="IP_Addr1",port=3001,peeraddr="172.24.233.25",
peerport=3001,svc="nassrv3",desc="link 1 to NAS3"
prov-add:iplnk:name="nlink4",if="en0",ipaddr="IP_Addr1",port=3001,peeraddr="172.24.233.27",
peerport=3001,svc="nassrv4",desc="link 1 to NAS4"
prov-add:files:name="BCFile",file="bcsm",action="import"

```

Adding an Adjacent Point Code

An adjacent point code (APC) defines an SS7 STP through the MGC that it connects to. The APC is the SS7 network address of the STP. Its MML name is APC.

For information on point code parameters, refer to [Table 2-1 on page 2-4](#).

Procedure

Command	Purpose
<code>mml>prov-add:apc:name="STP-A",netaddr="214.111.0",desc="STP A pointcode",netind=2</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: apc NAME: STP-A NETADDR: 214.111.0 NETIND: 2 DESCRIPTION: STP A pointcode

Verify

Use the **PROV-RTRV** command to verify.

Adding a Linkset

A linkset is the group of all signaling links between two point codes. Its MML name is LNKSET.

For information on linkset parameters, refer to [Table 2-2 on page 2-4](#).

Procedure

Step	Command	Purpose
1	<code>mml>prov-add:lnkset:name="linkset1",desc="linkset 1 to STP-A",apc="STP-A",type="IP",proto="SS7-ANSI"</code>	Uses the prov-add command to add the component and required parameters: Component: lnkset APC: STP-A PROTO: SS7-ANSI TYPE: IP Name: linkset 1 Description: 1 linkset 1 to STP-A
2	<code>mml>prov-add:lnkset:name="linkset2",desc="linkset 2 to STP-B",apc="STP-B",type="IP",proto="SS7-ANSI"</code>	Uses the prov-add command to add the component and required parameters: Component: lnkset APC: STP-B PROTO: SS7-ANSI TYPE: IP Name: linkset 2 Description: linkset 2 to STP-B

Verify

Use the **PROV-RTRV** command to verify.



Tip

Setting up linksets is a two-step process that consists of first adding the linkset and then adding links to the linkset.

Adding an SS7 Subsystem

The SS7 subsystem is a logical entity that mates two STPs. When two STPs are defined as mates within the MGC, the controller can use either STP for communications to a destination device. The SS7 subsystem MML name is SS7SUBSYS.

For information on SS7 subsystem parameters, refer to [Table 2-5 on page 2-10](#).

Procedure

Step	Command	Purpose
1	<code>mml>prov-add:ss7subsys:name="mate1",svc="STP-A",matedapc="STP-B",proto="SS7-ANSI",pri=1,desc="mate STP-A to STP-B"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ss7subsys NAME: mate1 SVC: STP-A PROTO: SS7-ANSI MATEDAPC: STP-B PRI: 1 SSN: 0 (default) TRANSPROTO: SCCP (default) DESC: mate STP-A to STP-B
2	<code>mml>prov-add:ss7subsys:name="mate2",apc="STP-B",matedapc="STP-A",proto="SS7-ANSI",pri=2,desc="mate STP-B to STP-A"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ss7subsys NAME: mate2 SVC: STP-B PROTO: SS7-ANSI MATEDAPC: STP-A PRI: 2 SSN: 0 (default) TRANSPROTO: SCCP (default) DESC: mate STP-B to STP-A

Verify

Use the **PROV-RTRV** command to verify.



Tip

Protocol families must be the same for mated subsystems. If one pair of STPs handles both ITU and ANSI variants, you must configure two pairs of STPs: one for ITU and the other for ANSI.

Adding Subsystem Numbers

You can also use the SS7 subsystem to define an SCP using TCAP. For TCAP applications, TRANSPROTO is set to TCPIP, and the subsystem number is set to a value greater than 0 to support AIN. You also must set STPSCPIND to route to the appropriate SCP. For information on SS7 subsystem parameters including STPSCPIND, refer to [Table 2-5 on page 2-10](#).

Procedure

Step	Command	Purpose
1	<code>mml>prov-add:ss7subsys:name="LNP-1",svc="stpa",transproto="SCCP",proto="SS7-ANSI",pri=1,ssn=231,desc="LNP231 for STP A"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ss7subsys NAME: LNP-1 SVC: stpa transproto: SCCP proto: SS7-ANSI pri: 1 SSN: 231 DESC: LNP231 for STP A
2	<code>mml>prov-add:ss7subsys:name="AIN-1",svc="stpb",transproto="SCCP",proto="SS7-ANSI",pri=1,ssn=241,desc="AIN8xx for STP B"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: ss7subsys NAME: AIN-1 SVC: stpb transproto: SCCP proto: SS7-ANSI pri: 1 SSN: 241 DESC: AIN8xx for STP B

Verify

Use the **PROV-RTRV** command to verify.

Adding an SS7 Route

An SS7 route is a path from the MGC to another MGC or SSP switch. Its MML name is SS7ROUTE. For information on SS7 route parameters, refer to [Table 2-6 on page 2-11](#).

Procedure

Step	Command	Purpose
1	<code>mml>prov-add:ss7route:name="rte1DPC1",opc="OPC",dpc="DestSW1PC",lnkset="linkset1",pri=1,desc="route 1 to DestSW1 thru STP-A"</code>	Use the prov-add command to add the component and required parameters: Component: ss7route Name: rte1DPC1 OPC: OPC DPC: DestSW1PC LNKSET: linkset1 PRI:1 Description: route 1 to DestSW1 thru STP-A
2	<code>mml>prov-add:ss7route:name="rte2DPC1",opc="OPC",dpc="DestSW1PC",lnkset="linkset2",pri=1,desc="route 2 to DestSW1 thru STP-B"</code>	Uses the prov-add command to add the component and required parameters: Component: ss7route Name: rte2DPC1 OPC: OPC DPC: DestSW1PC LNKSET: linkset2 PRI:1 Description: route 2 to DestSW1 thru STP-B

Verify

Use the **PROV-RTRV** command to verify.



Tip

You must create a route for each DPC-OPC combination.

Adding an SS7 Signaling Service

An SS7 signaling service specifies the protocol variant and the path that the MGC uses to communicate with a remote switch (SSP) sending bearer traffic to the MGWs. Its MML name is SS7PATH.

For information on signaling service parameters, refer to [Table 2-7 on page 2-11](#).

Procedure

Command	Purpose
<pre>mml>prov-add:ss7path:name="ss7svc1",mdo="ANSI SS7_STANDARD",dpc="dpc1",desc="SS7 svc to dpc1"</pre>	<p>Use the prov-add command to add the component and required parameters:</p> <p>COMPONENT: ss7path NAME: ss7svc1 SIDE: default (network) MDO: ANSI SS7_STANDARD DPC: dpc1 CUSTGRPID: default (0000) CUSTGRPTBL: default (0101) ASPPART: default (N) DESCRIPTION: SS7 service to dpc1</p>

Verify

Use the **PROV-RTRV** command to verify.



Tip

Do not change the default values for CUSTGRPID and CUSTGRPTBL; they are used for DPNSS feature transparency.

CUSTGRPID also associates variants and dial plans. Use **rtrv-variants** to see valid variants.

Adding a TCAP over IP Signaling Service

TCAP over IP signaling service is the path to an STP/SCP. Its MML name is TCAPIPPATH.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Command	Purpose
<code>mml>prov-add:TCAPIPPATH:name="tcapiprv1",extnode="scp2",desc="TCAPIP Service to SCP 2"</code>	Use the prov-add command to add the component and required parameters: Component: tcapipath Name: tcapiprv1 EXTNODE: scp2 Description: TCAPIP Service to SCP 2

Verify

Use the **PROV-RTRV** command to verify.



Tip

When defining a new ITU service for SCCP/TCAP, you must first enter a “dummy” DPC. This DPC is not used for routing calls, but is used internally by the software.

To define a dummy DPC, refer to the [“Adding a Point Code” section on page 6-2](#) and add a DPC with the following parameters:

- MML name: itudummy
- Description: Dummy Point Code for ITU service
- Network address: 9.9.9
- Network Indicator: National network

You also select this dummy DPC when you are provisioning the SS7 route for the ITU service. Refer to the [“Adding an SS7 Route” section on page 6-9](#).

Adding a FAS Signaling Service

This is the facility associated signaling (FAS) service or signaling path to a particular destination when you are using either ISDN-PRI or DPNSS. Its MML name is FASPATH.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Step	Command	Purpose
1	<pre>mml>PROV-ADD:FASPATH:NAME="FASPATH1",SIDE="network",MDO="ETSI_300_102",ASPPART="N",DESC="FASPATH 1",ABFLAG="a",CRLEN=1</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: faspath Name: FASPATH1 Side: network MDO: ETSI_300_102 CUSTGRPID: CUSTGRPTBL: ABFLAG: a CRLEN: 1 ASPPART: N Description: Destination Switch 1 - Jane</p>

Verify

Use the **PROV-RTRV** command to verify.

Adding Signaling Link Components

After configuring the SS7 signaling routes, you need to configure the signaling link components. These components link the MGC to the STPs and to the MGWs. You will typically do this by:

- [Adding Two Ethernet Cards, page 6-12](#)
- [Adding Two Ethernet Interfaces, page 6-13](#)
- [Adding a C7 IP Link, page 6-14](#)
- [Adding a TDM Interface, page 6-15](#)
- [Adding a TDM Link, page 6-16](#)

Adding Two Ethernet Cards

These are network card or adapter cards that are operating in the MGC. Its MML name is CARD.

For information on card parameters, refer to [Table 2-10 on page 2-20](#).

Procedure

Step	Command	Purpose
1	<code>mml>prov-add:card:name="Ethernet1",desc="Ethernet Card 1",type="EN",slot=0,</code>	Uses the prov-add command to add the component and required parameters: <ul style="list-style-type: none"> • COMPONENT: card • DESCRIPTION: Ethernet Card 1 • NAME: Ethernet 1 • TYPE: EN • SLOT: 0
2	<code>mml>prov-add:card:name="Ethernet2",type="EN",slot=1,desc="Ethernet Card 2"</code>	Uses the prov-add command to add the component and required parameters: <ul style="list-style-type: none"> • COMPONENT: card • NAME: Ethernet 2 • TYPE: EN • SLOT: 1 DESCRIPTION: Ethernet Card 2

Verify

Use the **PROV-RTRV** command to verify.



Tip

You must configure the adapter card before you configure its corresponding interface.

Adding Two Ethernet Interfaces

The Ethernet interface provides the physical line interface between a MGC Ethernet network card/adaptor and the physical Ethernet network. You configure parameters that control communications between the network card/adaptor and the Ethernet. Its MML name is ENETIF.

For information on Ethernet interface parameters, refer to [Table 2-11 on page 2-21](#).

Procedure

Step	Command	Purpose
1	<code>mml1>prov-add:enetif:name="EtherIF1", desc="Ethernet IF 1",card="Ethernet1"</code>	Uses the prov-add command to add the component and required parameters: <ul style="list-style-type: none"> • COMPONENT: enetif • NAME: EtherIF1 • DESCRIPTION: Ethernet IF 1 • CARD: Ethernet1
2	<code>mml1>prov-add:enetif:name="EtherIF2", desc="Ethernet IF 2",card="Ethernet2"</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: enetif NAME: EtherIF2 DESCRIPTION: Ethernet IF 2 CARD: Ethernet2

Verify

Use the **PROV-RTRV** command to verify.



Tip

You must configure the adapter/card before configuring the interface.

Adding a C7 IP Link

A C7 IP link component identifies a link between a Cisco SLT IP address and port and the SS7 network (SSP or STP). Its MML name is C7IPLNK.



Tip

For SS7 provisioning, keep the following points in mind.

- A maximum of 6 OPCs that can be supported.
- Enter routing information for the OPC before creating the C7 IP link.
- For each OPC added, you must specify a different local port for each C7 IP link.
- Provision a maximum of 32 links per local port number. Specify another port number for each additional group of 32 links.

For information on C7 IP link parameters, refer to [Table 2-14 on page 2-23](#).

Procedure

Command	Purpose
<pre>mml>prov-add:c7iplnk:name="lkset1SLC0",if="EtherIF1", ipaddr="IP_Addr1",port=7000, peeraddr="10.15.4.23", peerport=32767,pri=1,slc=0, lnkset="linkset1",desc= "linkset1 SLC 0 thru SLT-23 ser0/0",timeslot=0</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>COMPONENT: c7iplnk NAME: lkset1SLC0 IF: EtherIF1 PORT: 7000 PRI: 1 PEERADDR: 10.15.4.23 PEERPORT: 32767 IPADDR: IP_Addr1 SLC: 0 LNKSET: linkset1 TIMESLOT: 0 DESC: linkset1 SLC 0 thru SLT-23 ser0/0</p>

Verify

Use the **PROV-RTRV** command to verify.



Tip

Use this component only when the MGC uses Cisco SLTs to communicate SS7 messages over IP.

The peerport must be set to 32767. The peerport value is contained in the XECfgParm field stPort. For more information, refer to the *Cisco Media Gateway Controller Software Release 7 Installation and Configuration Guide*.

Adding a TDM Interface

The TDM interface provides the physical line interface between a MGC TDM network card/adaptor and the physical TDM network. Its MML name is TDMIF.

For information on TDM interface parameters, refer to [Table 2-19 on page 2-31](#).

Procedure

Command	Purpose
<pre>mml>prov-add:tdmif:name="card1lif1",desc="V35 LIF 1", card="card1", lifnum=2, sigtype="V.35", datarate=64</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: tdmif Name: card1lif1 CARD: card1 LIFNUM: 2 RESIST: default (0) DATARATE: default (64) CLOCK: default (EXT) DTEDCE: default (DTE) CODING: default (NA) FORMAT: default (NA) SIGTYPE: V.35 HDLC:(DEFAULT) DESC: V35LIF 1</p>

Verify

Use the **PROV-RTRV** command to verify.



Tip

[Table 6-1](#) shows typical line interface parameters based on card type.

Table 6-1 Typical Line Interface Parameters

Card Type	LIFNUM	RESIST	Data Rate/ Clock	DTEDCE	Line Coding	Format/ Framing	Signal Type	I/HDLC
ITK (T1)	1	75		NA	B8ZS	ESF	T1	IHDLC
ITK (E1)	1	120		NA	HDB3	CRC4	CEPT	IHDLC
V.35	2	0	64/EXT	DTE	NA	NA	V.35	DEFAULT

Adding a TDM Link

A TDM link is a communications link between a TDM interface card on the MGC and a TDM hardware element. For each link, you need to specify the card interface to which the link connects. Its MML name is TDMLNK.

For information on TDM link parameters, refer to [Table 2-15 on page 2-24](#).

Procedure

Command	Purpose
<pre>mml>prov-add:tdmlnk:name="tdmlink1",if="card1lif1",pri=2,slc=2,svc="ls-1",desc="signal link 1"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: tdmlink Name: tdmlink1 IF: card1lif1 PRI: 2 SLC: 2 SVC: ls-1 Description: signal link 1</p>

Verify

Use the **PROV-RTRV** command to verify.

Adding MGW Control Links

Now you need to configure MGW control links. The MGC uses these links to control the bearer traffic that passes between each MGW. You typically add MGW control links by:

- [Adding an External Node, page 6-17](#)
- [Adding a Card, page 6-18](#)
- [Adding an Ethernet Interface, page 6-18](#)
- [Adding an EISUP Signaling Service, page 6-18](#)
- [Adding an SGCP Signaling Service, page 6-19](#)
- [Adding an IPFAS Transport Service, page 6-20](#)
- [Adding and Modifying an MGCP Signaling Service Property, page 6-20](#)
- [Adding a NAS Signaling Service, page 6-21](#)
- [Adding an IP Link, page 6-22](#)

Adding an External Node

An external node is an MGW with which the MGC communicates. Its MML name is EXTNODE. For information on external node parameters, refer to [Table 2-16 on page 2-28](#).

Procedure

Command	Purpose
<code>mml>prov-add:extnode:name="mgx-8260", type="MGX8260" desc="MGX 8260"</code>	Use the prov-add command to add the component and required parameters: Component: extnode Name: mgx-8260 Type: MGX8260 Description: MGX 8260

Verify

Use the **PROV-RTRV** command to verify.



Tip

You must create an external node for each media gateway.

Adding a Card

The card being referred to is a network card or adapter that is operating in the MGC. Its MML name is **CARD**.

Adding an Ethernet Interface

The Ethernet interface provides the physical line interface between an MGC Ethernet network card/adapter and the physical Ethernet network. You configure parameters that control communications between the network card/adapter and the Ethernet. Its MML name is **ENETIF**.

Each SS7 link in the node must be associated with an Ethernet interface component, which must be associated with a network card. The Ethernet interface represents a physical network connection on the network card.



Note

In the MGC, the same cards and interfaces can be used for communication with Cisco SLTs and media gateways. When configured this way, separate links are assigned for Cisco SLT and media gateway communications.

Adding an EISUP Signaling Service

The EISUP signaling service or signaling path is the signaling path to an externally located MGC (destination). Its MML name is **EISUPPATH**.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Command	Purpose
<code>mml>prov-add:eisuppath:name="eisupsrv1",extnode="extseq1",desc="EISUP Service to Ext Seq Node1"</code>	Use the prov-add command to add the component and required parameters: Component: eisuppath Name: eisupsrv1 EXTNODE: extseq1 Description: EISUP Service to Ext Seq Node1

Verify

Use the **PROV-RTRV** command to verify.



Note

To ensure correct failover operation in a configuration with two local MGCs (one active and one standby) and a remote MGC, you need a minimum of two EISUP links from the remote VSC to each MGC redundant pair.

Adding an SGCP Signaling Service

The SGCP signaling service is the protocol path between the MGC and the MGW. Its MML name is SGCPPATH.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Command	Purpose
<code>mml>prov-add:SGCPPATH:name="sgcpcsr1",extnode="mgw2",desc="SGCP Service to mgw 2"</code>	Uses the prov-add command to add the component and required parameters: Component: sgcpcsr1 Name: sgcpcsr1 EXTNODE: mgw2 Description: SGCP Service to mgw 2

Verify

Use the **PROV-RTRV** command to verify.

For SGCP 1.0 only.

Adding an IPFAS Transport Service

The FAS over IP transport service or signaling path is the transport service from a Gateway to an MGC. Its MML name is IPFASPath.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Command	Purpose
<pre>mml>prov-add:ipfaspath:name="ipfassvc1",extnode="nas1",desc="PRI Backhaul Service to NAS1", mdo="ETSI_300_172", custgrpId="1111", custgrptbl="0101", asppart="Y", abflag="a", crlen=1</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: ipfaspath</p> <p>Name: ipfassvc1</p> <p>EXTNODE: nas1</p> <p>SIDE: network (default)</p> <p>MDO: ETSI_300_172</p> <p>CUSTGRPID: 1111</p> <p>CUSTGRPTBL: 0101 (default)</p> <p>ABFLAG: a</p> <p>CRLEN: 1</p> <p>ASSPART: Y</p> <p>Description: PRI Backhaul Service to NAS1</p>

Verify

Use the **PROV-RTRV** command to verify.

Adding and Modifying an MGCP Signaling Service Property

Adding an MGCP Signaling Service Property

The MGCP signaling service or signaling path is the signaling service to a trunking gateway. Its MML name is MGCPPATH.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Command	Purpose
<code>mml>prov-add:mgcppath:name="mgcpsrv1",extnode="cu1",desc="MGCP Service to CU 1"</code>	Uses the prov-add command to add the component and required parameters: Component: mgcppath Name: mgcpsrv1 EXTNODE: cu1 Description: MGCP Service to CU 1

Verify

Use the **PROV-RTRV** command to verify.

Modifying an MGCP Signaling Service Property

The MGCP signaling service property is the signaling service to a trunking gateway. The following is an example of how to change the codec used between an ingress and egress MGW. Its MML name is `GWDefaultCodecString`.

Procedure

To change an MGCP signaling service property to the media gateway configuration, use the **PROV-ED** command as follows:

Command	Purpose
<code>mml>prov-ed:sigsvccprop:name="mgcsrv1",GWDefaultCodecString="G711_U,G729",desc="MGC Signaling Service to MGW1"</code>	Uses the prov-ed command to change the component and required parameters: Component: sigsvccprop Name: mgcsrv1 GWDefaultCodecString: G711_U,G729 Description: MGC Signaling Service to MGW1

Verify

Use the **PROV-RTRV** command to verify the MGCP signaling service was changed.

Adding a NAS Signaling Service

The NAS signaling service is the Q.931 protocol path or NAS signaling service between the MGC and the MGW. Its MML name is `NASPATH`.

**Note**

If you are configuring a redundant system, you must define two redundant link manager links between each MGC and MGW. Each redundant link manager group must be associated with a different port number and a different NASPATH, but the same EXTNODE.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

Procedure

Command	Purpose
<pre>mml>prov-add:naspath:name="nassrv1",extnod="nas1",desc="Service to NAS1",mdo="BELL_1268_C3"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: naspath Name: nassrv1 EXTNODE: nas1 MDO: BELL_1268_C3 Description: Service to NAS1</p>

Verify

Use the **PROV-RTRV** command to verify.

**Tip**

For the NASPATH component, there is only one protocol: Bell_1268_C3.

Adding an IP Link

The IP link is an IP connection between an MGC Ethernet interface and an MGW. Its MML name is IPLNK.

For information on IP link parameters, refer to [Table 2-21 on page 2-33](#).

Procedure

Command	Purpose
<pre>mml>prov-add:iplnk:name="Iplink1",if="en-1lif1",ipaddr="IP_Addr1",port=3001,peeraddr="192.12.214.10",peerport=3001,svc="ipfassvc1",sigslot=1,sigport=1,desc="IP link for IPFAS service to NAS1"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: iplnk</p> <p>Name: Iplink1</p> <p>IF: en-1lif1</p> <p>PORT: 3001</p> <p>PRI: 1 (default)</p> <p>PEERADDR: 192.12.214.10</p> <p>PEERPORT: 3001</p> <p>IPADDR: IP_Addr1</p> <p>SIGSLOT: 1</p> <p>SIGPORT: 1</p> <p>SVC: ipfassvc1</p> <p>Description: IP link for IPFAS service to NAS1</p>

Verify

Use the **PROV-RTRV** command to verify.



Tip

When configuring two IP links to the same NAS, you need to configure two different Ethernet IP addresses on both the MGC and the NAS.

Adding an IP Link Requiring a Subnet Address (Release 7.4(10))

The IP link is an IP connection between an MGC Ethernet interface and an MGW. Its MML name is IPLNK. When the IP link is to another subnet, the optional NEXTHOP and NETMASK parameters are recommended.

For information on IP link parameters, refer to [Table 2-21 on page 2-33](#).

Procedure

Command	Purpose
<pre>mml>prov-add:iplnk:name="Iplink1",if="en-1lif1",ipaddr="IP_Addr1",port=3001,peeraddr="192.12.214.10",peerport=3001,svc="ipfassvc1",sigslot=1,sigport=1,desc="IP link for IPFAS service to NAS1",nexthop="172.24.235.1",netmask="255.255.255.0"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: iplnk</p> <p>Name: Iplink1</p> <p>IF: en-1lif1</p> <p>PORT: 3001</p> <p>PRI: 1 (default)</p> <p>PEERADDR: 192.12.214.10</p> <p>PEERPORT: 3001</p> <p>IPADDR: IP_Addr1</p> <p>SIGSLOT: 1</p> <p>SIGPORT: 1</p> <p>SVC: ipfassvc1</p> <p>Description: IP link for IPFAS service to NAS1</p> <p>NEXTHOP: 172.24.235.1</p> <p>NETMASK: 255.255.255.0</p>

Verify

Use the **PROV-RTRV** command to verify.



Tip

When configuring two IP links to the same NAS, you need to configure two different Ethernet IP addresses on both the MGC and the NAS.

All IP links and C7 IP links with a PEERADDR on the same network must have the same NEXTHOP. For each IP link and C7 IP link, the NEXTHOP must be on the same network address as the IPADDR. When NEXTHOP is set to 0.0.0.0, the IP routing feature is disabled.

A value of 0.0.0.0 for the NETMASK is not allowed.

As a binary number, the NETMASK cannot have any 1 bits less significant than the most significant 0 bits. For example, a NETMASK of 0.0.255.255 is invalid.

Adding Multiple FAS Signaling Services and IP Links

To configure the D channels on a Cisco MGX8260, you have to provision one IPFAS path and two IP links for each D channel. The MML component for provisioning multiple IPFAS paths and associated IP links on the Cisco MGX8260 is MLTIPFAS.

For information on signaling service parameters, refer to [Table 2-17 on page 2-29](#).

The adding multiple FAS signaling services and IP links procedure adds two IPFAS paths using FAS type signaling and four IP links (two per IPFAS path).

Procedure

Step	Command	Purpose
1	<pre>mm1>PROV-ADD:MLTIPFAS:NAME="BSC1",PATHSIZE=2, MDO="BELL_1268",EXTNODE="MGX-BH",PORT=7007,PE ERPORT=7007,SIGPORT=1,SIGSLOT=10,IF1="ENIF1", IF2="ENIF2",IPADDR1="IP_Adr1",IPADDR2="IP_Ad dr2",PEERADDR1="10.15.26.20",PEERADDR2="10.15 .27.20",SIDE="network",CUSTGRPID="V123"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: mltipfas</p> <p>Name: bsc1</p> <p>pathsiz: 2</p> <p>mdo: bell_1268</p> <p>extnode: mgx-bh</p> <p>port: 7007</p> <p>peerport: 7007</p> <p>sigport: 1</p> <p>sigslot: 10</p> <p>if1:enif1</p> <p>if2: enif2</p> <p>ipaddr1: ip_addr1</p> <p>ipaddr2: ip_addr2</p> <p>perraddr1: 10.15.26.20</p> <p>peeradr2: 10.15.27.20</p> <p>side: network</p> <p>custgrp: v123</p>

Verify

Use the **PROV-RTRV** command to verify.

MLTIPFAS supports add and delete operations only.

Removing Multiple FAS Signaling Services and IP Links

You must specify only the NAME and PATHSIZE parameters to remove several IPFAS paths and associated IP links. In the following example, two IPFAS paths and associated IP links are deleted.

Procedure

Command	Purpose
<code>mml>prov-dlt:mltipfas:name="bh531",pathsize="2"</code>	Uses the prov-dlt command to delete the component and required parameters: COMPONENT: mltipfas NAME: bh531 PATHSIZE: 2

Verify

Use the **PROV-RTRV** command to verify.

Adding Trunks, Trunk Groups, and Routing

You now need to configure trunks, trunk groups, and routing. The MGC uses this information for determining the call traffic on each trunk between the switches and the MGWs. You typically do this using the following:

- [Adding Files, page 6-26](#)
- [Adding a Nailed Trunk \(Bearer Channel\), page 6-27](#)
- [Routing, page 6-28](#)

Adding Files

The FILES component consists of customer-specific flat files that you can use to provision trunk groups, routing, trunks, and dial plans. The MML name is FILES.

For information on routing parameters, refer to the [“Provisioning Trunk Groups and Trunks” section on page 2-34](#).

Procedure

Command	Purpose
<code>mml>prov-add:files:name="BCFile",file="trunkCust.dat",action="import"</code>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: files</p> <p>Name: BCFile (a bearer channel file type)</p> <p>File: trunkCust.dat</p> <p>Action: import</p>



Note

When you are importing screening files, for example AWhite list or BBlack list, the import file name must be one of the following: <custGrpId>.awhite, <custGrpId>.bwhite, <custGrpId>.ablack, or <custGrpId>.bblack.

Verify

Use the **PROV-RTRV** command to verify.

Adding a Nailed Trunk (Bearer Channel)

The nailed trunk component is for adding individual nailed bearer channels in a Dial Access configuration. Its MML name is NAILEDTRNK.

For information on routing parameters, refer to the [“Provisioning Trunk Groups and Trunks”](#) section on page 2-34.

Procedure

Command	Purpose
<code>mml>prov-add:nailedtrnk:name="101",srsvc="ss7svc1",srctimeslot=101,dstsvc="nassrv1",dstspan=3,dsttimeslot=1</code>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: nailedtrnk</p> <p>Name: 101 (ID)</p> <p>SRCSVC: ss7svc1</p> <p>DSTSVC: nassrv1</p> <p>SRCPAN: default (0)</p> <p>DSTSPAN: 3</p> <p>SRCTIMESLOT: 101</p> <p>DSTTIMESLOT: 1</p>

Verify

Use the **PROV-RTRV** command to verify.



Tip

Use the FILES component with flat files to provision trunks; use the NAILEDTRNK component with an individual trunk.



Note

If you have already provisioned the spans on a network access server (NAS) that are to be associated with the trunks being added, remove the spans from service on the NAS using the shutdown command before provisioning trunks on the MGC.

Adding a Trunk Group

The trunk group component is for provisioning individual trunk groups. Its MML name is TRNKGRP.

For information on TRNKGRP parameters, refer to [Table 2-23 on page 2-36](#).

Procedure

Command	Purpose
<pre>mml>prov-add:trngrp:name="1000",clli="tttt-s s-bb-xxx",svc="ss7svc1",type="tdm_gen", selseq="lidl",qable="n" Virtual Switch Controller 2000-11-10 15:32:25 M COMPLD "TRNKGRP"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: trngrp Name: 1000 CLLI: tttt-ss-bb-xxx SERVICE: ss7svc1 TYPE: tdm_gen SELSEQ:lidl (select sequence) QABLE: n (time slot)</p>

Verify

Use the **PROV-RTRV** command to verify.

Routing

This section is used to configure the routing file. Three components are necessary to configure routing. Their MML names are RTTRNKGRPT, RTTRNK, and RTLIST.

For information on routing parameters, refer to the [Table 2-26 on page 2-41](#).

Procedure

Step	Command	Purpose
1	<code>mml>prov-add:rttrnkgrp:name="501910",type=7,reattempts=1,queuing=0,cutthrough=2</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: rttrnkgrp NAME: 501910 TYPE: 7 REATTEMPTS: 1 QUEUING: 0 CUTTHROUGH: 2
2	<code>mml>prov-add:rttrnk:name="rt513913",trnkgrpnum=513913</code>	Uses the prov-add command to add the component and required parameters: COMPONENT: rttrnk NAME: rt513913 TRNKGRPNUM:513913
3	<code>mml>prov-add:rtlist:listindex=1,name="rtlist501910",rtname="rt501910",carrierid=333</code>	Uses the PROV-ADD command to add the component and required parameters: Component: rtlist LISTINDEX: 1 NAME: rtlist501910 RTNAME: rt501910 CARRIERID: 333

Verify

Use the **PROV-RTRV** command to verify.



Tip

All the route lists, route trunks, and route trunk groups information can be retrieved by using the **prov:rtrv:rtlist:"ALL"** command. The ALL option cannot be used with other parameters.

Overriding the Trunk Group Property

The trunk group component is used for provisioning trunk group properties. Its MML name is TRNKGRPPROP. The following command overrides the trunk group property NPA for trunk group number 1000.

For information on TRNKGRPPROP properties, refer to [Table 2-24 on page 2-37](#).

Procedure

Command	Purpose
<pre>mml>prov-add:TRNKGRPPROP:NAME="1000",NPA="703" Virtual Switch Controller 2000-11-10 15:32:25 M COMPLD "TRNKGRPPROP"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: trnkgrp</p> <p>Name: 1000</p> <p>NPA: 703</p>

Verify

Use the **PROV-RTRV** command to verify.

Adding A Switched Trunk (Multiple Switched Trunks)

The trunk (switched bearer channel) component is used for provisioning multiple switched trunks. Its MML name is SWITCHTRNK.

For information on SWITCHTRNK parameters, refer to the [“Creating the Trunk Group” section on page 2-36](#).

The following command adds the six switched trunks shown in [Table 6-2 on page 6-31](#).

Procedure

Command	Purpose
<pre>mml>prov-add:switchtrnk:name="1",trnkgrpnum=" 1000",span="ffff",cic=25,cu="gw1",spansize=6, endpoint="S0/DS1-1/6@li-5300-3" Virtual Switch Controller 2000-11-30 08:54:50 M COMPLD Virtual Switch Controller 2000-11-10 15:32:25 M COMPLD "switchtrnk"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: switchtrnk</p> <p>Name: 1</p> <p>TRUNKGRPNUM: 1000 (trunk group number)</p> <p>SPAN: ffff</p> <p>CIC: 25 (circuit identifier code)</p> <p>CU: gw1 (coding unit)</p> <p>SPANSIZE: 6</p> <p>ENDPOINT: S0/DS1-1/6@li-5300-3</p>

Table 6-2 Result of Adding a Switched Trunk Command

Trunk Group Number	Trunk Group Member	Span	CIC	Endpoint	CLI
1000	1	ffff	25	S0/DS1-1/7@li-5300-3	gw1
1000	2	ffff	26	S0/DS1-1/8@li-5300-3	gw1
1000	3	ffff	27	S0/DS1-1/9@li-5300-3	gw1
1000	4	ffff	28	S0/DS1-1/10@li-5300-3	gw1
1000	5	ffff	29	S0/DS1-1/11@li-5300-3	gw1
1000	6	ffff	30	S0/DS1-1/12@li-5300-3	gw1

Verify

Use the **PROV-RTRV** command to verify.



Note

If you have already provisioned the spans on a network access server (NAS) that are to be associated with the trunks being added, remove the spans from service on the NAS using the shutdown command before provisioning trunks on the MGC.

Retrieving Multiple Switched Trunks

To retrieve multiple switched trunks based on the trunk group number, span, or CU name, use the **PROV-RTRV** command.

Procedure

Command	Purpose
<pre>mml> prov-rtrv:switchtrnk:trnkgrpnum="1000" Virtual Switch Controller 2000-11-29 10:13:10 M RTRV "session=cujo:switchtrnk" /* NAME SPAN CIC TRNKGRPNUM CU ENDPOINT 25 ffff--25---1000----gw1-S0/DS1-1/1@li5300-3 26 ffff--26---1000----gw1-S0/DS1-1/2@li5300-3 27 ffff--27---1000----gw1-S0/DS1-1/3@li5300-326</pre>	Retrieves all the components associated with trunk group 1000.

Verify

Use the **PROV-RTRV** command to verify.

Adding Multiple Nailed Trunks

To add multiple nailed trunks based on source svc, source span, dest-svc, and dest-span, use the PROV-ADD command.

Procedure

Command	Purpose
<pre>mml>prov-add:nailedtrnk:name="100",srcsvc="SC-1", dstsvc="PC-7-200-7",srcspan="0",dstspan="ffff", srctimeslot="1",dsttimeslot="4065",spansize=6 Virtual Switch Controller 2000-11-30 08:54:50 M COMPLD "nailedtrnk"</pre>	Adds six nailed trunks.

The previous command adds the six nailed trunks shown in [Table 6-3](#).

Table 6-3 Result of Adding a Nailed Trunk Command

Name	SRCSVC	SRCSPAN	SRCTIMESLOT	DSTSVC	DSTSPAN	DSTTIMESLOT
1	SC-1	0	1	PC-7-200-7	ffff	4065
2	SC-1	0	2	PC-7-200-7	ffff	4066
3	SC-1	0	3	PC-7-200-7	ffff	4067
4	SC-1	0	4	PC-7-200-7	ffff	4068
5	SC-1	0	5	PC-7-200-7	ffff	4069
6	SC-1	0	6	PC-7-200-7	ffff	4070

Verify

Use the **PROV-RTRV:nailedtrnk:srcsvc="sc-1"** command to verify.

Removing Multiple Nailed Trunks

To remove multiple nailed trunks based on source svc, source span, dest-svc, and dest-span, use the PROV-DLT command.

Procedure

Command	Purpose
<pre>mml>prov-dlt:nailedtrnk:name="100",srcsvc="SC-1", dstsvc="PC-7-200-7",srcspan="0",dstspan="ffff", srctimeslot="1",dsttimeslot="4065",spansize=6 Virtual Switch Controller 2000-11-30 08:54:50 M COMPLD "nailedtrnk"</pre>	Deletes six nailed trunks.

The previous command deletes the six nailed trunks.

Verify

Use the **PROV-RTRV:nailedtrnk:srcsvc="sc-1"** command to verify.

Retrieving Multiple Nailed Trunks

To retrieve multiple nailed trunks, use the **PROV-RTRV** command.

Procedure

Command	Purpose
<pre>mml>prov-rtrv:nailedtrnk:srcsvc="SC-1" Virtual Switch Controller 2000-11-29 12:47:41 M RTRV "session=xxx:nailedtrnk"</pre>	Retrieves all nailed trunks associated with the source service SC-1.

Verify

Observe the screen to verify the command.

Only one source service, destination service, source span, destination span is allowed at a time.

Adding Multiple Trunk Groups and Bearer Channels

The multiple trunk group and bearer channel components are for provisioning multiple PRI trunk groups and bearer channels. Its MML name is MLTRNKGRP.

Procedure

Command	Purpose
<pre>mm1>prov-add:mlttrnkgrp:name="1000",svc="bsc1", cli="5300E4011",numtrnkgrp=2,spansize=4,trnkmemum= 1,span=0,cic=1,endpoint="S10/DS1-0/1@mgx-8260,cu="m gx-east"</pre>	<p>Uses the prov-add command to add the component and required parameters:</p> <p>Component: mlttrnkgrp</p> <p>Name: 1000</p> <p>Svc: bsc1</p> <p>Cli: 5300E4011</p> <p>NUMTRNKGRP: 2</p> <p>SPAN SIZE: 4</p> <p>TRNKMEMUM: 1</p> <p>SPAN: 0</p> <p>CIC: 1</p> <p>ENDPOINT: S10/DS1-0/1@mgx-8260</p> <p>CU: mgx-east</p>

Verify

Use the **PROV-RTRV:mlttrnkgrp:name="1000",nexttrtrnkgrp=2** command to verify.

You cannot provision other trunk group types (for example, TDM or IP) with MLTTRNKGRP.

Removing Multiple Trunk Groups and Bearer Channels

You need to specify only the NAME and NUMTRNKGRP parameters to remove several multiple trunk groups and associated bearer channels.

Procedure

Command	Purpose
<pre>mm1>prov-dlt:mlttrnkgrp:name="1000",numtrnkgrp=2</pre>	<p>Uses the prov-dlt command to delete the component and required parameters:</p> <p>Component: mlttrnkgrp</p> <p>Name: 1000</p> <p>NUMTRNKGRP: 2</p>

Verify

Use the **PROV-RTRV** command to verify.

Scaling System Components

After you have configured your system components, you can begin scaling your system. Keep the following in mind when scaling.



Tip

A maximum of 6 OPCs can be supported per MGC.

Enter routing information for the OPC before creating the C7 IP link.

For each OPC added, you must specify a different local port for each C7 IP link.

Provision a maximum of 32 links per local port number. Specify another port number for each additional group of 32 links. As many as 192 links can be supported per MGC.

Planning for future network expansion by spreading the linksets evenly across the Control Channels is suggested. Failure to do so will require the linksets to be removed from service to add more links.

As many as 256 NASs can be supported. When creating IP links to the NASs, increment the MGC port number after 32 links have been added. Be sure to set the NAS RLM to match the MGC RLM port value.

Dynamically Configuring the Input/Output Channel Controller

When dynamically configuring the IOCC, evenly distribute number of channels associated with one channel controller. For different signaling service, there are different rules when associating channels with channel controllers. The number of links associated with a channel controller is configurable on a protocol family basis through parameters contained in XECfgParm.dat. If the number of links exceeds the limit defined in XECfgParm.dat, a new instance of channel controller is created.

The naming convention for creating a new channel controller is the first five characters of the protocol family, plus a dash (-), and <num>, where num is number of channel controllers per protocol family created so far.

Table 6-4 Scaling Links per Protocol Family

Signaling Type	Protocol Family	Criteria for a Unique IOCC	Criteria for a Valid Link (Channel)	Parameter in XECfgParm.dat (Default maximum number of links)
NAS	PRIIP	Port number. Number of links. When a channel controller is created, the RLM port number is created as the property port for this channel with the value of the actual port number (minus 1) in properties.dat. The format is: <IOCC MML Name>.port = <port number> - 1	Local port and peer port must be the same. The port number must always be an odd number. The number of links on the same port cannot exceed the maximum number of links specified in XECfgParm.dat. Links associated with the same signaling service must use the same port number. (that is, redundant links). Redundant links do not count when validating the maximum number of links per IOCC.	MaxNumLinks (32)
IPFAS	PRIL3	Number of links. Links associated with the same port number cannot be split over different channel controllers.	The number of links on the same port cannot exceed the maximum number of links specified in XECfgParm.dat. Links associated with the same signaling service must use the same port number. (that is, redundant links). Redundant links do not count when validating the maximum number of links per IOCC.	MaxNumPRIL3Links (168)
MGCP	MGCP	Number of links. Links associated with the same port number cannot be split over different channel controllers.	The number of links on the same port cannot exceed the maximum number of links specified in XECfgParm.dat. Links associated with the same signaling service must use the same port number. (that is, redundant links). Redundant links do not count when validating the maximum number of links per IOCC.	MaxNumMGCPLinks (1000)
SGCP	SGCP	Number of links.		MaxNumLinks (32)
EISUP	EISUP	Number of links.		MaxNumLinks (32)
FAS	ISDNPRI DPNSS	Number of links.		MaxNumLinks (32)
TCAP OverIP	TCAP OverIP	Number of links.		MaxNumLinks (32)

Table 6-4 *Scaling Links per Protocol Family*

Signaling Type	Protocol Family	Criteria for a Unique IOCC	Criteria for a Valid Link (Channel)		Parameter in XECfgParm.dat (Default maximum number of links)
			Protocol Family	Switch Type	
S77	SS7-ANSI SS7-UK SS7-ITU SS7-China SS7-Japan	Protocol Family Switch Type OPC Number of links.	Protocol Family	Switch Type	MaxNumLinks (32)
			SS7-ANSI	0	
			SS7-China	0, 5	
			SS7-ITU	0, 5	
			SS7-Japan	0, 10	
			SS7-UK	0, 5	

Table 6-5 *Maximum Scaling Limits for the SS7 Components*

Component	Scaling Limit
SS7 IOCC Instances	6
Linksets per SS7 IOCC	16
Links per SS7 IOCC	32
DPCs per SS7 IOCC	100
True OPCs per SS7 IOCC	1
Routes per SS7 IOCC	200
Protocol families per SS7 IOCC	1
Switch types per SS7 IOCC	1
Links per MGC*	192
Linksets per MGC*	96
True OPCs per MGC*	6
DPCs per MGC*	600
Routes per MGC*	1200

* Indicates the component must be spread evenly across the maximum number of IOCC instances.

